Energy-efficient Wireless Sensors: Fewer Bits, Moore MEMS

by

Fred Chen

B.S. in Electrical Engineering, University of Illinois, Urbana-Champaign, 1997

M.S. in Electrical Engineering, University of California, Berkeley, 2000

Submitted to the Department of Electrical Engineering and Computer Science

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2011

Author .....
Department of Electrical Engineering and Computer Science
August 31, 2011

Certified by .....
Vladimir M. Stojanović
Emanuel E. Landsman Associate Professor of Electrical Engineering
Thesis Supervisor

Certified by .....
Anantha P. Chandrakasan
Joseph F. and Nancy P. Keithley Professor of Electrical Engineering
Thesis Supervisor

Accepted by .....
Leslie A. Kolodziejski
Chair, Department Committee on Graduate Students

Energy-efficient Wireless Sensors: Fewer Bits, Moore MEMS

by

Fred Chen

Submitted to the Department of Electrical Engineering and Computer Science
on September 1, 2011, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Adoption of wireless sensor network (WSN) technology could enable improved efficiency across a variety of industries that include building management, agriculture, transportation, and health care. Most of the technical challenges of WSNs can be linked to the stringent energy constraints of each sensor node, where wireless communication and leakage energy are the dominant components of active and idle energy costs. To address these two limitations, this thesis adopts compressed sensing (CS) theory as a generic source coding framework to minimize the transmitted data and proposes the use of micro-electro-mechanical (MEM) relay technology to eliminate the idle leakage.

To assess the practicality of adopting CS as a source coding framework we examine the impact of finite resources, input noise, and wireless channel impairments on the compression and reconstruction performance of CS. We show that CS, despite being a lossy compression algorithm, can realize compression factors greater than 10X with no loss in fidelity for sparse signals quantized to medium resolutions. We also model the hardware costs for implementing the CS encoder and results from a test chip designed in a 90 nm CMOS process that consumes only 1.9 μW for operating frequencies below 20 kHz, verifies the models. The encoder is designed to enable continuous, *on-the-fly* compression that is demonstrated on electroencephalography (EEG) and electrocardiogram (EKG) signals to show the applicability of CS.

To address sub-threshold leakage, which limits the energy performance in CMOS-based sensor nodes, we develop design methodologies towards leveraging the zero leakage characteristics of MEM relays while overcoming their slower switching speeds. Projections on scaled relay circuits show the potential for greater than 10X improvements in energy efficiency over CMOS at up to 10-100 Mops for a variety of circuit sub-systems. Experimental results demonstrating functionality for several circuit building blocks validate the viability of the technology, while feedback from these results is used to refine the device design. Incorporating all of the design elements, we present simulation results for our most recent test chip design which implements relay-based versions of the CS encoder circuits in a 0.25 μm lithographic process showing 5X improvement over our 90 nm CMOS design.

Thesis Supervisor: Vladimir M. Stojanović
Title: Emanuel E. Landsman Associate Professor of Electrical Engineering

Thesis Supervisor: Anantha P. Chandrakasan
Title: Joseph F. and Nancy P. Keithley Professor of Electrical Engineering

Acknowledgments

Throughout my life I have been lucky, and my experience while at MIT has been no different. I've had the great fortune of having the guidance of both Professor Vladimir Stojanović and Professor Anantha Chandrakasan. I'm not sure if I could have had a better pair of mentors and am grateful for their support and patience during my six years here. I would like to thank Anantha for teaching me how to keep the big picture in mind, and how to stay healthy (I'm assuming that's why you recruited me to your GetFit team.) Your enthusiasm and devotion to teaching have been inspiring. I would like to thank Vladimir for giving me the freedom to explore what research areas interested me and entrusting me with his future that I would eventually find my way (like when our group was lost in the forest... at night). I have been lucky to have worked with you for over a decade now (wow!) and your energy and passion for research is a large part of what drew me back to school.

I would like to thank Professor Vivek Goyal for serving on my thesis committee. He was instrumental and proactive in defining many aspects of this thesis and I appreciate his patience with my general lack of theoretical rigor. I would like to thank Fabian Lim and Omid Abari for all of the hours spent debugging our terminology on the compressed sensing simulations.

I would also like to thank all of our collaborators on the MEM relay project whom without, much of this thesis would not have been possible: Prof. Elad Alon, Prof. Tsu-Jae King Liu, Prof. Dejan Marković, Hei (Anderson) Kam, Matt Spencer, Chengcheng Wang, Rhesa Nathanael, Jaeseok Jeon, Kevin Dwan, Vincent Pott, Abhinav Gupta, and "my boy", Hossein Fariborzi. This has been as rewarding and synergistic a collaborative project as I could have imagined. It has been a real pleasure working with everyone.

I would like to thank Prof. Hong Ma (and Prof. Alex Slocum) for developing the biomedical device design class at MIT. That was probably the impetus for taking the research direction that ultimately ended up being my thesis. I am also grateful to Professor Jing Kong and Professor Carl Thompson for the opportunities we had to collaborate during my early

years here. Although that work didn't make it into the thesis, I grew a newfound appreciation for research at the frontier of science. Thanks to the RLE and MTL support staff (Debb), and especially to Margaret Flaherty and Nora Luongo for all of their behind the scenes work.

I am grateful for all of the opportunities that have led me to MIT. I'd like to thank Jared Zerbe and the Rambus team for taking a chance on me and others, such as John Poulton, for planting the seed of curiosity for research (perhaps to their dismay).

Perhaps the best part of my 6 years here has been in the friendships made; the second floor of building 38 has been an entertaining collection of personalities. I'd especially like to thank all of the past and present members of Ananthagroup and ISG for keeping things fun while being a pool of knowledge and support; it has been like having two big families (too big to properly thank everyone!). To Dani, Yogesh, Marcus, John, Jason, Keith and Courtney, thanks for keeping me in shape. Viv, thanks for taking care of my car for so many years even though you didn't have your license. To Mike, Pat, Marcus, George, Viv, Joyce and Dani, I never thought that I would have so many Canadian friends that I'd start saying "A" at the end of sentences. Tony, thanks for taking time off from your PhD to hire me. To Byungsub and Sanquan, thanks for paving the way and distracting Vladimir long enough for me to get my act together. Ranko, you've been the best partner in crime that an incoming student could ask for, and could probably take over the Dos Equis job as the most interesting man in the world.

I'm grateful for the friends and family I have; I have been lucky to grow up with and meet the friends I've met. I'm lucky to have my wife Jes and thank her for all of her love, patience and understanding of what a "tapeout" means. I'd like to thank her mom for the hot dog I requested. And finally, I'd like to thank my Mom for providing every opportunity for me, and always trusting and supporting my decisions no matter what they were. This thesis is for all of your effort.

Contents

List of Figures

List of Tables

Acronyms

ADC	analog-to-digital converter
AIC	analog-to-information converter
ARQ	Automatic Repeat Request
AWGN	additive white Gaussian noise
BCH	Bose-Chaudhuri-Hocquenghen
BER	bit error rate
bpm	beats per minute
BPSK	binary phase-shift keying
CF	compression factor
CS	compressed sensing
CRC	Cyclic-redundancy check
DAC	digital-to-analog converter
dB	decibels
DFT	discrete Fourier transform
DRC	design rule checker
EEG	electroencephalography
EKG	electrocardiogram
FEC	forward error correction
FO4	fanout-of-4

FOM	figure-of-merit
IC	integrated circuit
LE	Logical Effort
LSB	least significant bit
LVS	layout versus schematic
MEM	micro-electro-mechanical
MOSFET	metal-oxide-semiconductor field-effect-transistor
MSB	most significant bit
MSE	mean squared error
MSFF	master-slave flip-flop
MSR	mean squared error ratio
NEF	noise efficiency factor
NF	noise figure
OTA	operational transconductance amplifier
poly-SiGe	Polycrystalline silicon-germanium
PRBS	pseudo-random bit sequence
PRD	percent root-mean-square difference
PRDN	normalized percent root-mean-square difference
S/H	sample-and-hold
SNDR	signal-to-noise and distortion ratio
SNR	signal-to-noise ratio
UWB	ultra-wideband
VTC	voltage transfer characteristic
VLSI	very-large-scale integration
WSN	wireless sensor network

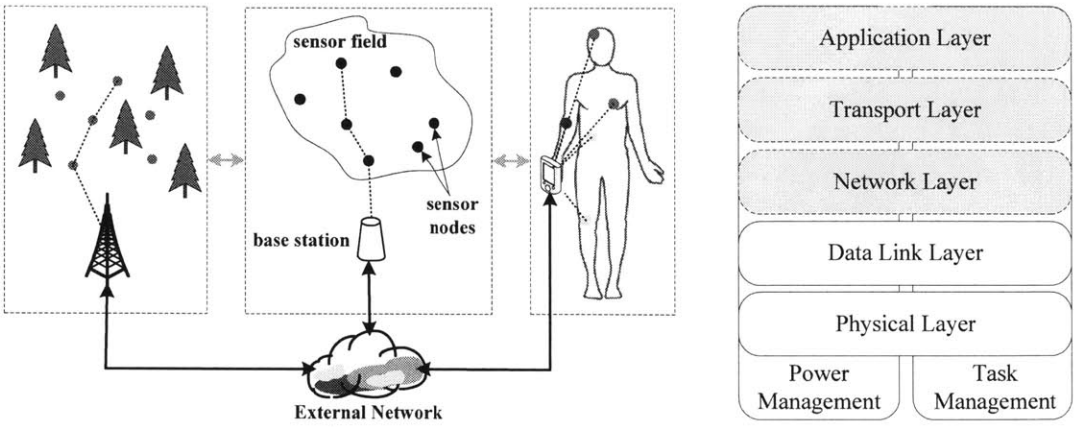
CHAPTER 1

Introduction

In order for any technology to be relevant and practical, it must also be cost effective; this usually implies scalability. The development of the semiconductor industry has revolved around this principle and it has been a major reason behind the success of CMOS. Over the past few decades, CMOS scaling and advancements in integrated circuit (IC) design have progressively reduced the cost of data collection, computation, and communication. The wireless sensor network (WSN) field has been one of the many applications enabled by the miniaturization of CMOS, and now have emerged as a vital "green" technology that could enable efficiency across a variety of industries including building management, agriculture, transportation, and health care [27–29]. However, to become a broad enabling technology like CMOS, WSNs must also become a cost effective one, which requires design optimization across several design layers—from system protocols to device technology.

■ 1.1 Wireless Sensor Networks

Figure 1-1a illustrates the components of a generic WSN architecture. WSNs span a diverse and broad application space that ranges between harsh (military) and benign (building monitoring) environments, large (habitat monitoring) and small (implantable medical devices) spatial coverage, and static (home automation) and dynamic (animal tracking) network configurations [30–36]. The number of nodes in the network could be as small as 1 or in excess



(a) (b)

Figure 1-1: The wireless sensor network model (1-1a) and communication protocol stack (1-1b).

of 1000 and the protocol stack (Figure 1-1b) is as complicated as any other communications system but with far fewer available resources.

Although the applications are diverse, many of the technical challenges facing the field are similar: data reliability, security, dynamic network configuration and node/network lifetime. From the protocol layer down to the circuit level most of these challenges can be linked to the stringent energy constraints of each sensor node [28,29,36–38]. Problems such as reliable data delivery and security are common among many communications systems, but many of the existing strategies for insuring reliable and secure data transfer are not directly applicable to WSNs given the energy limitations at each sensor node [28,36]. In most applications, the energy constraints are determined by either the cost or utility of the sensor node. Higher energy consumption in the node results in more frequent node maintenance/replacement, larger batteries or a shorter useful lifetime–all of which equate to higher cost. For example, even with a sensor lifetime of 10 years, a network with 4000 nodes, such as in a large office building, requires on average a battery changed per day [39]. For patients who require implantable medical devices, limiting the battery size and frequency of replacement reduce costly surgeries and improves the quality of life. With the energy density of modern portable batteries in the range of 1 W-hr/cc (Table 1.1), a 10 year device lifespan requires a sensor

Table 1.1: Energy Density of Batteries

Primary Battery Type	**W·hr/kg**	**W·hr/cm^3**	**Rechargeable Battery Type**	**W·hr/kg**	**W·hr/cm^3**
Alkaline-Mn [40]	143	0.40	NiMH [40]	60	0.17
Lithium-Mn [40]	230	0.55	NiCd [40]	49	0.12
Zinc-Air [40]	411	1.75	Lithium [41]		0.3
Alkaline [41]		0.33	NiMH [41]		0.24
Lithium [41]		0.8	NiCd [41]		0.18
Zinc-Air [41]		1.05			

node to consume on the order of 10 μW of average power per cubic centimeter of battery volume–roughly the size of a lithium ion coin battery. Since the battery typically dominates the physical node size, energy consumption can impose physical limitations on the range of applications even when cost is not an issue.

■ 1.2 Area of Focus: Wireless Sensor Nodes

The technical challenges in WSNs are commonly due to the finite energy resources at the sensor node. In its most primitive form, the function and purpose of a sensor node is simply to sense and communicate information. The energy that is required to perform this function can be viewed as having an active component proportional to the information it must communicate and an overhead or idle component. This thesis attempts to reduce the cost of both of these components through a combination of techniques spanning signal processing, circuit design and device technology.

■ 1.2.1 Energy Consumption in Wireless Sensor Nodes

Figure 1-2 shows the typical functional subsystems within a wireless sensor node. The energy costs for the major components are plotted in Figure 1-3 where results from both commercial and academic sources are shown. As Figure 1-3 shows, in most modern wireless sensor applications a majority of the energy consumption is attributable to communication; the cost to wirelessly transmit data is orders of magnitude greater than for any other function [28,42–

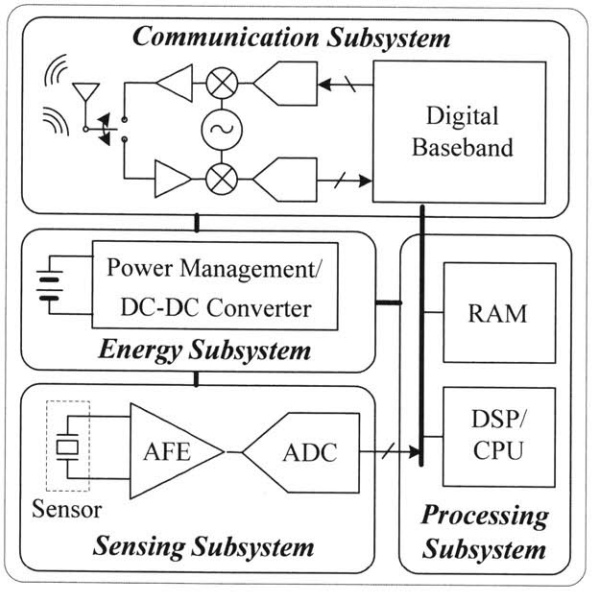
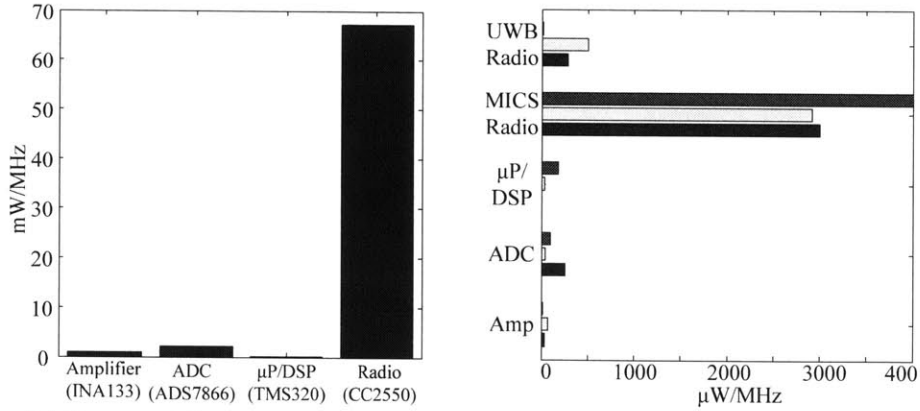


Figure 1-2: Block diagram of common functional blocks in a wireless sensor node



(a) Commercial sensor components

(b) Academic sensor components

Figure 1-3: Example energy-efficiencies of commercial [1–4] and academic sensor components [5–18].

44]. With the exception of ultra-wideband (UWB) radios, which have limited range (< 10 m) and networking challenges, state-of-the-art radio transmitters exhibit energy efficiencies in the nJ/bit range while every other component consumes at most only 10's of pJ/bit. To combat the relative expense of communication, energy saving schemes such as node duty

cycling, data aggregation and adaptive node selection can be employed at higher layers of the protocol stack [45]. However, reducing energy costs in the underlying hardware will result in savings and benefits that propagate throughout the system. The cost disparity between the communication subsystem and the rest of the components indicates that whenever possible, communication should be traded for computation. This implies that the adoption of some computationally efficient data reduction strategy at the sensor node could have a dramatic impact on sensor node energy consumption.

■ 1.2.2 Data Reduction in Wireless Sensor Nodes

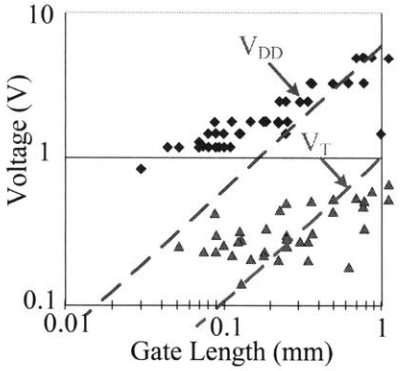
In many WSN applications, the rate of information in the data that each sensor node captures is low, so there is typically an opportunity for data reduction. The existing strategies for implementing data reduction at the sensor node can be classified into data filtering and data compression techniques. Data filtering approaches largely revolve around detecting and extracting signal specific characteristics [46–49]. However, since such data reduction techniques require decision making at the sensor, the filtered data often contains limited information. Approaches such as feature extraction require training, are usually signal specific and typically provide only inferred macro level decisions based on the original signals [5, 50]. Meanwhile, traditional data compression algorithms, both lossless and lossy, can require significant hardware resources (memory and computation cycles) and error protection overhead to insure reliable decoding [43]. For any strategy considered, there is a trade-off between data reduction, robustness, implementation cost, and the granularity of information captured. The goal of any scheme should be to minimize the net cost of data reduction and transmission while reliably preserving the necessary signal information.

■ 1.2.3 Technology Limitations

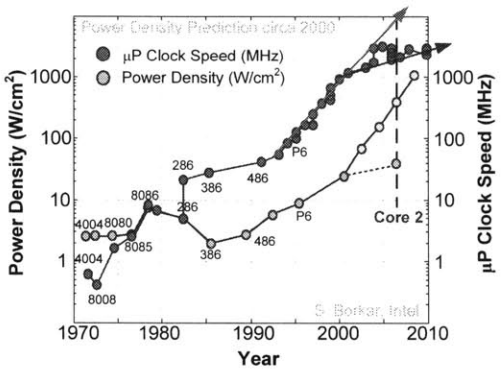
Even when the information-proportional energy can be minimized, there will still be a technology imposed limit on the minimum overhead energy to capture and process the data; leakage power in memories alone can consume the sensor node's power budget [44]. To date, the improvements in performance, cost and energy efficiency of ICs have been driven by

Table 1.2: Constant Field Scaling for MOSFETs

Device or Circuit Parameter	Scaling Factor
Device dimension: t_{ox}, L, W	$1/\kappa$
Supply Voltage (V)	$1/\kappa$
Current (I)	$1/\kappa$
Capacitance (C): $\epsilon A/t$	$1/\kappa$
Delay: VC/I	$1/\kappa$
Power: VI	$1/\kappa^2$
Power Density: VI/A	1
Energy: CV^2	$1/\kappa^3$



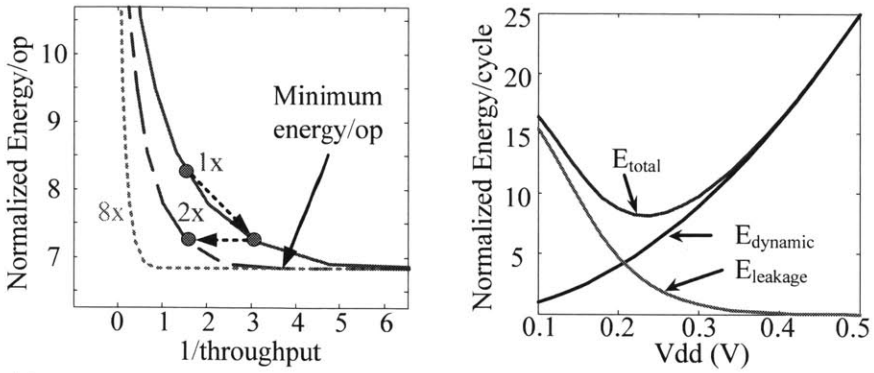
(a) Threshold voltage (V_T) and supply voltage (V_{DD}) scaling [53]



(b) Clock frequencies and power density of Intel microprocessors [54].

Figure 1-4: The slowdown of threshold and supply voltage scaling in CMOS (Figure 1-4a) corresponded with the increase in power density in microprocessors during the mid-to-late 1990's (Fig. 1-5b).

Gordon Moore's early projections of CMOS device scaling [51]. Until recently, the metal-oxide-semiconductor field-effect-transistor (MOSFET) has roughly followed a constant field scaling (Table 1.2) where the power density remained constant across technology nodes while the energy scaled cubicly [52]. However, for technology nodes at 90 nm and below, threshold voltages (V_T) stopped scaling in order to limit the increasing proportion of sub-threshold leakage current while supply voltages (V_{DD}) stopped scaling in order to preserve performance (Figure 1-4a). As a result, energy has scaled slowly with each subsequent technology node while power density has actually increased.



(a) Energy per operation versus throughput for a given functional unit.

(b) Energy per operation versus supply voltage for a given functional unit and technology.

Figure 1-5: Example illustrating the impact of parallelism and future scalability. Parallelism enables lower energy per operation at the same throughput or more throughput for the same power (Figure 1-5a). The limit to which parallelism can improve energy efficiency is determined by the minimum energy point for a given function (Figure 1-5b).

The inability to scale supply voltage coupled with a tightening circuit design space has forced a move to architectural parallelism to increase system performance while maintaining energy efficiency. As Figure 1-5a illustrates, the basic principle behind parallelism is to operate each functional unit at a lower supply voltage/energy and utilize multiple functional units in parallel to recoup performance. This lowers the energy cost for each function while maintaining the overall performance. Alternatively, for the same total power budget, a higher system throughput could be achieved. However, even when the desired function can be fully parallelized, there is a limitation to this approach in CMOS due to sub-threshold leakage. As Figure 1-5b shows, for a given CMOS functional unit, there is a well defined minimum energy point [55] at which an incremental decrease in leakage current is exactly offset by a corresponding increase in delay. For applications targeting low power, such as wireless sensors, this means that there is a lower bound to the energy consumption per operation regardless of how slow the circuits are allowed to run [44]. For applications targeting high-performance, it means that there is an upper bound on throughput for a given power budget, regardless of how much parallelism is employed. For applications that are amenable to *sleep*

modes, power gating can also be used to improve the active versus leakage trade-off; power gating has its own overhead costs (*wake-up* energy, decreased performance) however, and can not eliminate the need to balance leakage energy when active. So as long as sub-threshold leakage is present, parallelism, power gating, and other energy saving techniques will eventually fail to provide a solution to power constrained CMOS designs. To address this limitation, an alternative device technology with better energy trade-offs is required.

1.3 Thesis Contributions

This thesis develops circuit architectures based on analysis that spans algorithm, circuit and device level hierarchies to address the energy consumption and lifetime requirements of wireless sensor nodes. The main contributions in this thesis fall into two general categories: data compression based on compressed sensing (CS) theory and circuit design with micro-electro-mechanical (MEM) relays.

1.3.1 Application of Compressed Sensing for Data Compression

The first half of this thesis presents the design and implementation of a sensor architecture based on the theory of CS that combines the positive qualities of existing data acquisition and compression systems: it provides a flexible and general interface like an analog-to-digital converter (ADC), yet still enables data compression proportional to the signal information content, which is consistent with the performance of source coding. For wireless sensor applications, this combination of characteristics is particularly attractive as it would enable a single hardware interface across many applications while simultaneously addressing the energy cost of wireless transmission by minimizing the data transmitted.

The existing literature in CS has largely focused on developing theoretical bounds and establishing the necessary and sufficient conditions under which the theory is applicable [56–63]. However, there are only a handful of efforts aimed towards practical applications [64–68]. Of these works proposing CS for practical applications, only two [64, 66] even consider the hardware costs and only the latter [66] describes the details of hardware implementation

in an integrated circuit context. The contributions of this thesis fill the gaps in existing literature in regards to determining the practical system performance trade-offs between reconstruction and compression performance and their connections to implementation costs over the broader design space. The specific contributions of this thesis towards this topic are as follows:

1. **Noise and CS Data Compression**

   This thesis first examines the practical applicability of CS as a general source encoder. Trade-offs between signal quality (reconstruction performance) and transmission energy (compression performance) are examined with practical implementation impairments such as signal noise, signal quantization and measurement quantization taken into consideration.

2. **CS Over a Noisy Channel**

   Since many adaptive source coding algorithms are sensitive to channel errors, this thesis investigates the robustness of CS signal recovery to channel noise. In particular, we propose less complex channel coding strategies that are specifically compatible with CS and show their effectiveness. Although most of the ideas and foundation for this exploration had already been conceived, the results presented in this chapter would not have been possible without the help of Fabian Lim and Omid Abari who helped develop the simulation framework, collect the data and analyze the results.

3. **Circuit Modeling Framework and Hardware Implementation of CS Front-ends**

   To understand the implementation costs of CS, a circuit modeling framework is developed that connects hardware costs to system specifications and technology parameters and is applicable to any IC application seeking to adopt CS theory [69]. The circuit models are then used to design and develop a CS encoding IC test chip in a 90 nm CMOS process [25, 70]. The design of the test chip also introduces an efficient way to dynamically generate the encoding/measurement matrix for CS. The measured results verify the CS architecture's ability to do *in-place* compression (no buffering or memory required) and signal recovery for various bio-physical signals, even with an

imperfect ADC front-end. The measured results also correlate well with circuit model predictions and demonstrate the practicality of the CS based architecture.

■ 1.3.2 Circuit Design with MEM Relays

The second half of this thesis focuses on addressing the energy efficiency limitations of CMOS designs by exploring the use of MEM relays/switches as an alternative switching device. From the standpoint of energy efficiency, MEM switches are attractive because they exhibit immeasurably low leakage in the *off*-state so there is no equivalent to sub-threshold leakage in CMOS against which dynamic energy must be balanced. However, to date MEM switches have not garnered much attention for general purpose logic because their switching speeds are many orders of magnitude slower than CMOS transistors. The work in this thesis develops a methodology and infrastructure towards leveraging the leakage characteristics of MEM switches while overcoming their slower switching speeds.

This effort towards developing a MEM switch technology and its supporting infrastructure is a multi-university collaboration of over a dozen students and faculty members spanning circuit and device specialties.[1] The results described here would not have been possible without their help. Specific contributions from individuals will be acknowledged as appropriate in the text. The contributions of this thesis towards this project are described as follows:

1. **Design and Analysis of MEM Switch Circuits**

   This work first revisits the application space for MEM relays to show that MEM relay based circuits could enable 10X improvements in energy efficiency over CMOS across a variety of VLSI systems. We then reexamine the device characteristics of MEM

[1]The collaborators include students and faculty from MIT, UC-Berkeley and UCLA.

- Faculty: Elad Alon (Berkeley), Tsu-Jae King Liu (Berkeley), Dejan Marković (UCLA), Vladimir Stojanović (MIT)
- Students:
  Berkeley: Hei (Anderson) Kam, Rhesa Nathanael, Jaeseok Jeon, Matt Spencer, Abhinav Gupta, Vincent Pott
  UCLA: Chengcheng Wang, Kevin Dwan
  MIT: Fred Chen, Hossein Fariborzi

switches to develop a a set of circuit design principles that are specifically geared towards relay devices [26]. We show that by adopting this approach we can reduce the performance and area gap with respect to CMOS by over an order of magnitude while maintaining the energy benefits. This general design approach lays the groundwork for building a variety of other circuit sub-systems which exhibit similar energy-efficiency benefits.

2. **Circuit Driven Device Design**

   Experimental verification showing functionality of several circuit building blocks is demonstrated [23]. As a result of the experimental results, evolutionary changes to the original MEM switch device are proposed that improve performance and reduce the relative overhead of MEM based circuits. Based on the developed design principles and evolved device design, the CS encoder circuits are designed in a 0.25 μm relay process and their simulated performance are shown to meet or exceed current CMOS metrics despite being in an older technology and at a higher operating voltage.

■ 1.4 Thesis Overview

Chapter 2: CS as a Source Encoder

This chapter first provides some background on CS theory, and then examines the effects of finite quantization and the number of compressed measurements on the reconstruction performance. We show that there is a trade-off between compression performance and the fidelity at which we can recover the signal; for sparse signals where the signal model is known, CS generally performs as well as lossless source encoding algorithms for a reconstruction fidelity of up to 8 bits of quantization.

Chapter 3: CS Over a Wireless Channel

Since many of the existing adaptive source coding algorithms are sensitive to channel errors, this chapter investigate how CS performs in the presence of channel noise. We show that the energy cost to insure a certain quality of recovered signal scales proportionally to sending raw uncoded data. So in this regard, CS performs similarly to uncoded samples only with

far fewer bits sent. The application of any known channel coding scheme is thus expected to have the same performance when applied to CS samples. However, we observe the fact that CS inherently sends redundant information so we also propose some simpler error detection schemes that perform as well as some more complex error correction codes.

Chapter 4: CS Encoder Architectures

In the proposed application for CS, the encoder is at the sensor node so in this chapter, we look specifically at the hardware costs associated with implementing the CS encoder. We develop circuit models for both analog and digital realizations of the CS framework and tie these circuit costs to system requirements such as measurement resolution and compression factor. The result of the analysis falls in line with conventional wisdom in that higher precision systems are more efficiently implemented with digital hardware while analog processing is more suitable for lower resolutions. As we briefly describe, the results are also directly applicable for analyzing analog-to-information converter (AIC) applications as well. For WSNs, the energy performance for the analog and digital systems are set by amplifier noise and sub-threshold leakage respectively.

Chapter 5: Hardware Implementation of a CS Encoder

Based on the models from Chapter 4 a digital CS encoder is designed and fabricated on a 90 nm CMOS test chip. The design of the test chip includes a novel approach to dynamically generating the encoding/measurement matrix needed for CS that is orders of magnitude more efficient than the alternatives. This dramatically reduces the cost of the CS encoder and the results of the test chip show both functionality and good correlation with the hardware models while demonstrating the practical levels of compression in an experimental setting. This chapter also discusses CS in the context of an EKG application and highlights the importance of understanding the end usage with respect to system cost.

Chapter 6: Integrated Circuit Design with MEM Relays

The energy performance of our CS encoder, which was representative of many low-rate digital systems, showed that we were operating in an extremely leakage limited regime. In response, this chapter introduces the idea of using MEM relays as a logic/computing ele-

ment to overcome the leakage limitations of CMOS. We first describe the relay technology characteristics and then discuss the design strategies that can best utilize those characteristics. A comparison of (roughly) equivalently scaled CMOS and relay technologies show over an order of magnitude improvement in energy for operating frequencies up to 10's of MHz.

Chapter 7: MEM Relay Design for Integrated Circuits

This chapter discusses some of the early experimental results from implementing relay based circuits and discusses the key device enhancements that were a product of early circuit testing. The design and simulated results of relay-based CS encoder circuits that incorporate the most recent device and circuit advancements is then presented. Finally some conclusions and challenges are identified for realizing the potential benefits.

Chapter 8: Conclusions

In this chapter, we review the results and the impact that both CS and MEM relays could have for sensor applications. We also offer some final thoughts regarding the general prospects, opportunities and challenges of both CS and MEM relay technologies.

Chapter 2

CS as a Source Encoder

For decades, data acquisition architectures have been based on the principles of Shannon's sampling theorem where sampling at a rate greater than twice the maximum frequency of the signal being sampled (the Nyquist rate) guarantees signal recovery [71]. CS theory challenges conventional sampling paradigms by leveraging known signal structure to acquire sampled data at a rate proportional to the information content rather than the frequency content of a signal [59]. In theory, this would enable far fewer data samples than traditionally required when capturing signals with relatively high bandwidth, but a low information rate. As shown in Table 2.1, many biophysical signals of interest fall into this category where their required sampling rates far exceed the information rate (frequency of event occurrences). The same is typically true of other sensor signals as well. Although these particular examples are in the

Table 2.1: Characteristics of common measured bio-signals [24]

Signal	**Sampling Rate**	**Frequency of Events**	**Event Duration**	**Duty Cycle (%)**
Extracellular APs	30 kHz	10-150 Hz	1-2 ms	2 to 30
EMG	15 kHz	0-10 Hz	0.1-10 s	0 to 100
EKG	250 Hz	0-4 Hz	0.4-0.7 s	0 to 100
EEG,LFP	200 Hz	0-1 Hz	0.5-1 s	0 to 100
O_2, Ph, Temp.	0.1 Hz	0.1 Hz	N/A	Very low

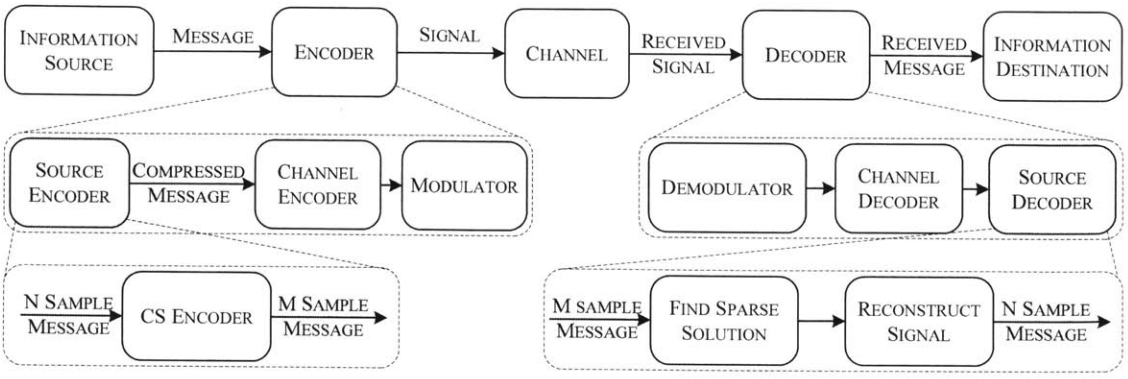


Figure 2-1: In the context of data compression, CS can be viewed as a source coding scheme.

context of medical applications, CS can be generally applied to any field where the signals of interest are sparse.

This chapter introduces the concept of applying CS theory as a general purpose data compression scheme. In the context of wireless sensors, the proposed use of CS is most closely related to the role of a source encoder in typical point-to-point communications systems, as shown in Figure 2-1. The role of the source encoder is to represent the source information as efficiently as possible, which is synonymous with the objectives of data compression. The remainder of this chapter discusses in more detail the performance trade-offs and limitations of adopting CS for this role.

■ 2.1 Compressed Sensing Background

This section first presents an overview of the basic principles of compressed sensing and the relevance of each principle as it applies to data compression in a wireless sensor system. CS is based on the following key concepts which will be discussed hereafter: signal sparsity, signal reconstruction and incoherent sampling.

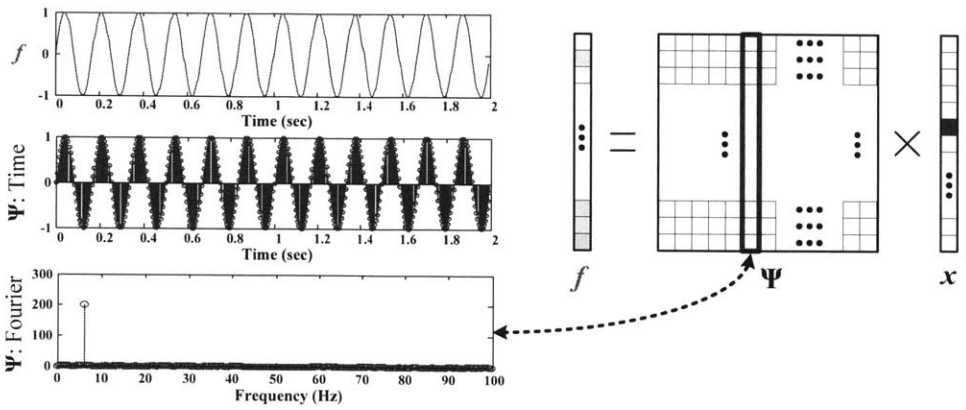


Figure 2-2: A sparsity example: the sine wave is not sparse when represented in the time domain (delta spikes) but is a 1-sparse signal when represented in the frequency domain where $\boldsymbol{\Psi}$ is the inverse DFT matrix.

■ 2.1.1 Signal Sparsity

Compressed sensing theory relies first and foremost on the signal of interest, $\boldsymbol{f}$, having a sparse representation in some basis, $\boldsymbol{\Psi} = [\psi_1 \psi_2 ... \psi_L]$ such that $\boldsymbol{f} = \boldsymbol{\Psi x}$ or:

$$\boldsymbol{f} = \sum_{i=1}^{L} x_i \psi_i. \tag{2.1}$$

where $\boldsymbol{x}$ is the coefficient vector for $\boldsymbol{f}$ under the basis $\boldsymbol{\Psi}$. For $\boldsymbol{f}$ to be sparse in $\boldsymbol{\Psi}$, the coefficients, x_i, must be mostly zero or insignificant such that they can be discarded without any perceptual loss. If $\boldsymbol{f}$ has the most compact representation in $\boldsymbol{\Psi}$, then $\boldsymbol{f}$ should be compressible when it is captured under another basis. So sparseness also implies compressibility and *vice versa*. A familiar example of such a signal, shown in Figure 2-2, is a sine wave which requires many coefficients in time to represent, but requires only one non-zero coefficient in the Fourier domain. Fortunately, many sensor signals have sparse representations. For example, the bio-signals from Table 2.1 are sparse in either the Gabor or wavelet domains [72–74] thus making them suitable for data compression using CS.

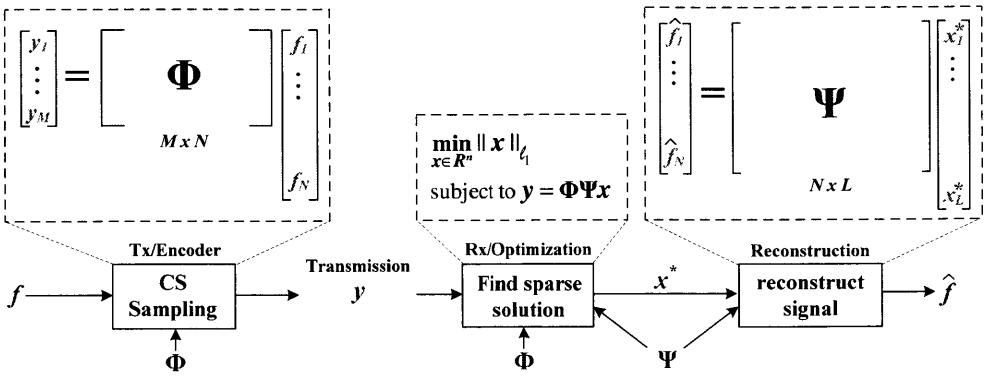


Figure 2-3: CS Sampling Framework

■ 2.1.2 Signal Recovery from Incomplete Measurements

CS theory also proposes that rather than acquiring the entire signal and then compress, it should be possible to construct a sampling or "sensing" framework to capture only the useful information in the sparse signal to begin with. This generalized sampling framework is shown in Figure 2-3 where the N-dimensional input signal, $\boldsymbol{f}$, is encoded into an M-dimensional set of measurements, $\boldsymbol{y}$, through a linear transformation by the $M \times N$ measurement matrix, $\boldsymbol{\Phi}$, where $\boldsymbol{y} = \boldsymbol{\Phi f}$. When $M = N$ then $\boldsymbol{f}$ can be reconstructed exactly from $\boldsymbol{y}$ assuming no singularities exist in $\boldsymbol{\Phi}$. In the case where $M > N$, then the system is said to be overdetermined but a best fit solution can be determined using least squares.

In order to reduce the number of samples, we are interested in when $M < N$; in this case, the system is underdetermined (fewer equations than unknowns) and there is an infinite number of feasible solutions. However, in cases where the signal to be recovered is known to be sparse, the sparsest solution (fewest non-zero x_i) is often the correct solution. The sparse solution can be found by solving the combinatorial optimization problem:

$$\min_{x \in R^L} \| \boldsymbol{x} \|_{\ell_0} \quad \textit{subject to } \boldsymbol{y} = \boldsymbol{\Phi \Psi x} \tag{2.2}$$

where $\boldsymbol{\Psi}$ is the $N \times L$ basis matrix, and $\boldsymbol{x}$ is the coefficient vector from (2.1). This results in minimizing the number of non-zero terms in $\boldsymbol{x}$. However, for most practical problems, this

approach generally requires an intractable combinatorial search so a more practical approach is to use the ℓ_1-norm of $\boldsymbol{x}$ as a proxy objective function for finding the sparse solution [57]

$$\min_{x \in R^L} \| \boldsymbol{x} \|_{\ell_1} \quad subject\ to\ \boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{\Psi}\boldsymbol{x} \tag{2.3}$$

where $\| \boldsymbol{x} \|_{\ell_1} = \sum_{i=1}^{L} |x_i|$. The recovered signal is then $\hat{\boldsymbol{f}} = \boldsymbol{\Psi}\boldsymbol{x}^*$ where $\boldsymbol{x}^*$ is the optimal solution to (2.3). Substituting the ℓ_1-norm as an objective function turns the problem into a convex optimization problem that can be solved efficiently [75]. In this work the signals are largely reconstructed using the lasso algorithm [76] and by solving the equality constrained ℓ_1 minimization in (2.3).[1] It should be noted that the signal can be reconstructed using a variety of different algorithms besides what we chose [80–85]. The main point is that these reconstruction algorithms have practical implementations for estimating the sparse solution. The practical feasibility of solving (2.3) (or similar) implies that an N-dimensional signal can be recovered from a lower number of samples, M, provided that the signal is sparse under some basis. This principle is the foundation for how CS can be used to reduce the data that the sensor must transmit; the ratio N/M is essentially the data compression factor (CF) realized by a CS system and is proportional to the radio power that would be saved.

■ 2.1.3 Incoherent Sampling

In addition to sparseness, CS also relies on incoherence between the sensing matrix ($\boldsymbol{\Phi}$) and the signal model ($\boldsymbol{\Psi}$) to minimize the number of measurements (M) needed to recover the signal. Coherence measures the largest correlation between any row of $\boldsymbol{\Phi}$ and column of $\boldsymbol{\Psi}$ and can be defined by the operator μ as:

$$\mu(\boldsymbol{\Phi}, \boldsymbol{\Psi}) = \max_{1 \le k,j \le N} |\langle \phi_k, \psi_j \rangle| \tag{2.4}$$

where $\mu^2(\boldsymbol{\Phi}, \boldsymbol{\Psi})$ can range between 1 and N [63] when ϕ_k and ψ_j are unit vectors. The less coherence between $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$, the fewer the number of measurements needed to recover

[1] The Matlab toolboxes used for each reconstruction approach came from these sources: [77–79]

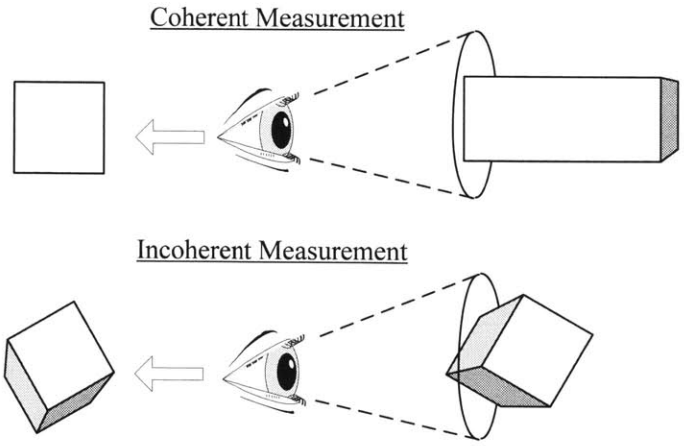


Figure 2-4: An illustration of incoherent sampling in 3-dimensional space.

the signal. Since a sparse signal occupies very few dimensions in the basis space, the intuitive interpretation of wanting incoherence is that each measurement should intersect as many dimensions of the basis space as possible to find which subset of dimensions the signal lies in with fewer tries. A trivial example of this concept is illustrated in Figure 2-4 for a 3-dimensional space. A conservative estimate of the lower bound on the number of measurements, M, needed to recover the overwhelming majority of terms in an S-sparse signal (a signal with S significant non-zero terms) was shown to be:

$$M \geq C \cdot \mu^2(\mathbf{\Phi}, \mathbf{\Psi}) \cdot S \cdot \log N \tag{2.5}$$

where N is the dimensionality of the signal to be recovered and C is some small positive constant (empirically ~2-2.5 [82]). Since S is a measure of the information in the signal, (2.5) indicates that the number of samples required to recover a signal in the CS framework is proportional to the information content of the signal.

From the standpoint of hardware cost and complexity, it is desirable if the signal basis, $\mathbf{\Psi}$, does not need to be known *a priori* in order to determine a viable sensing matrix, $\mathbf{\Phi}$. Fortunately, random matrices with sufficient sample size exhibit low coherence with any fixed basis [61]. As suggested in [61], this means that a random sensing matrix can be employed

as a universal encoder and acquire the sufficient measurements needed to enable signal reconstruction of any sparse signal without knowing *a priori* what the proper basis (Ψ) for the signal is. This principle is leveraged to build a generic infrastructure for data acquisition and compression that is agnostic to the type of signals being acquired, provided that they are sparse. In the context of wireless sensors, a generic measurement infrastructure enables hardware re-use across different applications. Just as important, it also provides flexibility within the same application in situations where the signal information is dynamically changing (e.g. inside the body, mobile sensor). In this regards it is similar to an ADC and can be advantageous when compared to data filtering techniques that require a learning period.

■ 2.2 Analysis Framework

In order to evaluate the compression performance of CS, a set of metrics and a framework under which to compare it to incumbent systems must be established. Here we describe these metrics and provide the context and nomenclature for subsequent experiments and analysis.

■ 2.2.1 System Models

Figure 2-5 shows the context under which the CS framework is applied to the wireless sensor environment. Since CS is intended to be a source coder, this analysis will focus on the quantization and compression components of the sensor node. Figure 2-6 shows the baseline and CS system models that will be used to evaluate the compression performance of CS.

Baseline System Model

As shown in Figure 1-2, the infrastructure of a wireless sensor node typically requires an ADC to sample and quantize the data before processing and data transmission. In the most simplified system, the data would be transmitted with no additional processing. If the quantized data is received error-free, then an optimally designed system would choose the resolution of the quantization to precisely meet the need of the application. Since any practical system will require the quantization to be finite, we treat the error introduced

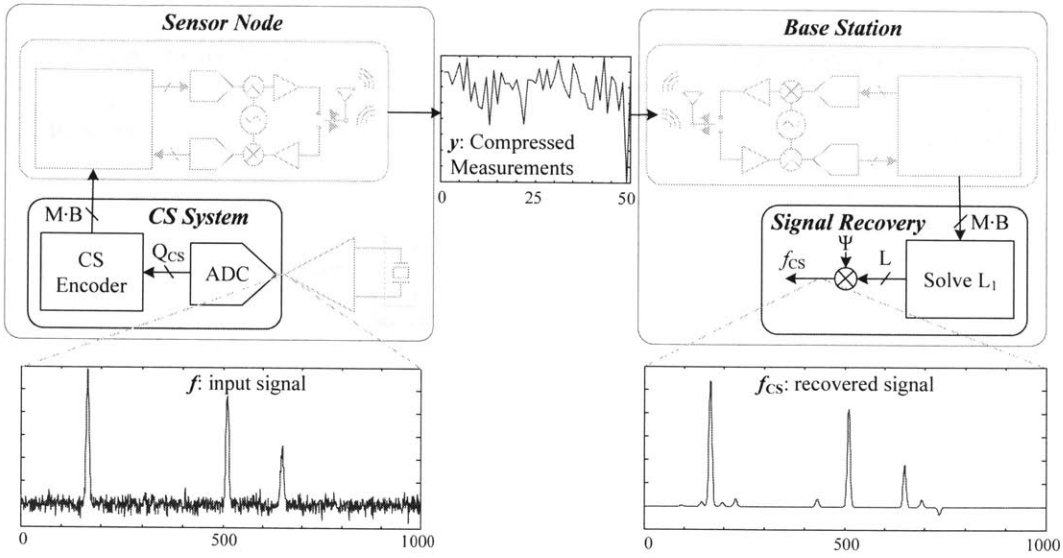


Figure 2-5: CS sampling framework in the context of a wireless sensor system.

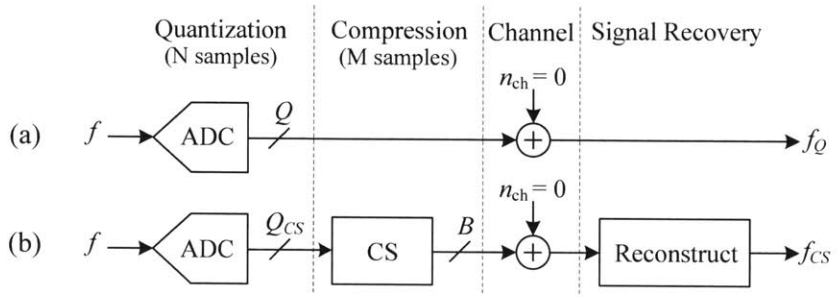


Figure 2-6: System models for an input signal block of N samples: (a) a baseline system with only quantization error, (b) a CS system with quantization noise, and compression error.

by quantization to Q bits as the baseline system performance with which to compare the recovered signal quality of subsequent systems. This baseline system model is illustrated in Figure 2-6a and the recovered signal representing that system is denoted by $\boldsymbol{f_Q}$.

CS System Model

As will be clear later, the CS framework does not necessarily require quantization of the input samples. However, to preserve the signal information, it does require that the input to the CS sampling block maintain the dynamic range of the input signal. So for the CS system model, it is assumed that the input has a dynamic range corresponding to Q_{CS}

bits of resolution, regardless of whether or not the input is actually quantized. Recall from Section 2.1 that the output of the CS sampling block is a set of M measurements. Much like the input samples, the compressed measurements must have some finite resolution and range which we represent with B bits.

■ 2.2.2 Performance Metrics

In the models shown in Figure 2-6, errors in the signal estimate for each respective system are limited to effects from quantization noise and compression error. In subsequent chapters, errors due to signal noise and channel noise will be included as well. The effect of all of these errors on the signal estimate can be captured by the mean squared error (MSE) metric:

$$MSE = \frac{1}{N}\sum_{n=1}^{N} |f[n] - \tilde{f}[n]|^2 \tag{2.6}$$

where $\boldsymbol{f}$ is an arbitrary N-sample input signal and $\tilde{\boldsymbol{f}}$ is the estimate of the input signal. Here we will also introduce another metric, the mean squared error ratio (MSR), which we define as:

$$MSR_{x,Q} = \frac{MSE_x}{MSE_Q} = \frac{\sum_{n=1}^{N} |f[n] - f_x[n]|^2}{\sum_{n=1}^{N} |f[n] - f_Q[n]|^2} \tag{2.7}$$

The MSR simply quantifies the relative error of an estimate, $\boldsymbol{f}_x$, compared to the MSE of the baseline system performance with a resolution of Q bits. In other words, when the MSR is less than 1, the performance is better than quantization of the input to Q bits alone.

Although MSE captures the absolute error, it is not always clear what the relative impact of that error is. So for this purpose, we also adopt the percent root-mean-square difference (PRD) measure in order to capture the relative signal distortion caused by compression. PRD is a commonly used metric in quantifying information loss in biomedical signals [86] and is defined as:

$$PRD = 100 \cdot \sqrt{\frac{\sum_{n=1}^{N} |f[n] - \tilde{f}[n]|^2}{\sum_{n=1}^{N} |f[n]|^2}} \tag{2.8}$$

An additional metric, normalized percent root-mean-square difference (PRDN), can be used

to remove the dependence on the signal mean or DC signal component ($\bar{\boldsymbol{f}}$):

$$PRDN = 100 \cdot \sqrt{\frac{\sum_{n=1}^{N} |f[n] - \tilde{f}[n]|^2}{\sum_{n=1}^{N} |f[n] - \bar{\boldsymbol{f}}|^2}} \tag{2.9}$$

From equations (2.6) and (2.8), it can be seen that for any given signal, MSE can be mapped to PRD (and vice versa). Likewise, PRD can be mapped to other metrics such as signal-to-noise and distortion ratio (SNDR) where $PRD = 100 \cdot \sqrt{1/SNDR}$. In subsequent discussions, different metrics may be used to highlight certain points, but regardless of the choice of metric, there is a one-to-one mapping between metrics so the results of any relative comparisons will be preserved.

■ 2.3 CS Reconstruction and Quantization Error

CS is generally a lossy compression scheme whose reconstruction performance is dependent on a number of different factors: the number of significant features (S) in the signal, the block size of signal samples to compress (N), the resolution of each signal sample (Q_{CS}), the number of measurements (M), and the resolution of each measurement (B). For any combination of a random measurement matrix, $\boldsymbol{\Phi}$, and input signal, $\boldsymbol{f}$, there is some probability that the input signal will be reconstructed with low error. Likewise, there is also some non-zero probability that there will be an ill-conditioned pair of $\boldsymbol{f}$ and $\boldsymbol{\Phi}$ that results in poor reconstruction error.

To illustrate these dependencies, random 4-sparse input signals of length N=1000 are constructed from an over-complete dictionary of Gaussian pulses to pass through the CS system. In these examples, the over-complete dictionary (which we use as $\boldsymbol{\Psi}$) simply consists of N sample-shifted unit-amplitude copies of a single Gaussian pulse. The signals are generated by drawing on a uniform random distribution to assign the sign, magnitude and position for each of the four Gaussian pulses in the signal, where the maximum amplitude of the signal is normalized to 1. For each signal block, the resolution of each compressed measurement (B) is allowed to be as large as needed to accommodate the measurement range,

but at a resolution equivalent to the input signal quantization (Q_{CS}). So for example, if each input sample, $f[n]$, is quantized to 4 fractional bits, then each measurement, $y[n]$, will also have only 4 fractional bits of resolution but will extend the number of integer bits as needed to handle the additional range from accumulating N input samples. The measurement matrix, $\mathbf{\Phi}$, is a random Bernoulli matrix where each entry in the matrix is ± 1. For the following results, each measurement (each row of $\mathbf{\Phi}$) is generated from a randomly seeded 31-bit pseudo-random bit sequence (PRBS) generator. As we will discuss later, we choose a random Bernoulli matrix since it enables hardware simplicity, and as suggested in [87], there is no benefit to adopting a Gaussian measurement matrix (which is commonly used) as opposed to a Bernoulli matrix.

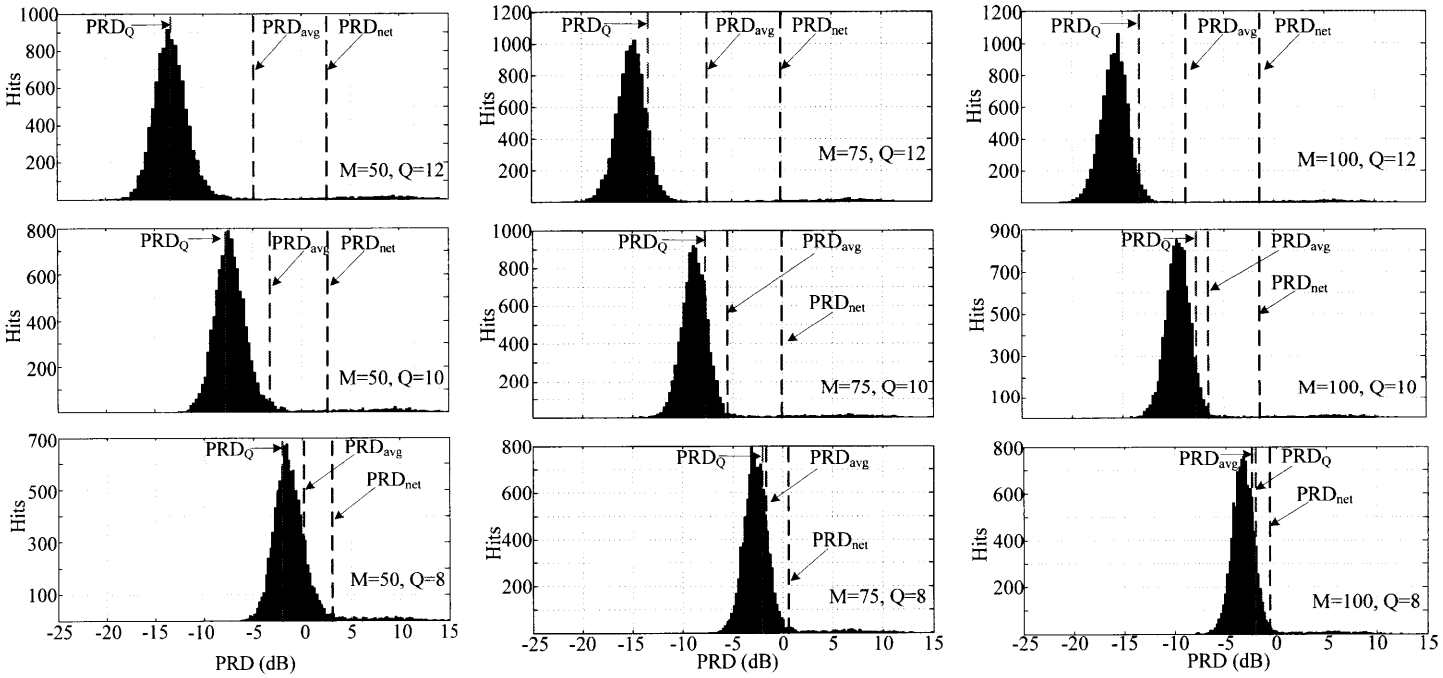


Figure 2-7: Distribution of PRD over 10,000 input signals of length N=1000 where Q_{CS}=8, 10 and 12, and (a) M=50, (b) M=75, and (c) M=100.

Figure 2-7 shows an example distribution of PRD (in dB) resulting from the reconstruction of 10,000 different input signals of length 1000 for system configurations of M=50, 75, and 100, and Q_{CS}=8, 10, and 12. Figure 2-7 also shows the PRD thresholds corresponding to the net PRD (PRD_{net}), average PRD (PRD_{avg}) and the PRD that would have resulted from simply quantizing the input to Q_{CS} bits (PRD_Q). The PRD_{net} is the equivalent PRD if all 10,000 input blocks were considered as a single input signal or in other words:

$$PRD_{net} = 100 \cdot \sqrt{\frac{\sum_{k=1}^{K}\sum_{n=1}^{N}|f_k[n] - \tilde{f}_k[n]|^2}{\sum_{k=1}^{K}\sum_{n=1}^{N}|f_k[n]|^2}} \tag{2.10}$$

where K is the total number of input blocks (10,000 in this case). The distribution highlights the fact that a small subset of signal blocks can result in reconstruction errors many orders of magnitude greater than the majority and that these signals largely dictate the value of PRD_{net}. The distributions also show that for resolutions between 8 and 12 bits, the inherent reconstruction error of this subset of signal blocks is much greater than the error due to quantization and is largely independent of signal resolution. As shown in Figure 2-7, the only way to improve the reconstruction error for these signals is to increase M, which tightens the distribution.

Meanwhile, PRD_{avg} represents what the expected PRD for one block of an S-sparse, N sample signal would be:

$$PRD_{avg} = \frac{1}{K}\sum_{k=1}^{K} 100 \cdot \sqrt{\frac{\sum_{n=1}^{N}|f_k[n] - \tilde{f}_k[n]|^2}{\sum_{n=1}^{N}|f_k[n]|^2}} \tag{2.11}$$

As the distribution plots for different input quantizer resolutions show, the majority of the 4-sparse signals are limited by quantization noise so PRD_{avg} depends somewhat on Q_{CS}. To help distinguish the importance of these two measures, Figure 2-8 plots the PRD of the input blocks on a time axis as if the signal blocks were part of a single, continuous stream of 4-sparse data. PRD_{avg} is equivalent to the time averaged error performance, whereas PRD_{net} is the accumulated error performance. Depending on the application, one metric

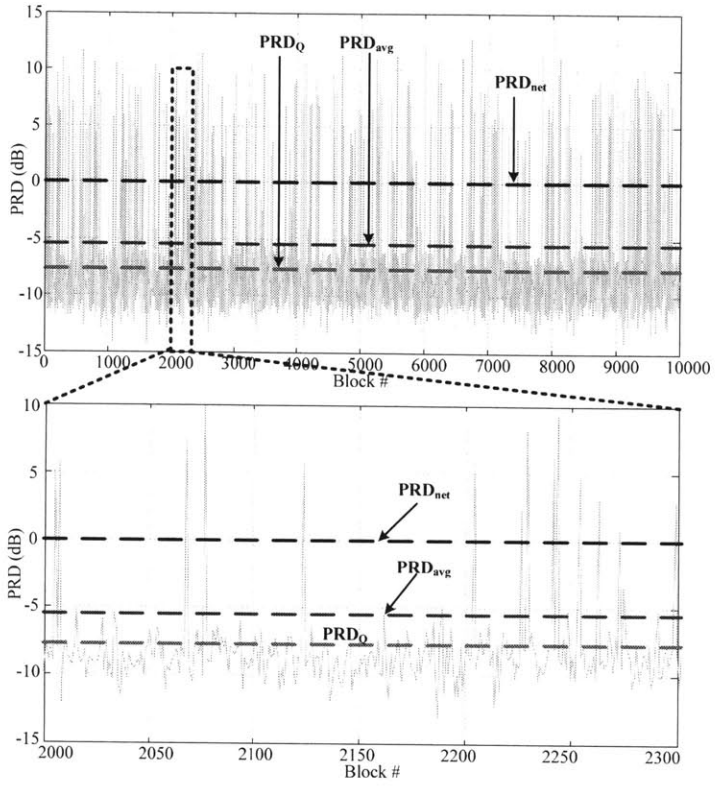


Figure 2-8: Time view of the PRD of a 4-sparse input signal with 10,000 input blocks of length (N) 1000 where M=75 and Q_{CS}=10.

may be more relevant than the other. These results are consistent with the discrepancy shown between worst case and average case analysis of CS reconstruction error [87]. In subsequent discussion, PRD_{avg} signals will refer to signals that fall in the bulk of the distribution while PRD_{net} signals will refer to signals in the tail.

The reconstruction error dependencies are highlighted in Figure 2-9 where signals representing PRD_{avg} and PRD_{net} are plotted over the M and Q_{CS} design space. Figure 2-9 captures the performance limitations imposed on CS due to reconstruction error and the quantization of the input signal. Below a certain number of measurements ($\sim$50) the PRD_{avg} recovered signal error is dominated by reconstruction error. However, for larger M, the recovered signal becomes limited by the quantization noise of the input. The same is true for the PRD_{net} signal only the threshold at which this crossover occurs is at a higher value of

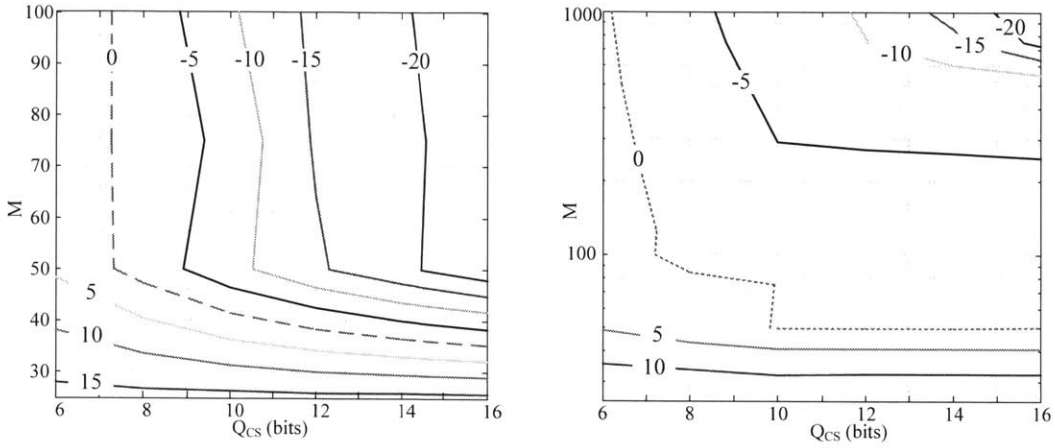


Figure 2-9: PRD contours (in dB) of representative (a) PRD_{avg} and (b) PRD_{net} 4-sparse signals across the input quantization (Q_{CS}) and CS measurement (M) design space.

M. As a point of reference, the original input and reconstructed waveforms corresponding to a PRD of 0.1% (-10 dB) and 10% (10 dB) are plotted in Figure 2-10; they show that while the quantitative difference is significant between signal recovery for a signal in the bulk of the distribution and one in the tail, the perceptual difference is small in either case.

■ 2.4 Compression Performance

Since the goal is to leverage CS theory for source coding, the compression performance of CS is discussed and compared against the Huffman [88], and LZW [89, 90] source coding algorithms. The comparisons will be limited to these lossless compression alternatives since their recovered signal error is equivalent to just the quantization error. The coding efficiency of each option is measured in bits per sample which is just the number of transmitted bits divided by the number of input samples those bits represent. Figure 2-11 plots the resulting compression performance for the PRD_{avg} and PRD_{net} signals versus their level of quantization. For every case except CS, the signals are each quantized to Q bits before compression. For the CS system, M and Q_{CS} are chosen such that the MSE of the reconstructed signal is equal to or less than the MSE from quantization to Q bits (i.e. $MSR_{CS,Q} < 1$). The coding

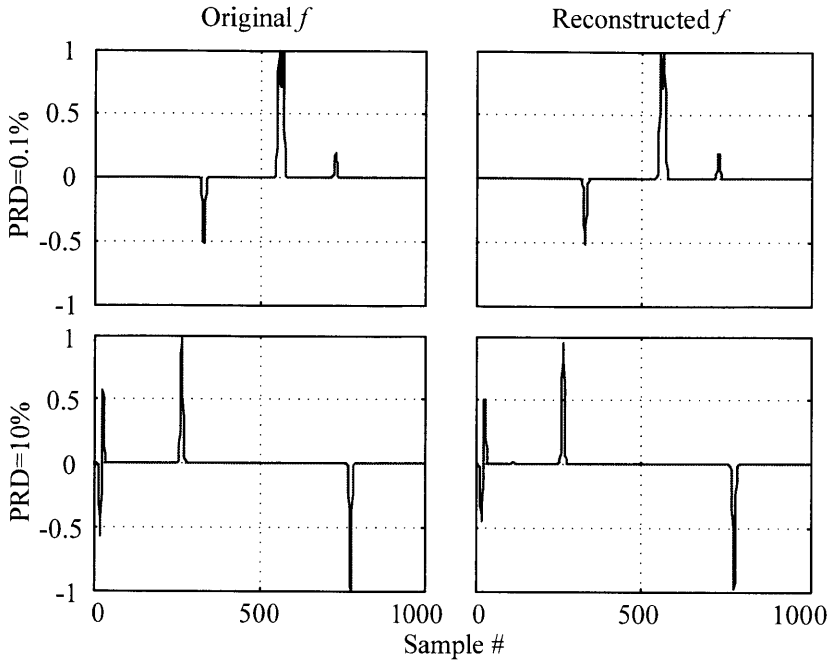


Figure 2-10: Original and reconstructed signals for when the PRD is 0.1% (-10 dB) and when the PRD is 10% (10 dB) for M=75, Q_{CS}=10.

efficiency of the CS system is $M \cdot B/N$ where B is ~4 bits larger than Q_{CS}. Additionally, the theoretical entropy per sample is also calculated and plotted for each signal where the sample entropy is defined as:

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i) \tag{2.12}$$

where p is the probability mass function of X.

For the example signals used in Figure 2-11, the coding efficiency of CS roughly tracks the sample entropy and generally outperforms Huffman and LZW when the reconstruction error is not dominant. This should be somewhat expected as CS is lossy whereas Huffman and LZW are both lossless algorithms. For the PRD_{net} signal in Figure 2-11b, the coding efficiency of CS degrades rapidly beyond 8 bits of quantization when the reconstruction error begins to dominate the quantization error and M must be increased to meet the

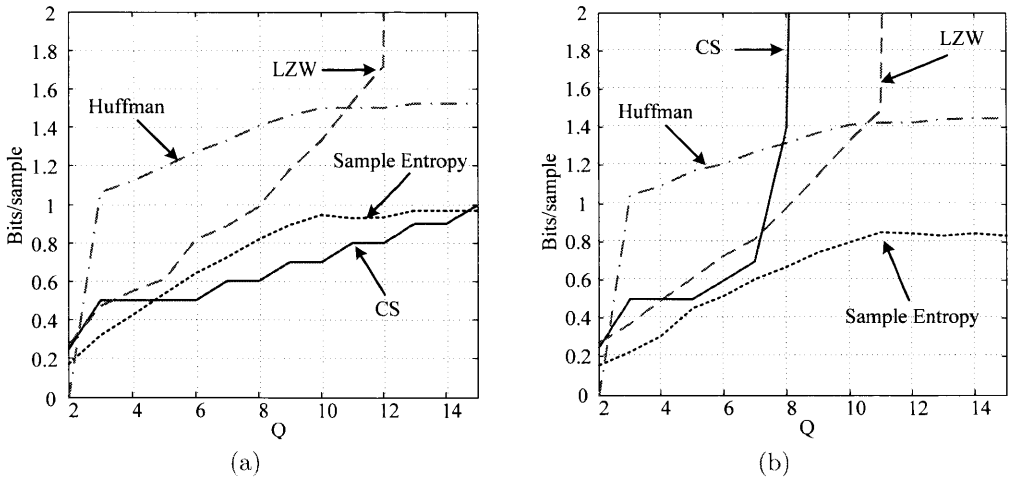


Figure 2-11: Number of bits per sample corresponding to the sample entropy, Huffman coding, and LZW coding for the 4-sparse (a) PRD_{avg} and (b) PRD_{net} signals when quantized to Q bits. For CS, the bits per sample represent $M \cdot B/N$ where M and Q_{CS} are chosen such that the reconstructed MSE matches the output of the baseline system for each target Q value (i.e. $MSR_{CS} < 1$ at each Q) and $B \sim Q_{CS} + 4$ to accommodate the sum.

reconstruction error target. It should be noted, that the results for the Huffman coding algorithm assume that all N samples of the signal have been stored in order to calculate the sample statistics. In both cases, LZW does poorly at higher resolutions since there is less sequence repetition, requiring a larger, less efficient code book. Both the Huffman and LZW algorithms would expect to perform somewhat better with longer input sequences, however such a choice has consequences on the required implementation hardware, which will be discussed later.

The compression performance plotted in Figure 2-11 is for a noiseless, sparse input signal which is why the sample entropy is so low when compared to the Huffman and LZW algorithms. Figure 2-12 plots the same compression performances when Gaussian random noise is added to the input such that the input signal PRD is 1% (0 dB). This corresponds roughly to the quantization noise of an 8-bit uniform quantizer. As the plots show, the input is now sufficiently random such that the performance of LZW and Huffman now both track the sample entropy. The relative performance of CS improves dramatically, even for PRD_{net}

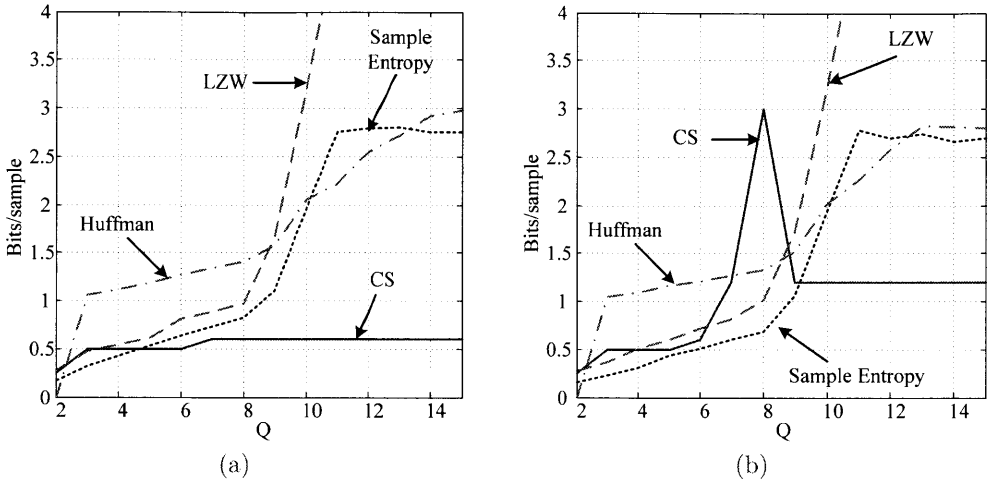


Figure 2-12: Number of bits per sample corresponding to the sample entropy, Huffman coding, and LZW coding for noisy 4-sparse (a) PRD_{avg} and (b) PRD_{net} signals when quantized to Q bits. The added signal noise sets the PRD of the input signal to be 1% (0 dB).

signals when noise is added to the input.[2] The explanation for this is that the baseline quantizer system is unbiased and happily quantizes the noise. Consequently, the error performance target that CS must recover the signal to is now relaxed. This is the appropriate comparison since the decompressed output of the LZW and Huffman compression will also include the signal noise. In most sensor (or real world) applications, it is expected that the captured signals will not be noiseless. If the input noise level is known, and is constant, then we would expect that the quantizer would never be over-designed, so in this example, that means limiting the quantization to 8-9 bits. However, if the noise is either unknown or dynamically changing, then the quantizer would need to be designed to reflect the worst case assumptions. In that case the quantization levels at which we would compare will be higher than 8-9 bits. In Chapter 3 we will discuss the impact of signal noise on the energy required to recover a certain quality signal, but from Figure 2-12 we can see that CS has a relative advantage for such inputs.

[2]The reason for the large inflection point in the CS compression performance is because the baseline error due to quantization to 8-bits actually decreases at that point. This is due to the quantization step being near the input noise level so some of the noise actually gets suppressed. Thus the resulting compression performance is not solely a reflection of how much the CS error increases, but rather a combination of both effects.

■ 2.5 Summary

CS for Wireless Sensors

In this chapter, we introduced the notion of using CS as a general source coding framework. The basic building blocks of CS theory were discussed in this context, and each had a practical implication for WSNs:

Signal Sparsity

In order to leverage CS, the signals that we are trying to recover must be sparse in some signal domain. For wireless sensors, this is often the case; many signals of interest are often time sparse or have already been shown to be sparse in some known basis.

Signal Recovery from Incomplete Measurements

If the signal we are trying to recover is sparse, then we can recover it from far fewer samples than its original signal dimension implies. This is the key idea that allows us to compress the data representation of the signal.

Incoherent Sampling

To minimize the number of measurements we need to recover the signal, we want our mode of measurement to be incoherent (uncorrelated) with the signal basis. Random matrices have been shown to be incoherent with any fixed basis so employing a "random" measurement framework enables hardware generality across a variety of signal types.

Reconstruction Error

An important point that was discussed in this chapter is that CS is a lossy compression scheme whose performance is probabilistic. We showed that there is an inherent reconstruction error associated with CS, and that there is a distinction between the average and aggregate (worst case) reconstruction error. For a given sparsity, there is some small percentage of signals whose inherent reconstruction error is high and essentially independent of signal resolution and only dependent on the number of measurements (M). However, for the bulk of the signals, the inherent reconstruction error is extremely low and thus dependent on the resolution at which the signal is captured. The difference between designing for the

bulk of the signals and the tail of the distribution can be likened to designing for the average and worst case. If we design for the worst case, then it will certainly cost us.

Compression Performance

Lastly, we quantify the ability of CS to compress the data. For this, we compare the compression performance against some well known lossless source coding schemes (Huffman, LZW). To insure that the comparison is on level terms, we require that the reconstruction error in CS is below some target performance level. For the average signal, we show that CS outperforms the other coding schemes across the range of target performances and is even more efficient than the sample entropy of the signal. For the worst case signals, the CS compression performance degrades rapidly (along with LZW) for higher resolutions since many more measurements are needed to reduce the inherent reconstruction error. However, if we consider that the input signal we are trying to recover is corrupted with noise (before we sample), then CS once again outperforms both lossless schemes for both average and worst case input signals.

Now that we have shown that a CS based framework *can* perform well as source encoder, we will next discuss if it is robust to non-idealities such as input and channel noise, and then see if the system is actually realizable.

CHAPTER 3

CS Over a Wireless Channel

In the communication system shown in Figure 2-1, the source encoder/decoder rely on channel coding and error control protocols to insure that the message received by the source decoder is error free. This is common for adaptive/variable length encoding algorithms such as Huffman or LZW which are prone to propagating errors in the message [91]. For comparison, the performance sensitivity of CS to front-end noise and channel errors is examined and presented in this chapter.

■ 3.1 Cost of Transmission Errors

Regardless of whether a compression scheme is employed or not, there is a cost to insuring a certain level of performance in the presence of channel noise. In this section, we first quantify the cost of channel errors for both the uncoded and CS systems when no additional channel coding is applied.

■ 3.1.1 Noisy System Models

The system models from Figure 2-6 have been updated to include input signal noise and channel noise and are shown in Figure 3-1. The system in Figure 3-1b corresponds to the uncoded baseline system in Figure 3-1a only including the impact of both input noise (n_f) and channel noise (n_{ch}). The corresponding recovered signal is denoted by $\boldsymbol{f_{QE}}$; the resolution of

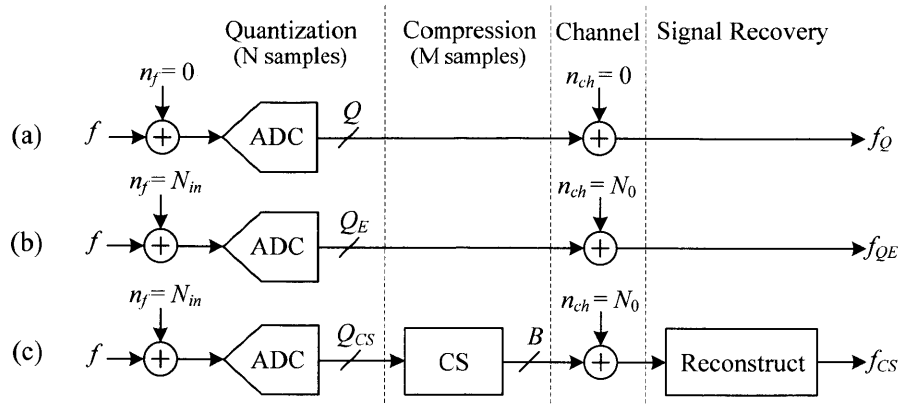


Figure 3-1: System models for an input signal block of N samples: (a) a baseline system with only quantization error, (b) a system with input noise, quantization noise, and channel noise and (c) a CS system with input noise, quantization noise, compression error and channel noise.

this system, Q_E, is different than Q because it is expected that channel errors will degrade the recovered signal and thus the quantization error must be smaller to begin with in order to achieve the same performance as the baseline system. Thus, the two parameters are kept independent for the sake of clarity.

■ 3.1.2 Channel Model

To capture the general effect of channel errors, the wireless channel is simply modeled as an additive white Gaussian noise (AWGN) channel with noise, N_0. We choose this for the simplicity of correlating signal-to-noise ratio (SNR) to bit error rate (BER), but our results should apply to other channel models as well (e.g. Rayleigh fading) where the final energy results will simply be shifted by their respective error distributions. For both uncoded and CS systems, the modulation scheme is assumed to be such that the baud rate matches the bit rate (i.e. there is only one bit per symbol such as with binary phase-shift keying (BPSK)), and that the BER is determined by the SNR where $SNR = E_b/N_0$ and E_b is the energy per bit and N_0 is the channel noise. The total number of bit errors per block will depend on how many bits are transmitted per block of N samples. In the uncoded system this is $N \cdot Q_E$ whereas for the CS system this is $M \cdot B$. Likewise, the total energy cost for each system

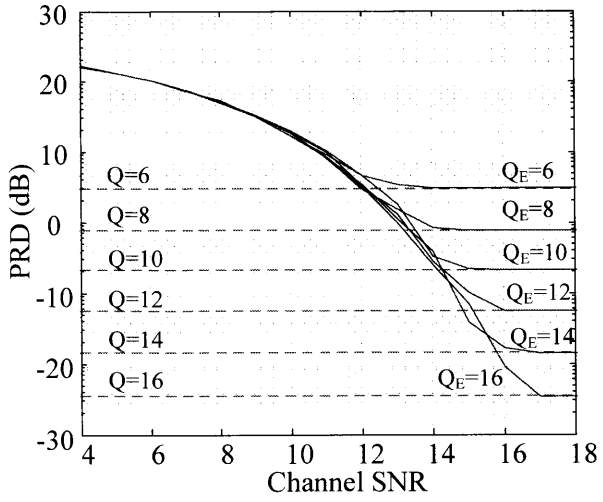


Figure 3-2: PRD versus received SNR for an uncoded ADC based system.

will also depend on the number of bits transmitted per block and the required SNR per bit. To avoid confusion in the future, it should be pointed out that the channel SNR, both in this context and in future discussion/plots, is not rate adjusted so it refers to the energy per transmitted bit (as opposed to energy per transmitted information bit). This convention is chosen in part because for CS, even with no additional channel code, the delineation between information and redundancy is not clear.

■ 3.1.3 PRD vs. Channel Noise

For an uncoded data stream, any channel errors will degrade the MSE and thus the PRD of the received signal. This relationship between channel errors and PRD is plotted in Figure 3-2 where PRD_{QE} is plotted against the channel SNR for varying values of Q_E. The signals used to generate the plots are drawn from the sparse set described in Section 2.3. At high channel SNRs, the PRD_{QE} values converge to their respective quantization noise limits. As Figure 3-2 shows, to preserve the PRD of the system (i.e. $PRD_{QE} = PRD_Q$) in the presence of any appreciable channel noise, the resolution of the system must increase ($Q_E > Q$) to counteract the errors due to channel noise. But regardless of the choice of Q_E, there is a minimum SNR that is required to meet a certain PRD performance.

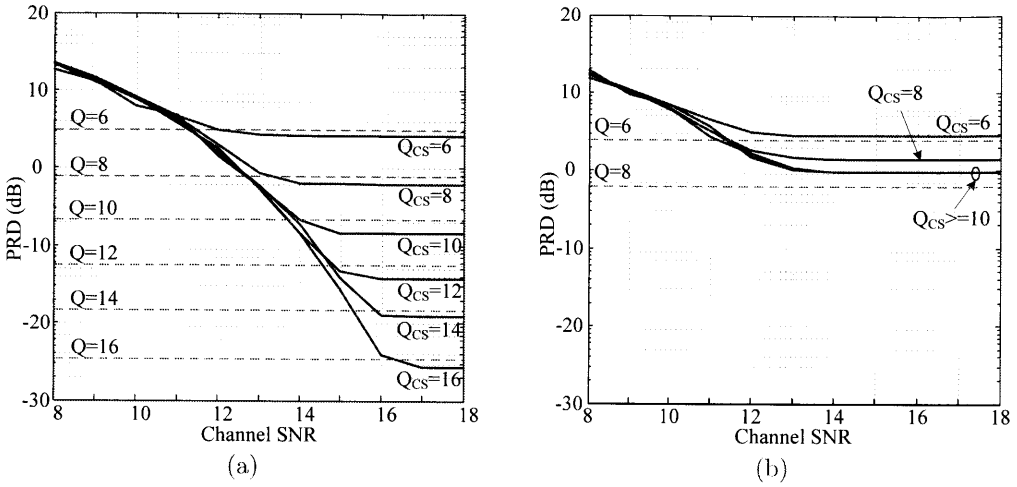


Figure 3-3: PRD versus received signal SNR for a (a) PRD_{avg} and (b) PRD_{net} signal in the CS system using 50 measurements (M).

For CS, a similar exercise can be performed to extract the relationship between the reconstructed signal PRD (PRD_{CS}) and channel SNR. However, for CS there are additional degrees of freedom, such as M, N, and B that affect the reconstruction performance. Figure 3-3 plots a snapshot of this relationship for both PRD_{avg} and PRD_{net} signals for M=50, N=1000 and a B that scales with Q_{CS}. The corresponding distortions from just quantizing the signal (PRD_Q) are also plotted for reference. As Figure 3-3 shows, the signal that is representative of the PRD_{net} reconstruction performance essentially has a higher noise floor due to reconstruction error. As was shown in Figure 2-9b, to improve the noise floor, the number of measurements (and total energy) must be increased.

■ 3.1.4 Energy Cost of Channel Errors

In Figures 3-2 and 3-3, there are several system specifications that can achieve a targeted PRD performance. However, they are not all equally energy efficient. For transmitting the raw quantized samples, the normalized energy cost per sample is simply $Q_E \cdot SNR_{min,E}$, where $SNR_{min,E}$ is the minimum channel SNR that enables the system to meet a target PRD. For each CS configuration, the equivalent normalized energy cost per sample is: $M \cdot B \cdot SNR_{min,CS}/N$.

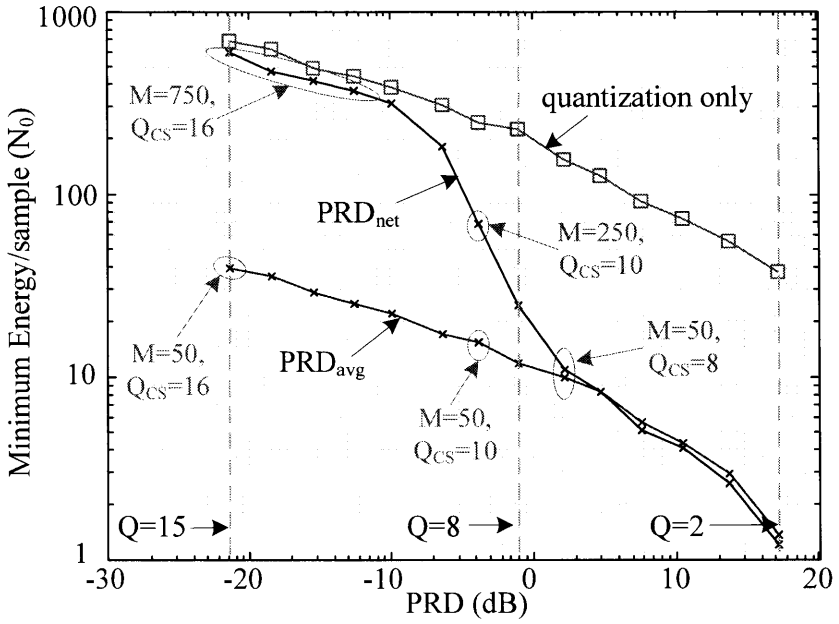


Figure 3-4: Minimum energy per sample in units of channel noise (N_0) for each required PRD performance for both the uncoded system and the CS system for PRD_{avg} and PRD_{net} 4-sparse signals.

Figure 3-4 plots the minimum energy cost curves over a range of target PRD performances for both systems, where the energy is in units of channel noise (N_0). As a point of reference, several of the minimum energy configurations for the CS signals are labeled. Figure 3-4 shows the incremental energy cost of additional resolution in both systems and shows the order of magnitude energy savings that CS can provide through data compression. However, achieving a low PRD (<1%) or high resolution for the worst case signals (PRD_{net}) requires an energy cost on par with the uncompressed quantized samples. This is because more measurements are needed to improve the net reconstruction error and hence more energy is required. In contrast, if occasional performance degradation is acceptable over brief blocks of time, as with the time average PRD (PRD_{avg}), then the energy savings from compression can be realized over the entire range of PRD specifications when compared to transmitting the raw quantized samples. Just as importantly, the channel induced bit errors in the CS measurements do not on average result in catastrophic signal reconstruction as they would

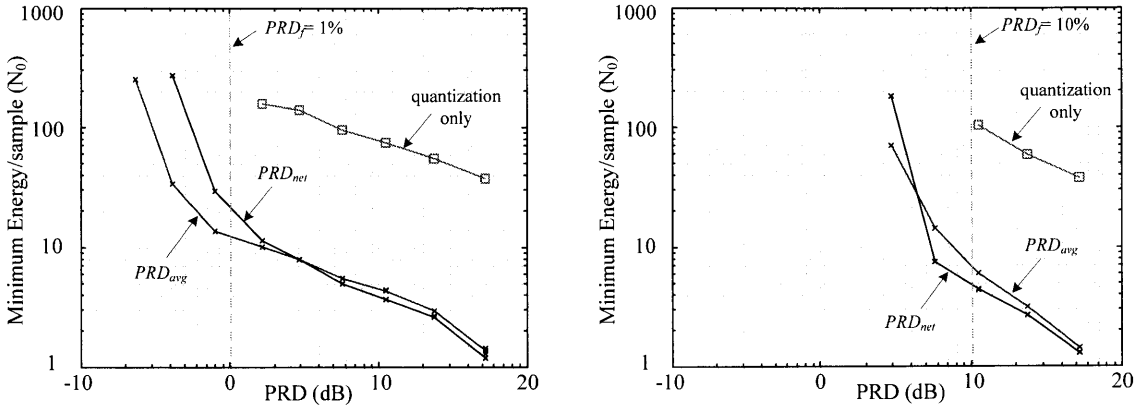


Figure 3-5: Minimum energy per sample vs. required PRD performance for both the traditional system and the CS system for PRD_{avg} and PRD_{net} 4-sparse signals corrupted with noise such that the input signal PRD equals (3-5a) 1% and (3-5b) 10%.

for LZW or Huffman coding.

■ 3.1.5 Effect of Signal Noise

So far, the discussion has been limited to noiseless input signals which suggests that the energy/performance trade-offs described to this point are minimum energy scenarios. Any real sensor input signal will have noise. To account for signal noise, a white Gaussian noise (n_f) is added to the inputs of the systems in Figure 3-1b and Figure 3-1c in order to capture the impact of input noise on the energy/performance trade-offs. The variance of n_f is chosen to correspond to input signal SNDRs of 20 dB and 40 dB which correspond to PRDs of 10% and 1% respectively. The resulting minimum energy curves after adding the two different noise powers are shown in Figure 3-5. As expected, the traditional quantizer system is limited by the PRD of the input. The CS system, however, filters some of the input noise during reconstruction and is thus able to achieve both lower energy and better PRD results. So while the achievable reconstruction performance of CS does not compare favorably with a simple quantizer for noiseless inputs [92], the opposite is true when it comes to more practical noisy sensor inputs.

Another important observation is that the achievable PRD of the CS system is less

limited by the sensor input noise than by the quantization noise (as shown in Figure 3-3). The reason is that the input noise is uncorrelated with the input signal (and thus, the signal basis) so it is largely ignored during reconstruction whereas the quantization noise is highly correlated, as is typical of oversampled inputs [93]. This result is interesting as it has potential hardware implications; for example, an amplifier that drives an ADC could be allowed to have a significantly higher noise floor compared to the quantization noise of the ADC, which is normally avoided in non-CS systems.

■ 3.2 Channel Coding for CS

We showed in the previous section that, unlike traditional source codes, the performance of CS is more robust to channel errors which has enabled the energy analysis and comparisons thus far. Despite the graceful performance degradation in CS, channel coding can still further improve the energy/performance trade-off for the system. Communication protocols generally revolve around two channel coding strategies or hybrids thereof: error correction and error detection. In error correction protocols, some form of forward error correction (FEC) is applied and sent with the data to detect and correct any errors during transmission, while error detection schemes, such as Automatic Repeat Request (ARQ), rely on detecting errors at the receiver and communicating back to the sender what corrupted data to resend. In both cases, there is energy overhead to offering some data reliability. Since channel coders are often designed without bias towards the message bits, we expect the coding performance of any existing error correction or detection schemes to work equally well with CS measurements. However, since single bit errors in the measurement matrix can cause relatively large reconstruction errors, it is not necessarily clear how coding gain translates into energy efficiency. Here we propose modifications to some common existing coding strategies for use with CS and then examine the resulting energy performance after applying the codes.

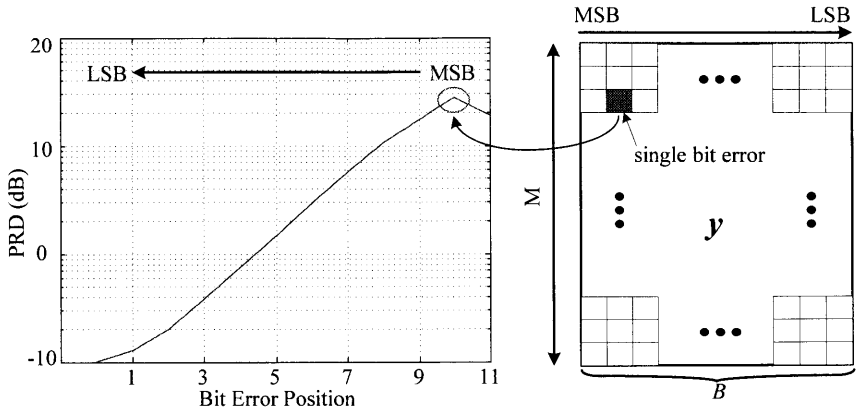


Figure 3-6: PRD_{avg} versus bit position of a single bit error in the transmitted data.

■ 3.2.1 Bit Errors in CS

The performance of channel coding is typically evaluated in terms of BER versus normalized channel SNR where a lower SNR to achieve the same BER indicates coding gain. However, this type of analysis assumes that each transmitted bit carries the same importance. Ideally, a source code would insure this condition such that the signal error is proportional to the BER of the channel. However, in the case of CS measurements, and even the raw quantized samples, this is not the case. The binary representation of each sample and measurement, while efficient, means that some bits carry more information than others. To illustrate the uneven impact of a single bit error, PRD_{avg} versus bit position error for CS is plotted in Figure 3-6, where higher bit positions indicate MSBs (the highest bit is a sign bit). As Figure 3-6 shows, a single bit error in the wrong bit position can result in significant distortion of the reconstructed signal. Given the uneven distribution of information in the CS measurements, the task is to find the most efficient coding strategy.

■ 3.2.2 Error Correcting Block Codes

As seen in Figure 3-6, errors in the least significant bit (LSB)s of the measurement matrix have less impact on the reconstruction error. Thus, if the intention is to maximize the performance gain from error correction, it is most efficient to protect the most significant

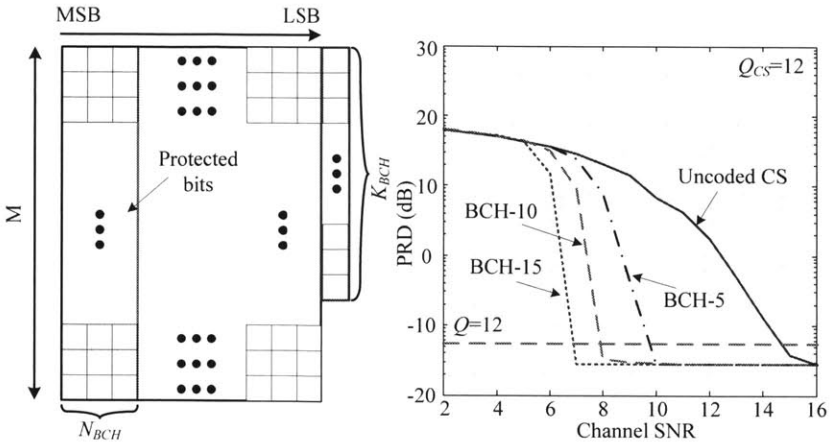


Figure 3-7: Example of the proposed BCH based coding scheme and the resulting PRD (for a PRD_{avg} signal) from applying the error correction coding scheme to the CS measurements when M=75 and Q_{CS}=12.

bit (MSB)s of the matrix first. To assess the effectiveness of this approach, we apply Bose-Chaudhuri-Hocquenghen (BCH) block codes across the M measurements starting with the most significant bits of each measurement.

Figure 3-7 shows the idea behind the coding scheme as well as the reconstructed PRD versus channel SNR for the uncoded and the BCH coded measurement block for different group codes when Q_{CS}=12. In the plots, BCH-T refers to a BCH code with T-bits of error correction capability. The actual BCH code that is chosen is determined by the number of bits in the measurement matrix and the desired error correction capability. The most efficient (lowest overhead) BCH(n, k) code is chosen whose message size (k) will try to cover as many of the measurement bits $(M \cdot B)$ without requiring filler bits. So for the example shown in Figure 3-7, the codes used for BCH-15, BCH-10, BCH-5 are BCH(511,376), BCH(511,421), and BCH(511,466) respectively. Figure 3-8 shows the resulting minimum energy plots both with and without coding. As the results show, there is a noticeable improvement in energy-efficiency over the uncoded data; for increased error correction capability the improvement in PRD performance from coding is significant (Figure 3-7) but the improvement in energy consumption for a given PRD performance diminishes at higher redundancies (e.g. BCH-10→BCH-15). As such, adding additional error correction capability will yield diminishing

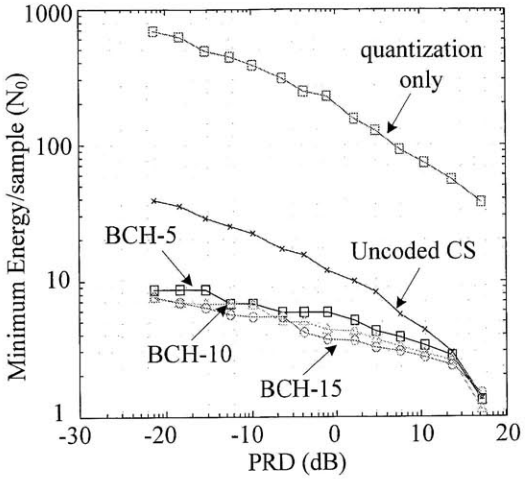


Figure 3-8: Minimum energy per sample versus required PRD performance for uncoded CS measurements and with BCH-5, BCH-10, BCH-15 error correction codes for a PRD_{avg} signal.

returns.

■ 3.2.3 Cyclic-Redundancy Check Coding for CS

Cyclic-redundancy check (CRC) codes are a set of error detection codes that are popular for their ease of implementation and efficiency in detecting errors. In typical error detection schemes the transmission energy overhead for a bit error can be significant since an entire data packet must be retransmitted. However, CS is uniquely suited for error detection schemes as data re-transmission may not be necessary. Recall from Figure 2-9 that there is a soft degradation in reconstruction performance as the number of measurements, M, is decreased. In the cases where the signal sparsity is lower than expected, then it is possible that there is no significant performance penalty for a decrease in M. This is attributable to the redundancy of information captured in the compressed measurements.

To take advantage of this characteristic, the CRC code is used to add parity bits per measurement (for every B bits) as shown in Figure 3-9. At the receiver, when an error is detected, rather than request a retransmission those measurements are simply thrown away and not used in the signal reconstruction. In this scheme, channel errors effectively reduce M, which is less destructive than reconstructing the signal with corrupted measurements

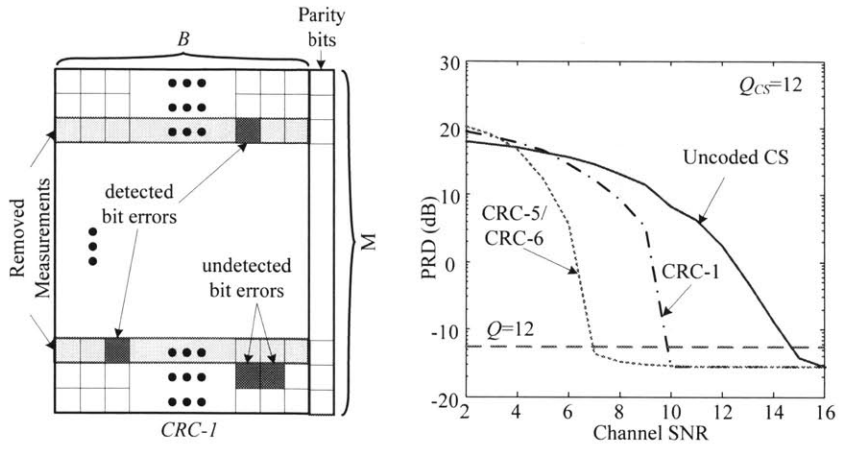


Figure 3-9: Example of the proposed CRC-1 based coding scheme and the resulting PRD for a PRD_{avg} signal after applying the error detection coding scheme to the CS measurements when Q_{CS}=12 and M=75.

and far more energy efficient than retransmitting the packet. The results of this coding scheme for CRC-1 (detect odd bit errors), CRC-5 (detect all bit errors ≤ 2) and CRC-6 (detect all bit errors ≤ 3) codes are shown in Figure 3-9 and Figure 3-10. Figure 3-9 shows the coding gain in regards to PRD for the coded and uncoded cases while Figure 3-10 shows the resulting energy/performance trade-offs for the PRD_{avg} and PRD_{net} signals. As seen in Figure 3-10, there is roughly a 2X energy savings over the uncoded measurements from adding the initial parity bit with the CRC-1 code and roughly another 2X to go to CRC-5. However, like the BCH codes, there is a diminishing return in regards to energy efficiency for higher order error detection. This indicates that any additional error detection applied in this scheme will result in little improvement in the energy/performance trade-off.

■ 3.3 Summary

Channel Noise

In this chapter we first looked at the effect of channel noise on compressed samples. We saw that the effect of channel errors on the recovered signal was similar for CS as for a raw quantized signal, only scaled by the number of bits transmitted. This showed that unlike

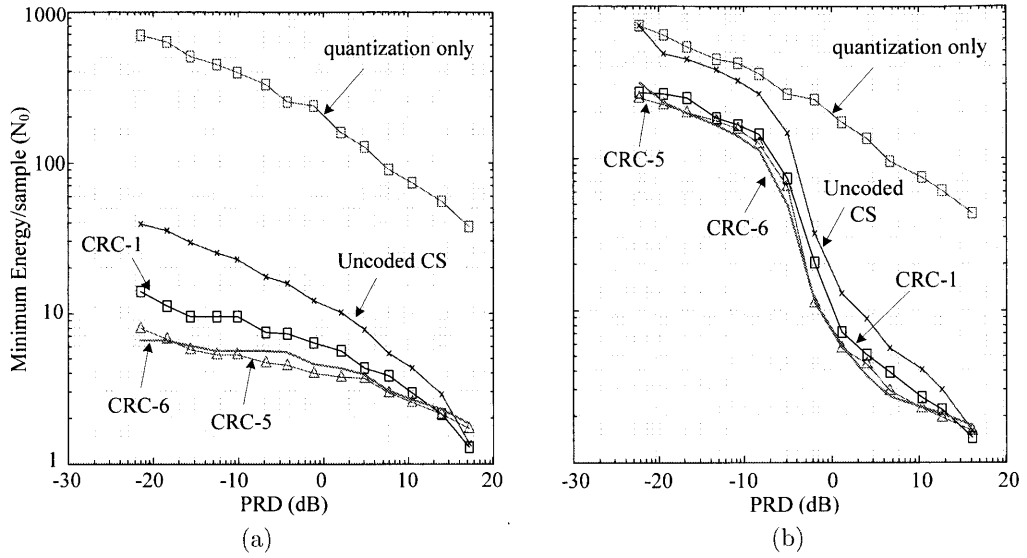


Figure 3-10: Minimum energy per sample versus required PRD performance for uncoded CS measurements and with CRC-1 and CRC-5 error detection codes for both (a) PRD_{avg} and (b) PRD_{net} signals.

some other source coding schemes that there is no propagation or magnification of channel induced errors. Because of this, we could see that the net transmission energy savings that CS could offer, even in the presence of channel noise, was proportional to the data compression factor (>10X).[1]

Channel Coding

Although traditional channel codes would work just as well with CS measurements as with other source data, we also proposed some channel coding schemes that leverage the natural redundancy in CS measurements. The coding results are summarized in Figure 3-11 where the minimum energy results for a PRD_{avg} signal protected with the BCH-5, BCH-15, CRC-1, and CRC-5 codes are shown. As the plot indicates, an interesting finding was that error detection (CRC-5) can be just as effective as more complex error correction (BCH-15) in regards to the energy/performance trade-offs. This would not be the case in a typical system

[1]It should be mentioned that these results for bit errors due to channel noise are different than the experiment described by [81] where noise was added to the measurements before quantization. The experiment described in [81] is nearly equivalent to adding input signal noise.

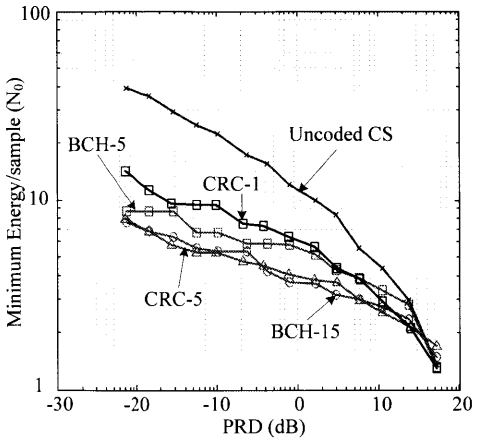


Figure 3-11: Minimum energy per sample versus required PRD performance for uncoded CS measurements and with BCH-5, BCH-15, CRC-1 and CRC-5 error codes for a PRD_{avg} signal.

that would require packet retransmissions on detected errors. Our proposed error detection scheme has potential benefits in multi-hop wireless networks as well where data needs to be retransmitted between nodes. In the event of detected errors, those particular measurements could be dropped and not rebroadcast on the next hop.

Input Signal Noise

In addition to showing robustness to channel noise, we also introduced signal noise at the input. This input signal noise can be representative of environmental noise, sensor measurement noise or even circuit noise. While the performance of a simple quantizing system will be limited by such noise inputs, it was shown that the CS compression and signal reconstruction process filters out some of this random noise. This is an important observation as this result is different than what we saw in Chapter 2 where the average PRD performance was limited by the quantization noise of the input. The primary reason for this is that the signal noise is uncorrelated with the signal whereas the quantization noise, when oversampled, is correlated. Thus, when implementing the system, it may be possible to relax the noise constraints on circuits preceding signal quantization which would further improve the power-efficiency of the front-end hardware.

Having shown that CS is both effective as a data compression scheme and robust under

wireless transmission, we will now examine if the complexity required to implement the system is practically feasible or not.

Chapter 4

CS Encoder Architectures

In the previous chapters, we saw that CS can encode sparse data in a wireless sensor to the signal's bit entropy or below, which in our examples showed over an order of magnitude improvement in communication energy cost. However, in order to improve the energy efficiency of the entire system, the overhead to process and compress the data cannot outweigh the energy savings gained at the transmitter from data reduction [43]. Thus, for CS to be a practical solution for wireless sensor nodes, an efficient hardware implementation of the encoder must be realized. Given the relative immaturity of the field, there have been few works that discuss the trade-offs or costs of realizing CS in hardware [64,66,68,94] and even fewer measured results [94]. In this part of the thesis, we try to bridge this gap and connect system performance to circuit implementation and technology costs for CS. In particular we develop circuit models that capture the energy costs associated with IC implementations of the CS system so that we can predict overall system performance.

The CS parameters described in Chapter 2 can be translated into a set of required hardware. As shown in Figure 4-1, the CS encoder essentially amounts to performing a linear projection from the N-dimensional input, $\boldsymbol{f}$, to an M-dimensional set of measurements, $\boldsymbol{y}$, using the matrix, $\boldsymbol{\Phi}$. In the context of implementation, this amounts to encoding every block of N samples of $\boldsymbol{f}$ into M measurements ($\boldsymbol{y}$). As defined in Chapter 2, Q_{CS} and B are the bits needed to represent the dynamic range of each sample in $\boldsymbol{f}$ and $\boldsymbol{y}$ respectively. Thus,

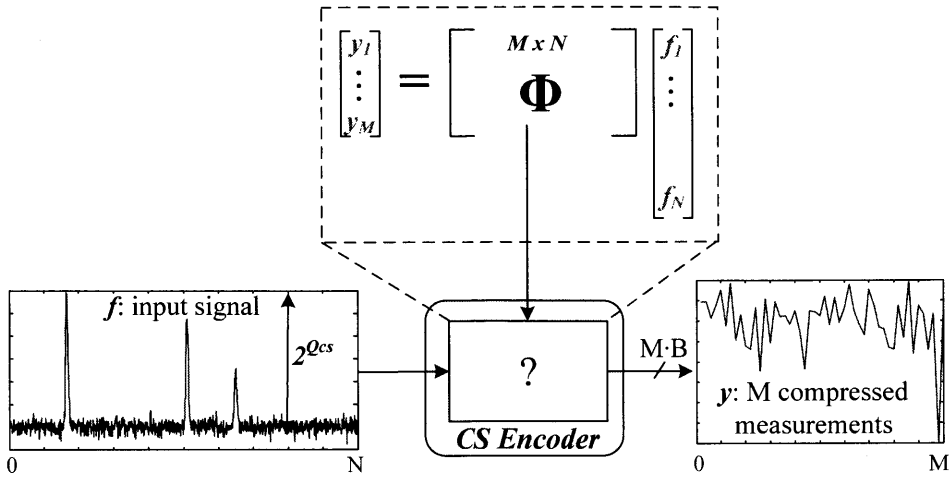


Figure 4-1: CS sampling framework at the sensor node consists of a matrix multiply between $\mathbf{\Phi}$ and $\boldsymbol{f}$.

the effective compression factor is $(N \cdot Q_{CS})/(M \cdot B)$. A common approach to facilitate an efficient hardware implementation of $\mathbf{\Phi}$ is to use a pseudo-random Bernoulli matrix where each entry, $\Phi_{m,n}$, is ± 1 [66,68]. This minimizes the size of $\mathbf{\Phi}$ and subsequent matrix-multiply operations by representing each matrix entry with only a single bit. Any other choice of a full rank $M \times N$ matrix would result in significant additional circuit complexity, data storage, and computation requirements. Since $\mathbf{\Phi}$ is a common input into the CS encoder regardless of how the encoder is implemented, the details of its implementation will be discussed later. However, it bears mentioning that we will show that the matrix can be generated at a fraction of the cost of the encoder such that the following analysis/comparison is representative of the CS encoder cost.

As with traditional signal processing algorithms, the CS encoding can be implemented in either the analog or digital domains. In early proposed applications of CS, the linear projection was applied in the analog domain prior to digitization either because the dominant consumer of power was the sensing mechanism [65] or to reduce the required sampling frequency of the ADC [66, 68]. However, unlike previously proposed applications for CS, wireless sensor applications are rarely limited by ADC performance. Thus, the next two sections will model the dependencies and costs of both systems. In an effort to provide a

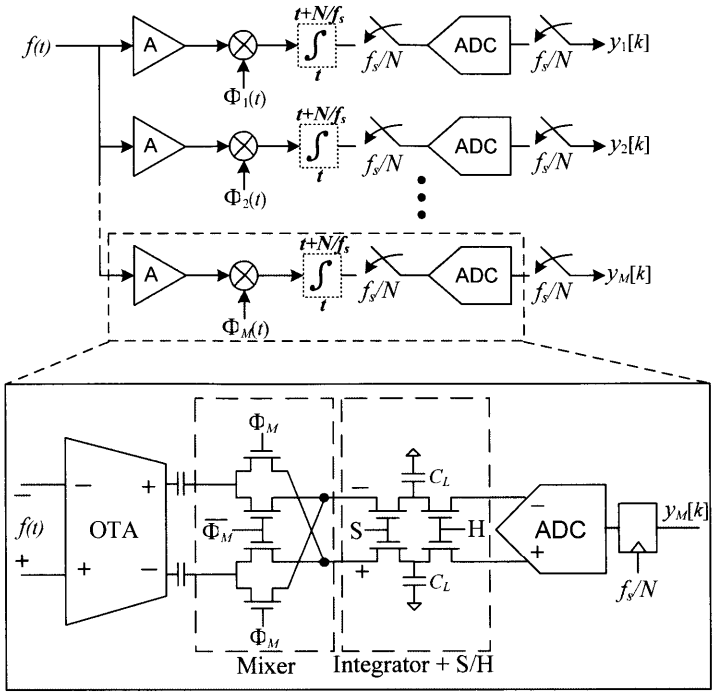


Figure 4-2: Block diagram and example circuitry for an analog implementation of the CS linear transformation. The passive mixer is driven by the matrix coefficients at a rate of fs. During the sample phase (S=1), the sample-and-hold (S/H) circuit also acts as a passive integrator.

level comparison between these implementations, the analysis of circuit costs are described in terms of the common required system parameters: N, M, Q_{CS}, B, and the signal bandwidth (BW_f) in Hertz.

■ 4.1 Analog CS Encoder Power Model

Figure 4-2 shows the block diagram and example circuits for an analog implementation similar to those described in [68] and [66]. In the circuits shown, the input is amplified through an operational transconductance amplifier (OTA) while the multiplication is realized with a double-balanced passive mixer. The sample-and-hold (S/H) circuit following the mixer acts as an integrating (summing) stage as well as the S/H input to the ADC. Although there are many possible alternative circuit realizations, the analysis for the example provided is

representative of how hardware costs in this architecture will scale. To maintain readability, many of the details and basis for these modeling equations have been left out of this section but can be found in the Appendix in Section A.1.

■ 4.1.1 ADC Power

One of the appealing aspects of the architecture shown in Figure 4-2 is that the sampling frequency of each ADC only needs to be f_s/N where $f_s > 2BW_f$ and N is the number of integration samples per compression block. The output of each ADC produces one measurement result, $y_i[k]$, so the resolution of the ADC should be equal to the required measurement resolution, B. The resulting power of the array of M ADCs is then:

$$P_{ADC} = (M/N) \cdot FOM \cdot 2^B \cdot f_s \tag{4.1}$$

where the figure-of-merit (FOM) of the ADC is a design specification. In subsequent analysis, the FOM used to show tradeoffs is 100 fJ/conversion step, which is consistent with the general performance of modern ADCs over a wide range of resolutions and sampling speeds [95]. For sensor applications, it is assumed that the required resolution and bandwidth of the ADC are low enough so that the ADC efficiency is not noise limited such that the ADC power will scale 2X with resolution rather than 4X (i.e. the FOM remains constant) [95].

■ 4.1.2 Integrator and Sample/Hold Power

The simplified Norton and Thevenin equivalent noise circuits in Figure 4-3 show that the constraints on the sampling circuit are partially dictated by the mixer and OTA. When the sampling circuit is tracking the input, the noise bandwidth of the system is set by the sampling capacitor and the series resistance of the CMOS switch (R_{sw}), mixer (R_{mix}) and the OTA output resistance (R_o). For practical purposes, R_o should be dominant to insure that the OTA looks like a current source and so that the combined circuit acts like an integrator where the appropriate noise model more closely resembles the Norton equivalent model. As described in [66], if the S/H is assumed to be a perfect integrator, then the

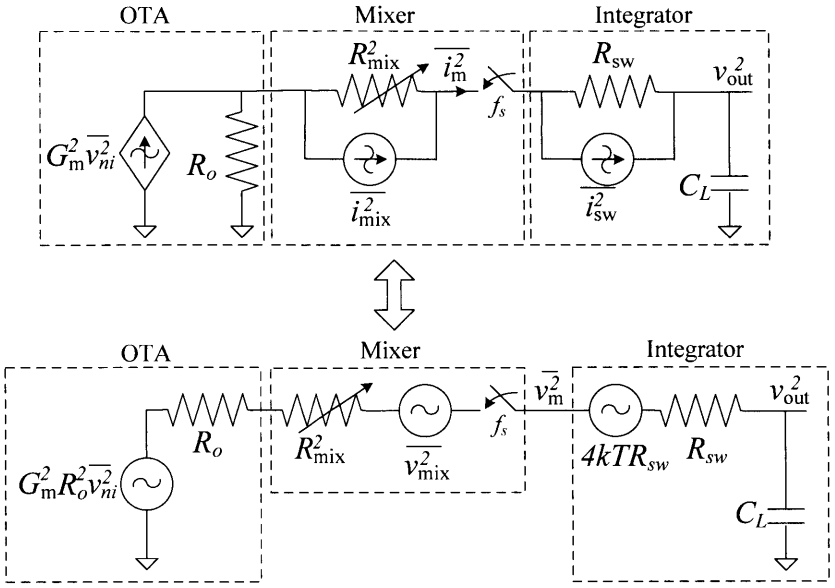


Figure 4-3: Simplified Norton and Thevenin equivalent noise models for the OTA, mixer and integrator.

frequency response of the integrator is a sinc pulse

$$H_i(f) = \frac{N}{f_s C_L} \cdot sinc\left(\frac{\pi f N}{f_s}\right) \tag{4.2}$$

where the integration period is N/f_s. Thus, the gain (G_I) and noise bandwidth (BW_N) of the integrator can be expressed as:

$$G_I^2 BW_N = \int_0^\infty \mid H_i(f) \mid^2 df = \underbrace{\left(\frac{N}{f_s C_L}\right)^2}_{gain} \cdot \underbrace{\left(\frac{f_s}{2N}\right)}_{bandwidth} \tag{4.3}$$

However, to the extent that R_o is not infinite then the equivalent noise model moves closer to the Thevenin equivalent model where the noise bandwidth is given by the low-pass filter response over the integration period:

$$G_I^2 BW_N = \int_0^\infty \mid H_i(f) \mid^2 df = \underbrace{R_o^2}_{gain} \cdot \underbrace{\frac{1}{4R_o C_L} \cdot (1 - e^{-2N/R_o C_L f_s})}_{bandwidth} \tag{4.4}$$

where it is assumed that the equivalent resistance seen by the capacitor is dominated by R_o. The circuit properly approximates an ideal integrator for integration periods where $N/f_s < 0.1R_oC_L$. The bandwidth of the unloaded OTA, assumed to be set by a single dominant pole $1/(2\pi R_oC_p)$, should at least match the required bandwidth of the signal, $BW_f = f_s/2$, so the lower bound on the size of the integrating capacitor to functionally act as an integrator can be described by:

$$C_L > \frac{10 \cdot N}{R_o f_s} = 5\pi \cdot N_{max} \cdot C_p \tag{4.5}$$

where C_p is the capacitance at the dominant pole, and N_{max} is the maximum number of samples to compress. The extent to which the integrator is non-ideal will essentially introduce errors in the applied matrix entries (weights of each input sample) and require a means to back out the actual matrix applied as in [66]. The power due to switching the integrator and S/H circuits is then modeled by:

$$P_i = \frac{M}{N} \cdot V_{DDA}^2 \left(\frac{C_L}{16} + C_G \right) \cdot f_s \tag{4.6}$$

where C_G is the total gate capacitance of the switches. In (4.6) it is assumed that the single-ended voltage swing is between $\frac{1}{4}V_{DDA}$ and $\frac{3}{4}V_{DDA}$, and that the common mode reset voltage is at $\frac{1}{2}V_{DDA}$. Even if C_p is unrealistically assumed to consist of only parasitics and wiring, a reasonably useful value of $N_{max} = 100$ would still require C_L to be on the order of 3 pF in most modern processes. Thus, the power attributed to switching the switches themselves is negligible compared to C_L.

■ 4.1.3 Mixer Power

The passive mixer shown in Figure 4 is described in [96] where it is shown to have a theoretical voltage conversion gain (G_C) ranging between -3.92 dB and -2.1 dB and a measured noise figure (NF) of 3.8 dB. The primary impact of the mixer performance is its impact on the specifications for the OTA. For a $G_C = -3$ dB and a $NF = 3.8$ dB, the current noise density

at the output of the mixer, $\overline{i_m^2}$, is then:

$$\overline{i_m^2} = \overline{i_{OTA}^2} \cdot 10^{(G_C/2+NF)/10} = 1.7 \cdot \overline{i_{OTA}^2} \tag{4.7}$$

where i_{OTA}^2 is the noise current density out of the amplifier (into the mixer). For a PRBS of N samples, the resulting noise accumulated during an integration window is N times the output noise of a single sample, where the output noise density is filtered by the gain and effective noise bandwidth of an integrator with $1/N$th the integration period. The total integrated output noise needs to be less than the quantization noise of the ADC leading to:

$$\overline{v_{out,rms}^2} \simeq 1.7 \cdot \overline{i_{OTA}^2} \cdot G_I^2 BW_N \leq \frac{V_{DDA}^2}{12 \cdot 2^{2B}} \tag{4.8}$$

where the noise term due to the sampling switch of the S/H circuit has been omitted since it will be insignificant for any practical values of R_{sw} and R_o.

In the architecture shown in Figure 4-2, the matrix coefficients, $\Phi_i(t)$, need to be applied at the Nyquist frequency, f_s, of the signal or higher in order to avoid aliasing [66]. The actual mixer power is dominated by the clocking and logic needed to generate the sequence of matrix coefficients. However, as mentioned earlier, since the problem of matrix generation is common to the choice of encoder, its implementation cost will be discussed on its own.

■ 4.1.4 Amplifier Power

The lower bound on power consumption in the amplifier is typically set by the input referred noise $v_{ni,rms}$ requirement. A figure of merit that captures the relationship between $v_{ni,rms}$ and power consumption in the amplifier is the noise efficiency factor (NEF) which was first introduced in [97] and captures the effective number of transistors contributing noise:

$$NEF = v_{ni,rms}\sqrt{\frac{2I_{amp}}{\pi \cdot V_T \cdot 4kT \cdot BW_{amp}}} \tag{4.9}$$

where I_{amp} is the total amplifier current, V_T is the thermal voltage (kT/q) and BW_{amp} is the bandwidth of the amplifier. Measured NEFs in state of the art low-noise amplifiers fall in between 2 and 3 [98–100]. For future analysis, a NEF of 3 will be used and the required power for the array of amplifiers can then be calculated by rewriting (4.9) in terms of the amplifier current:

$$P_{amp} = M \cdot V_{DDA} I_{amp} \geq M \cdot V_{DDA} \cdot \frac{(NEF)^2}{v_{ni,rms}^2} \cdot \frac{\pi \cdot V_T \cdot 4kT \cdot BW_f}{2} \tag{4.10}$$

The output noise constraint in (4.8) can then be rewritten in terms of $v_{ni,rms}$ such that:

$$\underbrace{\left(0.92\sqrt{N} \cdot \frac{G_m}{f_s C_L}\right)^2}_{G_A} \cdot \underbrace{\overline{v_{ni}^2} f_s}_{v_{ni,rms}^2} \leq \frac{V_{DDA}^2}{12 \cdot 2^{2B}} \tag{4.11}$$

where G_A is the total voltage gain from the input of the amplifier to the input of the ADC. The required value of G_A varies by application and the expected dynamic range of the input signal, but a common specification used in previously published bio-sensor applications is 40 dB [46, 98, 100]. This constraint, however, assumes that the total gain is set such that the input range of the ADC is perfectly accommodated. Since we are integrating over N samples, the instantaneous voltage (variance) on the integrator can be expected to rise or fall by a factor of $\sqrt{N}$ and cannot be allowed to exceed the available headroom such that $v_{in} G_A \sqrt{N} \leq 2V_{DDA}$.[1] This constraint reduces the available headroom and thus the required noise floor for a given resolution. Combining this additional constraint with (4.10) and (4.11) results in the minimum amplifier power required:

$$P_{amp} = 2BW_f \cdot 3M \cdot N \cdot 2^{2B} \cdot \frac{G_A^2 NEF^2}{V_{DDA}} \cdot \frac{\pi (kT)^2}{q} \tag{4.12}$$

[1] See A.1.4

■ 4.1.5 Analog CS Encoder Summary

The total power for the analog implementation of the CS encoder, excluding the matrix generation and mixer (multiply) cost, can be summarized as:

$$P_{CS,a} = 2BW_f \left[\underbrace{\frac{M}{N} \cdot FOM \cdot 2^{2B}}_{ADCs} + \underbrace{\frac{M \cdot V_{DDA}^2 \cdot 10\pi \cdot N \cdot C_p}{16}}_{integrators} + \underbrace{3M \cdot N \cdot 2^{2B} \cdot \frac{G_A^2 NEF^2}{V_{DDA}} \cdot \frac{\pi(kT)^2}{q}}_{amplifiers} \right] \tag{4.13}$$

As expected, the costs of all components scale with the number of measurements, M, but they are also dependent on the input signal bandwidth. So even if the number of samples in the CS framework is independent of the signal bandwidth, the cost to implement the circuits is not.

■ 4.2 Digital CS Encoder Power Model

Figure 4-4 shows the block diagram and circuits for an equivalent digital implementation of the CS encoder. The input signal is first amplified and then digitized by a single ADC sampling at the Nyquist rate, f_s. The ADC output is passed to M parallel accumulators that accumulate the incoming sample based on their respective sequence of matrix coefficients, $\Phi_i[n]$. Recall that the coefficient matrix is a Bernoulli random matrix where all elements are ± 1. Thus, the multiplication function can be simply implemented with an XOR gate and the carry-in input of the accumulator. The output of the accumulator is then captured every N samples at which time the accumulator is reset. Like for the analog model, many of the details for these modeling equations have been left out of this section but can be found in the Appendix in Section A.2.

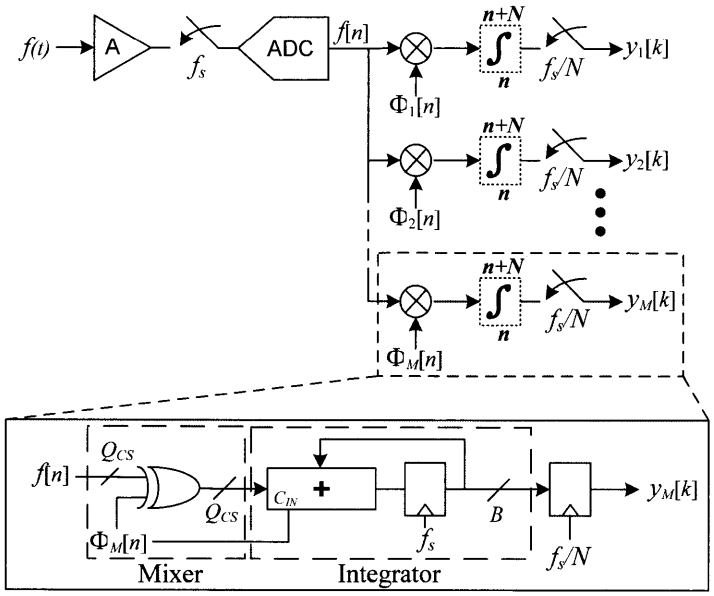


Figure 4-4: Block diagram and circuitry for a digital implementation of the CS encoder.

■ 4.2.1 Accumulator and XOR

Each measurement, y_i[k], requires an accumulator with at least B bits of resolution which results in B flip-flops and XORs, and a B-bit adder. In order to model the delay and energy costs associated with these circuits, a Logical Effort (LE) [101] model is adopted to determine the sizing of each gate and the methodology for sizing the adder is similar to [102]. A slightly simplified version of the alpha-power law delay model used in [103] is used to map the normalized delay of the LE model to real delay. The LE delay of the accumulator is used to scale V_{DD} until the timing constraint is just met, resulting in the following minimum operating V_{DD}:

$$V_{DD,min} \simeq \frac{\alpha_d V_{th}}{1 - 2.5 \cdot K_d \cdot (D_{FF} + D_{ADD,B}) \cdot f_s} \tag{4.14}$$

where K_d and α_d are technology fitting parameters and D_{FF} and $D_{ADD,B}$ are the LE delay of the flip-flop and the critical path of a B-bit adder. The dynamic power consumption can be calculated by accounting for all of the gate and parasitic capacitances at each node along with the switching activity at those nodes. So for M B-bit accumulators and XORs

operating at $V_{DD,min}$ this results in a dynamic switching energy of:

$$P_{accum,dyn} = M \cdot \left[27 \cdot (B + log_2\sqrt{N}) + 2\right] \cdot f_s \cdot V_{DD,min}^2 \cdot C_{inv} \tag{4.15}$$

where C_{inv} is the capacitance of the reference inverter. The $log_2\sqrt{N}$ term is added to avoid overflow in the accumulators during integration and is analogous to the headroom constraint reflected in (4.12) of the analog model. Unlike the analog integrator, the dynamic range of the digital accumulator can be expanded by extending the headroom rather than lowering the noise floor such that it does not impact the noise or resolution requirements of the OTA and ADC.[2] Parasitics such as source/drain capacitance and local wire capacitance are also accounted for in the LE model. It is assumed that all wires are local in the design, however, since we expect to be in a leakage limited operating regime, small errors in the parasitic estimates will not significantly alter the results.

One component of energy consumption that LE does not explicitly model is the sub-threshold leakage current. To account for leakage, an additional normalized parameter is added that captures the relative leakage current in each gate compared to the reference inverter. Similar to how the normalized delay in LE was used to model delay, we use the normalized leakage parameter to arrive at a power consumption expression due to leakage:

$$P_{accum,leak} = M \cdot \left[22.5 \cdot (B + log_2\sqrt{N}) + 4\right] \cdot V_{DD,min} \cdot I_{leak,ref}(V_{DD,min}) \tag{4.16}$$

where $I_{leak,ref}(V_{DD,min})$ is the leakage of the reference inverter at a supply voltage of $V_{DD,min}$.

■ 4.2.2 ADC and Amplifier

The constraints on the ADC and amplifier for the digital CS encoder system are similar to those in the analog system discussed earlier. For the ADC, it is now dependent on signal resolution (Q_{CS}) instead of measurement resolution (B) and samples at the Nyquist rate

[2]This is actually not a real requirement in the digital system so long as the accumulators are allowed to overflow and then underflow. i.e. so long as the final result is within the range of the accumulator, no errors will occur. However, we add this constraint to "insure" the validity of the data.

such that:

$$P_{ADC,D} = FOM \cdot 2^{Q_{CS}} \cdot f_s \tag{4.17}$$

Similarly for the amplifier, the noise constraint on the amplifier is now only determined by the quantization noise of the ADC such that:

$$P_{AMP,D} = G_A^2 (NEF)^2 \cdot \frac{12 \cdot 2^{2Q_{CS}}}{V_{DDA}} \cdot \frac{2 \cdot \pi (kT)^2 \cdot BW_f}{q} \tag{4.18}$$

with the same assumptions regarding G_A and NEF as before.

■ 4.2.3 Digital CS Encoder Power

The total power for the digital implementation of the CS encoder, excluding the matrix generation cost, is summarized below where $B^* = B + log_2\sqrt{N}$.

$$\begin{aligned} P_{CS,D} = 2BW_f \Bigg[& \underbrace{FOM \cdot 2^{Q_{CS}}}_{ADC} + \underbrace{12 \cdot 2^{2Q_{CS}} \cdot \frac{G_A^2 (NEF)^2}{V_{DDA}} \cdot \frac{\pi (kT)^2}{q}}_{amplifier} \Bigg] \\ & + M \cdot V_{DD,min} \Bigg[\underbrace{(22.5B^* + 4) \cdot I_{leak,ref}(V_{DD,min})}_{leakage current} \\ & + \underbrace{(54B^* + 4) \cdot BW_f \cdot V_{DD,min} \cdot C_{inv}}_{switching current} \Bigg] \end{aligned} \tag{4.19}$$

■ 4.3 Analog vs. Digital CS Encoders

To determine which implementation is most suitable for wireless sensor applications, the power models developed in Sections 4.1 and 4.2 are used to map the power costs based on technology parameters extracted from the 90 nm CMOS process intended for the test chip fabrication (90* in Table 4.1). For wireless sensor applications, systems are typified by low sampling frequencies, medium resolutions and small amplitude input signals. Since the purpose of the CS encoder is for data compression, a desirable target is for 10X compression,

Table 4.1: CMOS technology parameters from predictive technology models [20] and for the 90 nm CMOS process used on our test chip (90*) [25]. C_g is calculated from an inverter load, R_{on} reflects the equivalent on resistance of an NMOS device and I_{off} is the leakage of an NMOS device with the drain at V_{NOM}. The unit-less parameters n and γ are the sub-threshold slope factor and fitting parameter to model DIBL.

CMOS process (nm)	V_{NOM} (V)	C_g (fF/μm)	R_{on} (k$\Omega \cdot \mu$m)	I_{off} (A/μm)	n (unit-less)	γ (unit-less)
32LP	1	1.5	1.46	33 p	2.3	0.115
45LP	1	1.5	1.96	21 p	2.1	0.092
32	1	1.5	0.55	358 n	2.3	0.321
45	1	1.6	0.67	105 n	2.1	0.226
65	1	1.9	0.80	47 n	1.9	0.159
90	1	2.1	0.95	26 n	1.7	0.115
130	1.2	2.6	1.05	22 n	1.5	0.082
90*	1	1.4	1.7	800 p	1.7	0.133

which means that we should target $N/M \geq 10$. Minimally, the system should be designed to recover a 1-sparse signal, but a more practical choice is to build in margin. Based on Figure 2-9, reconstructing 4 significant terms per block requires the following range of specifications for the system: $M = 50$, $N > 500$, $BW_f = 0.1 - 10$ kHz, $Q_{CS} = 8$, $B > 8$ and $G_A > 100$ (40 dB).

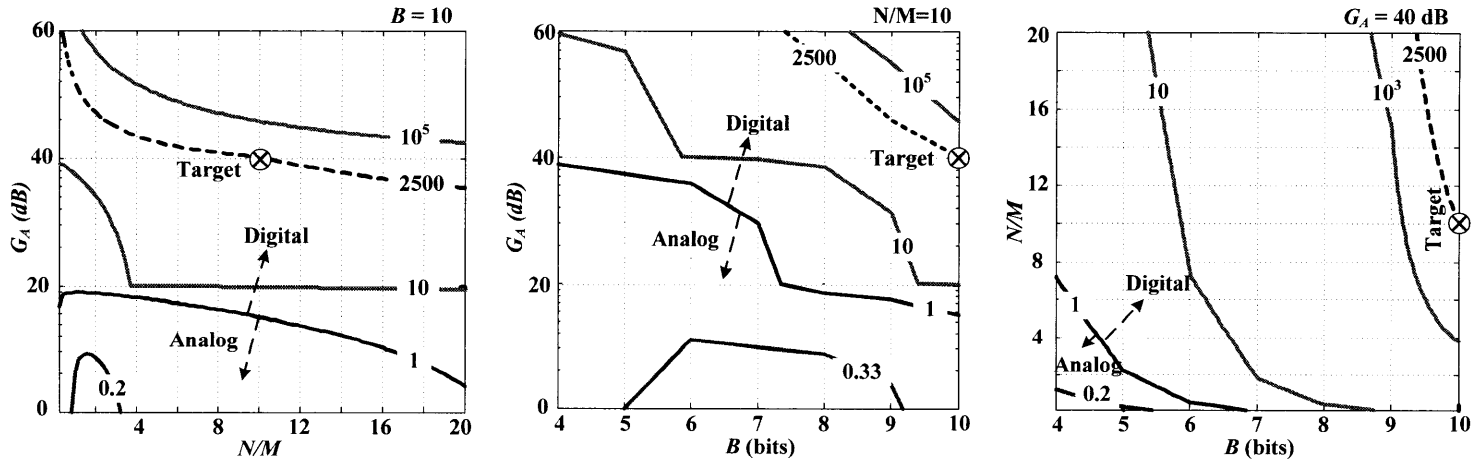


Figure 4-5: Relative power cost of analog vs. digital CS encoder implementations ($P_{CS,a}/P_{CS,D}$) across the specification space for (a) the compression factor (N/M) and amplifier gain (G_A), (b) the measurement resolution (B) and G_A, and (c) B and (N/M). In each plot M is fixed so the compression factor is really a sweep of N. The targeted specification of M = 50, N = 500, G_A = 40 dB, Q_{CS} = 8, B = 10, and BW_f = 200 Hz is shown on each plot along with its corresponding cost contour. All power calculations are based on the 90 nm CMOS process used for fabrication (90* in Table 4.1).

The summary results are captured in Figure 4-5 which plots the relative power ($P_{CS,a}/P_{CS,D}$) of the analog CS encoder versus the digital CS encoder over the range of specifications. To help visualize this multi-dimensional design space, each sub-plot (Figure 4-5a-c) captures the dependencies across only two of the three most sensitive parameters: N, G_A and B. The remaining parameters, when not swept, are kept at a specification of $M = 50$, $N = 500$, $G_A = 40$ dB, $Q_{CS} = 8$, $B = 10$, and $BW_f = 200$ Hz, which is shown on each plot as the target specification. To provide a point of reference, the digital encoder power is predicted to be $\sim$0.4 μW at the target specification for the 90 nm CMOS process we designed in (90* in Table 4.1). The general conclusion that can be drawn from the plots in Figure 4-5 is that the digital implementation is more efficient at higher signal gains (G_A), compression ratios (N/M) or measurement resolutions (B). For the target specification, even potential inaccuracies in the power models cannot account for several orders of magnitude in power difference, so a digital implementation clearly presents the more power efficient option in the low-rate, high dynamic range operating regime.

The common power limitation of the analog implementation stems from the noise and headroom requirements of the amplifier. In each case, higher signal gain, compression (larger N), and resolution translates into a lower input referred noise requirement. The steep power cost for low noise in the OTA is then multiplied by the number of parallel measurements, M. So even though the analog CS architecture relaxes the requirements and power of the ADC(s), the front-end requires M amplifiers that each must still maintain the signal bandwidth and noise floor of the single amplifier in the digital CS architecture. The noise requirement is further exacerbated by the reduced effective headroom due to integration. These two factors are the biggest contributors to the analog CS cost. The amplifiers power dominance is shown in Figure 4-6 where the power breakdown of both the analog and digital CS implementations is plotted across operating frequencies (input signal bandwidth). As expected, the digital implementation is limited by leakage at low sampling rates and the ADC and OTA at higher sampling rates.

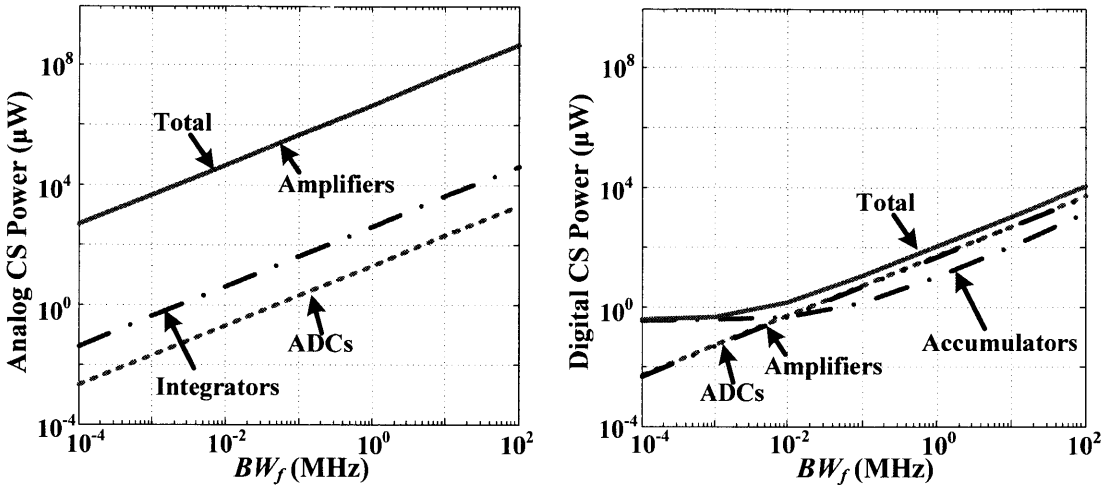


Figure 4-6: Power breakdown of the (a) analog CS implementation and the (b) digital CS implementation over input signal bandwidth for a 90 nm CMOS technology where $M = 50$, $N = 500$, $B = 10$, $Q_{CS} = 8$, and $G_A = 40$ dB.

■ 4.4 Modeling Discussion and Extensions

In this section, we discuss some general implications, limitations and possible extensions of our modeling framework and the CS architecture.

■ 4.4.1 Additional Design Considerations

The models used to derive the power dependencies and relationships for the analog and digital implementations have been idealized as to provide a lower bound on power cost. However, there are other practical implementation issues that should also be considered when choosing the appropriate architecture for a given application.

Circuit Non-idealities

For the analog system, there are some built-in assumptions to the model that will generally produce optimistic power numbers. For one, it is assumed that the circuit components perform ideally such that the integrator and mixer perform perfect accumulation and multiplication like their digital counterparts. In reality this will not be true, so when comparing the digital and analog systems at the same specifications, the resulting system performance

will not be identical. Any non-linearity in the mixer will degrade the effective resolution of the signal while any clock jitter in applying the matrix coefficients, $\phi_i(t)$, will introduce additional noise. A byproduct of these two effects is that the system will need to measure or calibrate the actual matrix coefficients applied in order to reconstruct the signal as was done in [66]. In the case of jitter, this may ultimately limit the performance of the analog CS system akin to the jitter limitations of ADCs. Any linearity error in the amplifier or ADC will distort the measured signal for both the analog and digital implementations. However, for the analog implementation, linearity errors that differ across the banks of ADCs and amplifiers will effectively translate into matrix coefficient errors requiring the same type of calibration as with the mixer non-idealities. Meanwhile, the digital implementation will see only static errors in the signal due to the amplifier and ADC non-linearity.

Area

In regards to area, the digital implementation, which has fewer circuit sub-blocks that scale with the number of measurements, will scale better with M as well as with technology. Similarly, because the integrator is digitally implemented, there is little to no area cost associated with increasing N, whereas the integrating capacitor must increase with N in the analog case. So compared to the analog implementation, whose area is dominated by the integration capacitors and the ADCs, the digital implementation will be more compact.

Process Technology

For the digital implementation, the choice of technology has a large impact on power at low sampling rates due to the dominant effect of leakage. Reducing the leakage will expand the range of resolutions and gains for which a digital CS implementation would be more efficient than the analog CS implementation. As shown in Figure 4-7a, scaled technologies with raised threshold voltages, such as the 32 nm low-power predictive process described in Table 4.1, heavily favor the digital implementation as the digital circuits benefit from feature size scaling while also maintaining low leakage currents. Meanwhile, choosing older technologies with greater voltage headroom will benefit the analog implementation somewhat as it helps relax the noise constraints on the amplifiers, but as Figure 4-7b shows, the voltage headroom

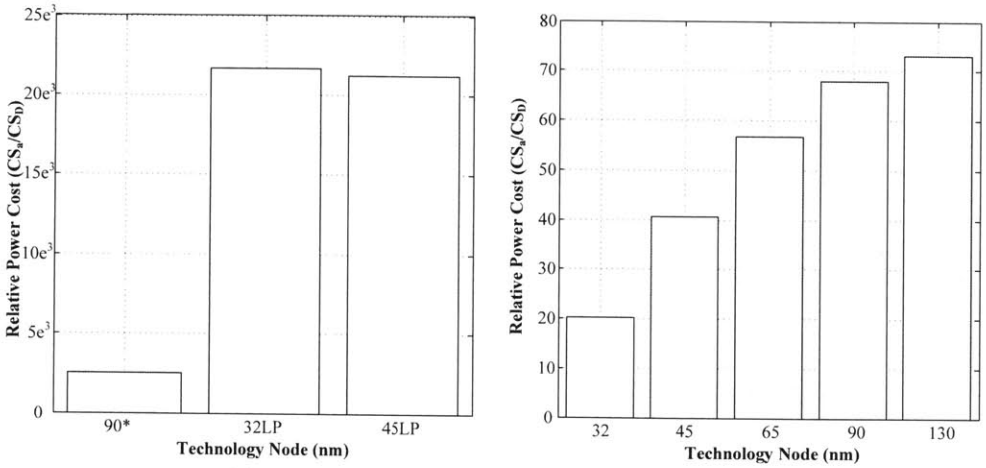


Figure 4-7: Relative analog encoder vs. digital CS encoder power at the target specification of $M = 50$, $N = 500$, $G_A = 40$ dB, $Q_{CS} = 8$, $B = 10$, and $BW_f = 200$ Hz versus (a) predictive model low power, low leakage nodes along with the actual 90 nm CMOS process designed in, and (b) standard predictive model technology nodes.

does not reverse-scale as quickly as the digital leakage decreases.

Application Flexibility

In the comparison between the analog and digital implementations, each design is assumed to be designed for a specific set of system parameters: M, N, G_A, B, Q_{CS}, and BW_f. However, in a practical system, it would be useful to adapt some of these parameters. For example, when the signal basis is still unknown, the N/M ratio could be initially set to 1 to acquire the raw signal input as a way to adaptively build the basis at the receiver, after which the compression factor can be slowly increased. Also, flexibility in the encoder allows for a single hardware interface to be reused across a range of application specifications.

An advantage of the digital implementation is that the value of N can be programmable with no impact on the power consumption or functionality of the encoder. Similarly, the measurement resolution and signal resolution can be altered by truncating bits yet they have no functional impact on the CS encoder. Meanwhile, the functionality and performance of the circuits in the analog implementation are entirely dependent on system parameters such as f_s and the integration period N/f_s. Consequently, the system must be designed for the

worst case system operating conditions and incurs overhead when operating otherwise.

■ 4.4.2 Extensions: Analog-to-Information Converters

The inputs to the power modeling framework presented consist only of technology parameters, circuit performance specifications and system specifications. So to the extent that these inputs are well defined, the model is applicable to any CS application. One clear extension of the model is to analyze the power tradeoffs for AIC applications. AICs, which are identical to the analog CS encoder described in Section 4.1, have been proposed as a way to reduce the required sampling frequencies of ADCs [68]. In applications where sampling at the Nyquist rate is not an option, AICs offer an alternative. However, even in AICs some functions such as mixing or amplification still need to operate at the Nyquist rate, so it stands to question whether AICs, even when the signal to be sampled is sparse, are generally advantageous over traditional ADCs.

The cost of the digital CS encoder is greater than a single ADC at higher frequencies so the AIC comparison would likely yield similar conclusions regarding the application space as those presented in Figure 4-5. Furthermore, any arguments that suggest that lower operating frequencies would enable relaxed circuit design might not be totally accurate. The performance in high-speed ADCs is often limited by sampling jitter [95]; AICs will see a similar limitation in the mixer block since the matrix coefficients must still be applied at the Nyquist rate [66].

■ 4.5 Summary

In this chapter, we examined the hardware and computation costs associated with implementing the CS encoder. To enable the analysis, we developed two circuit level models: one where the encoding was performed with analog components and one where the encoding is done digitally. The initial circuit architectures of each system are based on using a Bernoulli random matrix for the encoding, which we explain should yield the lowest cost system. For each model, the underlying circuit performances are described by either a system

specification (e.g. Q_{CS}), a technology parameter (e.g. leakage), or a commonly published performance metric (e.g. FOM of an ADC). Although we only present the results for the wireless sensor application, the model is generally applicable to CS related IC designs such as AICs.

As we saw in previous chapters, the performance of the encoder varies with both measurement resolution and the number of measurements. Based on these models, we were able to explore the same design space but instead look at the hardware costs. The comparison between the two implementation options yielded a couple of key results. First, aside from any circuit non-idealities, the energy efficiency of the analog encoder is limited by the noise requirements of the input amplifier which is intricately tied to the performance of the downstream circuits. Perhaps the more important limitation is that as N is increased to improve the compression factor, the noise requirement on the amplifiers gets tighter due to an increasing variance at the integrator which effectively demands a larger dynamic range for the same supply voltage. This combination of constraints along with the practical system complexity issues with non-ideal computing components led us to choose the digital encoder design which does not have the same limitations. For the targeted specification, the digital encoder was predicted to consume less than 0.4 μW, which is over three orders of magnitude less than the predicted analog encoder power and certainly insignificant compared to the transmission energy that would be saved. Now that we have built both the system performance and hardware models, we will try to validate both by fabricating a test chip to demonstrate the merits of CS.

Chapter 5

Hardware Implementation of a CS Encoder

In Chapters 2 and 3, the transmission energy benefits of adopting CS as a source encoding scheme were shown. In this chapter, the design and measured results of a fabricated CS encoder test chip based on the analysis from Chapter 4 is presented. In particular, the challenge of generating the matrix and some of the practical aspects of implementing the encoder are discussed. The measured test chip serves to demonstrate the functionality of the system while validating the power models.

■ 5.1 Matrix Generation

The problem of generating the entries in the measurement matrix ($\mathbf{\Phi}$) is a common problem regardless of how the encoder is implemented. In many cases it can be the limiting factor for power and area. As discussed in previous chapters, the measurement matrix should be a random Bernoulli matrix to both satisfy generality of the interface and to simplify the matrix-multiply operation. In this section, the design requirements and trade-offs for several implementation options are discussed. We limit our discussion to deterministic methods of generating "random" matrices because we will need to generate the same matrix at the receiver to recover the signal (and obviously we do not want to transmit the matrix itself).

Thus, true random number generators that are based on physical entropy sources (e.g. thermal noise), such as the one proposed for use in [64] are not practical for CS in the context of wireless sensors.

Memory Lookup Table

Since the matrix needs to approximate a random matrix, one straightforward approach is to use a look-up table or a memory to implement the matrix. However, to get compression factors on the order of 10X or more with an M of just 50 requires an N greater than 500 which equates to at least a 25 kb ($>$ 3 kB) memory. Although this may seem like a small amount of memory, it would dwarf the area of the accumulators, ADC and AFE combined and also dominate the power consumption since it is both large (leaky) and needs to run at the Nyquist rate. Even in a relatively low leakage process (130 nm), roughly every 1 kB of memory burns 1 μW of leakage power [104]. Additionally, the size of the memory would cap the value of $N \cdot M$ and thus limit the maximum achievable compression factor of the encoder.

Independent PRBS Generators

Another approach, which was adopted in [66], is to use an independent PRBS generator for each measurement, $y_i[k]$. While this is much more compact than the memory implementation, it still roughly doubles the size of the accumulator array when the length of the PRBS generator polynomial is close to the resolution of each measurement (B). For example, for an M of 50, generating an independent 2^{15}-PRBS sequence for each measurement would require 750 (15$\times$50) flip-flops. This does not include the PRBS seed registers either, which would double the total number of registers if the seed were to be fully programmable. Although both blocks need to run at the Nyquist rate, the activity factor of the PRBS registers is much higher than in the accumulator banks. Even in [66], where the number of measurements is relatively small, the PRBS generators and associated clocks were still the largest contributor to power consumption.

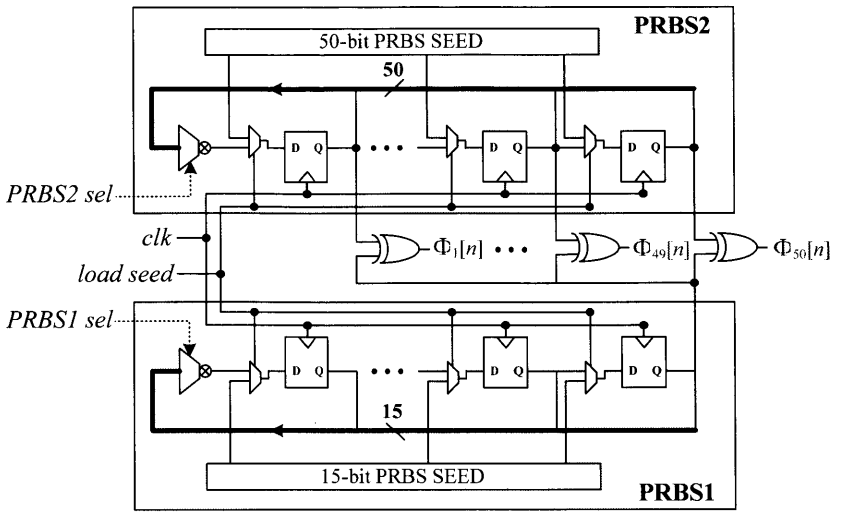


Figure 5-1: Block diagram of the measurement matrix generation block. The PRBS seeds are loaded every Nth sample in conjunction with the resetting of the accumulators.

Mixed PRBS Matrix Generation

Since power consumption is paramount in sensor applications, we propose an alternative realization of the matrix generation that requires only two PRBS generators [25]. As shown in Figure 5-1, the matrix generation circuit consists of the state of one PRBS generator XORd with the output of a second PRBS generator to create the columns of $\mathbf{\Phi}$ on a sample by sample basis. The seed and sequence length of each PRBS is programmable to enable the synthesis of a wide variety of pseudo-random matrices. Not taking into account the additional overhead to seed the PRBS generators and state, the resulting implementation requires only 65 flip-flops for an M of 50 compared to what would have been 750 flip-flops to enable PRBS sequences with the same run length for the approach described in [66]. If seed registers are included, the disparity is 130 to 1500 memory elements. With these improvements, the active matrix generation power is reduced to less than 10% of the digital CS encoder (accumulators) power.

One property of full length 2^N-PRBS sequences is that shifted versions of the same sequence are maximally uncorrelated with one another (i.e. $1/2^N$). Based on this property, it may seem that only a single PRBS generator could be used to generate the entire matrix (i.e.

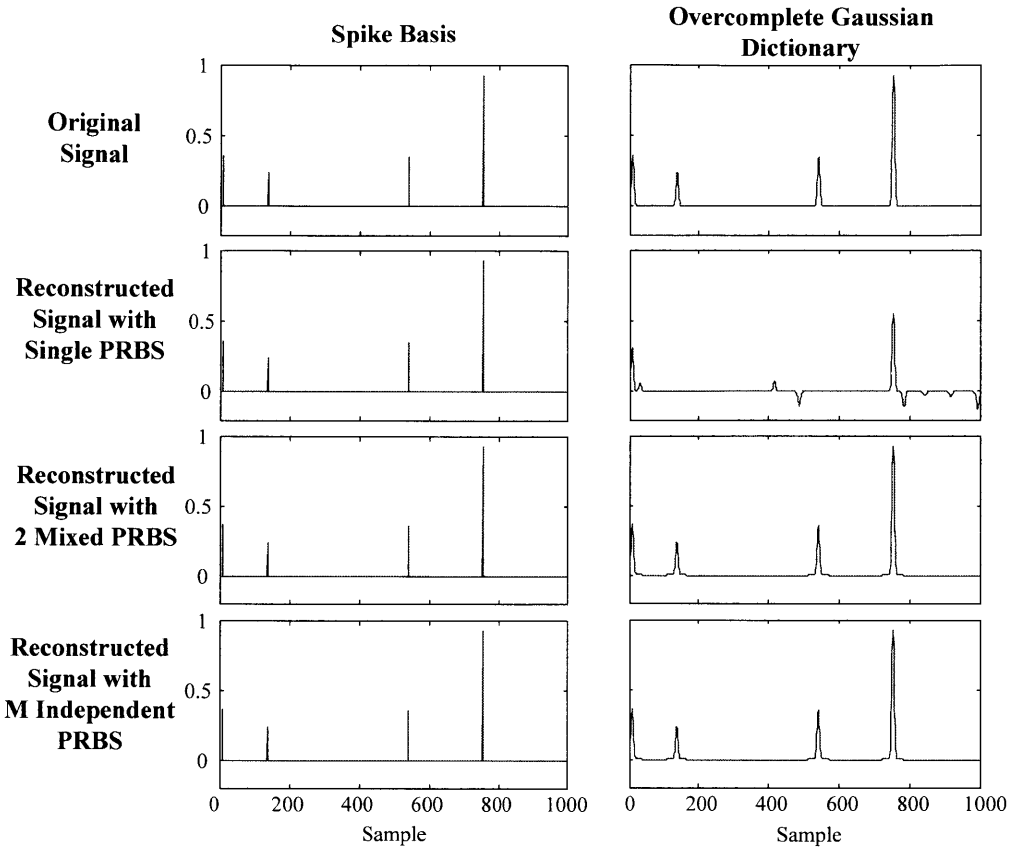


Figure 5-2: Example of original and reconstructed signals using different measurement matrices (Φ).

that PRBS1 in Figure 5-1 is unneeded). The positive results from using Toeplitz-like matrix structures as described in [105] would suggest that this is possible too. However, in [105], the reconstructed signals were delta spikes, and spikes in time reflect a critically sampled input whereas the input is typically oversampled. When it is oversampled (requiring an overcomplete dictionary and not an orthogonal basis), the inner product of a measurement matrix that is derived from a single shifted PRBS sequence and the input will appear correlated over several sample times. The net result is that consecutive input samples which are similar provide ambiguous information when captured with a bit shifted version of a single PRBS sequence. In other words, the same measurement result could have been produced by any one of a number of phases of the same sparse signal.

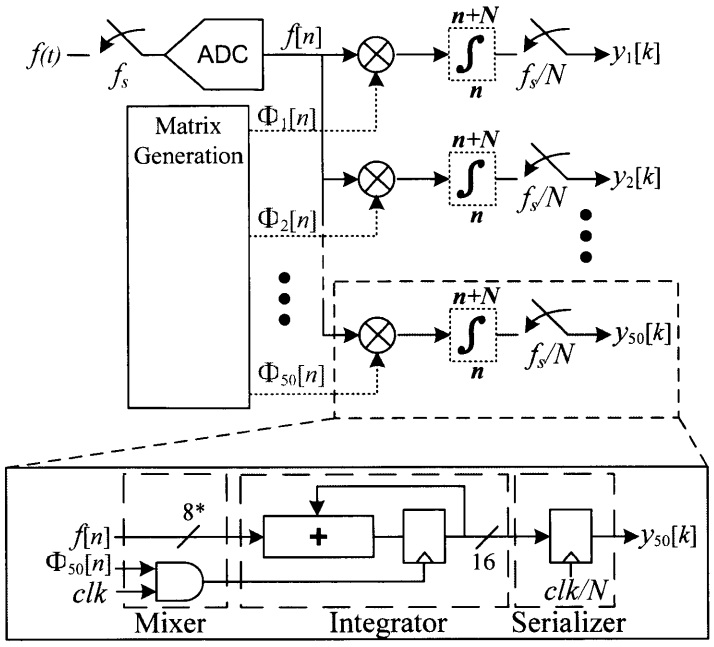


Figure 5-3: Block diagram of the test chip.

To illustrate this point, as well as the functionality of the mixed PRBS matrix, Figure 5-2 plots the original and reconstructed signals for when the input signal is spike based (critically sampled) and when it is a series of Gaussian pulses. In all cases, the size of the measurement matrix is the same (75×1000) and the coefficient vector, $\boldsymbol{x}$, is the same for both signals. When reconstructing the spike input, the signal basis, $\boldsymbol{\Psi}$, is orthogonal and is the identity matrix whereas $\boldsymbol{\Psi}$ is an overcomplete dictionary (sample shifted versions of the same basis vector(s)). As Figure 5-2 shows, the mixed PRBS solution produces the same result as the matrix generated by the M independent PRBS generators. However, when using the single PRBS to generate the matrix, only the spike signal is faithfully reproduced.

■ 5.2 Test Chip Architecture

Based on the power modeling results from Chapter 4, the digital CS encoder architecture is adopted for design and fabrication as it clearly presents the more energy-efficient option for the targeted specifications. The block diagram of the resulting test chip architecture is

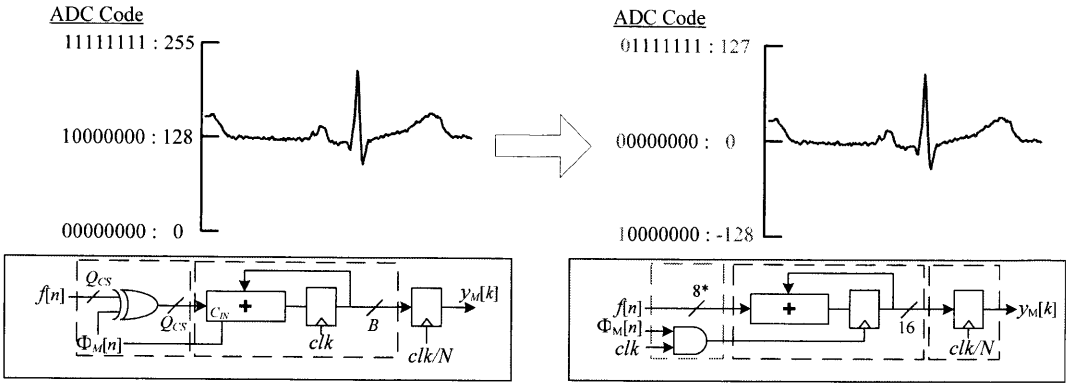


Figure 5-4: Recoding of the ADC output to enable clock gating of the accumulators.

shown in Figure 5-3. The architecture is more or less the same as the digital CS encoder described in Section 4.2.

Accumulator Clock Gating

The primary difference from the digital CS encoder described in Section 4.2 is that the ADC output is recoded into twos complement format, so that the output is nominally zero when there is no input differential signal. This assumes that the amplifier stage nominally biases the output in the center of the available range. The net result of this change is that it allows clock gating of the accumulators on -1 entries instead of performing subtractions. This approach saves roughly half the dynamic power of the accumulators since the matrix coefficients are evenly split between ± 1. The logical and circuit transformations are shown in Figure 5-4. The resulting matrix has 1/0 entries and is functionally equivalent to its ± 1 counterpart. The only drawback is that the accumulators may saturate for a smaller N if the input has a large DC component, which is why the ADC output was recoded to begin with.

■ 5.3 Test Chip Measurements

In order to validate the predicted hardware costs and demonstrate the system, the encoder circuits shown in Figure 5-3 were fabricated in a 90 nm CMOS process. The test chip consists of an 8-bit SAR ADC [70] and the digital CS encoder block [25]. Figure 5-5 shows the die

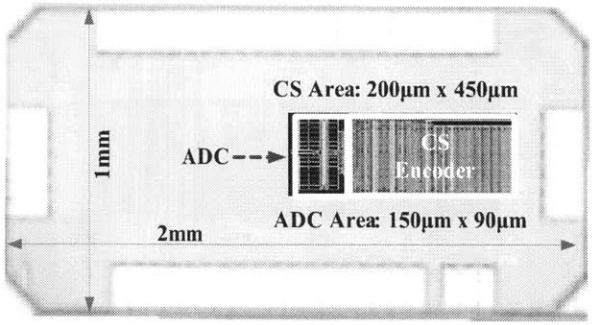


Figure 5-5: Testchip Die Photo.

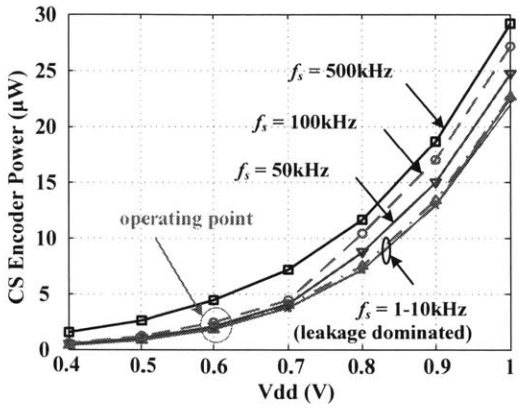


Figure 5-6: Measured power consumption of the CS encoder.

photo of the chip with the layout superimposed. The measured power of the CS encoder is shown in Figure 5-6. The digital CS encoder, including control circuitry, matrix generation and clock power, consumes only 1.9 μW at 0.6 V for sampling frequencies below 20 kS/s. As expected, the measured power is largely dominated by leakage for the sampling frequencies of interest. Considering that the operating point is in the leakage limited regime, the results correlate well with the model developed in Chapter 4 which predicts $\sim$0.6 μW of power consumption for the digital backend and matrix generation (not including clocks, buffers or control circuitry) under the same operating conditions.

The block diagram of the test infrastructure, both on chip and off chip, is shown in Figure 5-7. For testing, pre-recorded sensor signals were either driven into the ADC from an external DAC or passed directly as digitized data into the CS block through an on-chip deserializer. The output of the ADC can be observed synchronously with the output of the

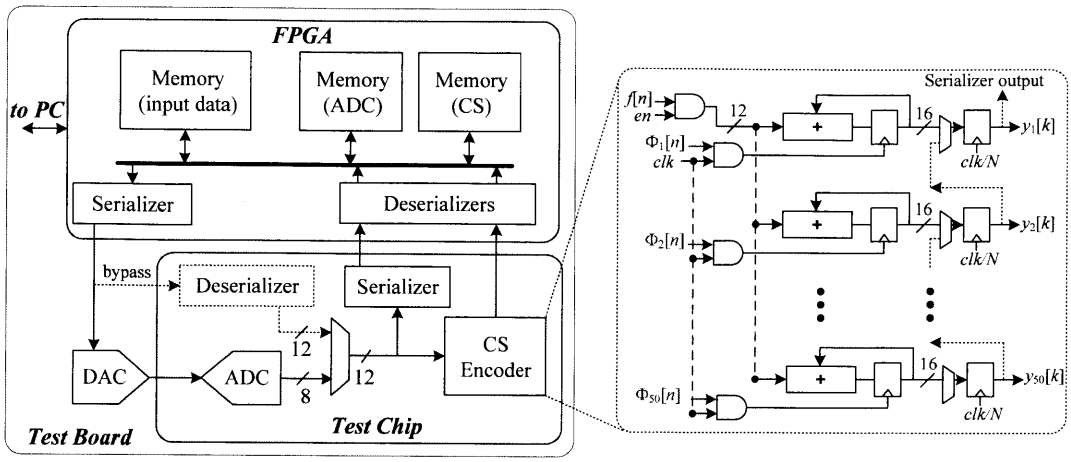


Figure 5-7: Testing infrastructure for the CS encoder test chip.

CS encoder block to enable a comparison between the quantized and reconstructed signals. Figure 5-8 shows an example of a continuous data acquisition for an N/M ratio of 1000/50. In this example, a pre-recorded EEG signal [106] driven by the off-chip DAC is sampled, compressed and then reconstructed off-line. The input is quantized by the ADC and continuously compressed from 1000 8-bit ADC samples into 50 16-bit accumulator measurements netting an effective compression factor of 10. As Figure 5-8 shows, the reconstructed signal faithfully represents the distinguishing features of the original ADC output despite being somewhat lossy.

From Chapter 2 we know that the quality of the recovered signal depends on the signal sparseness, compression factor (N/M) and the resolutions of the ADC and CS encoder. However, in prior analysis, the ADC was assumed to be perfectly linear while the CS encoder resolution was always sized to accommodate the full dynamic range of the measurements. To explore the necessity of each requirement, a synthetic EEG signal with over a dozen non-zero elements is created and driven by the off-chip DAC into the test chip. The measured SNR for the reconstructed signal under varying compression factors and resolutions is plotted in Figure 5-9. Since the number of non-zero elements exceeds what can be reconstructed from only 50 measurements, it is expected that the reconstruction will not be perfect. However, in each case, the largest amplitude spike signal is well recovered which is indicative of the CS

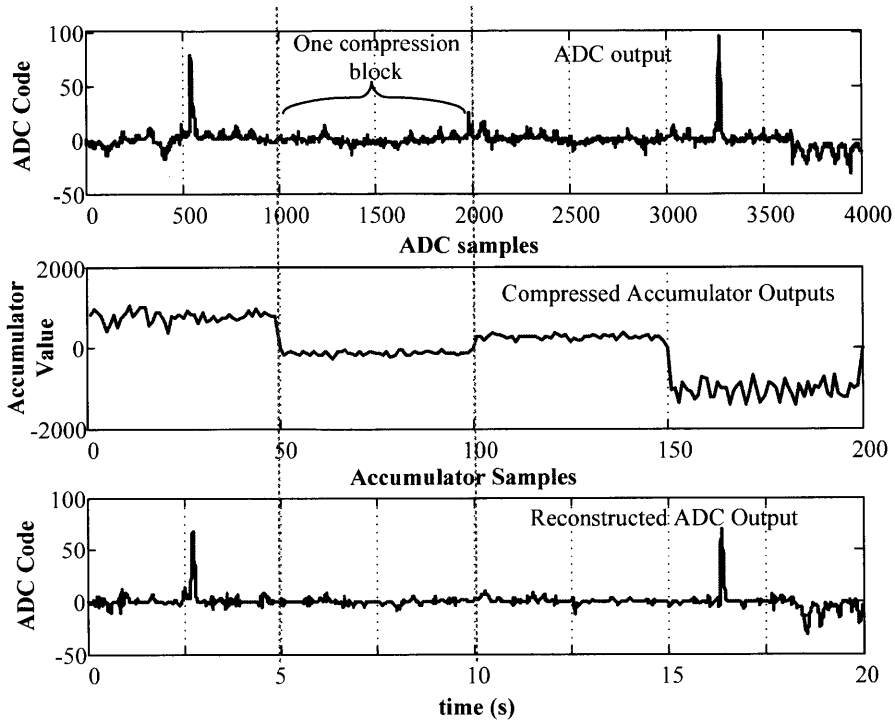


Figure 5-8: Measured result showing continuous data acquisition of an EEG signal (driven by an off-chip DAC) for the ADC output, compressed measurements, and reconstructed waveform for $N = 1000$ and $M = 50$.

reconstruction process being more robust when recovering higher energy components of the signal. The effect of having a lower resolution ADC, is emulated by masking out the ADC's LSBs in hardware while the effect of transmitting fewer bits from the CS encoder is mimicked by dropping bits during reconstruction. As the plots show, there is little perceptual difference between the reconstructed signal from an 8-bit and 5-bit ADC output. The same is true when the measurement resolution is reduced to 8-bits by dropping LSBs in the accumulator. If both resolution requirements could be relaxed, it would further lower the costs of the ADC and OTA as well as improve the compression factor (by transmitting fewer bits). It is also interesting to note that at lower resolutions, the reconstruction error from the on-chip ADC output is lower than from an ideal ADC. This is likely due to lower quantization error introduced by non-linearities (non-uniform quantization) in the on-chip ADC. This is not a wholly unexpected result as uniform quantizers are not necessarily optimal for CS signal

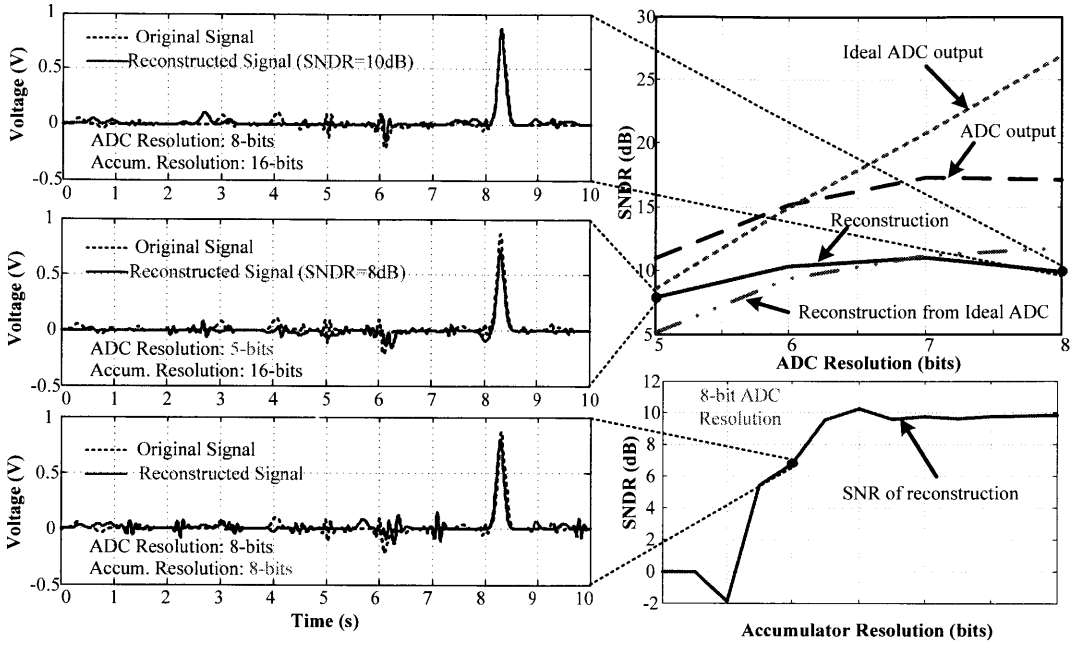


Figure 5-9: SNR of the ideally and actual quantized signals and associated reconstructed signals for each versus measurement resolution (B) and ADC resolution (Q_{CS}). Select accompanying waveforms provide relative points of reference for the quality of the reconstructed signals.

recovery [84, 107].

■ 5.4 Assessing Signal Quality: an EKG Example

A common problem with evaluating lossy compression schemes is that the metrics for success are often subjective and at the discretion of an individual's perception (e.g. image compression). The reconstructed signals shown in Figure 5-9 may be perfectly acceptable to one set of users and not to another. However, when designing a system where there are significant costs associated with over design, it is important to delineate between what is and what is not enough *information*. To try and highlight the importance of this question, and at the same time demonstrate the practical applicability of CS, an electrocardiogram (EKG) monitoring application is used as an example. The block diagram of the experiment is shown in Figure 5-10. In this example, a pre-recorded EKG signal from the MIT-BIH arrhythmia

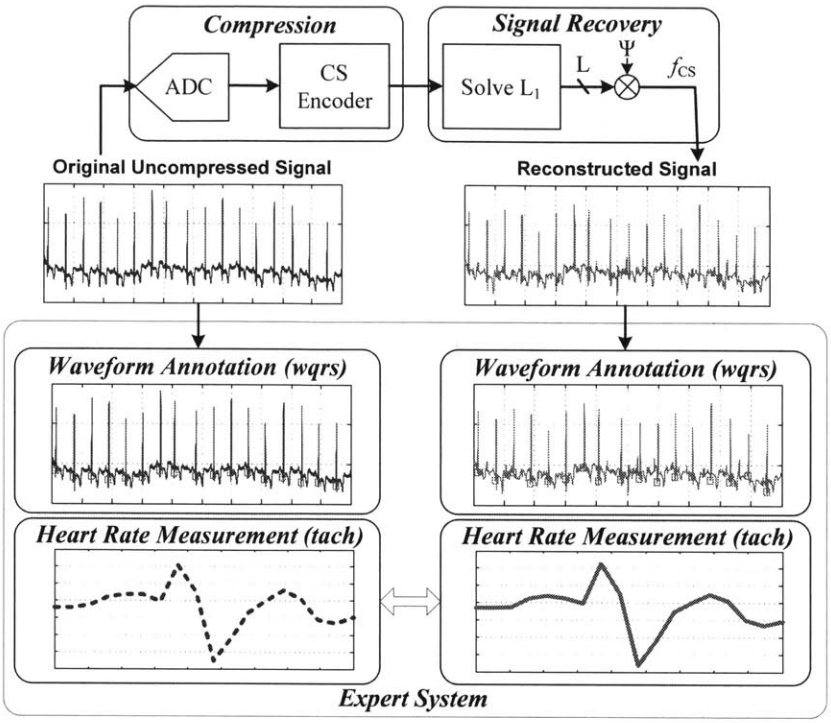


Figure 5-10: Block diagram describing the expert system which annotates QRS waves from an EKG waveform and calculates a running estimate of the heart-rate based on the annotations. Results from the CS system are compared to results based on the original uncompressed waveform.

patient database was used as the input [108] to the CS encoder. The reconstructed signals are then processed with an automated QRS wave detection algorithm whose output is used to determine the instantaneous heart rate of the patient via another automated algorithm [109]. The results from each reconstruction are compared against the same decision algorithms based on the original uncompressed signals.

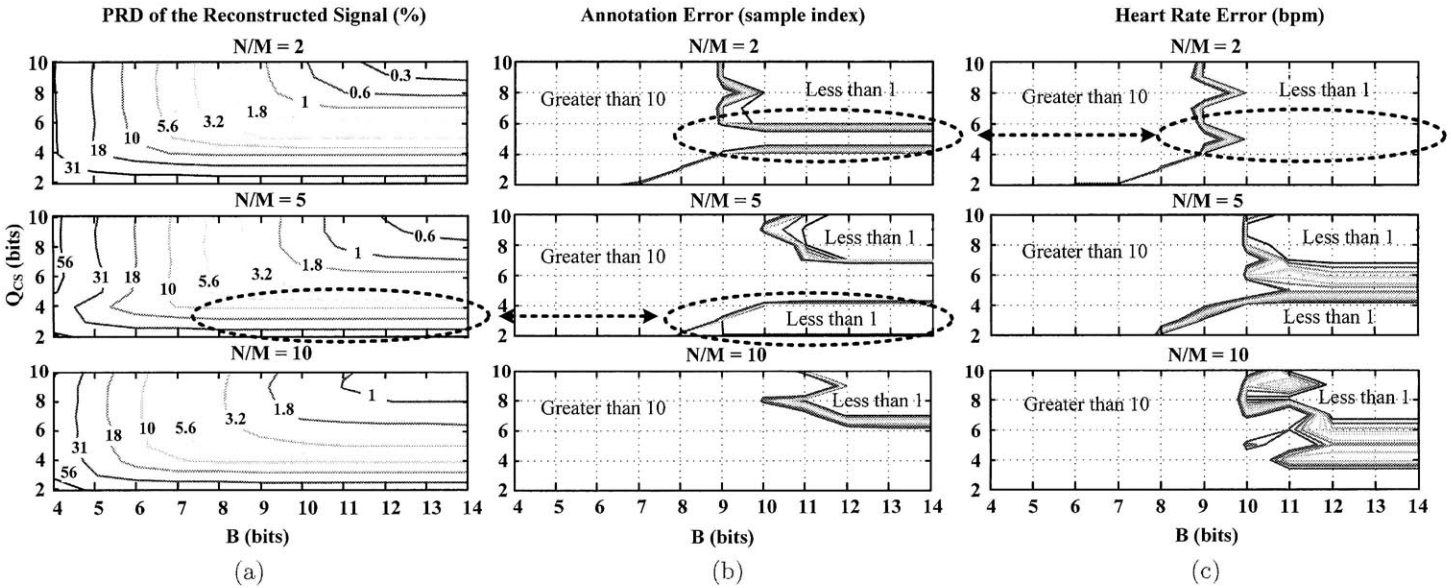


Figure 5-11: Contour plots of (a) the PRD of the reconstructed signal and (b) the corresponding annotation error and (c) heart rate error produced by that reconstructed signal. The annotation error and heart rate error are calculated with respect to the uncompressed signal. Highlighted regions indicate scenarios where high level information is preserved (not corrupted) despite lower level metrics indicating otherwise.

Figure 5-11 plots the result of this experiment where the ADC resolution and CS encoder resolution are swept at three different compression factors to introduce a range of signal distortions. Figure 5-11a shows the PRD of the reconstructed signal across this operating space. The EKG signals are reconstructed using an over-complete Gabor dictionary. A more optimized signal basis/dictionary can be found but in general the reconstruction performance is on par with other reported results for the same database signals [110] even if the compression performance is not quite optimal. Figure 5-11b plots the error between QRS wave annotations extracted from the CS reconstructed signal and annotations extracted from the uncompressed quantized input. The annotation error is calculated for every annotation (false positive and missed positive) and is specified in number of samples where errors greater than 10 samples are truncated to provide a useful visual dynamic range. Similarly, Figure 5-11c plots the error in the instantaneous heart rate calculated from the QRS wave annotations where the units are in beats per minute (bpm). Again, errors exceeding 10 bpm have been truncated for visual purposes. At a compression factor (N/M) of 10, the metrics provided by each of the plots are consistent indicating that a measurement resolution greater than 10 bits and an ADC resolution greater than 8 bits are needed to produce the same detected heart rate as the uncompressed signal. However, the results at compression factors of 5 and 2 illustrate the dependency on the application context. For example, the PRD calculated for very low ADC resolutions for an $N/M = 5$ would indicate that significant signal information was lost. However, the annotation and heart rate decisions derived from these reconstructed signals say otherwise. Similarly, when $N/M = 2$ there is an intermediate range of ADC resolutions that produce larger annotation errors, however, these errors do not significantly corrupt the calculated heart rate. So depending on what level of information is required, there is a possibility to drastically reduce the hardware requirements for certain end applications which would further reduce the total system power. Thus, developing a framework to optimize metrics beyond PRD, MSE, and SNDR, such as described in [111], could be extremely beneficial for lowering system costs.

Table 5.1: An example of average sensor power with and without CS data compression.

System (f_s=20 kS/s)	**AFE** (NEF=3)	**ADC** (100 fJ per conv. step)	**Data Reduction**	**Tx** (3 nJ/bit)	**Total**
ADC Only	v_n =2.8 μV_{rms}	10 bits	–	200 kb/s	
	7.6 μW	1 μW	–	600 μW	608.6 μW
ADC with CS	v_n =2.8 μV_{rms}	10 bits	10X	20 kb/s	
	7.6 μW	1 μW	1.9 μW	60 μW	70.5 μW

■ 5.5 Summary

In this chapter we discussed the design and test of the CS encoder test chip. To our knowledge, this is the first demonstrated IC implementation of such a design. Test measurements demonstrate the system benefits of using CS as a source encoder, both in regards to hardware cost and compression performance as predicted in Chapters 2, 3, and 4. The measured power results correlate well with the model predictions and confirm that the minimum energy of the encoder is limited by sub-threshold leakage. An example of the energy benefits from compression along with the low overhead cost of the CS encoder are summarized in Table 5.1.

In the leakage dominant regime, any alternative source coding/data compression strategy must have on the order of 1000 flip-flops or less to be competitive with CS in terms of energy cost. So for example, in Huffman and LZW, the typical memory requirements for their dictionaries alone would exceed 1000 elements, (typically >4 kb for 8-bit words), even in optimized hardware [112]. When not limited by leakage, the computational complexity of any alternative must be on the order of $M \cdot N$ additions to roughly match the cost of CS. This does not take into account the latency of the algorithm either. In the proposed implementation of CS, there is no latency penalty to encode after the N^{th} sample has been captured whereas LZW and Huffman have the overhead of searching their code books; whether done in parallel or iteratively there will be a power cost to maintain some fixed latency/throughput.

The primary enabler for the low leakage and computational cost for CS was the devel-

opment of a compact and efficient matrix element generator that created each column of $\mathbf{\Phi}$ dynamically per sample. The resulting matrix generator is less than $1/10^{\text{th}}$ the cost of the matrix itself and is likewise an order of magnitude less energy than the next best generator option (M independent PRBS). One of the important findings in developing the matrix generator is that a single PRBS generator is not functionally suitable for use in CS applications when the input signal is oversampled. To solve this problem, we 'mixed' in the output of a second PRBS generator and showed that the reconstruction performance using our new matrix was equivalent to the matrix constructed from the M independent PRBS generators.

Finally we show results for an EKG application where it is shown that signal information can be preserved across different application levels, which highlights the importance of understanding the end application and usage model as it may enable more efficient underlying hardware.

Chapter 6

Integrated Circuit Design with MEM Relays

With the application of CS, we have shown that it is possible to create a generic hardware interface that can efficiently capture and represent sparse information signals with a low overhead cost of implementation. If we consider the duties of a wireless sensor node, this essentially minimizes the active energy that a wireless sensor node must consume. However, as we noticed with the leakage limited CS encoder design, there is an energy price paid even when the sensor is doing nothing. In the following chapters, we will try to address the dominant inactive energy cost of wireless sensor nodes with the adoption of a new underlying technology: MEM relays.

As we discussed in the introduction, for sensor node lifetimes to be measured in years without incurring high battery costs (large batteries or battery replacements), the average power consumption in each node must be in the range of 10 μW. However, even when sensor nodes are deeply duty-cycled such that they are essentially inactive, the power budget may be consumed entirely by idle memory [44]. Even in the most aggressively scaled sub-threshold memory designs, leakage energy alone can still consume on the order of 1 μW per 32 kB of memory [113, 114]. However, device variation makes such aggressively voltage scaled designs difficult to achieve at scale. Although the implementation costs of our CS encoder

are attractive in this respect when compared to more resource demanding source encoding algorithms, the energy consumption is still limited by device leakage.

As described in Section 1.2.3, the fundamental limitation with CMOS is that there is always a non-zero minimum energy per circuit function where leakage energy consumption matches the dynamic energy consumption; the fact that the leakage current is dependent on the thermal voltage (kT/q), which does not scale, has caused threshold voltages (V_T) to stop scaling as well. Thus, for energy constrained applications such as WSNs (and really most IC applications), it is of interest to investigate the use of alternative device technologies that offer better energy trade-offs; to outperform CMOS, this requires a higher I_{on}/I_{off} relationship and steeper subthreshold slope across operating conditions where I_{on} and I_{off} correspond to the device *on*-state and *off*-state currents [115].

MEM Relay Overview

One promising class of devices that offers nearly ideal switching characteristics (high I_{on}, low I_{off}, near-infinite sub-threshold slope) are electro-statically actuated MEM relays (or switches) [116]. The fact that MEM relays exhibit zero leakage current when switched off makes them extremely attractive for low power applications. Since the energy efficiency of many CMOS designs (such as our CS encoder) are limited by leakage current, we naturally are interested in exploring the viability of adopting MEM relays as a switching technology for very-large-scale integration (VLSI) applications. To date, the cited drawbacks for MEM relays as a logic element have been that they are too big and too slow to be considered competitive with CMOS. On the device level, when comparing a single MEM relay to a single CMOS transistor, this is generally true. However, we are not building single device systems; what we are more interested in is comparing the technologies at the system or functional level which then allows us to provide useful feedback to the device technologists.

To assess the viability of MEM relays as a potential logic element, we first examine and model the switching characteristics of relays. An important observation is that MEM relays benefit from scaling in a similar fashion to CMOS. The miniaturization of mechanical structures is in fact what has allowed the MEMS field to exist in the first place. So if device

features can continue to be shrunk, both performance (speed) and energy will continue to improve–only there won't be any leakage current to balance against!

Based on the device switching characteristics, we then develop a circuit design methodology that is specifically catered to MEM relays [26]. In this way, we can compare optimized relay designs against optimized CMOS designs. In particular we exploit the large discrepancy between mechanical and electrical time constants of the switch by implementing large complex logic gates instead of staged logic. The results of adopting this circuit design methodology improves the circuit level performance of logic by over an order of magnitude, which helps narrow the performance gap with CMOS and expand the range of potential applications beyond the traditional RF and power gating applications. As we will show, the resulting design paradigm is also used to relax and improve the reliability of the device design.

Based on this design methodology, several circuit sub-systems that span typical VLSI building blocks, such as logic, memory and I/O, are designed and simulated to show that the energy-efficiency of MEM relay circuits can be over an order of magnitude below what is achievable in CMOS, provided that we can scale the physical dimensions. Based on the promise of these results, we then present some experimental data that demonstrate the feasibility of relay based circuits [22, 23]. Although most of the fabricated devices were in a feature size larger than ultimately desired, the circuits and testing performed at this "macro" level proved to be extremely beneficial in identifying, and in many cases improving, any issues at the device level. Finally, we present the most recent device in the evolution of the technology and discuss the projected performance of several circuit blocks relevant to sensor nodes.

■ 6.1 MEM Relay Technology

In general terms, a MEM relay can be thought of as any electrical connection that is completed by some electrically actuated mechanical movement. A generic lumped-parameter spring-mass-damper model for such a device is shown in Figure 6-1 where x is the displace-

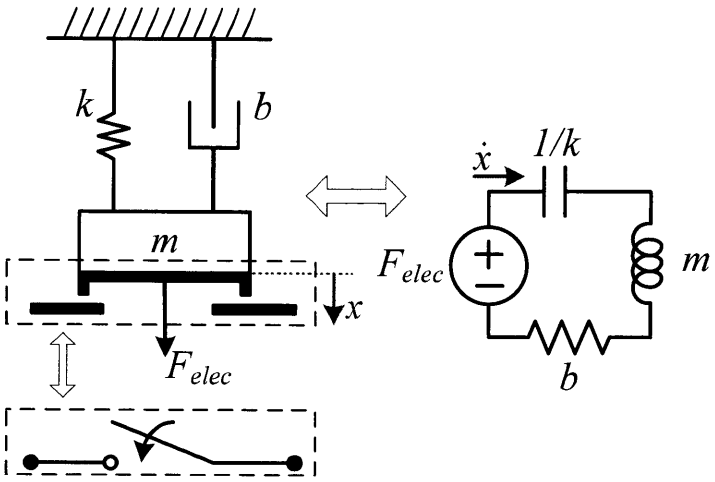


Figure 6-1: A generic lumped-parameter electro-mechanical model of a MEM relay.

ment, k is the effective spring constant, b is the damping force, m is the effective inertial mass, and F_{elec} is the applied electrostatic force. There are numerous ways to realize this type of switching function and the model parameters for each realization will be determined by the switch's physical dimensions and structural materials [19, 116–120]. The design methodologies and results that we will present are generally applicable across different MEM relay designs. For example, our initial analysis in this area developed the design insights based on a MEM cantilever relay (Figure 6-2a) [26]. However, the majority of the experimental work [23] was based on a later design: a 4-terminal (4T) folded spring MEM relay (Figure 6-2b) [19].[1] The results presented in this thesis have origins associated with both device types, and although their absolute performance (speed, voltage, energy) is different, their device operation and models are essentially the same, so we will describe these devices in tandem and differentiate where appropriate.

■ 6.1.1 MEM Relay Structures

Figure 6-2 shows a diagram of the layout and cross-sectional views of the 4-terminal cantilever relay and the more recently designed 4T folded spring relay. The device dimensions

[1]The design, fabrication and device test data for the cantilever and folded spring design were done by Hei (Anderson) Kam and Rhesa Nathanael, respectively–both at the UC-Berkeley Microlab.

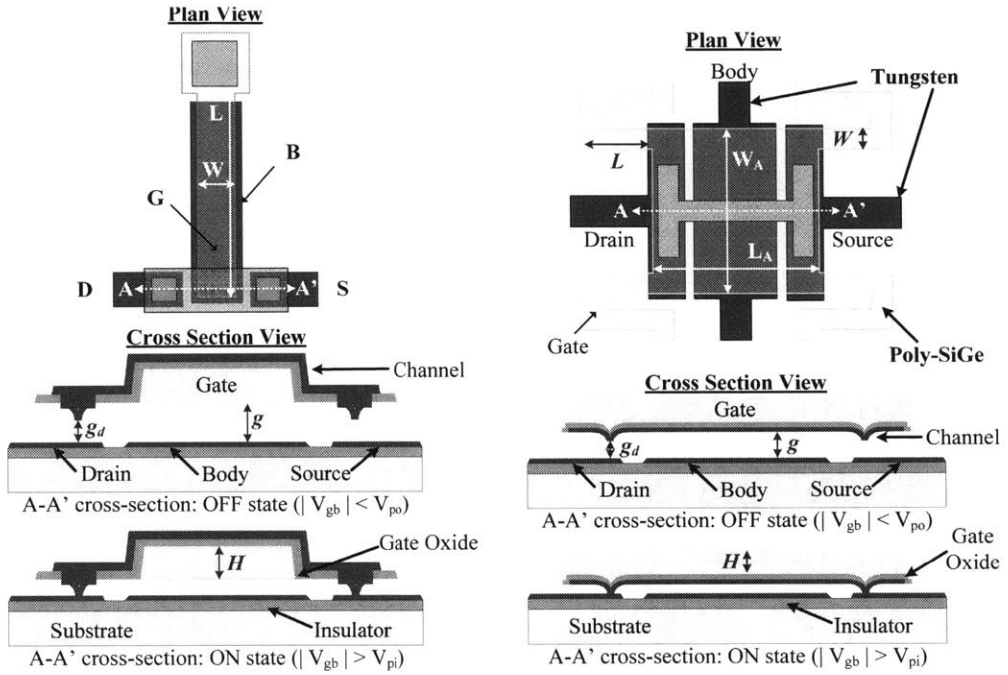


Figure 6-2: Layout and cross-section views of a 4-terminal cantilever MEM relay (a) and a folded flexure (crab leg) MEM relay (b). In general we will use 4T to refer to the crab leg design in (b)

associated with the different fabricated devices are listed in Table 6.1. The general composition and function of the relay is similar to a MOSFET in that there are four terminals; both relays consist of a of a gate terminal that suspends a floating channel, separated by a gate oxide, over the body, drain and source electrodes. The folded spring design helps mitigate the effects of any residual stress in the device that may cause out of plane bending/curling [121]. This robustness to residual stress was one of the reasons for later moving to the folded spring design as this was one failure mechanism of the cantilever design. Polycrystalline silicon-germanium (poly-SiGe) is used as the gate's structural material because of its thermal compatibility with backend CMOS processes [122] while tungsten is chosen for the channel, drain, source and body electrodes due to its compatibility with HF-vapor etching [123, 124] and its physical and electrical durability as a contact material [125]. As we will discuss later, the insight that aided the choice of tungsten as a contact material emerged

Table 6.1: 4T device dimensions used in experimental data [23, 26].

Device	$W_A \times L_A$	W	L	H	g	g_d
Cantilever (5 μm)	N/A	5 μm	30 μm	1 μm	500 nm	250 nm
Cantilever (2 μm)	N/A	2 μm	12 μm	400 nm	200 nm	100 nm
Cantilever (1 μm)	N/A	1 μm	6 μm	200 nm	100 nm	50 nm
Folded spring	27×30 μm	5 μm	27 μm	1 μm	200 nm	100 nm

from circuit level analysis. For the folded spring (cantilever) relay, the channel is suspended beneath (above) the gate's center plate by an insulating aluminum oxide (Al_2O_3) layer which acts as a gate dielectric. The actual contact points, or contact dimples, protrude out from the channel and restrict the distance that the gate can be displaced. For the folded spring design, after the entire structure has been released, it is coated with a thin layer (~1-2 Å) of titanium oxide (TiO_2) to reduce current densities at the contacts and slow down any native tungsten oxide formation.

■ 6.1.2 Static Switching Characteristics

The basic operating states of the relay are also shown in Figure 6-2. The device is actuated by applying an electrostatic force (F_{elec}) between the movable gate electrode and the body electrode beneath it. When a sufficient electrostatic force is applied to overcome the mechanical spring force of the relay, the entire gate and channel structure will be "pulled-in" such that the channel completes the connection between the source and drain (as shown by the ON state cross-sections). To open the switch, the applied voltage must be reduced until the gate's spring restoring force returns it to its original resting position (the OFF state). The measured leakage of the folded spring 4T relay in this OFF state is shown in Figure 6-3. The figure shows both of the attractive aspects of MEM relays: the leakage is beneath the noise floor of the measurement instrumentation and the current drops by over 10 orders of magnitude over a 1 mV input voltage.

The critical voltage at which the switch closes is called the pull-in voltage (V_{pi}) and it gets its name from the fact that for a fixed gate-to-body voltage, the electrostatic force increases as the gate gets closer to the body. When the applied voltage is such that the spring force

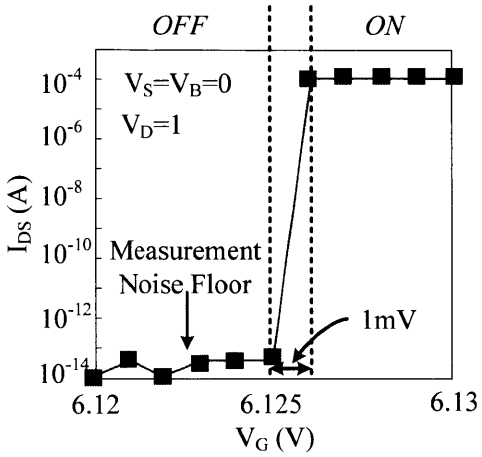


Figure 6-3: Measured leakage of a 4T folded spring MEM Relay (data courtesy of Rhesa Nathanael) [19].

is not sufficient to balance out the electrostatic force, then positive feedback occurs and the gate abruptly gets "pulled-in". The resulting derived value for V_{pi} is [126]:

$$V_{pi} = \sqrt{\frac{8}{27} \cdot \frac{kg_0^3}{\epsilon_{eff} A}} \tag{6.1}$$

where g_0 is the actuation gap when $F_{elec} = 0$, A is the actuation area, ϵ_{eff} is the effective dielectric seen between the center plate (or beam) and the body electrode ($\sim \epsilon_0$, air), and k is the effective spring constant of the relay. More details on this derivation can be found in A.3. The analytic expression for the effective (flexural) spring constant is:

$$k = \frac{\gamma E W H^3}{L^3} \tag{6.2}$$

where γ is a constant dependent on the device structure, E is the Young's modulus of the gate, and W, L and T are the width, length and thickness of the spring arms/cantilever beam. For the cantilever design, $\gamma = 0.25$, while for the folded spring structure, $\gamma = 2$. For the folded spring design, a more precise calculation that includes both flexural and torsional components is described in [127]. Once the device has been pulled in, the amount

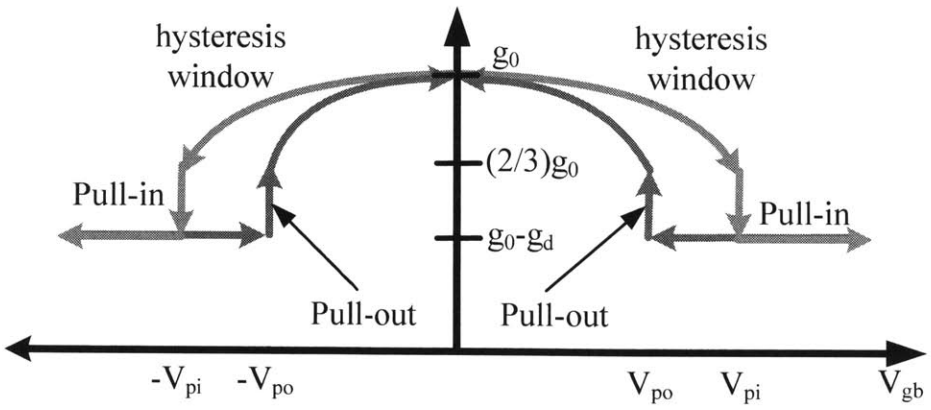


Figure 6-4: Illustration of the ambipolar pull-in and pull-out for the relays where gap distance (g) is plotted against the applied gate-to-body voltage.

of electrostatic force needed to keep the pulled-in state, V_{po} (the "pull-out" voltage), is less than V_{pi} since the actuation gap is now smaller. A similar exercise as used to determine V_{pi} can be used to find V_{po}.

$$V_{po} = \sqrt{(kg_d - F_A) \cdot \frac{2(g_0 - g_d)^2}{\epsilon_{eff} A}} \tag{6.3}$$

In (6.3), g_d is the gap between the dimple contacts and the drain/source electrodes and F_A accounts for the surface adhesion force at the contact points. V_{pi} and V_{po} are analogous to the threshold voltage in MOSFETs only that there are separate thresholds for turning the switch on and off. This is illustrated in Figure 6-4. The difference between V_{pi} and V_{po} results in a switching hysteresis that can limit the functional operating voltage since the voltage swing at the gate/body must at least span the two thresholds. For the folded spring dimensions shown in Table 6.1, the resulting analytic values for V_{pi} and V_{po} are ~8 V and ~7 V respectively.

■ 6.1.3 Dynamic Switching Characteristics

Similar to second order electrical systems, the switching delay of the relay is inversely proportional to the resonant frequency, $\omega_0 = \sqrt{k/m}$. The precise delay of the switch was modeled by Hei Kam and simulated in ANSYS [115]. The result of this analysis produced

Table 6.2: Electrical parameters for the 5 μm wide 4T relays in Table 6.1

Device	R_{con}	R_{pox}	C_{gb}	C_{gc}	C_{gd}, C_{gs}	C_{cb}
Cantilever	$\sim 0.25\Omega$	N/A	5.3 fF	50 fF	<1 fF	<1 fF
Folded spring	$\sim 0.1\Omega$	$500 - 1000\Omega$	32 fF	150 fF	6 fF	7 fF

the following analytical model for the switching delay:

$$t_d \cong \alpha\sqrt{\frac{m}{k}}\left(\frac{g_d}{g_0}\right)^{\gamma}\left(\frac{V_{dd}}{V_{pi}} - \chi\right)^{-\beta} \quad for\ 5V_{pi} \geq V_{dd} > 1.1V_{pi},\ g_d \geq g/3 \tag{6.4}$$

where $\chi \simeq 0.8$, and α, β, and γ are fitting parameters that are a function of the quality factor (Q) of the system. As shown in [127], these values saturate for values of $Q > 1$ so there is little performance to be gained by increasing Q. Unlike RF MEM switches, which are designed to resonate and thus prefer high Q, these switches are actually designed for low Q to avoid contact bounce and long settling times when turning off. A low Q also means that vacuum packaging solutions may not be necessary to insure the switching speed of the relay.

Like with transistors, the speed of the MEM switch is dependent on how much gate-to-body voltage is applied in excess of the pull-in voltage. For the folded spring dimensions listed in Table 6.1 and a relatively small overdrive voltage ($V_{dd}/V_{pi} = 1.2$), the mechanical delay (t_{mech}) is roughly 30 μs. At this point we should point out that this only represents the *turn-on* delay ($t_{d,on}$) of the relay. The *turn-off* delay ($t_{d,off}$), or the time it takes to open the switch, is much shorter than $t_{d,on}$ since the relay only needs to travel enough distance ($\sim$1 nm) to break the contacts rather than the entire gap distance.

■ 6.1.4 Electrical Characteristics of the Relay

The switch capacitance and resistance for the 4T folded spring relay are shown in the electrical model of the relay in Figure 6-5. The model for the cantilever is nearly identical with all of the same components (only different values). Of the major contributors to device

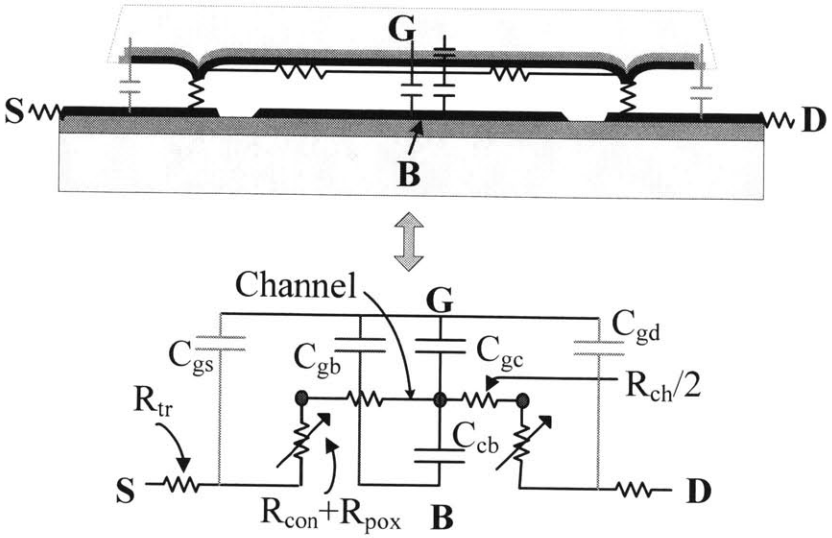


Figure 6-5: Electrical model of the folded spring MEM relay.

capacitance, only the gate-to-body capacitance (C_{gb}) is desirable (for actuation) while the other capacitances are unwanted parasitics. In both the cantilever and crab leg designs, C_{gb} and the gate-to-channel capacitance (C_{gc}) are the largest components where C_{gb} and C_{gc} are simply:

$$C_{gb,cantilever} \simeq \frac{\epsilon_0 W L}{g}, \quad C_{gb,crableg} \simeq \frac{\epsilon_0 W_A L_A}{g} \tag{6.5}$$

$$C_{gc} = \frac{\kappa_{ox}\epsilon_0 (W_{ch} L_{ch})}{t_{ox}} \tag{6.6}$$

where t_{ox} and κ_{ox} are the gate oxide thickness and relative permittivity and W_{ch} and L_{ch} are the dimensions of the channel. When the relay is switched off, C_{gb} is the dominant component, and when the switch closes, C_{gc} is also seen at the gate. C_{gc} is relatively large because it is only separated from the gate by a thin layer of Al_2O_3 whose dielectric constant is ~9. As a point of reference, Table 6.2 summarizes the worst case (switch closed) electrical parameters for the 5 μm devices from Table 6.1.

The total *on*-resistance (R_{on}) of the relay is comprised of the wire trace resistance (R_{tr}), the contact resistance (R_{con}), channel resistance (R_{ch}) and the resistance of the passivating TiO_2 layer (R_{pox}). Of these components, R_{con} and R_{pox} are by far the largest components

as their contact area is limited by surface asperities [128]. For the scale of devices tested, the component attributable to the conductive oxide resistance was dominant where the total measured *on*-resistance ranged from 500-1000 Ω. Based on these values, the electrical time constant of one switch driving itself (analogous to f_T in MOSFETs) is roughly 100-200 ps (and even lower for the cantilever design). However, as the device scales, it is expected that R_{con} will become increasingly significant. The contact resistance can be expressed as [129]:

$$R_{con} = \frac{4\rho\lambda}{3A_r} \tag{6.7}$$

where A_r is the effective area of the contact, and ρ and λ are the resistivity and electron mean free path of the contact material respectively. For tungsten, the values for ρ and λ are 55 nΩ·m and 33 nm respectively. The effective area of the contact, which is typically dominated by asperities, is a function of the loading force which is dependent on the applied electrostatic force (F_{elec}), material hardness (H), and the deformation coefficient (ζ) at the contact:

$$A_r \approx \frac{F_{elec}}{\zeta H} \tag{6.8}$$

ζ is a measure of the plasticity of the contact asperities [130] and is < 0.3 for elastic deformation, between 0.3 and 0.75 for elastoplastic deformation and < 1 for plastic deformation. For a 100X scaled device R_{con} grows to the 1 kΩ range which is roughly equivalent to what R_{pox} is currently. So even if the passivating oxide is no longer needed, switch resistance is still expected to be in the kΩ range for device scales of interest.

■ 6.1.5 MEM Device Scaling

At these dimensions, clearly the size and performance of the MEM switch is not comparable to current CMOS technologies. However, like CMOS, MEM relays also benefit from device scaling. For example, Table 6.3 shows that if all of the physical dimensions (W, H, L, g, etc...) of the relay are simply scaled linearly, then V_{pi} should scale linearly too. As a result, the delay and energy scale quadratically and cubicly. Experimental verification of V_{pi} scaling for the cantilever relay is shown in Figure 6-6. There are more energy-efficient

Table 6.3: Constant Field Scaling for MEM relays showing the corresponding device dimensions and parameters for a folded spring relay scaled by 10X.

Device or Circuit Parameter	**Constant Scaling Factor**	**With Scaling Factor = 10**
Physical dimensions: L_A, W_A, W, H	$1/S$	2.7,3,0.5,0.1 μm
Beam Length: L	$1/S$	2.7 μm
Gap Distance: g, g_d	$1/S$	20,10 nm
Actuation Area:	$1/S^2$	4.5 μm^2
Surface Forces: F_A	$1/S^2$	4 nN
Pull-in Voltage: V_{pi}, V_{po}	$1/S$	0.67 V
Gate Capacitance: C_{gc}, C_{gb}, $C_{gd,gs}$	$1/S$	4.7, 4.0 fF
Speed: t_{mech}	$\sim 1/S^2$	400 ns
Energy: $V_{dd}^2 \cdot C_g$	$1/S^3$	5.7 fJ

scaling strategies [127], but in general these scaling trends indicate that there is a direct path towards improving the performance and energy of MEM relays. It is also for this reason, that the device was designed to actuate in the z-direction such that the actuation gap, which is the smallest dimension of the device, would not have to be defined (and be limited) by lithography. Naturally, not all of these dimensions are as easily scalable as described; these and other challenges will be discussed later.

■ 6.2 MEM Relays for VLSI Applications

One of the reasons for choosing the design of the 4T relay to be so similar to CMOS is so that it can leverage as much of the design infrastructure that has been built up around CMOS as possible. And although the 4T MEM relay shares many similar characteristics with MOSFETs, in this section, we will discuss the fundamental differences in device characteristics that motivate an alternative set of circuit design strategies compared to what is conventional in CMOS.

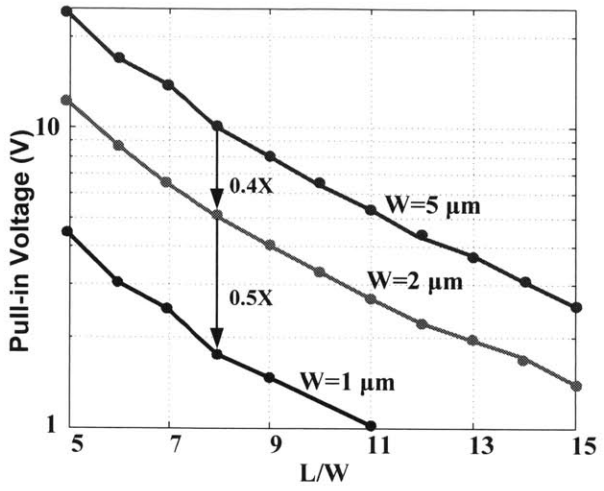


Figure 6-6: Measured cantilever relay pull-in voltages as device widths are scaled. (Data courtesy of Hei Kam.)

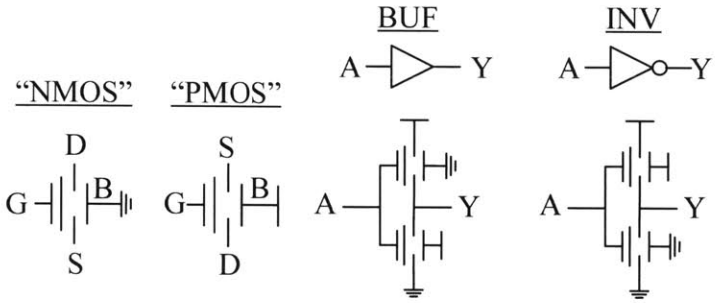


Figure 6-7: MEM relays as logic elements.

■ 6.2.1 MEM Switches as a Logic Element

As was described in Section 6.1, the operation of the MEM relay is such that when a sufficient electrostatic force is applied between the gate and body terminals, the switch will close. The voltage at which this occurs (V_{pi}) is analogous to the threshold voltage (V_{th}) in MOSFETs, so the logical function of the relay is the same as a transistor except for two key distinguishing features. First, for MEM relays the electrostatic force can be applied by either a positive or negative gate-to-body voltage. Thus, for a gate input that swings between ground and V_{dd}, the same relay can act as either a PMOS or NMOS equivalent device by simply biasing the

body node to either V_{dd} or ground respectively. This is illustrated in Figure 6-7 which shows the use of the MEM relay corresponding to the two logic switch elements in CMOS. Also shown in Figure 6-7 is the other distinguishing characteristic. Unlike CMOS devices, V_{pi} is (nominally) independent of the source and drain voltages so both "PMOS" and "NMOS" devices are equally good at pulling up to supply as they are at pulling down to ground. Consequently, implementation of both inverting and non-inverting logic gates are possible.

■ 6.2.2 Logic Styles for MEM Relays

While MEM switches can implement the same logic styles as CMOS, their electrical characteristics and behavior are significantly different than that of CMOS transistors. Consequently, relay circuit topologies should be tailored to optimize their device characteristics rather than follow those of CMOS. Since CMOS circuits are dominated by their electrical time constant, there is a quadratic delay penalty for stacking devices in series. Consequently, the proper way to design in CMOS is to buffer and distribute the logical and electrical effort over many stages of simpler logic gates [101, 131]. In contrast, the delay of a MEM relay is dominated by the time it takes to mechanically actuate the gate. In the example from Section 6.1, the mechanical delay ($\sim$30 μs) is orders of magnitude larger than the electrical time constant ($\sim$200 ps for $R_{on} = 1k\Omega$) associated with charging or discharging the output once the switch closes.

Given this large disparity between the mechanical and electrical delays of the switch, the observation we made is that relay-based circuits should optimally be designed such that all mechanical movements would happen simultaneously–even if this drastically increases the *on*-resistance of the logic gate [26]. To illustrate this point, Figure 6-8 plots the delay of a cascaded series of N 2-input AND gates implemented with both transistors and MEM relays using static CMOS and pass-transistor logic styles. As the plot shows, adopting a static CMOS logic style is a poor choice for MEM relays since the mechanical delay is incurred at every stage and thus the circuit performance tracks the device performance. Likewise, pass transistor logic is ill suited to take advantage of MOSFET characteristics. However, when pass-transistor logic is used with relays, the incremental electrical delay due

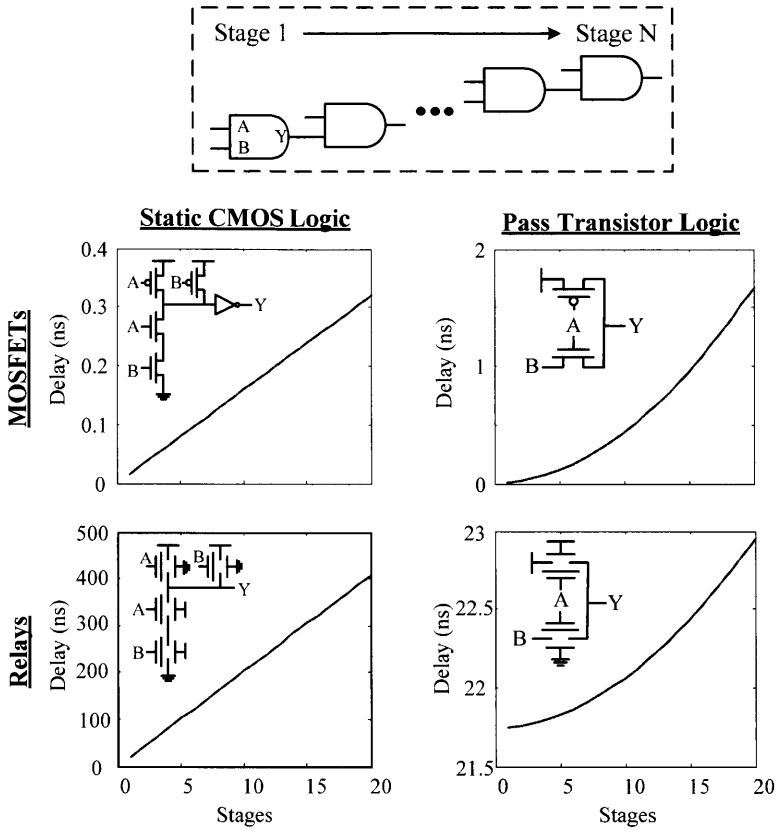


Figure 6-8: Delay of N 2-input AND gates implemented using transistors and relays in both static CMOS and pass transistor logic styles (90 nm CMOS [20], folded spring MEM Relay scaled 20X)

to stacking devices in series is small when compared to the mechanical delay of the relay and thus the performance gap between CMOS and MEM relays begins to shrink for longer logic depths. The extent to which relays can be stacked in series depends on the relative delay between mechanical and electrical delays. Eventually, the number of series devices will be large enough such that the electrical delay approaches the mechanical delay at which point a buffer should be added. Even in scaled versions of the device, this cross-over point does not happen until the device stack exceeds 100's of relays. Most practical logic functions have a lower fan-in than this, so in general, relay based circuits should be designed as a single complex gate when possible.[2]

[2]This should really come as no surprise since many of the macro sized relay circuits used before transistors

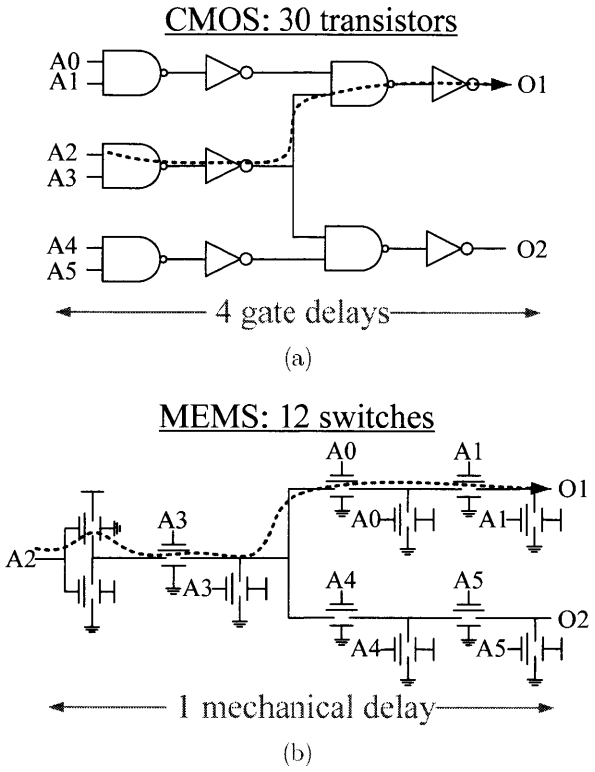


Figure 6-9: Circuit level comparison between CMOS and MEM relays.

■ 6.2.3 Relay vs. CMOS Paradigms

Since each device technology has its own preferred logic style, the proper level of abstraction at which CMOS and MEM switches should be compared is not at the device level, but rather at the circuit or functional level. As shown in Figure 6-9, the performance of a relay based circuit is essentially bound by the relay's mechanical delay; relay circuits pay very little penalty in total delay by adding complexity as long as all of the relays can switch simultaneously. Thus, as the complexity of the function grows, the disparity in delay between MEM switches and CMOS diminishes. By implementing complex pass transistor logic, the number of relays needed to implement a function is typically fewer than the number of CMOS transistors so the area overhead is also not as dramatic as it may initially seem. Additionally, since the electrical delay is relatively insignificant, there is no notion or need for tapered gate

were designed in a similar fashion [132].

sizing in relay design so all relays can be minimally sized.

■ 6.3 Energy/Performance of Relay Circuits vs. CMOS

Having made this observation about how logic circuits using relays should be designed, we want to quantify the relative circuit level performance between a MEM relay and CMOS circuit to see what the real trade-offs are. For our initial exploration [26], we chose to compare a 32-bit adder implemented using technology scaled 4-terminal cantilever relays to a CMOS adder optimized for energy-efficiency [21]. The adder is chosen because it is a moderately complex block that is representative of the performance of most random digital logic; conveniently, it is also the core circuit of the CS encoder described in Chapter 5. Both designs are compared for lithographic feature sizes definable by a 90 nm equivalent CMOS technology. To estimate the cantilever relay performance, we developed a simple Verilog-A circuit model based on the electrical and mechanical parameters described in Section 6.1 for simulation in Spectre.

■ 6.3.1 Relay Adder Design

Figure 6-10 shows the design of the MEM relay adder. Adhering to the design philosophies discussed earlier, each full adder cell is designed so that all of the switches nominally turn on/off simultaneously so that only one mechanical delay is incurred.[3] Since the MEM relay can be actuated with either positive or negative polarities, an XOR gate can be implemented with a single switch. This property is exploited to calculate the sum and propagate signals efficiently. The entire cell is designed with both true and complementary signals to avoid the need for inversion during the add, which would incur an additional mechanical delay. The full 32-bit adder is created by cascading each full adder cell in a carry-ripple configuration. The critical path of the 32-bit relay adder is through the cascaded carry chain (similar to a

[3]If the signals all arrive simultaneously, this condition will not occur since the *turn-on* and *turn-off* delays are not the same. However, since relays turn off faster, short circuit conditions should never occur as we are always "breaking" before "making" connections

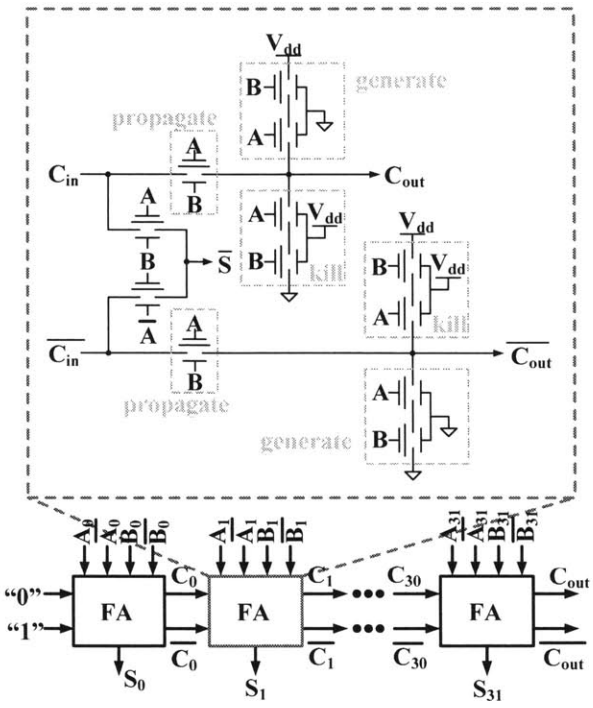


Figure 6-10: Schematic of a 32-bit Manchester carry chain adder implemented using MEM relays. Each full adder (FA) cell is a single differential complex gate.

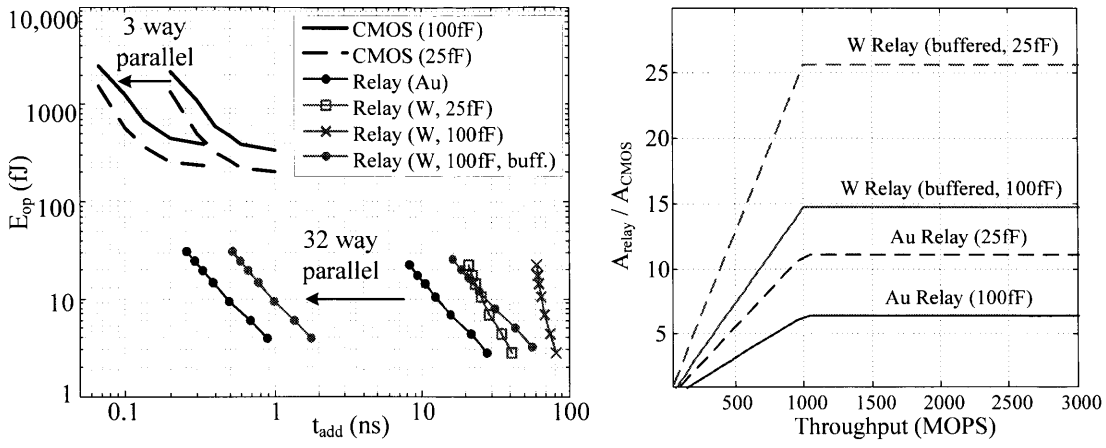
Manchester Carry chain) and can be approximated by:

$$t_{add} = t_{mech} + \frac{32 \cdot 33}{2} R_{on} C_{g,tot} + 32 R_{on} C_L \tag{6.9}$$

where C_L is the output load capacitance on the sum outputs ($\bar{S}$). The delay of the 32-bit adder is still nominally one mechanical delay since the additional electrical delay caused by cascading the full adder cells is fairly insignificant. In the event that the output node is heavily loaded, a buffer relay can be added at the output of the adder such that the resulting delay becomes:

$$t_{add,max} = 2t_{mech} + \frac{32 \cdot 33}{2} R_{on} C_{g,tot} + R_{on} C_L \tag{6.10}$$

This delay is essentially the upper bound on the adder delay for any value of C_L (since we can always add parallel devices at the output).



(a) Energy-delay curves without load energy

(b) Area Overhead

Figure 6-11: (a) Energy-throughput comparison of 32-bit adders after sizing and V_{dd} scaling: a static CMOS Sklansky design [21] versus a cantilever MEM relay adder. (b) Area-throughput tradeoffs in relay adders targeting an E_{op} of 20 fJ in comparison to CMOS adders with 25 fF or 100 fF of load for the same performance (throughput).

■ 6.3.2 Cantilever Adders: Energy vs. Throughput vs. Area

The results of the energy/performance comparison for the cantilever relay adder are plotted in Figure 6-11a where the energy per operation is plotted against the delay of a single add operation for both designs. Both the relay adder and CMOS adder performances are plotted for two different output load scenarios (25 fF and 100 fF). The energy to drive the final output load is not included in the plots, so only the load's effect on delay is captured. We also plot the energy-delay curves for a buffered relay output as described in calculating $t_{add,max}$. It is interesting to note that buffering the output load is more energy efficient for larger loads and at higher voltages (when the relative mechanical delay overhead is smaller).

To gauge the impact of contact resistance at small scales, Figure 6-11a also includes the performance of a relay adder assuming that the contact material is gold (Au); gold is typically regarded as the best contact material for minimizing contact resistance in low force applications [133]. For Figure 6-11a, there is only one plot for the gold contact relay adder since the load capacitance has almost no effect on the total delay. The relative performance penalty from using tungsten over gold contacts is roughly 2X. However, the softness of gold

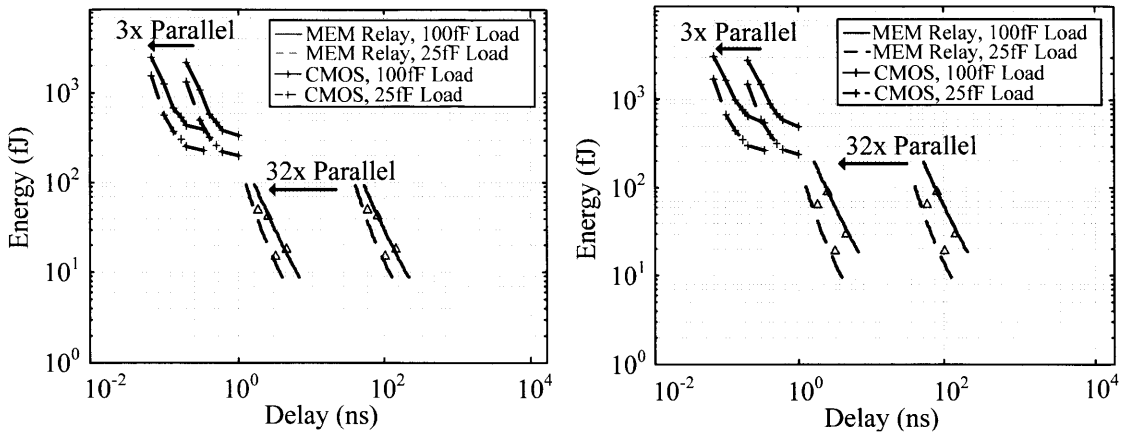
can lead to problems such as stiction and low endurance which makes this penalty relatively small considering the reliability that a harder metal such as tungsten offers.

It is important to note that the final operating point on the CMOS curve represents the minimum energy point of the design, so even if the adder is run any slower, the energy per operation will not decrease. The area of all of the relay adders is estimated as slightly less than 500 μm^2 which is roughly equivalent to the CMOS adder when it is driving a 25 fF load [26]. Thus, for operating frequencies below ~100 MHz, the MEM relay adder will offer nearly an order of magnitude more energy-efficient option at the same performance and area as CMOS.

The energy efficiency offered by such a relay adder could then be extended to higher throughputs by trading off an increase in area to utilize parallelism. This area/performance trade-off is plotted in Figure 6-11b for a constant E_{op} of 20 fJ (~10X improvement over CMOS) while maintaining performance parity with CMOS. As the plots show, the area overhead is bounded at throughputs above the minimum energy point of CMOS, at which point the CMOS adder will need to be parallelized as well to maintain the same energy efficiency. A large portion of the energy gains (~30X) are due to the lower gate and drain/source capacitance per device which result from a larger gate-to-body gap (10 nm vs. 2nm) and lower dielectric (air vs. gate oxide). Also, there are fewer devices required to implement the relay-based adder than for its CMOS counterpart. The remaining energy gains can be attributed to the additional supply scaling that is not possible or is not beneficial for CMOS.

■ 6.3.3 4T Relay Adders: Energy vs. Delay

The energy efficiency results described above provided the impetus for further pursuing MEM relay technology. However, because of reliability and yield issues, the actual relay device that we designed with and that yielded experimental results evolved into the folded spring relay design. Since this design was experimentally more stable, even at larger scale, we are likewise interested in its potential energy efficiency at scaled dimensions. This effort was led by Matt Spencer from UC-Berkeley who refined the Verilog-A circuit model such that it was more stable near contact discontinuities and could be used for larger scale circuit simulations; this



(a) Energy-delay curves without load energy (b) Energy-delay curves including load energy

Figure 6-12: Energy-throughput comparison of an optimized CMOS 32-bit adder versus a scaled MEM relay 32-bit adder in a 90 nm lithographic process [22].

model was then applied with a similar set of experiments as used on the cantilevers only using a scaled 4T folded spring relay instead [22]. The results of this study are plotted in Figure 6-12 which shows the energy performance of the scaled relay both with and without the load energy included. As the plots show, we see a similar energy performance trade-off and still order of magnitude gains in energy efficiency at lower supply voltages. The speed and energy of the scaled folded spring relay is not quite as good as predicted by the cantilever design since there is overhead (energy) for the thicker center plate and additional spring legs. However, the comparison between the projected cantilever and folded spring design is a good indicator of the reliability performance trade-off in the device design.

■ 6.4 Relay Circuits for Sensors

In the last section we showed that MEM relays have the potential to improve on the energy efficiency of current CMOS logic designs. Since the origin of these gains come from decreased capacitance, supply scaling and the absence of leakage, many of these advantages extend beyond datapath functions and should have similar benefits in memory, and mixed-signal components as well [26]. In this section we will discuss relay based designs for some of these

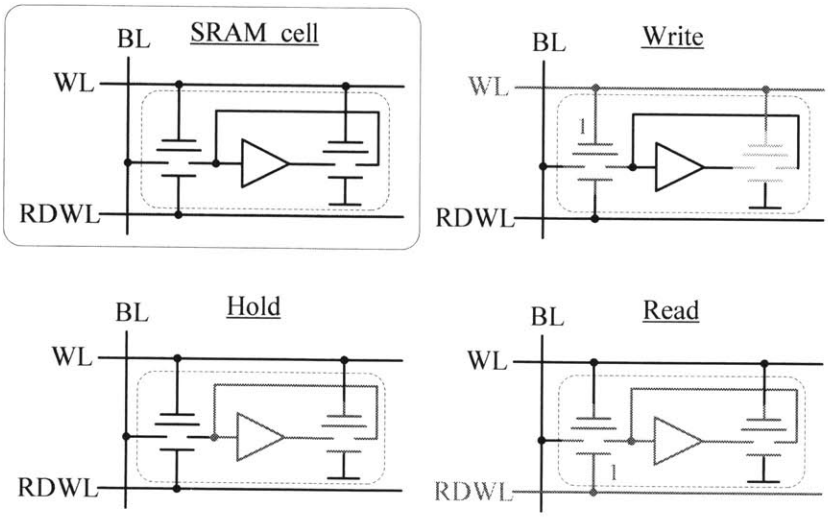


Figure 6-13: A 4 relay (4R) SRAM cell showing each of the legal operating states.

circuit sub-systems that are relevant to sensor nodes.

■ 6.4.1 Relay Memories

As we have mentioned, one of the primary limitations to sensor lifetime is the idle leakage current in memories, and in particular SRAMs. SRAMs often need to retain their data based on requirements outside of their own circuit sub-system so operating at the minimum energy point or periodically shutting down are often not available leakage reduction options [113]. In this regards, MEM relays present a nearly perfect solution. However, the typical 6T SRAM is a circuit not conducive to relay design. This is because an SRAM cell works on the principle of ratioed logic where internal transistors in the memory need to be overpowered in order to write to the cell. Although in principle relay based designs could implement a similar circuit (by placing many relays in parallel), this would then require the variation of R_{on} for relays to be well controlled, which works against many of the advantages of relays that we have discussed so far.

Instead, for relays we can implement memory by taking advantage of the fact that we can implement non-inverting logic and make use of having two control terminals (gate and body). The resulting example of a relay SRAM cell is shown in Figure 6-13. The basic

SRAM cell is essentially a latch, only the cross-coupled inverters are replaced with a single relay buffer. In the example shown, the bit-line (BL) is shared between read (RDWL) and write (WL) word-lines to maximize density. The body terminal of the access relay is used to separately enable reads from writes. To enable simultaneous read/write access, an additional access relay placed at the output of the buffer and a separate read bit-line would be needed. Besides the fact that there is no static current path in this circuit, relay memories will not need the equivalent of a sense amp and in general, the power consumed by the memory will be entirely proportional to the access rate. The read delay of the SRAM is one mechanical delay while the write delay, to insure latched data, is two mechanical delays (one for the access relay and one for the buffer). Perhaps the major drawback when compared to CMOS is the density of the relay memory array which has 2/3 the number of devices but each device is ~10X the size of the transistor. However, as mentioned, this may be offset somewhat by a reduction in support and peripheral circuitry. Nonetheless, the leakage associated with whatever amount of CMOS SRAM that is replaced will be eliminated by moving to a MEM relay based memory.

■ 6.4.2 Relay DAC

In addition to memory and processing units, sensor nodes will still need a way to interface to analog inputs/outputs (I/O). Given the abrupt switching behavior of relays, the implementation of efficient analog and mixed-signal building blocks critical to I/Os is significantly less clear. Many of the ADC and digital-to-analog converter (DAC) architectures used in CMOS rely on the ability to operate transistors in their saturation region, providing high output impedance and the ability to generate relatively linear voltage gain with low drain-source voltage. For MEM relays, there is no such mode of operation and thus the challenge is to build mixed-mode circuits using these relays despite their ineffectiveness in performing traditional analog processing.

Recently even in CMOS there has been a transition to largely "digital" analog processing blocks to leverage the scalability of digital circuits [134, 135]. We leverage some of these design trends for relay circuits. Figure 6-14 shows one implementation of a DAC inspired

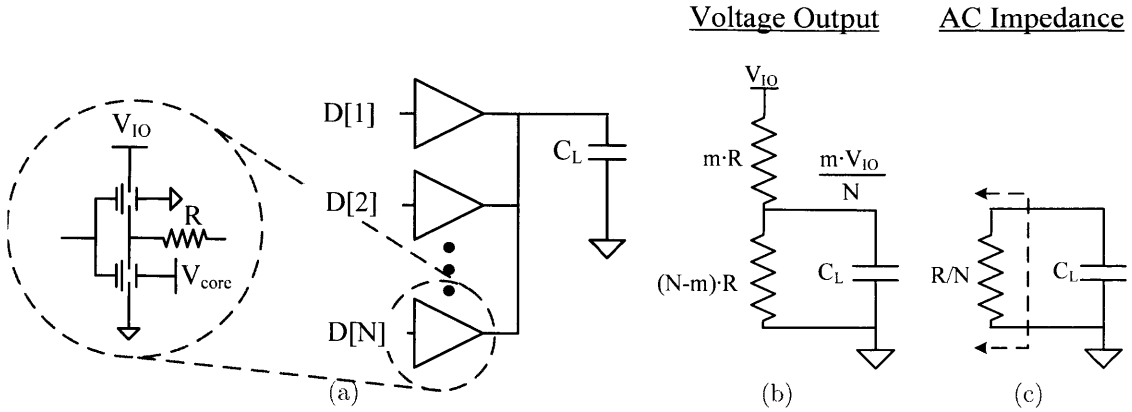


Figure 6-14: DAC topology, schematic and equivalent DC and AC circuits.

by an existing CMOS voltage mode driver [136]. In this example, each inverter/driver is driven by a thermometer encoded input where $N = 2k - 1$ and k is the bit resolution of the DAC. Each inverter is composed of a MEM relay-based buffer followed by a resistor; the resistor is necessary to provide both a constant controlled termination (R/N) and a means for intermediate voltage generation. It is also important to observe that the inputs of the relay DAC can operate at a core voltage that is independent of the I/O voltage, so in this respect it also acts as a level shifter.

Assuming that the contact resistance of the relays is unpredictable and varies over a wide range, for the DAC to operate as described, R should be at least an order of magnitude greater than the worst case expected contact resistance. For gold-based contacts this does not present a stringent constraint, but for tungsten-based contacts in a scaled technology, R would have to be at least 10 kΩ to ensure at worst a 10% error in any single DAC element. If we assume that the termination resistance is specified by a required output bandwidth (BW) for a given output load (C_L), then the power to drive the output is independent of the DACs resolution:

$$P_{IO} = \frac{V_{IO}^2 N}{4R} = \frac{V_{IO}^2}{4} BW \cdot C_L, \quad BW = \frac{N}{RC_L} \tag{6.11}$$

For example, a DAC with a 1 pF load, a voltage swing (V_{IO}) of 200 mV, and a desired

bandwidth of 1 GHz will require about 62.8 μW of power. Similarly, a 50 Ω termination will cost 200 μW. Assuming that we always design the output bandwidth to our desired signaling rate, or else use parallelism to signal on the channel up to its bandwidth, the energy per bit simply due to driving the output is:

$$E_{bit} = \frac{\pi}{2} V_{IO}^2 C_L \tag{6.12}$$

For the example given above, this translates to 62.8 fJ/bit which is more than an order of magnitude smaller than the output driver power reported in [137]. However, this only accounts for the load power, which generally dominates the DAC power, but even if we include switch power, than the worst case total DAC power is:

$$P_{DAC} = P_{IO} + NC_g V_{core}^2 f_s \tag{6.13}$$

where f_s is the signaling frequency (ideally this would be linked to BW). The power of switching the DAC itself exceeds the output load power when $NC_g > (V_{IO}/V_{core})^2 \cdot (\pi/2) C_L$ assuming that $BW = 2\pi f_s$. When $V_{IO} = V_{core}$, N is on the order of 1000 or more (~10 bits) for a load capacitance of 1 pF and scaled relay gate capacitance of ~2 fF. So even in this scenario, the relay DAC power is still 5X less than the CMOS transmitter.

■ 6.4.3 Relay-based Flash ADC

As we saw in the CS digital encoder design (Figure 4-6), we were limited by leakage at low sampling rates while at higher sampling frequencies we were limited by the ADC. If we adopt a MEM relay based digital backend, then we will no longer be limited by leakage and ADC power will become the limiting component. As we have shown so far, the energy-efficiency of relay-based circuits is maximized by topologies that leverage parallelism. Relays themselves do not provide any linear gain so architectures such as pipelined ADCs would need to rely on passive components to provide the gain and would also have a latency proportional to the number of bits times the mechanical delay. Meanwhile, any iterative or oversampling

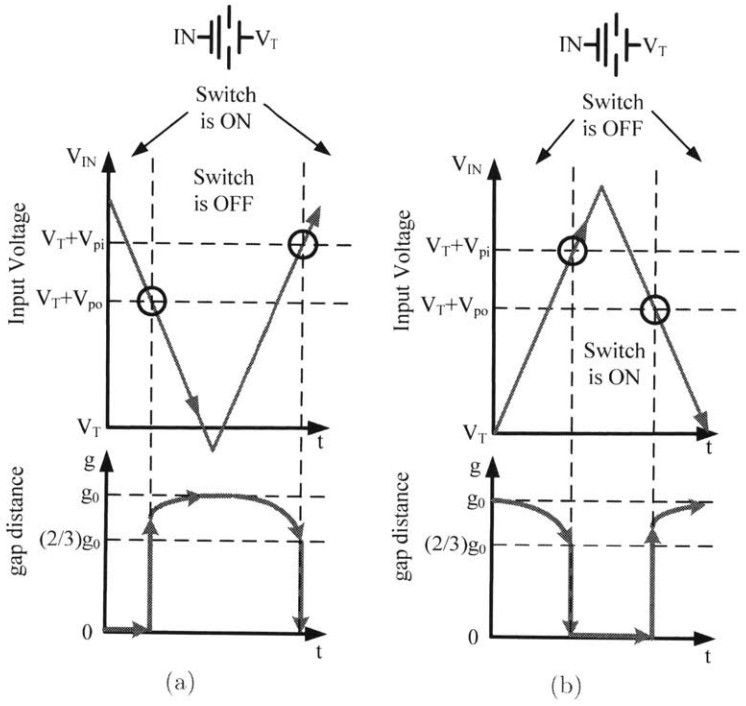


Figure 6-15: Relay comparator hysteresis when the relay is initially (a) on and (b) off.

converters will incur the mechanical delay of the relays many times over so their sampling frequencies will be limited. Consequently, the ADC architecture that is most amenable to relays is the Flash ADC topology.

Relay Comparators

As we have seen, MEM relays act as very good switches, and more importantly for a Flash ADC, the switching threshold is adjustable via the body terminal. Thus, the MEM relay lends itself to being a good comparator. However, two distinct characteristics of relays that must be kept in mind when designing a relay-based Flash ADC are:

1. MEM relays exhibit hysteresis with a larger turn-on threshold (V_{pi}) than turn-off threshold (V_{po}).
2. The relay will be actuated in response to voltage inputs that are either above *or* below its body voltage.

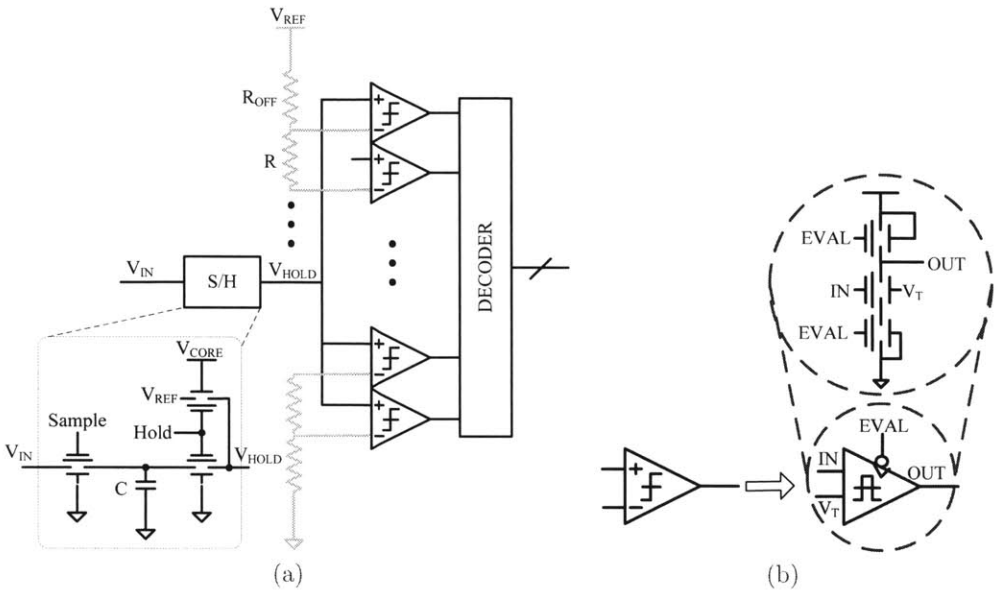


Figure 6-16: ADC block diagram and comparator schematic.

The effect of the switching hysteresis is shown in Figure 6-15. The difference in switching thresholds means that the state of the comparators *before* making a comparison must be the same across all comparators (i.e. we want to always use V_{pi} or V_{po} as our threshold) in order to get non-data-dependent results. If we want to use V_{pi} as the switching threshold, that means that all of the comparators in the comparator bank must be simultaneously turned off at some point. This essentially limits the input voltage range to be $2V_{po}$ at best (such that there is a voltage at which all comparators will be off), which in scaled processes will be quite small. Alternatively, if we use V_{po} as the threshold, then we just need to find a voltage where all comparators are turned on (e.g. $V_{T,max} + V_{pi}$). Since the latter option is more flexible and easier to guarantee, we chose to use that for our design.

In order to enable this state, we must pre-charge the input to the comparator bank to some known voltage (that will close all the switches). The resulting block diagram of the ADC to implement this function is shown in Figure 6-16a along with the schematic for the sample-and-hold (S/H) circuit and comparator. The converter consists of a bank of comparators that are each a dynamic inverter (or buffer) with varying body-bias thresholds generated by a resistor string. During the sample phase, the comparator bank is reset and the

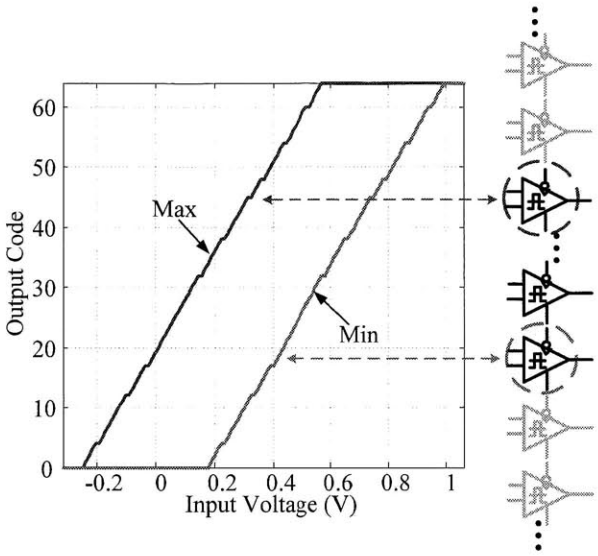


Figure 6-17: Simulated ADC output code vs. Input Voltage.

outputs of each comparator are pre-charged high. For the comparison phase, the Hold switch is enabled and the pull-up path of the comparators is also enabled. All of the comparators whose threshold is within one pull-out voltage of the input will remain off and keep their output nodes high.[4]

The ambipolar characteristic of relays presents a decoding challenge as it implies that the response of relay-based comparators to the input voltage will be non-monotonic; these comparators actually act as absolute value comparators. This means that for a given input voltage, there will be a band of comparators that are left off while comparators both above and below will be on. A 6-bit ADC was simulated in Spectre using the Verilog-A models mentioned earlier and the output of the ADC is plotted in Figure 6-17. The comparator outputs (and equivalent code) corresponding to the highest and lowest threshold comparator to stay off are plotted. The actual decoder can choose either curve as the output.

The energy consumed by this ADC design is dominated by the reference generation. For the example given, where $V_{CORE} = 300$ mV, $V_{REF} = 1$ V, $C = 500$ fF, $R = 4$ kΩ, and

[4]To enable performance optimization of the ADC, a separate signal is used for Samp, Hold, and Eval to allow separate control of timing to account for differences between the mechanical turn-on and turn-off delays.

R_{OFF} = 74 kΩ, the energy consumption in a single cycle is ~350 fJ, out of which 320 fJ is dissipated by the reference supply for threshold generation and reset. This translates to 5.5 fJ/conversion step for a 6-bit 10 MS/s converter, which is equivalent to the best reported modern CMOS based converter [138]. The energy for this converter, which is quadratically dependent on supply voltage, can be further reduced if the input dynamic range is reduced by scaling down V_{REF} or if the resistors in the reference ladder can be increased (smaller comparator kickback). In a more practical design, additional redundancy in the comparators will likely have to be added to deal with variation in a manner similar to [139]. Fortunately, additional comparators (for either redundancy or more resolution) in the converter comes at a relatively low penalty in energy since much of the energy is consumed in the reference generation.

■ 6.5 Summary

In this section we introduced and proposed the use of MEM relays as a fundamental switch technology for VLSI circuit applications. The primary motivation for exploring the use of MEM relays was because of their immeasurably low off-state leakage and nearly ideal switching transition between on and off states. MEMS benefit from scaling in the same way that CMOS does, but despite this their device switching speeds are still orders of magnitude slower than in CMOS. However, in this section, we showed how relay optimized circuit design can narrow the performance gap with CMOS by over an order of magnitude at the circuit and architecture levels. Additionally, we confirmed for moderate throughputs (<10-100 MOPS) that relay circuits offer an order of magnitude improvement in energy-efficiency for logic and for other circuit sub-systems as well. Here we summarize the following relay circuit design insights and guidelines that were developed:

Mechanical vs. Electrical Delay

1. MEM relay circuits should be designed as single complex gates (ala pass transistor logic) when possible. The electrical delay is orders of magnitude smaller than the mechanical delay allowing for many stacked devices with little delay penalty.

2. For logic, relay circuits only require a single minimum sized relay as a building block since electrical delay has little impact on performance, so there is no equivalent to gate sizing as in CMOS.
3. For large output loads, only a single stage buffer is necessary to reduce the effect on longer device stacks and improve the energy efficiency of the circuit.

Source/Drain Independent Actuation

1. Independence between the control terminals (gate/body) and output terminals (source/drain) enables both inverting and non-inverting logic gates to be implemented–minimizing the number of stages/relays to implement a logic function.
2. Independence between gate/body and source/drain also means that relays can be used as natural level shifting element as seen with the DAC.

Hysteretic and Ambipolar actuation

1. Ambipolar actuation enables the construction of both PMOS and NMOS-like relay equivalents.
2. Relays have different turn-on and turn-off thresholds and delays. Fortunately for logic, the turn-off delay is always shorter such that circuits will "break" before "make".
3. Relays act as absolute value comparators, so when used in circuits that require comparison, a single relay output may not give you sufficient information to tell you where the input is.
4. Hysteresis in the relay's switching characteristics also requires that the state of the relay is known before being used as a comparator, otherwise the threshold used for comparison will be unknown. Dynamic circuit logic styles are well suited for this as they allow for relays to be reset to a known state without prematurely evaluating the output.

CHAPTER 7

MEM Relay Design for Future Integrated Circuits

The potential of MEM relay technology presented in the previous section is attractive for energy constrained applications. However, the benefits need to be demonstrated and the device technology is still not yet mature. In this chapter we discuss some of the steps taken towards validating the design premises as well as refining the technology development to improve future relay circuit performance.

■ 7.1 Experimental Results

To verify the initial concept behind relay based integrated circuits, a test vehicle comprised of various small to medium sized functional blocks implemented using only relays was designed and fabricated [23]. The chip fabrication was done in a 1 μm lithographic process at the UC-Berkeley Microlab by Rhesa Nathanael. The design, test and layout of the circuits was a collaborative effort between students from MIT, UC-Berkeley and UCLA.[1]

The layout of the test chip (named CLICKR1) is shown in Figure 7-1 which contains logic, sequential memory elements, memory arrays, I/O and clocking circuits. The dimensions of

[1]The students besides myself that were involved were: Hossein Fariborzi (MIT), Matt Spencer (UC-Berkeley), and Chengcheng Wang (UCLA)

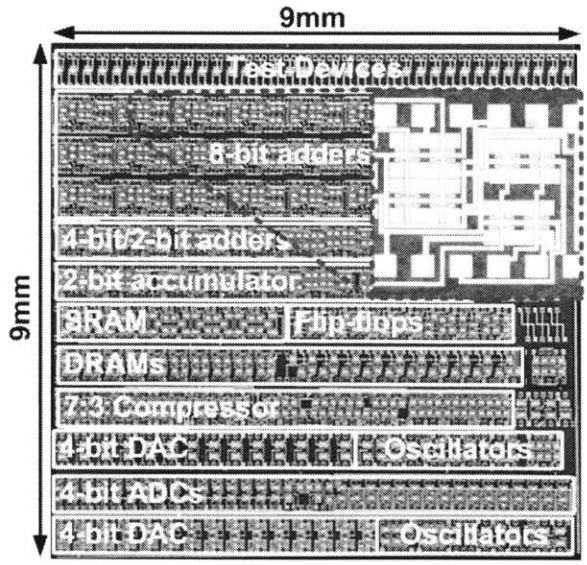


Figure 7-1: Die photo and circuit layout of the 9 mm × 9 mm CLICKR1 test chip [23].

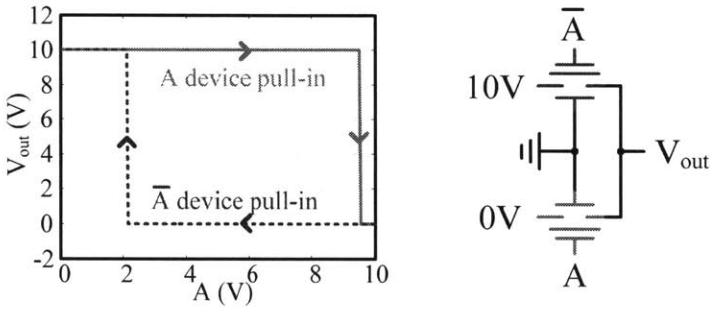


Figure 7-2: Measured VTC of a MEM relay inverter.

the relays on this test chip are the same as shown in Figure 6-2 and Table 6.1. Despite the large dimensions, some of the underlying design principles, such as the disparity in mechanical and electrical time constants, can still be demonstrated. Furthermore, building confidence in the device and circuit modeling infrastructure and demonstrating circuit functionality is an important step towards larger scale integration.

■ 7.1.1 DC Transfer Characteristic

One of the necessary requirements to enable cascaded levels of digital logic is that a device must be able to drive itself. Figure 7-2 shows the circuit and measured voltage transfer

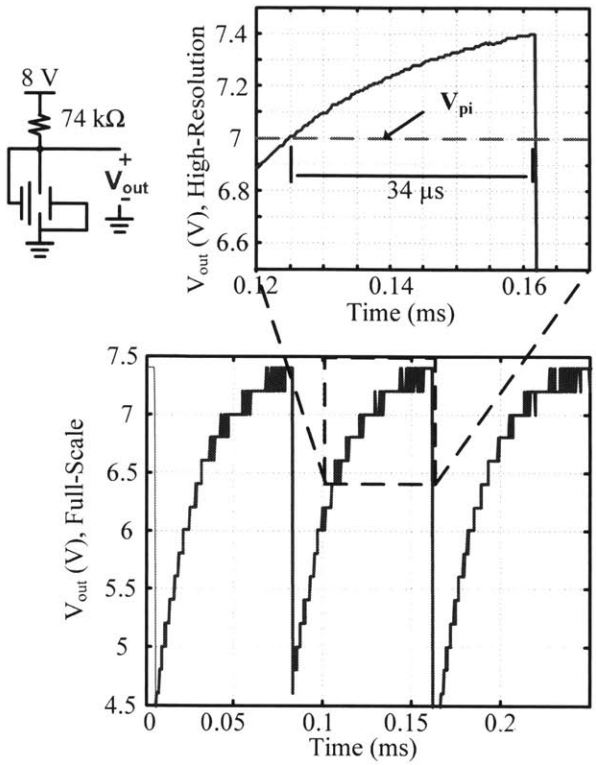
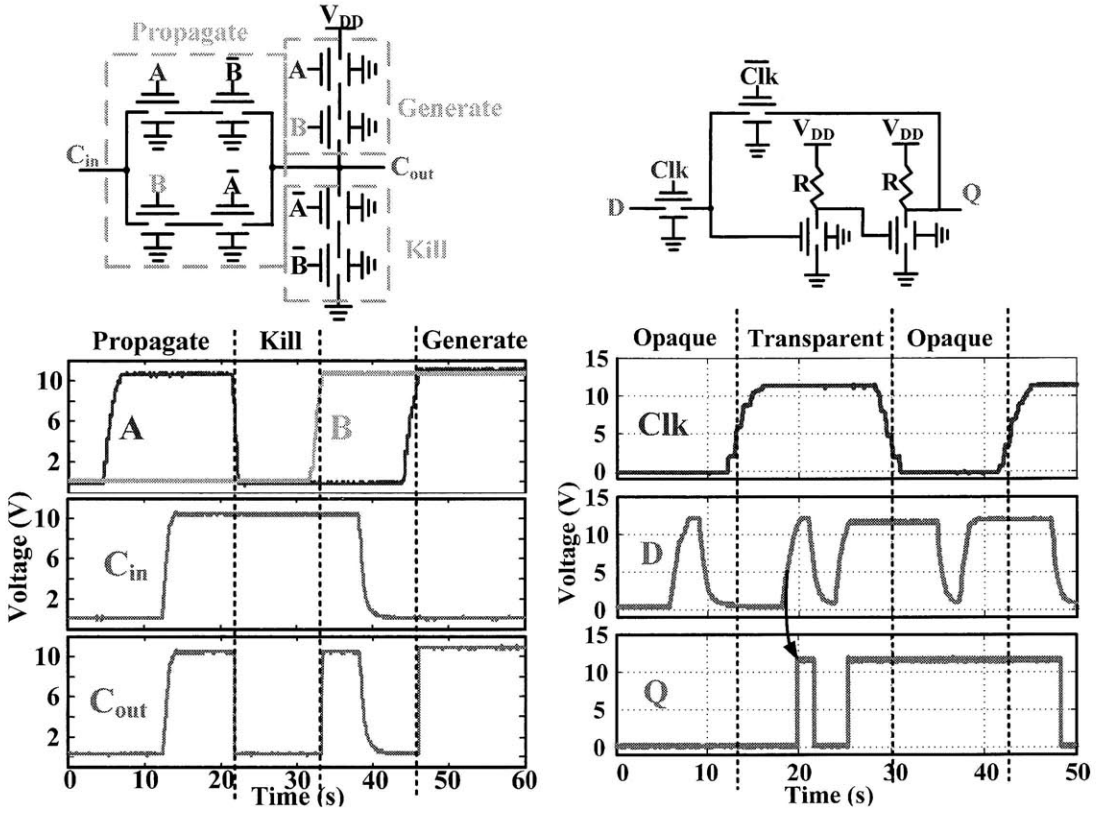


Figure 7-3: Measured output of a single relay pseudo-NMOS style oscillator.

characteristic (VTC) of a pass-gate style inverter/XOR circuit that demonstrates this property. In the plot, the $\bar{A}$ input is not shown but is swept as the complement of the A input (10 V - A). As the VTC shows, the device is capable of driving the necessary output voltages to actuate/de-actuate a copy of itself.

■ 7.1.2 Mechanical Delay vs. Electrical Delay

In Section 6.2.2, we discussed leveraging the difference between mechanical and electrical delays in motivating our choice of circuit topology. In Figure 7-3, a single relay pseudo-NMOS style oscillator demonstrates the difference. When the relay is off, the output rises at a rate set by the RC time constant of the 74 kΩ external load resistor and the capacitance due to the test infrastructure which is estimated to be about 55 pF. As shown in the figure inset, the mechanical delay can be extracted by monitoring the voltage around V_{pi} (previously characterized with a DC analyzer) to see when the relay switches on. The nature of this



(a) MEM relay PGK circuit.

(b) MEM relay latch circuit.

Figure 7-4: Measured functionality of (a) a PGK circuit and (b) a pseudo-NMOS style latch.

particular oscillator experiment is that the gate overdrive (V_{dd}/V_{pi}) when the switch turns on will always be small which means that the measured delay will be sensitive to changes in the overdrive voltage. This caused the delay measurements to vary (~25-35 μs) over successive measurements. When the switch does turn on, the output drops abruptly as the time constant of the falling edge is set by the relay's *on*-resistance (R_{on}) and the same external load capacitance. Based on the falling edge rate, the R_{on} is estimated around 1 kΩ which would result in an electrical time constant of ~300 ps for a self loaded (gate + parasitic) capacitance of 300 fF.

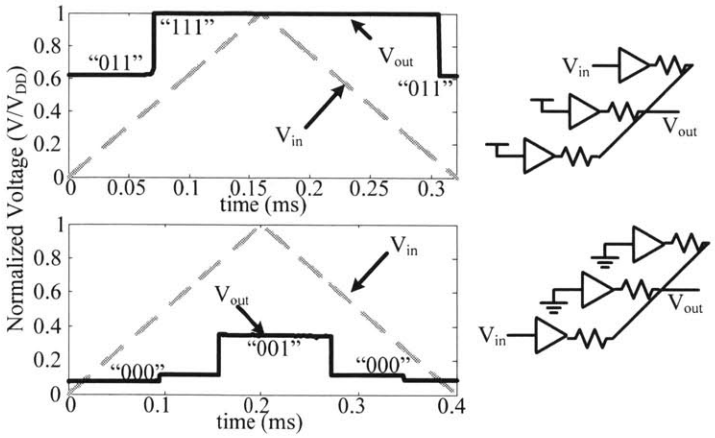


Figure 7-5: Measured functionality of a 2-bit DAC.

■ 7.1.3 Circuit Demonstrations

Earlier, the design example used an adder for analysis where the heart of the adder circuit is the propagate-generate-kill (PGK) circuit. Figure 7-4 shows the measured functionality of an "NMOS" only variation of a PGK circuit along with the functional demonstration of a pseudo-NMOS style latch. In the first iteration of devices and circuits, the circuit topology used for these experimental circuits were not identical to what was discussed in Section 6.2; only NMOS style devices were used due to some device and test infrastructure limitations that will be discussed shortly.

In addition to the PGK and latch, a small scale DRAM and DAC were also demonstrated in [23] and shown in Figure 7-5 and 7-6. The 2-bit experimental DAC shows the 4 output states corresponding to the thermometer encoded levels for the circuit described in Section 6.4. The 10-bit DRAM is configured like a NAND Flash such that the read delay out of the memory is determined by $t_{d,off}$ of the relay. The bypass path in all of the inactive rows of the memory are enabled during a read such that the bit-line is only pulled high if the active row cell holds a "1". The experimental results show a simultaneous read and write operation; this was enabled by replacing the pre-charge relay with a 100 kΩ external pull-down resistor. Subsequent test chips by the team have also demonstrated the full adder [22], relays in a CMOS power gating application [140], and multiplier blocks [141].

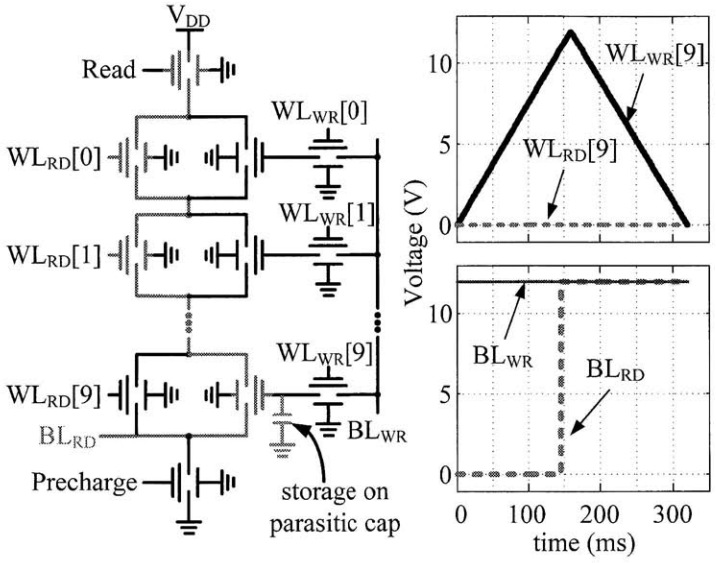


Figure 7-6: Measured functionality of a 10-bit DRAM.

■ 7.2 Circuit Driven Relay Design

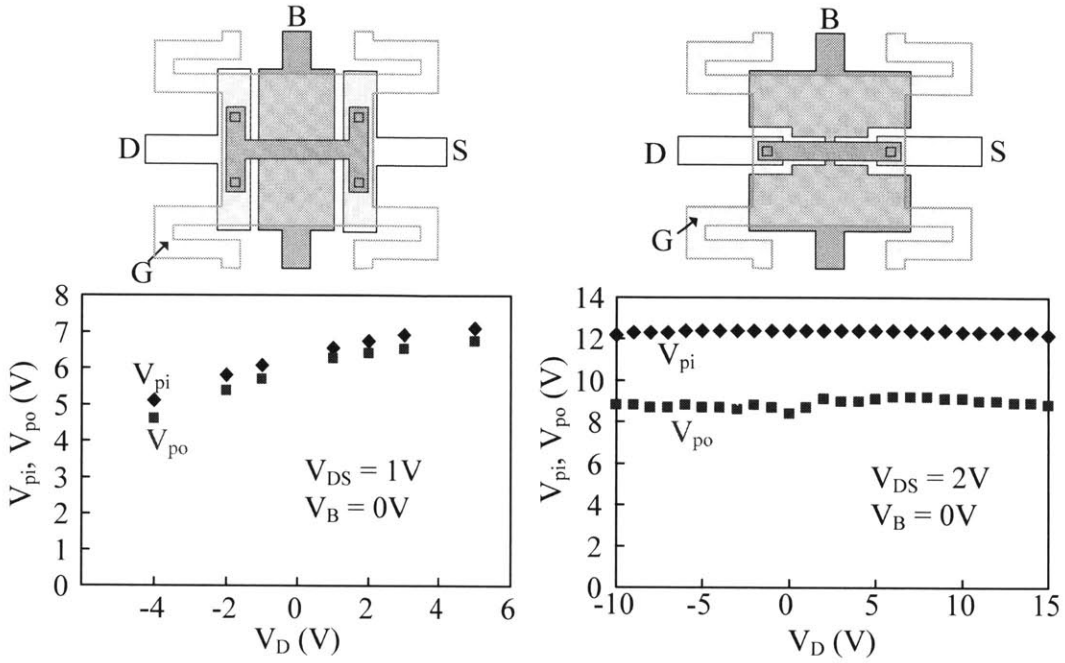
An important part of developing any technology is understanding the needs of the application. In the case of MEM relays, this entails understanding the impact of device parameters on circuit performance and functionality. In this section, we discuss some of the circuit insights, both analytic and experimental, that have been used to drive different aspects of the relay design.

■ 7.2.1 Device Layout

As mentioned earlier in Section 7.1, the experimental circuits on the CLICKR1 test chip were limited to NMOS-like circuit topologies. There were a few reasons for this limitation, all of which were device layout related.

V_{gb} vs. V_{bg}

First, it was observed that there was a significant discrepancy between pull-in voltages when the body was grounded and a gate voltage was applied versus when the opposite was performed. In the case when the voltage is applied at the body terminal, the required applied voltage to close the switch was much higher (>50%). Looking at the original CLICKR1



(a) CLICKR1 relay design.

(b) Revised relay design.

Figure 7-7: Measured V_{pi} and V_{po} versus drain voltage for (a) the original CLICKR1 device and (b) a revised, lower parasitcs relay (CLICKR2). Data courtesy of Rhesa Nathanael and Jaeseok Jeon.

device layout (Figure 6-2,7-7a), this can be attributed to the discrepancy in electrostatic force (F_{elec}) generated by just the body terminal to the gate versus the F_{elec} generated by the gate structure (including spring arms) to the body terminal and the substrate. Consequently, circuits that used the body terminal as an input port, such as the XOR in the full adder circuit (Figure 6-10), had pull-in voltages that were too high to be used.

Source/Drain Overlap

In the original device layout shown in Figure 7-7a, there is a relatively significant gate-drain and gate-source overlap area as well. When the body is at a fixed bias, this causes a drain/source dependent pull-in and pull-out voltage as shown in Figure 7-7a. As a result of this, PMOS-like relay devices would have different pull-in/pull-out voltages compared to their NMOS counterparts for the same source/drain voltages. In the subsequent device design shown in Figure 7-7b, the gate-drain and gate-source overlap areas were minimized

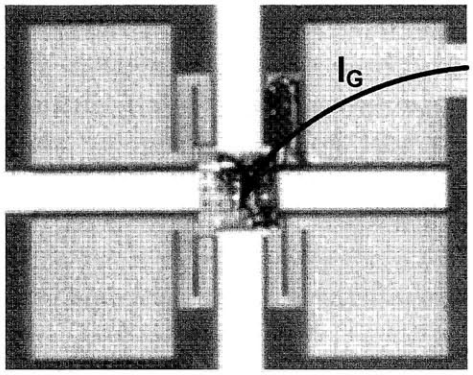


Figure 7-8: Example of a device failure from excessive gate current.

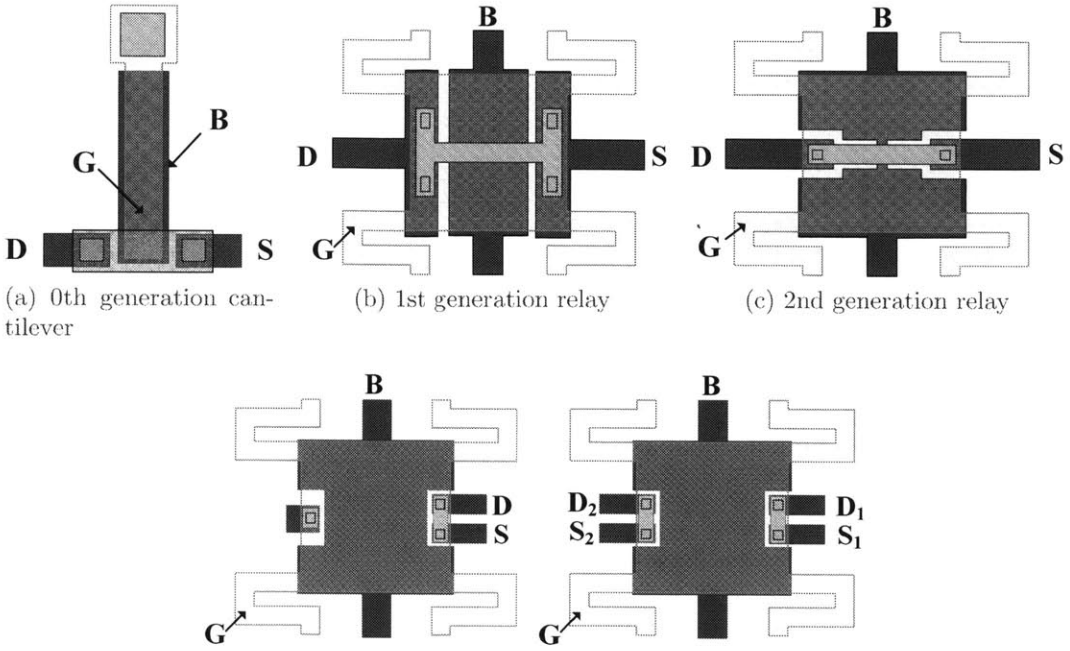
to resolve this discrepancy.

Channel Area

As was shown in Table 6.2, C_g is dominated by gate-to-channel capacitance since the gate oxide is relatively thin and has a high dielectric constant. So clearly reducing the channel footprint under the gate will have energy benefits. However, there are functional reasons to minimize the channel area as well. When the CLICKR1 device was being used in a source follower configuration (NMOS pulling up or PMOS pulling down), the channel-to-body coupling in addition to C_{gd} and C_{gs} were sufficient to keep the device actuated even after V_{gb} was reduced to 0. Consequently, the other major device revision shown in Figure 7-7b is to minimize the channel area and the channel to body overlap area.

Gate Current

One possible way to have overcome some of the layout problems initially would have been to increase the supply voltage. However, that has its own drawbacks as excessive gate voltage can cause large gate tunneling currents that can literally burn up the device, as shown in Figure 7-8. To help alleviate the potential for this sort of failure, later designs that have access to a via technology would tie all of the gate's anchor points together below the device such that there would be a distributed path for current.



(a) 0th generation cantilever

(b) 1st generation relay

(c) 2nd generation relay

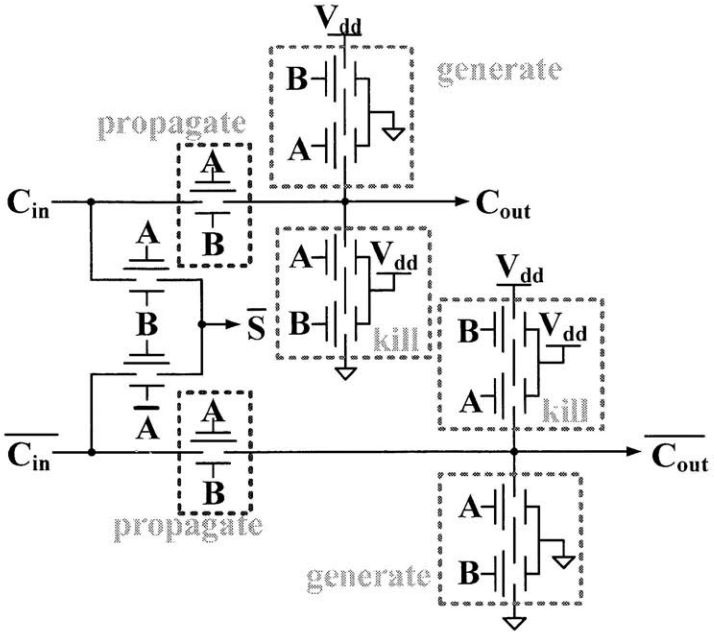
(d) Initial 3rd generation relay

(e) Implemented 3rd generation (6T) relay

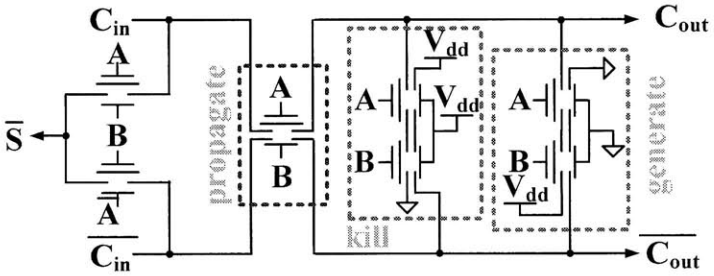
Figure 7-9: MEM relay device evolution.

■ 7.2.2 **6T Relays**

Figure 7-9 summarizes the evolution of the MEM relays designed for use in logic applications, including the original cantilever design discussed in [26]. In the most recent generations, the device in Figure 7-9d was proposed by the device designers as a way to minimize the channel area. The single dimple on the left was a mechanical stopper to prevent the device from tilting when landed. While developing the automated layout tool to generate this device, we realized that the stopper could be replaced by a second drain/source pair and that this would add functionality. The resulting 6-terminal (6T) relay device is shown in Figure 7-9e. The 6T relay halves the number of relays required to implement functions that have common control signals but different drain/source connections. For the circuit styles used in relays, this sort of structure occurs repeatedly. For example, Figure 7-10 shows the full adder before and after using the 6T device. In most cases the 6T relay halves the area cost and the gate



(a) Full adder using 4T devices (12 relays)



(b) Full adder using 6T devices (7 relays)

Figure 7-10: 6T relay based adder: halves the number of relays in the PGK circuit.

switching energy to implement a given function when compared to 4T designs. Obviously extensions to this idea can be made to include additional drain source pairs or to design the device to implement logic mechanically, but the basic takeaway is that there is room for additional gains if the devices and circuits can be jointly designed.

■ 7.3 Compressed Sensing MEM Relay Circuits

As a next step towards pushing the scale of MEM relays, we have designed MEM relay based circuits to implement the CS encoder blocks that incorporate all of the previous design and

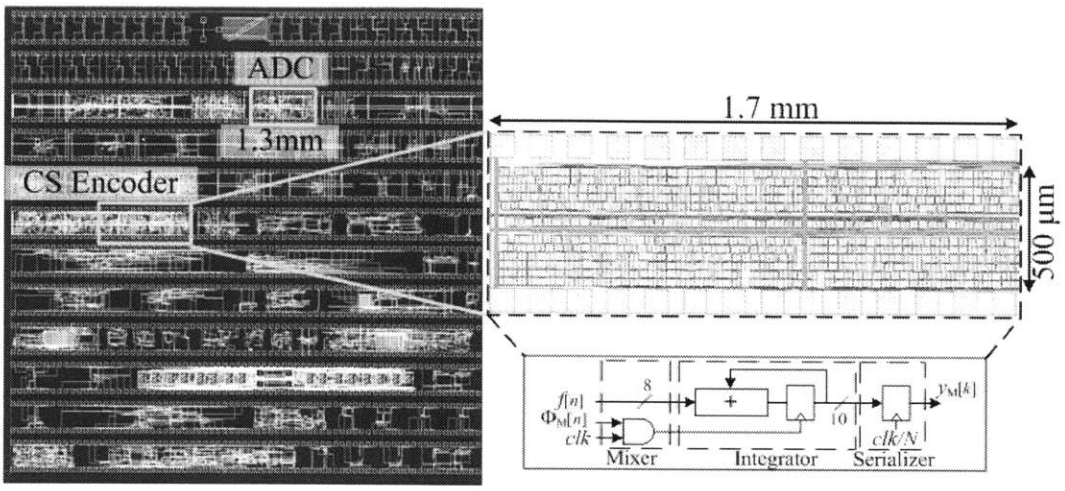


Figure 7-11: Layout of one CS measurement slice using the 6T relays in a 0.25 μm process testchip.

experimental feedback. For this design, we use the 6T relay design in a 0.25 μm lithographic process with device dimensions corresponding to W_A=7.5 μm, L_A=7.5 μm, W=1 μm, L=5 μm, and g=100 nm. The footprint of the new scaled device occupies 20×20 μm as opposed to the 120×150 μm footprint of the original device (including anchors).

CS Encoder Slice

As a reminder, Figure 7-11 shows one 10-bit measurement slice in the encoder. The test chip implements several measurement slices of the CS digital encoder using the scaled 6T MEM relays. The encoder slice is essentially just an accumulator so the 6T adder shown in Figure 7-10 is used along with a simple MSFF. Figure 7-11 also shows the corresponding layout for one measurement slice. In this most recent design iteration, we have adapted the existing CMOS place and route (P&R) tools to layout the MEM relay design. Again, since our performance is a weak function of electrical delay and parasitics, the automated tools do not yet need to be timing driven. The matrix is simulated using the Verilog-A model developed by Matt Spencer and estimates the power for a 50 measurement 10-bit array running at 20 kHz to be about 400 nW running off of a 2 V supply. To put this in perspective, this is already 5X lower than our CMOS test chip in a 90 nm CMOS process running off of a 0.6V supply. If the MEMS device and operating voltage can be further

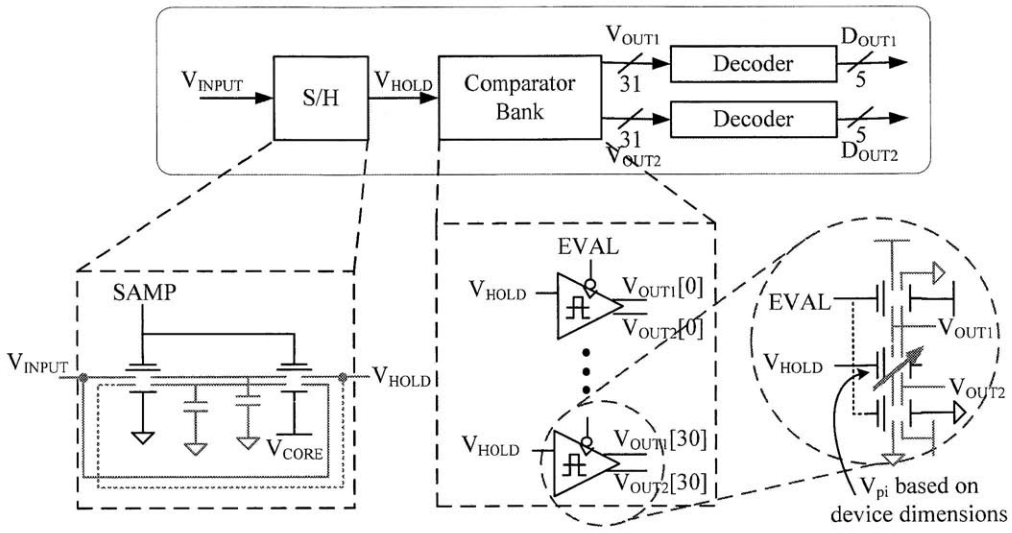


Figure 7-12: Schematic of the 5-bit 6T reference-less DDR Flash ADC.

scaled (e.g. W=100 nm, etc...) then we would expect to see another order of magnitude or two of energy improvement, which directly translates into battery lifetime.

5-bit Reference-less DDR ADC

In an effort to demonstrate the overall efficiency of the sensor node, we also implemented a 5-bit version of the Flash ADC architecture described earlier. However, a couple of key design changes have been made for this implementation. Figure 7-12 shows the block diagram and component schematics for the ADC. In the sample and hold block, we leverage the 6T device to enable sampling of the input on both edges of the sample clock with no device overhead. Along the same lines of thinking, the comparator cell is also enabled for double data rate (DDR) operation where the second set of drain/source pairs are used to pre-charge and evaluate on the negative edge of EVAL. The other major design change is that there is no resistor string to generate the reference levels for the comparators. Instead, we leverage the fact that device layout in MEM relays determine the relay's switching thresholds. So to create the different output levels the comparator bank is composed of relays with progressively sized spring lengths where the V_{pi} and V_{po} of each comparator is linearly skewed from the next. Thus, there is no static current in the ADC design at all. Again, a practical implementation of this design will likely require some comparator redundancy and energy overhead to manage

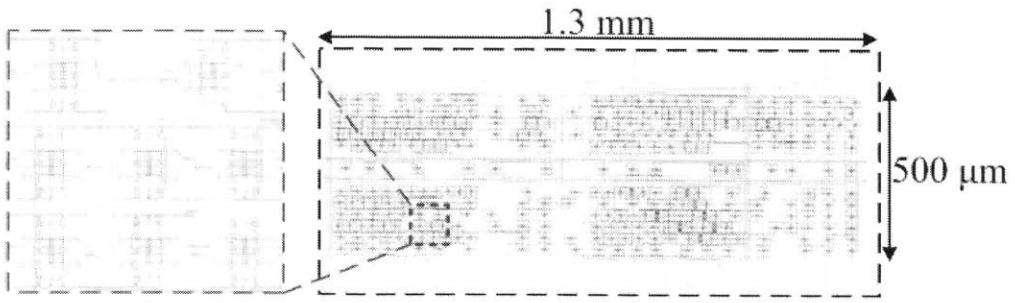


Figure 7-13: Layout of the 5-bit 6T reference-less DDR Flash ADC.

variations in V_{pi} analogous to those described in [139].

Two separate decoders, which are essentially large multiplexor trees, are used at the output to capture both conversion streams. To estimate the performance, we again use the Verilog-A model; the results of simulations show that for a V_{dd}=2 V and f_s=20 kS/s, the power in the ADC is roughly 45 nW. For 5-bits of resolution this results in an FOM of 70 fJ/conversion step. Again, this is already on par with state-of-the-art ADCs in CMOS but at a minimum device size of 1 μm and supply of 2 V, where the energy gains with respect to CMOS are expected to improve continually with additional device scaling. For example, even just a 4X dimensional scaling of the device would result in roughly 64X energy savings putting the FOM in the 1 fJ/conversion step range. Furthermore, since there is no source of DC energy consumption, this FOM will essentially be preserved as the frequency is scaled down. The layout of the ADC design is shown in Figure 7-13. Since there is no reliance on parasitics, ratioed logic, or electrical delays, the ADC was also placed and routed automatically which highlights a convenience in regards to mixed-signal design with MEM relays.

■ 7.4 MEM Relay Summary

What we have shown in the last two chapters is that MEM relays offer a unique combination of ideal switching characteristics in a CMOS compatible fabrication process. We have shown that the speed deficiencies of MEM relays can be partially mitigated at the circuit

design hierarchy by tailoring the design methodologies to suit the characteristics of relays. Meanwhile, projections on relay-based circuits show the potential for greater than 10X improvements over what CMOS can achieve. As we have discussed, early experimental results have provided invaluable feedback towards refining the relay device design which has also led to added functionality. Even at the scales of recently fabricated test chips, the energy benefits for applications such as WSNs where energy, and not performance, is the bottleneck could be relevant immediately.

■ 7.4.1 Challenges

However, there are clearly challenges that still need to be addressed before MEM relays can become widely adopted. It goes without saying that all of these challenges need to be overcome and the solutions must be reliable before MEM relay technology becomes widely utilized in practice.

Contact Engineering

First and foremost, contact engineering remains the next immediate challenge. Even in the current devices, the thin layer of TiO_2 and native tungsten oxide need to be "broken" through before any current flow is observed through the relay. This process requires a sufficiently high voltage (~4-6 V) to breakdown the oxide. Once the relay is operating the tungsten native oxide begins to form on time scales as short as a few seconds. This can ramp up the contact resistance from an initial value of ~1 kΩ to >10 kΩ after which the oxide will once again need to be broken to get any DC connection. Thus, in the long term, there will need to be a contact solution where either the contact materials do not form insulating oxides or else a packaging solution where such an oxide formation can not occur.

Device Scaling

The other clear challenge to making MEM relays relevant is how to reliably scale the devices. Without scaling, there are limited energy benefits to transitioning to relays. It has been shown that the energy limits of scaling relays is only determined by the surface forces at the contacts, and that this limit is many orders of magnitude away from the current designs

[127]. Most of the concerns associated with processing thin-films, such as residual stress and endurance, are relevant in regards to scaling relays as well. For example, even though V_{pi} is theoretically determined only by the device dimensions (which should be able to be defined with high accuracy) residual stress effects such as out-of-plane bending can cause dramatic changes in V_{pi}; thus, variations in residual stress will also lead to variations in device performance and energy. However, the greatest foreseeable obstacle to scaling may be scaling the gap dimension. Although it is not a challenge to fabricate sacrificial layers at sub-20 nm dimensions, reliably releasing the devices at these dimensions is a problem.

■ 7.4.2 Future Work

In the context of processing, the challenges described above are not wholly unknown which lends confidence that the scaling of relay technology is not all that distant.

Design Infrastructure

To take advantage of the technology when it does mature will require a significant support infrastructure. As we have shown, relays lend themselves nicely to much of the existing infrastructure developed for CMOS, but there are some differences in design paradigms that will require changes. For one, the difference in circuit styles will require synthesis and timing driven tools with a different set of optimization objectives. Kevin Dwan and Chengcheng Wang in Prof. Dejan Marković's group have already begun this effort. Although a circuit simulation model for the relays has been developed, this model is still largely analytic and optimized for a single relay design point. To really predict performance, more extensive circuit models that capture mismatch, noise, and device non-linearities will need to be developed. Along the same lines, tailored parasitic extraction tools and verification tools will be needed. We have already implemented some rudimentary design rule checker (DRC) and layout versus schematic (LVS) tools, but eventually we will want to allow designers freedom in choosing their device dimensions, so more sophisticated set of rules will be required.

Relay Circuit Design

What has been presented so far is analogous to the early work in the 1940's and 50's when transistors were first invented. Although we have presented the design of a number of circuit sub-systems, these are far from optimized designs. One of the immediate challenges is defining a robust clocking strategy that is not limited to the scale of a relay's mechanical delay. Basically any CMOS circuits that currently rely on linear gain, or ratios of device sizes need to find their relay equivalents. As we alluded to with our final ADC design, the other dimension of designing with MEM relays that we just touched upon is that in setting the device dimensions, the designer actually has control of the device performance. This would be akin to being able to select the doping profile and threshold voltage of every transistor in your design. This additional degree of design freedom should enable even more energy/performance optimized relay circuit designs.

Chapter 8

Conclusions

In this thesis we introduced the use of two new technologies to address the energy consumption in wireless sensor nodes and thus the reliability and cost of WSNs. Since the active energy of wireless sensors is dominated by communication cost, we first propose the use of compressed sensing as a generic source encoding algorithm to minimize the data that must be transmitted. We show that CS, despite being a lossy compression algorithm, is a more efficient encoder for the same reconstruction performance than lossless source coding techniques when the data being captured is quantized to moderate resolutions ($\sim$8 bits). Furthermore, the compressed measurements in CS show robustness in a noisy wireless environment and are amenable to low complexity error detection schemes that perform nearly as well as more complex error correction codes. For sparse signals, CS demonstrates over 10X lower energy cost to communicate the input samples compared to sending the raw quantized data. We develop circuit models that capture the implementation costs of the CS encoder with respect to system performance specifications and use these to guide our physical implementation. To demonstrate the practicality of CS for wireless sensor node applications, we designed and demonstrated a test chip that implements the CS encoding. The test chip was fabricated in a 90 nm CMOS processes and consumes 1.9 μW at 0.6 V for operating frequencies below 20 kHz. Unlike CS implementations that are coded on off-the-shelf signal processing hardware [142], the CS encoding here is implemented in such a way that the com-

pressed measurements are calculated *on-the-fly* for each sample so no additional memory or processing (and thus latency) is necessary before the data can be transmitted.

While data compression is an effective way to reduce active sensor cost, this does not address the energy overhead due to CMOS sub-threshold leakage when sensors are idle. To address this limitation, we introduced the idea of designing circuits with MEM relays, whose leakage is negligible. We first develop a relay circuit design methodology that reduces the performance and area gap with CMOS by over an order of magnitude. To verify some of the fundamental design concepts and the viability of the technology, we experimentally demonstrate several circuit building blocks in a larger MEM relay technology. We then show that in a scaled relay technology, MEM relays can provide a solution that is 10X more energy-efficient across a variety of circuit blocks.[1]

What we have shown (CS) and projected (MEMS) is that both the active (fewer bits) and idle (Moore MEMS) energy costs of wireless sensors can be reduced by over an order of magnitude. In addition to extending the lifetime of current sensor nodes to upwards of a decade, this could also pave the way for self-powered sensor nodes that don't require a battery at all [143]. The optimistic implication is that WSNs could become cost effective and pervasive enough to improve energy efficiency across all industries.

■ 8.1 Challenges

Despite the work in this thesis (or perhaps highlighted by this thesis) both CS and MEM relays are still relatively undeveloped technologies. For CS, one important point that we don't discuss much is the cost of reconstructing the signal. In wireless sensor applications, the node really only needs to be concerned with the data encoding. The base station, which is presumably much less energy constrained, is then tasked with reconstructing the signal. There have been some demonstrations that have shown real-time reconstruction algorithms

[1]The energy comparisons with CMOS that were shown all assumed that each circuit block was maximally utilized–i.e. the operating frequency perfectly accommodates the energy optimized delay of the circuit. However in practice, this is not really possible so if we consider the usage scenarios where there are long periods of inactivity, then the effective energy cost of CMOS functional units will continue to grow while relay based circuits will maintain their energy per operation cost.

running on an Apple iPhone for EKG recovery [67], so the challenge looking forward is not one of feasibility but of energy efficiency. The other challenge whose solution is not entirely clear is how to choose a proper basis for recovery. Again this is a task for the receiver and one method, as we alluded to is to train the receiver with uncompressed data first. However, this is probably not the ideal solution so this problem remains an open question.

In the final chapter on MEM relays, we discussed the immediate technology challenges as being contact engineering and device scaling. At a high level these both relate to the biggest challenge for MEM relay technology, which is not speed or energy, but rather reliability. If MEM relay technology could be made reliable, even at higher operating voltages (say 3-5 V), that would pave the way for some adoption which could help drive the scaling process.

■ 8.2 Discussion and Future Prospects

Thus far, we have described both CS and MEM relay technologies primarily in the context of wireless sensors. However, clearly both technologies are not strictly limited to this regime. For CS, its strengths are in its generality, and encoder hardware simplicity. It is particularly well suited for applications where there is an imbalance in available resources that is constrained by data–either energy or bandwidth. For many wireless sensor node applications, this constraint is energy, but for remote image or video capture, the constraint may be bandwidth. Also, one thing that should be pointed out is that CS adds an inherent level of security as the data is already "scrambled" in such a way that only knowing the encoding matrix and signal basis will enable signal recovery. However, it is not clear if CS is well suited for other proposed applications such as cognitive radio or AICs. The data rates where those applications will be relevant requires the reconstruction algorithm to be efficient and fast; and until the reconstruction costs are understood, it is not clear if this will be an energy efficient option.

As for MEM relays, the application space is clearly broad and only limited by the range of feasible operating frequencies. Although all levels of computation and communication stand to benefit, clearly mobile or embedded applications would have the most to gain

right now. The other less obvious (and less technical) implication of a VLSI MEM relay technology, is that it would alter the design model for MEMS and ICs in general. Currently there are dedicated MEMS foundries, but no fabless MEMS company uses an *off-the-shelf* process. The reason for this is because their added value is at the device level, which includes the process flow. Thus, each company develops its own process, but just uses the foundry's equipment. If MEM relays were to be used as logic, then MEMS foundries would be more like CMOS foundries today where a single process is used because the innovation is made at the system level. In turn, designers who were building systems, would find ways to build other MEMS devices (such as accelerometers, etc..) within this single process since integration breeds efficiency. This is akin to the development of RF components in CMOS or even more recently on-chip photonics. Hence, the bigger potential impact for MEMS, besides making circuits more energy efficient, is that it could enable a new breed of microsystem design that includes computation, communication, transduction and even energy conversion [144] components all on the same platform.

APPENDIX A

Modeling Details

A.1 Analog CS Encoder Power Model

In Section 4.1, a power model for the analog implementation of the CS encoder is presented. To maintain readiblility, the details of the model derivation were omitted; the derivations of those equations are now explained in more detail here.

A.1.1 Windowed Integrator Noise Bandwidth

The following section describes the derivation of (4.4). To quantify the noise bandwidth of a windowed single-pole low-pass filter, we assume that the time response of the low-pass filter (integrator) begins abruptly at time $t = 0$ and ends one integration period later at $t = T_i$. This is equivalent to windowing the low-pass filter response in time with a rectangular window of width T_i. Thus, the windowed step response, where the filter's time constant is $\tau = RC$, can be written as:

$$s_i(t) = (1 - e^{-t/RC})\,[u(t) - u(t - T_i)] \tag{A.1}$$

The Fourier transform of the impulse response is then:

$$H_i(j\omega) = \frac{1}{1 + j\omega RC}\left[1 - e^{-T_i/RC}e^{-j\omega T_i}\right] \tag{A.2}$$

So, to find the noise bandwdith, BW_N, we need to solve the following integral:

$$BW_N = \int_0^\infty \mid \frac{1 - Ke^{-jaf}}{1 + jbf} \mid \mathrm{d}f \tag{A.3}$$

where $a = 2\pi T_i$, $b = 2\pi RC$, and $K = e^{-T_i/RC}$. The integral is then solved as follows:

$$\begin{aligned}
BW_N &= \int_0^\infty \frac{1}{1 + b^2 f^2} \mid 1 - Kcos(af) + jKsin(af) \mid^2 df \\
&= \int_0^\infty \frac{1}{1 + b^2 f^2} \left[(1 - Kcos(af))^2 + K^2 sin^2(af)\right] df \\
&= \int_0^\infty \frac{1}{1 + b^2 f^2} \left[1 + K^2 - 2Kcos(af)\right] df \\
&= \int_0^\infty \frac{1 + K^2}{1 + b^2 f^2} - \frac{2Kcos(af)}{b^2(1/b^2 + f^2)} df \\
&= \left[\frac{1 + K^2}{b} tan^{-1}(bf)\right]_0^\infty - \frac{2K}{b^2}\left[\frac{\pi}{2} be^{-a/b}\right] \\
&= \frac{(1 + K^2)\pi}{2b} - \frac{K\pi}{b} e^{-a/b} \\
&= \frac{1 + e^{-2T_i/RC}}{4RC} - \frac{e^{-2T_i/RC}}{2RC} \\
&= \frac{1}{4RC}(1 - e^{-2T_i/RC})
\end{aligned} \tag{A.4}$$

Thus, as the integration time approaches infinity, the noise bandwidth approaches $1/4RC$ which results in the familiar total noise of kT/C for an equivalent resistor noise source of $4kTR$.

Solution via Parseval's equality

The same result can be found with less effort by invoking Parseval's equality and finding the differences between the noise bandwidth of the un-windowed response and the time shifted and scaled portion of the response that is removed by windowing. The corresponding solution

is as follows:

$$\begin{aligned} BW_N &= \int_0^\infty \frac{1}{1+b^2f^2}df - \int_0^\infty \frac{K^2}{1+b^2f^2}df \\ &= \left[\frac{1}{b}tan^{-1}(bf)\right]_0^\infty - \left[\frac{K^2}{b}tan^{-1}(bf)\right]_0^\infty \\ &= \frac{\pi}{2b} - \frac{\pi}{2b}K^2 \\ &= \frac{1}{4RC}(1-e^{-2T_i/RC}) \end{aligned} \tag{A.5}$$

■ A.1.2 Mixer Noise

As mentioned in Section 4.1, the primary impact of the mixer, not accounting for non-linearities, is on the required noise performance of the OTA. The mixer's noise figure (NF) and conversion gain (G_C) can be combined to specify the required input (output) noise power density at the mixer input (OTA output) per sample, where the NF is the ratio of the input SNR to the output SNR:

$$NF = \underbrace{(SP_{M,IN} - NP_{M,IN})}_{SNR_{IN}} - \underbrace{(SP_{M,OUT} - NP_{M,OUT})}_{SNR_{OUT}} \tag{A.6}$$

where all terms are expressed in decibels (dB). $SP_{M,IN}$ is the mixer input signal power, $NP_{M,IN}$ is the input noise power, $SP_{M,OUT}$ is the output signal power, and $NP_{M,OUT}$ is the output noise power. This expression can be rewritten to specify the required input noise power:

$$NP_{M,OUT} = NP_{M,IN} + G_C/2 + NF \tag{A.7}$$

where the conversion gain has been converted to an equivalent power gain ($G_C/2 = SP_{M,OUT} - SP_{M,IN}$). If we assume that the mixer bandwidth is wider than the OTA bandwidth, then we can describe the mixer's output current density as a function of its input current density (or the output current density of the OTA). This leads to (4.7) when evaluated for $G_C = -3$

dB and $NF = 3.8$ dB.

$$\begin{aligned}\overline{i_m^2} &= 10^{NP_{M,OUT}/10} \\ &= \overline{i_{OTA}^2} \cdot 10^{(G_C/2+NF)/10} \\ &= 1.7 \cdot \overline{i_{OTA}^2}\end{aligned} \tag{A.8}$$

■ A.1.3 Integrator Output Noise

The output noise density of the mixer is the input noise density to the integrator. Since the mixer is being modulated by a PRBS sequence over N samples, the resulting noise is a sum of the noise from all N samples where each sample has a noise bandiwidth that is $1/N$th of the integration period. Including the noise contributed from the sampling switch, the voltage noise power on the integrator at the end of an integration period is:

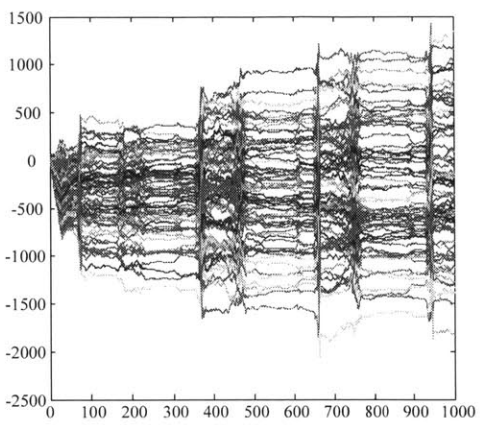
$$\begin{aligned}v_{n,int}^2 &= \frac{4kTR_{sw}}{R_o^2} \cdot G_I^2 BW_N + \left(\frac{1}{f_s C_L}\right)^2 \cdot \left(\frac{f_s}{2}\right) \cdot \sum_N \overline{i_m^2} \\ &= \left(\frac{4kTR_{sw}}{R_o^2} + \overline{i_m^2}\right) \cdot G_I^2 BW_N\end{aligned} \tag{A.9}$$

where $G_I = N/f_s C_L$ and $BW_N = f_s/2N$. For the switch to act like a good switch, R_{sw} should be small while R_o should be very large such that the OTA acts like a current source. Thus, in (A.9) it can be seen that the noise term due to the sampling switch should be negligible. Finally, the sampled noise on the integrator should be equal to or less than the quantization noise of the ADC in order to get the expected measurement resolution from the ADC which results in the following constraint on the integrator output noise.

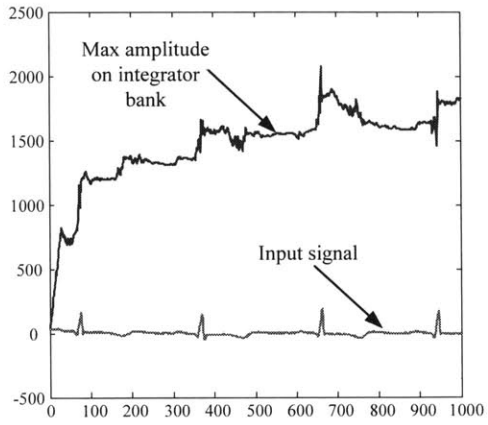
$$\overline{i_m^2} \cdot G_I^2 BW_N \leq \frac{V_{DDA}^2}{12 \cdot 2^{2B_y}} \tag{A.10}$$

■ A.1.4 Amplifier Power

Since the integrator noise is dependent on the mixer noise which is dependent on the OTA noise, the constraint in (A.10) also produces a constraint on the input referred noise ($v_{ni,rms}$)



(a) Distribution of integrator values after N samples



(b) Max. integrator amplitudes after N samples

Figure A-1: Dependence of integrator voltage on the number of accumulated samples.

of the OTA. We can start by rewriting (A.10) in terms of the mixer output current noise density and reshuffle the terms to get results in terms of OTA and system parameters.

$$\begin{aligned}
1.7 \cdot \overline{i_{OTA}^2} \cdot G_I^2 BW_N &\leq \frac{V_{DDA}^2}{12 \cdot 2^{2B_y}} \\
1.7 \cdot G_I^2 BW_N \cdot G_m^2 \overline{v_{ni}^2} &\leq \frac{V_{DDA}^2}{12 \cdot 2^{2B_y}} \\
0.85 \cdot N \cdot \frac{G_m^2}{f_s^2 C_L^2} \cdot \overline{v_{ni}^2} f_s &\leq \frac{V_{DDA}^2}{12 \cdot 2^{2B_y}} \\
\underbrace{0.92\sqrt{N} \cdot \frac{G_m}{f_s C_L}}_{G_A}{}^2 \cdot \underbrace{\overline{v_{ni}^2} f_s}_{v_{ni,rms}^2} &\leq \frac{V_{DDA}^2}{12 \cdot 2^{2B_y}} \\
v_{ni,rms}^2 &\leq \frac{V_{DDA}^2}{G_A^2 \cdot 12 \cdot 2^{2B_y}}
\end{aligned} \tag{A.11}$$

In (A.11), G_A represents the total in-band gain from the input of the OTA to the input of the ADC. The required gain will vary by application and the signals that are being acquired, but the general goal is to choose the gain such that the amplified and integrated signal exactly accommodates the input range of the ADC. However, since we are integrating over N samples modulated by a PRBS sequence, the variance of the voltage on the integrator will increase by a factor proportional to $\sqrt{N}$ (in the best case). An example showing the

voltage/value on the integrator bank after N samples have been integrated is plotted in Figure A-1a, while the maximum amplitude across the integrator bank is plotted against the input in Figure A-1b to show how the total dynamic range must increase as N increases. Thus, at a minimum, the required headroom for the integrated voltage value must increase by a factor of $\sqrt{N}$ to avoid saturating the integrators. Since the supply voltage cannot change, this implies instead that the input range of the ADC must shrink by $\sqrt{N}$ and thus the quantization noise must likewise drop which results in:

$$v_{ni,rms}^2 \leq \frac{4V_{DDA}^2}{N \cdot G_A^2 \cdot 12 \cdot 2^{2B_y}} \tag{A.12}$$

Recall that our assumption is that the input range to the ADC is differential, and set at V_{DDA}, so the instantaneous voltages could vary up to $2V_{DDA}$ differentially which accounts for the additional factor of 4. One could argue that the gain in the signal chain could merely drop to accomodate this range, and thus not incur the factor of $N/4$ penalty on noise power. However, that would assume that all of the noise contributions were due entirely from the input transistors, which even in just the amplifier is untrue. As noted in [98], the noise contribution from output transistors and feedback accounted for 60% of the output noise power. Furthermore, such a strategy would require that lowering G_m alone would sufficiently decrease the gain while preserving the OTA's bandwidth, output range, linearity, etc... For these reasons, the constraint placed on the amplifiers is that their noise must be reduced in proportion to the loss in headroom and this constraint is then substituted into the NEF equation to extract the minimum OTA power required.

$$\begin{aligned} P_{amp} &= M \cdot V_{DDA} I_{amp} \\ &= 2BW_f \cdot M \cdot V_{DDA} \cdot \frac{NEF^2}{v_{ni,rms}^2} \cdot \frac{\pi(kT)^2}{q} \\ &= 2BW_f \cdot M \cdot V_{DDA} \cdot NEF^2 \cdot \frac{N \cdot G_A^2 \cdot 12 \cdot 2^{2B_y}}{4V_{DDA}^2} \cdot \frac{\pi(kT)^2}{q} \\ &= 2BW_f \cdot 3M \cdot N \cdot 2^{2B_y} \cdot \frac{G_A^2 NEF^2}{V_{DDA}} \cdot \frac{\pi(kT)^2}{q} \end{aligned} \tag{A.13}$$

A.2 Digital CS Encoder Power Model

In Section 4.2, a power model for the digital implementation of the CS encoder is presented. To maintain readability, many details of the delay model and leakage model were omitted. The basis for those models and equations are now explained in more detail here.

A.2.1 Logical Effort Delay Model

In Section 4.2, a LE delay model was used for determining the minimum supply voltage and transistor sizing in the adder and flip-flops. The delay of the critical path (D) in LE can be calculated from the following equation:

$$D = G \cdot H + P: \qquad G = \prod_i g_i, \quad H = \prod_i h_i, \quad P = \sum_i p_i \tag{A.14}$$

where g_i, h_i and p_i represent the logical effort, electrical effort and parasitic delay of each gate in the path. Figure A-2 provides example calculations of these parameters for logic gates used in a ripple carry adder.

The output of the LE model is a delay normalized to the delay of a reference inverter in the technology of choice. This unit-less delay can be mapped to a real delay through the fanout-of-4 (FO4) delay of the technology through the relationship $1 \cdot FO4 = 5 \cdot \tau_{ref}$ where τ_ref is the normalized delay ($D = 1$) of the reference inverter which we model with a slightly simplified version of the alpha-power law delay model used in [103]:

$$\tau_{ref} \cdot d = \tau_{ref} \cdot g \cdot \left(h + \frac{p}{g}\right) = \frac{K_d \cdot V_{DD}}{(V_{DD} - V_{th})^{\alpha_d}} \cdot \left(\frac{w_{out}}{w_{in}} + \frac{w_{par}}{w_{in}}\right) \tag{A.15}$$

where K_d and α_d are technology fitting parameters and w_{out}/w_{in} and w_{par}/w_{in} are the electrical fan-out and intrinsic gate delay from (A.14). For each technology, K_d is approximated as the product $R_{on}C_g$ using values from Table 4.1 which approximates the delay of an unloaded inverter where $w_{out}/w_{in} = 0$ and $w_{par}/w_{in} = 2$ (parasitic and wiring). In the 90 nm predictive CMOS process [145], with $V_{th} = 0.4$ V and $\alpha_d = 1.3$, this results in a $\tau_{ref} = 7.7$ ps or a FO4 of 39 ps.

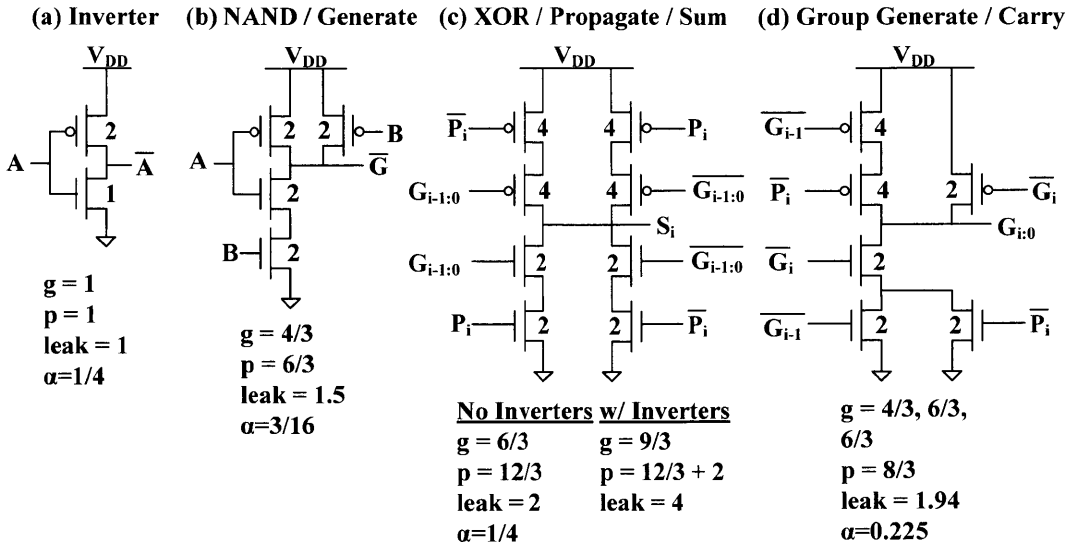


Figure A-2: Logical effort (g), parasitic delay (p) and normalized leakage (leak) for logic gate building blocks with normalized transistor widths used in the adder/encoder: (a) a reference inverter, (b) 2-input NAND gate, (c) 2-input XOR gate, (d) carry logic.

The critical path of the accumulator includes the delay of a flip-flop; for analysis, a master-slave flip-flop (MSFF) was chosen whose LE model is shown in Figure A-3. To account for both the clk-to-Q delay and setup time of the flip-flop, the calculated delay includes the entire path between the input and output. The resulting timing constraint on the accumulator is simply:

$$\tau_{ref} \cdot (D_{FF} + D_{ADD,B}) \leq \frac{1}{f_s} \tag{A.16}$$

where D_{FF} is the flip-flop normalized delay calculated in Figure A-3 and $D_{ADD,B}$ is the critical path of a B-bit adder as described in [102]. To provide a point of initial analysis, the adder topology chosen is a ripple carry adder whose normalized delay for a B-bit adder is:

$$\begin{aligned} D_{ripple,B} = & \underbrace{G_P H_P + (B + log_2\sqrt{N} - 1) \cdot G_C H_C + G_S H_S}_{Path\ Effort} \\ & + \underbrace{P_P + (B + log_2\sqrt{N} - 1) \cdot P_C + P_S}_{Parasitic\ Effort} \end{aligned} \tag{A.17}$$

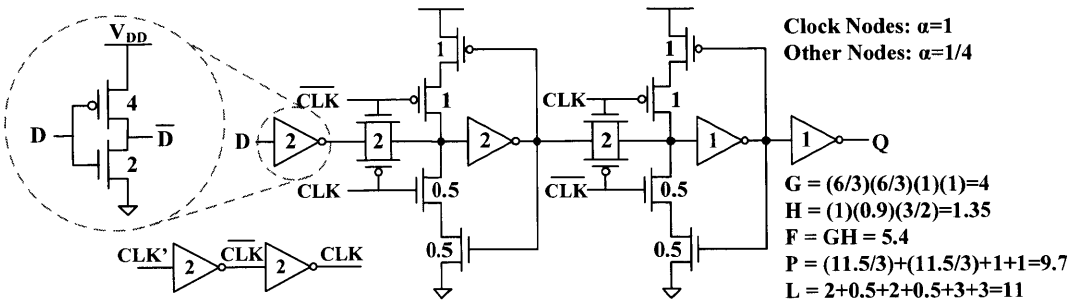


Figure A-3: Simplified logical effort based path effort (F), parasitic delay (P) and leakage (L) for a master-slave flip-flop (MSFF).

where $(G, H, P)_P$, $(G, H, P)_C$, and $(G, H, P)_S$ respectively denote the logical effort, electrical effort and parasitic effort associated with generating the propagate, carry, and sum signals in Figure A-2. The $log_2\sqrt{N}$ term is added to avoid overflow in the accumulators and is analogous to the headroom required in the analog model.

■ A.2.2 Switching Power/Energy Model

For results in our subsequent analysis, the normalized delay results are used to scale V_{DD} until the constraint in (A.16) is just met. This results in the following lower bound on V_{DD}:

$$V_{DD,min} \simeq \frac{\alpha_d V_{th}}{1 - 2.5 \cdot K_d \cdot (D_{FF} + D_{ADD,B}) \cdot f_s} \tag{A.18}$$

where $V_{DD,min}$ is used to estimate the power consumption of the digital blocks. The dynamic power consumption can be calculated by accounting for all of the gate and parasitic capacitances at each node in conjunction with the switching activity at those nodes. This can be captured in the standard equation for dynamic power:

$$P_{dyn} = f_s \cdot V_{DD}^2 \cdot C_{inv} \cdot \sum_i \alpha_i C_{d,i} \tag{A.19}$$

where C_{inv} is the capacitance of the reference inverter, and α and C_d are the activity and unit-less normalized node capacitance for each gate. These values for a single bit slice in the ripple carry adder have been enumerated in Table A.1. So for M B-bit accumulators

Table A.1: Switching activity and normalized node capacitance per bit in the ripple carry adder

Signal	(V) Activity Factor (α)	Normalized Load (C_d)
A_i	0.25	6.3
B_i	0.25	6.3
G_i	0.1875	7
P_i	0.25	10
Carry ($G_{i:j}$)	0.225	9.67
Sum	0.25	6

and XORs operating at the minimum required supply voltage ($V_{DD,min}$), this results in a dynamic switching energy of:

$$P_{accum,dyn} = M \cdot \left[27 \cdot (B + log_2\sqrt{N}) + 2\right] \cdot f_s \cdot V_{DD,min}^2 \cdot C_{inv} \tag{A.20}$$

■ A.2.3 Leakage Model

One component of energy consumption that LE does not explicitly model is the sub-threshold leakage current. In order to account for the leakage, an additional normalized parameter, leak, is added that captures the relative leakage current in each gate compared to the reference inverter. The simplifying assumptions made in determining this parameter are that the NMOS and PMOS leakage in the reference inverter are the same, and that the leakage current scales linearly with gate width and the number of off branches. The probability of the gate being in a certain leakage state is also taken into consideration. So for example, for the 2-input NAND in Figure A-2b, when the pull-down network is active there are two leakage paths–each with the same drive strength as the reference inverter. Thus, the relative leakage of the gate in this state is 2. However this state only occurs 1/4 of the time. The other 3/4 of the time the pull-up network is active and there is only one leakage path but with two additional states. When input A is high, the leakage is equivalent to only the lower NMOS being off and its leakage is twice that of the reference inverter. When the upper NMOS device is off, then the leakage is determined by the stack of devices. The total

relative leakage of the 2-input NAND gate is $leak_{nand2} = 1/4 \cdot (2) + 1/2 \cdot (1) + 1/4 \cdot (2) = 1.5$. Similar results for other gates are also provided in Figure A-2 and Figure A-3. Like the delay calculation, the absolute leakage of the circuit can be determined from the leakage of the reference inverter using a similar model as in [103]:

$$I_{leak,ref}(V_{DD}) = w_{n,ref} I_{leak} e^{\frac{-\gamma \cdot (V_{nom} - V_{DD})}{n \cdot kT/q}} \tag{A.21}$$

where $w_{n,ref}$ is the NMOS width of the reference inverter and I_{leak} is the transistor leakage per micron of gate width from Table 4.1. V_{NOM} is the nominal supply voltage for the process and also the drain voltage at which I_{leak} was measured. The parameter γ is used to account for DIBL and n accounts for the sub-threshold slope of the process. Again, we can combine this with the normalized leakage parameters for the MSFF, XOR, and adder to arrive at a power consumption expression due to leakage:

$$P_{accum,leak} = M \cdot \left[22.5 \cdot (B + log_2\sqrt{N}) + 4\right] \cdot V_{DD,min} \cdot I_{leak,ref}(V_{DD,min}) \tag{A.22}$$

where $I_{leak,ref}(V_{DD,min})$ is the leakage of the reference inverter at a supply voltage of $V_{DD,min}$.

A.3 Details of the MEM Relay Mechanical Model

This section provides the details regarding the calculation of the key MEM relay parameters.

A.3.1 Calculating the Pull-in Voltage

To determine pull-in voltage, V_{pi}, we can start with the mechanical model in Figure 6-1 which is a second-order system described by the equation

$$m\ddot{x} = F_{elec} - b\dot{x} - kx \tag{A.23}$$

where x is the downward displacement of the gate, k is the effective spring constant, b is the damping force, m is the inertial mass, and F_{elec} is the applied electrostatic force. The applied

force is a complex non-linear function of the displacement, x, for a fixed input voltage, but if we ignore effects such as fringe fields, then to first order it can be calculated as:

$$F_{elec} = \frac{\epsilon_0 A V_{gb}^2}{2(g_0 - x)^2} \tag{A.24}$$

where ϵ_0 is the permittivity of free space, A is the effective actuation area, V_{gb} is the gate-to-body voltage and g_0 is the gate-to-body gap with no applied electrical force. Since the electrostatic force increases quadratically with displacement while the spring force, kx only increases linearly, there is a critical displacement after which the electrical force will always be greater than the spring force causing the gap to close abruptly or "pull-in" [126]. At the displacement just before this occurs, the net force on the beam/gate must be zero, so this critical displacement can be found by setting the dynamic terms of (A.23) to zero so that

$$\frac{\epsilon_0 A V_{gb}^2}{2(g_0 - x)^2} - kx = 0 \tag{A.25}$$

The second condition that must be satisfied is that the displacement must be at a stable equilibrium point such that small disturbances do not generate positive feedback. This means that the derivative of the net force with respect to the displacement must be negative which requires:

$$\frac{\epsilon_0 A V_{gb}^2}{(g_0 - x)^3} - k < 0 \tag{A.26}$$

Now we can use (A.26) to substitute for k in (A.25) and solve for x.

$$\begin{aligned} \frac{\epsilon_0 A V_{gb}^2}{2(g_0 - x)^2} &= \frac{\epsilon_0 A V_{gb}^2}{(g_0 - x)^3} x \\ \epsilon_0 A V_{gb}^2 &= \frac{2\epsilon_0 A V_{gb}^2}{g_0 - x} x \\ 1 &= \frac{2x}{g_0 - x} \\ x &= \frac{1}{3} g_0 \end{aligned} \tag{A.27}$$

Now substituting this value of displacement back into (A.25) we get the gate-to-body voltage corresponding to this critical displacement, which is V_{pi}.

$$V_{pi} = \sqrt{\frac{8}{27} \cdot \frac{kg_0^3}{\epsilon_0 A}} \tag{A.28}$$

In this derivation, the effective dielectric seen between the gate's center plate and the body electrode is assumed to be entirely air, but if it is otherwise, then ϵ_0 should be replaced by some effective dielectric.

Bibliography

[1] TI, "INA333 Datasheet," 2008. [Online]. Available: http://focus.ti.com/lit/ds/symlink/ina333.pdf

[2] TI, "ADS7866 Datasheet," 2005. [Online]. Available: http://www.ti.com/lit/ds/symlink/ads7866.pdf

[3] TI, "TMS320C5504 Fixed-Point Digital Signal Processor Datasheet," 2011. [Online]. Available: http://www.ti.com/lit/ds/symlink/tms320c5504.pdf

[4] TI, "CC2550 Datasheet," 2011. [Online]. Available: http://www.ti.com/lit/ds/symlink/tms320c5504.pdf

[5] N. Verma, A. Shoeb, J. Bohorquez, J. Dawson, J. Guttag, and A. P. Chandrakasan, "A Micro-Power EEG Acquisition SoC With Integrated Feature Extraction Processor for a Chronic Seizure Detection System," *IEEE Journal of Solid-State Circuits*, vol. 45, no. 4, pp. 804–816, Apr. 2010.

[6] R. Yazicioglu, P. Merken, R. Puers, and C. Van Hoof, "A 200 W Eight-Channel EEG Acquisition ASIC for Ambulatory EEG Systems," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 12, pp. 3025–3038, 2008.

[7] J. L. Bohorquez, M. Yip, A. P. Chandrakasan, and J. L. Dawson, "A Biomedical Sensor Interface With a Sinc Filter and Interference Cancellation," *Journal of Solid-State Circuits*, vol. 46, no. 4, pp. 746–756, 2011.

[8] N. Verma and A. Chandrakasan, "An Ultra Low Energy 12-bit Rate-Resolution Scalable SAR ADC for Wireless Sensor Nodes," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 6, pp. 1196–1205, June 2007.

[9] A. Agnes, E. Bonizzoni, P. Malcovati, and F. Maloberti, "A 9.4-ENOB 1V 3.8W 100kS/s SAR ADC with Time-Domain Comparator," in *International Solid-State Circuits Conference*, 2008, pp. 246–248.

[10] M. Yip and A. P. Chandrakasan, "Marcus Yip, Anantha P. Chandrakasan," in *International Solid-State Circuits Conference*, 2011, pp. 822–823.

[11] M. W. Phyu, Y. Zheng, B. Zhao, L. Xin, and Y. S. Wang, "A real-time ECG QRS detection ASIC based on wavelet multiscale analysis," *2009 IEEE Asian Solid-State Circuits Conference*, pp. 293–296, Nov. 2009.

[12] J. Kwong, Y. Ramadass, N. Verma, M. Koesler, K. Huber, H. Moormann, and A. Chandrakasan, "A 65nm Sub-Vt Microcontroller with Integrated SRAM and Switched-Capacitor DC-DC Converter," in *International Solid-State Circuits Conference*, no. June, 2008, pp. 318–319.

[13] B. W. Cook, A. Berny, A. Molnar, S. Lanzisera, and K. S. J. Pister, "Low-Power 2.4-GHz Transceiver With Passive RX Front-End and 400-mV Supply," *Journal of Solid-State Circuits*, vol. 41, no. 12, pp. 2757–2766, 2006.

[14] J. L. Bohorquez, J. L. Dawson, and A. P. Chandrakasan, "A 350W CMOS MSK Transmitter and 400W OOK Super-Regenerative Receiver for Medical Implant Communications," in *2008 Symposium on VLSI Circuits Digest of Technical Papers*, 2008, pp. 32–33.

[15] S. Rai, J. Holleman, J. Pandey, F. Zhang, and B. Otis, "A 500 W Neural Tag with 2Vrms AFE and Frequency-Multiplying MICS/ISM FSK Transmitter," in *IEEE ISSCC Dig. Tech. Papers*, 2009, pp. 212–214.

[16] M. S. Chae, Z. Yang, M. R. Yuce, L. Hoang, and W. Liu, "A 128-channel 6 mW wireless neural recording IC with spike feature extraction and UWB transmitter." *IEEE transactions on neural systems and rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society*, vol. 17, no. 4, pp. 312–21, Aug. 2009.

[17] D. C. Daly and A. P. Chandrakasan, "An Energy-Efficient OOK Transceiver for Wireless Sensor Networks," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 5, pp. 1003–1011, May 2007.

[18] P. P. Mercier, D. C. Daly, and A. P. Chandrakasan, "An Energy-Efficient All-Digital UWB Transmitter Employing Dual Capacitively-Coupled Pulse-shaping Drivers," *Journal of Solid-State Circuits*, vol. 44, no. 6, pp. 1679–1688, 2009.

[19] R. Nathanael, V. Pott, H. Kam, J. Jeon, and T.-J. K. Liu, "4-Terminal Relay Technology for Complementary Logic," in *2009 IEEE International Electron Devices Meeting (IEDM)*. Ieee, Dec. 2009, pp. 1–4.

[20] "Predictive Technology Model." [Online]. Available: http://ptm.asu.edu/

[21] D. Patil, O. Azizi, M. Horowitz, R. Ho, and R. Ananthraman, "Robust Energy-Efficient Adder Topologies," *18th IEEE Symposium on Computer Arithmetic (ARITH '07)*, pp. 16–28, June 2007.

[22] M. Spencer, F. Chen, C. Wang, R. Nathanael, H. Fariborzi, A. Gupta, H. Kam, V. Pott, J. Jeon, T.-J. K. Liu, D. Markovic, E. Alon, and V. Stojanovic, "Demonstration of integrated micro-electro-mechanical switch circuits for VLSI applications," *IEEE Journal of Solid State Circuits*, vol. 46, no. 1, pp. 308–320, Feb. 2011.

[23] F. Chen, M. Spencer, R. Nathanael, C. Wang, H. Fariborzi, A. Gupta, H. Kam, V. Pott, J. Jeon, T.-J. K. Liu, D. Markovic, V. Stojanovic, and E. Alon, "Demonstration of integrated micro-electro-mechanical switch circuits for VLSI applications," in *2010 IEEE International Solid-State Circuits Conference - (ISSCC)*. Ieee, Feb. 2010, pp. 150–151.

[24] B. Gosselin and M. Sawan, "Circuits techniques and microsystems assembly for intracortical multichannel ENG recording," *2009 IEEE Custom Integrated Circuits Conference*, no. Cicc, pp. 97–104, Sept. 2009.

[25] F. Chen, A. P. Chandrakasan, and V. Stojanović, "A Signal-agnostic Compressed Sensing Acquisition System for Wireless and Implantable Sensors," in *IEEE Custom Integrated Circuits Conference*, 2010, pp. 1–4.

[26] F. Chen, H. Kam, D. Markovic, T.-J. K. Liu, V. Stojanovic, and E. Alon, "Integrated circuit design with NEM relays," in *2008 IEEE/ACM International Conference on Computer-Aided Design*. Ieee, Nov. 2008, pp. 750–757.

[27] OECD, "Smart Sensor Networks : Technologies and Applications for Green Growth," Organisation for Economic Co-operation and Development, Tech. Rep. December, 2009.

[28] I. F. Akyildiz and M. C. Vuran, *Wireless Sensor Networks*. Wiley, 2010.

[29] C.-y. Chong and S. P. Kumar, "Sensor networks: Evolution, opportunities, and challenges," *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1247–1256, Aug. 2003.

[30] G. Simon, M. Maroti, A. Ledeczi, G. Balogh, B. Kusy, A. Nadas, G. Pap, J. Sallai, and K. Frampton, "Sensor network-based countersniper system," *Proceedings of the 2nd international conference on Embedded networked sensor systems - SenSys '04*, p. 1, 2004.

[31] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications - WSNA '02*. New York, New York, USA: ACM Press, 2002, p. 88.

[32] P.-J. Chen, D. Rodger, S. Saati, M. Humayun, and Y.-C. Tai, "Microfabricated Implantable Parylene-Based Wireless Passive Intraocular Pressure Sensors," *Journal of Microelectromechanical Systems*, vol. 17, no. 6, pp. 1342–1351, Dec. 2008.

[33] A. Rowe, D. Goel, and R. Rajkumar, "FireFly Mosaic: A Vision-Enabled Wireless Sensor Networking System," in *28th IEEE International Real-Time Systems Symposium (RTSS 2007)*. Ieee, Dec. 2007, pp. 459–468.

[34] M. Corrà, M. Pozzi, D. Zonta, and P. Zanon, "Monitoring Heritage Buildings with Wireless Sensor Networks : The Torre Aquila Deployment," in *2009 International Conference on Information Processing in Sensor Networks*, 2009, pp. 277–288.

[35] P. Juang, H. Oki, Y. Wang, M. Martonosi, L.-s. Peh, and D. Rubenstein, "Energy-Efficient Computing for Wildlife Tracking : Design Tradeoffs and Early Experiences with ZebraNet," in *Proceedings of the 10th international conference on Architectural support for programming languages and operating systems*, 2002, pp. 96–107.

[36] D. Puccinelli and M. Haenggi, "Wireless sensor networks: applications and challenges of ubiquitous sensing," *IEEE Circuits and Systems Magazine*, vol. 5, no. 3, pp. 19–31, 2005.

[37] V. Rocha and G. Goncalves, "Sensing the world: Challenges on WSNs," in *2008 IEEE International Conference on Automation, Quality and Testing, Robotics*. Ieee, May 2008, pp. 54–59.

[38] A. Gaddam, S. C. Mukhopadhyay, G. Sen Gupta, and H. Guesgen, "Wireless Sensors Networks based monitoring: Review, challenges and implementation issues," in *2008 3rd International Conference on Sensing Technology*. Ieee, Nov. 2008, pp. 533–538.

[39] C. Links, "Wireless sensor networks: Maintenance-Free or Battery-Free?" *RTC Magazine*, vol. 2, pp. 18–21, 2009.

[40] Duracell, "Technical Bulletin." [Online]. Available: http://www1.duracell.com/oem/primary/default.asp

[41] S. Roundy, M. Strasser, and P. K. Wright, "Powering Ambient Intelligent Networks," in *Ambient Intelligence*, J. Rabaey, W. Weber, and E. H. L. Aarts, Eds. New York: Springer, 2005.

[42] N. Kimura and S. Latifi, "A survey on data compression in wireless sensor networks," *International Conference on Information Technology: Coding and Computing (ITCC'05) - Volume II*, pp. 8–13 Vol. 2, 2005.

[43] K. Barr and K. Asanovic, "Energy Aware Lossless Data Compression," in *Proceedings of MobiSys 2003: The First International Conference on Mobile Systems, Applications, and Services*, 2003, pp. 231–244.

[44] B. Cook, S. Lanzisera, and K. Pister, "SoC Issues for RF Smart Dust," *Proceedings of the IEEE*, vol. 94, no. 6, pp. 1177–1196, June 2006.

[45] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: A survey," *Ad Hoc Networks*, vol. 7, no. 3, pp. 537–568, May 2009.

[46] R. Harrison, P. Watkins, R. Kier, R. Lovejoy, D. Black, B. Greger, and F. Solzbacher, "A low-power integrated circuit for a wireless 100-electrode neural recording system," *IEEE Journal of Solid State Circuits*, vol. 42, no. 1, pp. 123–133, 2007.

[47] R. Olsson and K. Wise, "A three-dimensional neural recording microsystem with implantable data compression circuitry," in *2005 IEEE International Solid-State Circuits Conference, 2005. Digest of Technical Papers. ISSCC*, vol. vol, 2005, pp. 558–559.

[48] J. N. Y. Aziz, K. Abdelhalim, R. Shulyzki, R. Genov, B. L. Bardakjian, M. Derchansky, D. Serletis, and P. L. Carlen, "256-Channel Neural Recording and Delta Compression Microsystem With 3D Electrodes," *IEEE Journal of Solid-State Circuits*, vol. 44, no. 3, pp. 995–1005, Mar. 2009.

[49] C. Alippi, R. Camplani, C. Galperti, and P. Milano, "Lossless Compression Techniques in Wireless Sensor Networks: Monitoring Microacoustic Emissions," in *IEEE International Workshop on Robotic and Sensors Environments*, no. October, Ottawa, 2007, pp. 1–5.

[50] A.-T. Avestruz, W. Santa, D. Carlson, R. Jensen, S. Stanslaski, A. Helfenstine, and T. Denison, "A 5 W / Channel Spectral Analysis IC for Chronic Bidirectional Brain Machine Interfaces," *Journal of Solid-State Circuits*, vol. 43, no. 12, pp. 3006–3024, 2008.

[51] G. Moore, "Cramming More Components Onto Integrated Circuits," *Proceedings of the IEEE*, vol. 86, no. 1, pp. 82–85, Jan. 1998.

[52] R. H. Dennard, F. H. Gaensslen, H.-n. Yu, V. L. Rideout, E. Bassous, and A. R. LeBlanc, "Design of Ion-Implanted MOSFET's with Very Small Physical Dimensions," *Journal of Solid-State Circuits*, vol. SC-9, no. 5, pp. 256–268, 1974.

[53] E. Nowak, "CMOS devices below 0.1 μm: how high will performance go?" *International Electron Devices Meeting. IEDM Technical Digest*, pp. 215–218, 1997.

[54] Intel, "Microprocessor Quick Reference Guide," 2008. [Online]. Available: http://www.intel.com/pressroom/kits/quickreffam.htm

[55] B. Calhoun, A. Wang, and A. Chandrakasan, "Modeling and sizing for minimum energy operation in subthreshold circuits," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 9, pp. 1778–1786, Sept. 2005.

[56] R. Baraniuk, "Compressive sensing," *IEEE Signal Processing Magazine*, vol. 24, no. 4, p. 118, 2007.

[57] E. Candes and M. Wakin, "An Introduction To Compressive Sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

[58] J. Romberg and M. Wakin, "Compressed Sensing: A Tutorial," 2007. [Online]. Available: http://users.ece.gatech.edu/~justin/ssp2007

[59] D. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[60] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[61] E. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5406–5425, 2006.

[62] E. J. Candès, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure and Applied Mathematics*, vol. 59, no. 8, pp. 1207–1223, Aug. 2006.

[63] E. Candès and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Problems*, vol. 23, no. 3, pp. 969–985, June 2007.

[64] A. M. Abdulghani, A. J. Casson, and E. Rodriguez-villegas, "Quantifying the Feasibility of Compressive Sensing in Portable Electroencephalography Systems," in *5th International Conference on Foundations of Augmented Cognition, Neuroergonomics and Operational Neuroscience*. Berlin: Springer-Verlag, 2009, pp. 319–328.

[65] P. K. Baheti, "An ultra low power pulse oximeter sensor based on compressed sensing," in *Wearable and Implantable Body Sensor Networks, 2009*, 2009, pp. 144–148.

[66] X. Chen, Z. Yu, S. Hoyos, B. M. Sadler, and J. Silva-martinez, "A Sub-Nyquist Rate Sampling Receiver Exploiting Compressive Sensing," *IEEE Transactions on Circuits and Systems I*, vol. 58, no. 3, pp. 507–520, 2010.

[67] K. Kanoun, H. Mamaghanian, N. Khaled, and D. Atienza, "A Real-Time Compressed Sensing-Based Personal Electrocardiogram Monitoring System," in *Design Automation and Test in Europe*, 2010.

[68] J. N. Laska, S. Kirolos, M. F. Duarte, T. S. Ragheb, R. G. Baraniuk, and Y. Massoud, "Theory and Implementation of an Analog-to-Information Converter using Random Demodulation," *2007 IEEE International Symposium on Circuits and Systems*, pp. 1959–1962, May 2007.

[69] F. Chen, A. P. Chandrakasan, V. M. Stojanović, and F. Chen, "Design and Analysis of a Hardware-Efficient Compressed Sensing Architecture for Data Compression in Wireless Sensors," *submitted to Journal of Solid-State Circuits.*

[70] F. Chen, A. P. Chandrakasan, and V. Stojanović, "A Low-power Area-efficient Switching Scheme for Charge-sharing DACs in SAR ADCs," in *IEEE Custom Integrated Circuits Conference*, 2010, pp. 4–7.

[71] C. Shannon, "Communication in the presence of noise," in *Proc. Institute of Radio Engineers*, vol. 37, no. 1, 1949, pp. 10–21.

[72] M. Fira and L. Goras, "Biomedical signal compression based on basis pursuit," *Proceedings of the 2009 International Conference on Hybrid Information Technology - ICHIT '09*, vol. 14, pp. 541–545, 2009.

[73] S. Aviyente, "Compressed Sensing Framework for EEG Compression," *2007 IEEE/SP 14th Workshop on Statistical Signal Processing*, pp. 181–184, Aug. 2007.

[74] S.-G. Miaou and S.-N. Chao, "Wavelet-Based Lossy-to-Lossless ECG Compression in a Unified Vector Quantization Framework," *IEEE Transactions on Biomedical Engineering*, vol. 52, no. 3, pp. 539–543, July 2005.

[75] S. Boyd and L. Vandenberghe, *Convex Optimization.* Cambridge University Press, 2004.

[76] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B*, vol. 58, no. 1, pp. 267–288, 1996.

[77] G. Peyre, "Numerical Tours of Signal Processing," *Computing in Science & Engineering*, pp. 94–97, 2011. [Online]. Available: http://www.ceremade.dauphine.fr/~peyre/numerical-tour/

[78] M. Grant and S. Boyd, "Graph implementations for nonsmooth convex programs," in *Recent Advances in Learning and Control*, V. Blondel, S. Boyd, and H. Kimura, Eds. Springer-Verlag Limited, 2008, pp. 95–110. [Online]. Available: http://stanford.edu/~boyd/graph_dcp.html

[79] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 1.21," Apr. 2011. [Online]. Available: http://cvxr.com/cvx

[80] E. J. Candès, M. B. Wakin, and S. P. Boyd, "Enhancing Sparsity by Reweighted 1 Minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5-6, pp. 877–905, Oct. 2008.

[81] A. Zymnis, S. Boyd, and E. Candes, "Compressed Sensing With Quantized Measurements," *IEEE Signal Processing Letters*, vol. 17, no. 2, pp. 149–152, Feb. 2010.

[82] J. A. Tropp and A. C. Gilbert, "Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.

[83] L. Jacques, D. K. Hammond, and J. M. Fadili, "Dequantizing Compressed Sensing : When Oversampling and Non-Gaussian Constraints Combine," *IEEE Transactions on Information Theory*, vol. 57, no. 1, pp. 559–571, 2011.

[84] U. Kamilov, V. K. Goyal, and S. Rangan, "Optimal Quantization for Compressive Sensing under Message Passing Reconstruction," in *IEEE International Symposium on Information Theory*, 2011.

[85] G. Peyre, "Best Basis Compressed Sensing," *IEEE Transactions on Signal Processing*, vol. 58, no. 5, pp. 2613–2622, May 2010.

[86] M. Fira and L. Goras, "An ECG Signals Compression Method and its Validation Using NN's." *IEEE Transactions on Bio-medical Engineering*, vol. 55, no. 4, pp. 1319–1326, June 2008.

[87] S. Rangan, A. K. Fletcher, and V. K. Goyal, "Asymptotic Analysis of MAP Estimation via the Replica Method and Applications to Compressed Sensing," *to appear in IEEE Transactions on Information Theory*, pp. 1–18, 2009.

[88] D. A. Huffman, "Minimum-redundancy coding for the discrete noiseless channel," *IEEE Transactions on Information Theory*, vol. 7, no. 1, pp. 27–38, Jan. 1961.

[89] J. Ziv and A. Lempel, "A universal algorithm for sequential data compression," *IEEE Transactions on Information Theory*, vol. 23, no. 3, pp. 337–343, May 1977.

[90] T. A. Welch, "A Technique for High-Performance Data Compression," *Computer*, vol. 17, no. 6, pp. 8–19, 1984.

[91] J. Storer and J. H. Reif, "Error Resilient Optimal Data Compression," *SIAM Journal of Computing (SICOMP)*, vol. 26, no. 4, pp. 934–939, 1997.

[92] V. K. Goyal, A. K. Fletcher, and S. Rangan, "Compresseive Sampling and Lossy Compression," *IEEE Signal Processing Magazine*, no. March 2008, pp. 48–56.

[93] B. Widrow and I. Kollar, *Quantization Noise: Roundoff Error in Digital Computation, Signal Processing, Control and Communications*. Cambridge: Cambridge University Press, 2008.

[94] R. Robucci, J. D. Gray, J. Romberg, and P. Hasler, "Compressive Sensing on a CMOS Separable-Transform Image Sensor," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1089–1101, June 2010.

[95] B. Murmann, "A/D converter trends: Power dissipation, scaling and digitally assisted architectures," in *2008 IEEE Custom Integrated Circuits Conference*, vol. 1. Ieee, Sept. 2008, pp. 105–112.

[96] A. Shahani, D. Shaeffer, and T. Lee, "A 12-mW wide dynamic range CMOS front-end for a portable GPS receiver," *IEEE Journal of Solid-State Circuits*, vol. 32, no. 12, pp. 2061–2070, 1997.

[97] M. Steyaert, W. Sansen, and C. Zhongyuan, "A micropower low-noise monolithic instrumentation amplifier for medical purposes," *IEEE Journal of Solid-State Circuits*, vol. 22, no. 6, pp. 1163–1168, 1987.

[98] W. Wattanapanitch, M. Fee, and R. Sarpeshkar, "An energy-efficient micropower neural recording amplifier," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 1, no. 2, pp. 136–147, 2007.

[99] J. Holleman and B. Otis, "A sub-microwatt low-noise amplifier for neural recording." in *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, vol. 2007, Jan. 2007, pp. 3930–3.

[100] M.-Z. Li and K.-T. Tang, "A low-noise low-power amplifier for implantable device for neural signal acquisition." in *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, vol. 2009, Jan. 2009, pp. 3806–9.

[101] I. Sutherland, B. Sproull, and D. Harris, *Logical Effort: Designing Fast CMOS Circuits*, 1st ed. Morgan Kaufmann, 1999.

[102] D. Harris and I. Sutherland, "Logical effort of carry propagate adders," *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, vol. 1, pp. 873–878, 2003.

[103] D. Markovic, V. Stojanovic, B. Nikolic, M. Horowitz, and R. Brodersen, "Methods for true energy-performance optimization," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 8, pp. 1282–1293, Aug. 2004.

[104] J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits*, 2nd ed. Upper Saddle River, NJ: Prentice Hall, 2002.

[105] W. Bajwa, J. Haupt, G. Raz, S. Wright, and R. Nowak, "Toeplitz-structured compressed sensing matrices," *IEEE Workshop on Statistical Signal Processing (SSP), Madison, Wisconsin*, pp. 294–298, 2007.

[106] UCSD, "Swartz Center for Computational Neuroscience." [Online]. Available: http://sccn.ucsd.edu

[107] J. Z. Sun and V. K. Goyal, "Optimal quantization of random measurements in compressed sensing," *2009 IEEE International Symposium on Information Theory*, vol. 3, no. 1, pp. 6–10, June 2009.

[108] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals," *Circulation Electronic Pages*, vol. 101, pp. e215–e220, 2000. [Online]. Available: http://circ.ahajournals.org/cgi/content/full/101/23/e215

[109] PhysioToolkit, "(wqrs, tach)." [Online]. Available: http://www.physionet.org/physiotools/

[110] M. Fira, L. Gora, C. Barabasa, and N. Cleju, "On ECG Compressed Sensing using Specific Overcomplete Dictionaries," *Computer Engineering*, vol. 10, no. 4, pp. 23–28, 2010.

[111] V. Misra, V. K. Goyal, S. Member, and L. R. Varshney, "Distributed Scalar Quantization for Computing : High-Resolution Analysis and Extensions," *IEEE Transactions on Information Theory*, vol. 57, no. 8, pp. 5298–5325, 2011.

[112] W. Burleson, "Efficient VLSI for Lempel-Ziv compression in wireless data communication networks," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 6, no. 3, pp. 475–483, 1998.

[113] N. Verma and A. P. Chandrakasan, "A 256 kb 65 nm 8T Subthreshold SRAM Employing Sense-Amplifier Redundancy," *Journal of Solid-State Circuits*, vol. 43, no. 1, pp. 141–149, 2008.

[114] I. J. Chang, J.-j. Kim, S. P. Park, S. Member, and K. Roy, "A 32 kb 10T Sub-Threshold SRAM Array With Bit-Interleaving and Differential Read Scheme in 90 nm CMOS," *Journal of Solid-State Circuits*, vol. 44, no. 2, pp. 650–658, 2009.

[115] H. Kam, "MOSFET Replacement Devices for Energy-Efficient Digital Integrated Circuits," Ph.D. dissertation, University of California - Berkeley, 2009.

[116] K. Akarvardar, D. Elata, R. Parsa, G. C. Wan, K. Yoo, J. Provine, P. Peumans, R. T. Howe, and H.-S. P. Wong, "Design Considerations for Complementary Nanoelectromechanical Logic Gates," *2007 IEEE International Electron Devices Meeting*, pp. 299–302, Dec. 2007.

[117] P. Zavracky, S. Majumder, and N. McGruer, "Micromechanical switches fabricated using nickel surface micromachining," *Journal of Microelectromechanical Systems*, vol. 6, no. 1, pp. 3–9, Mar. 1997.

[118] G.-L. Tan and G. M. Rebeiz, "A DC-contact MEMS shunt switch," *IEEE Microwave and Wireless Components Letters*, vol. 12, no. 6, pp. 212–214, June 2002.

[119] J. Jeon, V. Pott, H. Kam, R. Nathanael, E. Alon, and T.-J. K. Liu, "Perfectly Complementary Relay Design for Digital Logic Applications," *IEEE Electron Device Letters*, vol. 31, no. 4, pp. 371–373, Apr. 2010.

[120] C. E. Schmitt, "Carbon Nanotube-Based Nanorelays for Low- Power Circuit Applications by," Ph.D. dissertation, 2009.

[121] W. Tang, T. Nguyen, and R. Howe, "Laterally Driven Polysilicon Resonant Microstructures," *Sensors and Actuators*, vol. 20, no. 1-2, pp. 25–32, Nov. 1989.

[122] C. W. Low, T.-J. King Liu, and R. T. Howe, "Characterization of Polycrystalline Silicon-Germanium Film Deposition for Modularly Integrated MEMS Applications," *Journal of Microelectromechanical Systems*, vol. 16, no. 1, pp. 68–77, Feb. 2007.

[123] K. Williams and R. Muller, "Etch rates for micromachining processing," *Journal of Microelectromechanical Systems*, vol. 5, no. 4, pp. 256–269, 1996.

[124] K. Williams, K. Gupta, and M. Wasilik, "Etch rates for micromachining processing-part II," *Journal of Microelectromechanical Systems*, vol. 12, no. 6, pp. 761–778, Dec. 2003.

[125] B. W. B. Ittner, D. Ph, and H. B. Ulsh, "The Erosion of Electrical Contacts by the Normal Arc," in *Proceedings of the IEE -Part B: Radio and Electronic Engineering*, no. 2248, 1957, pp. 63–68.

[126] S. D. Senturia, *Microsystem Design*. Boston: Springer, 2000.

[127] H. Kam, T.-j. K. Liu, V. Stojanovic, D. Markovic, and E. Alon, "Design , Optimization , and Scaling of MEM Relays for Ultra-Low-Power Digital Logic," *IEEE Transactions on Electron Devices*, vol. 58, no. 1, pp. 236–250, 2011.

[128] H. Kam, E. Alon, and T.-j. K. Liu, "A Predictive Contact Reliability Model for MEM Logic Switches," in *International Electron Devices Meeting*, 2010, pp. 399–402.

[129] R. Holm, *Electric Contacts*. Springer-Verlag, 1967.

[130] S. Bromley, B. Nelson, A. Inc, and M. Minneapolis, "Performance of microcontacts tested with a novel MEMS device," *Electrical Contacts, 2001. Proceedings of the Forty-Seventh IEEE Holm Conference on*, pp. 122–127, 2001.

[131] H. C. Lin and L. Linholm, "An optimized output stage for MOS integrated circuits," *IEEE Journal of Solid-State Circuits*, vol. 10, no. 2, pp. 106–109, Apr. 1975.

[132] W. Keister, "The Logic of Relay Circuits," *Transactions of the American Institute of Electrical Engineers*, vol. 68, no. 1, pp. 571–576, 1949.

[133] Z. Yang, D. Lichtenwalner, A. Morris, J. Krim, and A. Kingon, "Comparison of Au and AuNi Alloys as Contact Materials for MEMS Switches," *Journal of Microelectromechanical Systems*, vol. 18, no. 2, pp. 287–295, Apr. 2009.

[134] L. Brooks and H.-S. Lee, "A Zero-Crossing-Based 8-bit 200 MS/s Pipelined ADC," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 12, pp. 2677–2687, Dec. 2007.

[135] M. Z. Straayer and M. H. Perrott, "An efficient high-resolution 11-bit noise-shaping multipath gated ring oscillator TDC," in *IEEE Symposium on VLSI Circuits*, 2008, pp. 82–83.

[136] K.-L. Wong, H. Hatamkhani, M. Mansuri, and C.-K. Yang, "A 27-mW 3.6-Gb/s I/O Transceiver," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 4, pp. 602–612, Apr. 2004.

[137] R. Palmer, J. Poulton, W. J. Dally, J. Eyles, a. M. Fuller, T. Greer, M. Horowitz, M. Kellam, F. Quan, and F. Zarkeshvari, "A 14mW 6.25Gb/s Transceiver in 90nm CMOS for Serial Chip-to-Chip Communications," *2007 IEEE International Solid-State Circuits Conference. Digest of Technical Papers*, pp. 440–614, Feb. 2007.

[138] M. V. Elzakker, E. V. Tuijl, P. Geraedts, D. Schinkel, E. A. M. Klumperink, S. Member, and B. Nauta, "A 10-bit Charge-Redistribution ADC Consuming 1 . 9 W at 1 MS / s," vol. 45, no. 5, pp. 1007–1015, 2010.

[139] D. C. Daly and A. P. Chandrakasan, "Denis C. Daly, Anantha P. Chandrakasan," in *International Solid-State Circuits Conference*, 2008, pp. 554–556.

[140] H. Fariborzi, M. Spencer, V. Karkare, J. Jeon, R. Nathanael, C. Wang, F. Chen, H. Kam, V. Pott, T.-j. K. Liu, E. Alon, V. Stojanoviü, and D. Markoviü, "Analysis and Demonstration of MEM-Relay Power Gating," in *IEEE Custom Integrated Circuits Conference*, vol. 1, 2010.

[141] H. Fariborzi, F. Chen, R. Nathanael, J. Jeon, T.-J. K. Liu, and V. Stojanovic, "Design and Demonstration of Micro-Electro-Mechanical Relay Multipliers," in *to appear in Asian Solid-State Circuits Conference*, Jeju, Korea, 2011.

[142] H. Mamaghanian, N. Khaled, D. Atienza, and P. Vandergheynst, "Compressed Sensing for Real-Time Energy-Efficient ECG Compression on Wireless Body Sensor Nodes." *IEEE transactions on bio-medical engineering*, vol. 58, no. 9, pp. 2456–66, Sept. 2011.

[143] Y. K. Ramadass and A. P. Chandrakasan, "An Efficient Piezoelectric Energy Harvesting Interface Circuit Using a Bias-Flip Rectifier and Shared Inductor," *IEEE Journal of Solid-State Circuits*, vol. 45, no. 1, pp. 189–204, Jan. 2010.

[144] Y. K. Ramadass and A. P. Chandrakasan, "A Batteryless Thermoelectric Energy-Harvesting Interface Circuit with 35mV Startup Voltage," in *International Solid-State Circuits Conference*, 2010, pp. 296–297.

[145] ITRS, "Process Integration, Devices, and Structures," 2007. [Online]. Available: www.itrs.net