Compressed Sensing Receivers: Theory, design, and performance limits

Thesis by

Juhwan Yoo

In Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

California Institute of Technology

Pasadena, California

2012

(Defended May 24, 2012)

To my mother for teaching me determination,
my father for teaching me kindness,
and my brother for so enriching my life over the years.

Acknowledgements

I've always been fortunate; perhaps one of the most fortunate days of my life was the day I received undergraduate admission to Caltech. It's been a privilege to live the last 10 years of my life among this brilliant, talented, highly motivated, and endearingly eccentric group of people. This campus will always be dear to my heart and I will miss it greatly when I depart.

I would like to begin by expressing my profound gratitude to my advisor Professor Azita Emami-Neyestanak. What I learned and gained under her guidance goes well beyond academic mentorship. I am forever grateful for the significant impact she has had in both my development as a researcher as well as my personal growth. I am continually impressed by her clarity of thought and insight—all of which has hopefully rubbed off on me—as well as her passion for teaching and dedication to her students.

Working on the RMPI project afforded me the rare opportunity to collaborate with experts in both academia and industry. In particular, I'd like to acknowledge Dr. Stephen Becker's enormous contribution to the successful completion of the RMPI project. I learned an enormous amount from my interactions with him and cherish the friendship we have developed over the past 4 years. I'd also like to recognize the contributions of my colleagues and friends Matthew Loh and Manuel Monge, both of whom made significant contributions in both the design and testing of the RMPI. It was also my pleasure to interact with Drs. Emilio Sovero and Eric Nakamura of Northrop Grumman. In addition, although I neither knew him or even met him in person, I'd like to take this opportunity to acknowledge the contributions of Prof. Dennis Healy. He was the DARPA program manager who made the Analog-to-Information (A2I) program a reality. His sudden death during the course of this

endeavour had a profound impact on many involved. The people we knew in common always spoke of him as one of those rare people who both had the capability and took the time to understand all aspects of a problem. There is no doubt in my mind that it was my loss that we never met in person. I can only hope that he would've been satisfied with the outcome of the RMPI project.

I'm also particularly indebted to Dr. Edward Keehr, Prof. Joseph Bardin, Kaushik Sengupta and Dr. Amin Khajehnejad; much of what I learned came from my interactions with them (usually at hours when most civilized people are asleep). I could not have asked for better friends or colleagues to share this journey with.

I also owe special thanks to Prof. Ali Hajimiri. He has been a constant source of both encouragement and invaluable advice. I have always found him to be a source of inspiration and, more often than not, found myself with more motivation and enthusiasm after a conversation with him than I had before it.

I'd also like to thank my committee members, all of whom I've had a large number of interactions with: Sander Weinreb, Emmanuel Candès, Babak Hassibi, Joel Tropp, Ali Hajimiri, and Azita Emami-Neyestanak. I found the feedback I received from my committee valuable. I am indebted to each of them for not only providing valuable technical feedback, but supporting me in the pursuit of my academic interests and professional development. Other faculty I'd like to thank for their support and mentorship are: P. P. Vaidyanathan and Yu-Chong Tai.

In addition I must thank Profs. Aydin Babakhani and Hua Wang, as well as Michael McCoy and Alex Gittens, for their patience in the copious amount of questions they answered throughout the years. I'd also like to thank my floor-mates Steve Bowers, Kaushik Dasgupta, Amirreza Safaripour, Behrooz Abiri, Constantine Sideris, Meisam Nazari, Mayank Raj, Saman Saeedi, Dr. Florian Bohn, Dr. Firooz Aflatouni, Eyal EnGal, Jay Chen, Jennifer Arroyo, Manuel Godoy, Suyao Ji, Angie Wang, Lita Yang, Prof. Yu-jiu Wang, Prof. Sanggeun Jeon, Shohei Kousai, Tomoyuki Arai, Dr. Arun Natarajan, Damon Russell, Steve Smith, Hamdi Mani, and other colleagues/alumnus in both the MICS and CHIC groups (past and present), as well as the Hassibi group, for their camaraderie, shared humor, and most of all: for making my time here what it was.

I also wish to thank the many administrators and staff of Moore lab. In particular I'd like to thank: Michelle Chen, Christine Garske, Tanya Owen, Carol Sosnowski and Linda Dozsa for the many capacities in which they've helped me over the years.

Finally, anyone that's gone through the experience of a Ph.D. program can vividly remember at least one period during which they felt great frustration or experienced great difficulty. My friends from Caltech and the surrounding area saw me through several difficult periods that included a one month layover in a hospital bed. In particular, I'd like to thank Milo Lin, James Yoon, Haluna Gunterman, Alice Lin, Ruxandra Paun, Nadine Dabby, Jeremy Leibs, Joy Wong, Ryan Bates, and Miling Yan for their friendship and support.

Abstract

The past 50 years have seen tremendous developments in electronics due to the rise and rapid development of IC-fabrication technology [1]. In addition to the production of cheap and abundant computing resources, another area of rapid advancement has been wireless technologies. While the central focus of wireless research has been mobile communication, an area of increasing importance concerns the development of sensing/spectral applications over bandwidths exceeding multiple GHz. Such systems have many applications ranging from scientific to military. Although some solutions exist, their large size, weight, and power make more-efficient solutions desirable.

At present, one of the principal bottlenecks in designing such systems is the power consumption of the back-end ADCs at the required *digitization* rate. ADCs are a dominant source of power consumption; it is also often the case that ADC block specifications are used to determine parameters for the rest of the signal chain, such as the RF front-end and the DSP-core which processes the digitized samples [2]. Historically, increases in system bandwidth have come from developing ADCs with superior performance.

In contrast to improving ADC performance, this work presents a system-level approach with the goal of minimizing the required digitization rate for observation of a given effective instantaneous bandwidth (EIBW). The approach was inspired by the field of compressed sensing [3–5]. Loosely stated, CS asserts that samples which represent random projections can be used to recover *sparse* and/or *compressible* signals with what was previously thought to be insufficient information. The research in this thesis bridges the disparate areas of RF/Mixed-Signal IC design and CS; the primary contributions of this thesis include: the establishment of physical feasibility of CS-based receivers through implementation of the first-ever fully-integrated high speed CS-based front-end known as the random-modulation pre-integrator (RMPI) [6–9], and the development of a principled design methodology based on a rigorous analytical and empirical study of the system.

The 8-channel RMPI was implemented in 90 *nm* CMOS and was validated by physical measurements of the fabricated chip. The implemented RMPI achieves an EIBW of 2 *GHz*, with > 54 *dB* of dynamic range. Most notably, the aggregate digitization rate is $f_{agg} = 320$ *Msps*, 12.5× lower than the Nyquist rate.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Two principal goals in the design of modern electronic systems are to acquire large amounts of information *quickly* and *efficiently* (with little expenditure of resources). In the wireless technology sector, the goal of maximizing information throughput is illustrated by the strong interest in RF sensing and spectral applications that require instantaneous bandwidths of many *GHz*. Such systems have applications ranging from scientific instrumentation to electronic intelligence. Although some solutions already exist, their large size, weight, and power consumption make more-efficient solutions desirable. At present, bottlenecks in realizing high-bandwidth systems can be attributed to two principal challenges.

The first challenge comes from the power dissipation required in order to operate the back-end ADCs at the *Nyquist*-prescribed digitization rate. This issue is so significant that requirements on the other elements of the signal chain (RF front-end, DSP core, etc.) are dictated by the best performance achieved by an ADC compliant with the allocated power budget. Indeed, while converters with impressive speeds can be realized [12], their required power dissipation and relatively low resolution make their use impractical in many applications. For example, in most portable applications the power budget for the ADC is limited to a fraction of a watt [10]. Although several surveys [2, 10, 13, 14] of ADC technology note that the energy-efficiency of ADCs continues to improve, this may prove inconsequential for many emerging applications [15] that require increasingly larger effective instantantaneous bandwidths EIBW. This is due to the fact that while conversion-efficiency gains may considerably assist many existing applications, the same surveys also note that ADC speed-resolution product improves at an increasingly slower rate.

The second challenge comes from the logistical problems that arise from the need to store, compress, and post-process the ever-increasing volumes of data produced by such systems. For example, a system that acquires samples at a rate of 1 *Gsps* with 10-*bit*s of resolution will fill 1 *Gb* of memory in less than 1 *s*. This problem of "data déluge" combined with the slowing of process technology scaling, and the historically large disparity in the relative rates of converter and digital system performance strongly suggest that performance gains will have to be achieved by means beyond advances solely in ADC technology.

In contrast to the historic approach of realizing superior ADCs, this thesis addresses the stated challenges by utilizing results from the field of *compressed sensing* (CS) [3–5, 16]. The field of CS has recently emerged as an alternative paradigm to the Shannon-Nyquist sampling theorem which, at present, is implicitly used in the design of virtually all modern signal acquisition systems. CS theory makes the observation that many signals of interest display mathematical strucure, such as *sparsity/compressibility*, that can be exploited to reduce the *digitization* rate below that specified by the Shannon-Nyquist framework. In short, the theory states that signals with high overall bandwidth but comparatively low information level can be acquired very efficiently using randomized measurement protocols. The requisite sampling rate is merely proportional to the information level. Thus CS enables *sub-Nyquist rate* signal acquisition and provides a potential avenue to reduce both the power required for observation of a given bandwidth as well as data throughput by reducing the necessary digitization rate. Moreover, this approach allows the realization of systems with significantly higher EIBWs at any underlying level of mixed-signal technology. In addition, the rate at which a given system bandwidth can be achieved is accelerated as CS reduces the dependence of the system bandwidth on achievable ADC performance and ties it more directly to the level of sparsity.

1.1 Contributions

The principal contribution of this thesis is the first demonstration of the physical feasibility of CS-based receivers through the realization of the first-ever high-speed fully-integrated RF receiver capable of observing wide bandwidths at a sub-Nyquist rate. Specifically, this thesis presents the design, implementation, and measurement results of a prototype chip [6, 7] that employs a CS-inspired receiver architecture dubbed the random modulation pre-integrator (RMPI) [16, 17]. The RMPI is based upon the first CS-receiver architecture proposed in the academic literature known as the random demodulator (RD) [8, 9, 16, 18]. The presented RMPI was implemented in 90 *nm* CMOS and achieves 2 *GHz* EIBW while digitizing samples at an aggregate rate of just $f_{agg} = 320$ *Msps*: a factor of 12.5× below the Nyquist rate.

The prototype RMPI not only demonstrates the feasibility of CS-based approaches to wideband receiver design, but also represents the first work to address the void that exists between theory and practice. The material presented in this thesis provides the first quantitative answers to pragmatic questions such as how the theory of CS can be applied to a practical scenarios as well as when it is appropriate to do so. Other questions answered include the effects and limitations on achievable performance that result from physical nonidealities and design constraints. The answers to these questions are delivered in the form of rigorous analytical and empirical studies of the design space. Special emphases is placed on providing physical interpretations of the theory, in the context of wideband receiver design, to impart physical intuition that will aid potential future designers of RMPI-like systems. Although the setting of this work was in the field of RF systems design, many of the conclusions and insights will carry over to the design of other CS-based hardware platforms. An outline of the presented work is described in the next section.

1.2 Organization

Chapter 2 expands upon the discussion of challenges in the first section of this chapter and provides technical background for understanding both the motivation behind implementing

CS-based hardware and how the operation of CS receivers differs from their conventional counterparts. The background information briefly describes the function and operating principles of RF front-ends and ADCs. The chapter concludes with a brief review of the CS theory pertinent to the RMPI. The review begins with the insights and concepts at the heart of CS theory and concludes with an overview of the basic technical results.

Chapter 3 presents the operation of the ideal RMPI. The material in this chapter serves as a reference for the presentation of the material in succeeding chapters that describe the effects of departures from the ideal model and how to deal with them. The basic mathematical framework of the idealized RMPI including the matrix representation of the RMPI's operation and how it relates the measurements produced by the device and the linear inverse framework of the theory is given detailed treatment. The exposition begins by explaining how CS theory can be mapped to a physical sampling device and proceeds to describe the analytical framework of the RMPI. The chapter concludes with a discussion of the signal model of the RMPI and a review of other proposed CS systems that gives a glimpse of the different ways in which the theory can be applied.

Chapter 4 presents the results of design and feasibility studies that were conducted during the design of the physical chip. The procedure used to relax the idealized models into blocks with efficient and robust implementations is described. The effect of the most significant parameters of the system, their collective effect on system performance, as well as considerations in choosing these parameters is discussed. The nonidealities incorporated in the analysis are described and simulated performance is presented. The chapter concludes with an overview of the signal recovery algorithms used to reconstruct signals from the physically generated samples.

Chapter 5 describes the physical implementation details of the fabricated and tested 8-channel CMOS RMPI. Detailed transistor-level schematics and simulated performance of the chip are presented. A die-photo and performance breakdown in terms of more conventional RF-receiver metrics is given. Chapter 6 presents signal reconstructions obtained from the physical chip along with providing a description of the test setup. A summary of results, remaining open questions, and suggestions for future work are given in chapter 7.

Chapter 2

Background

The wireless communications technology that we take for granted every day required numerous advancements from a diverse set of fields including (but certainly not limited to): radio-frequency (RF) electronics, semiconductor device physics, digital computing, digital signal processing (DSP), numerical optimization, and sparse approximation. With respect to wireless transceivers these advancements came in rather uneven steps beginning in the latter part of the 19th century, with the prediction of electro-magnetic wave propagation by Maxwell, and continuing up to the present. As is the case with engineering any system that utilizes the results of multiple fields, significant advances (accomplished all at once or almost imperceptibly over time) in one field (or on occasion the creation of an entirely new one), often leads to the reexamination of implicit assumptions of the day that are used in the design of the system. The field of wireless systems engineering alone is replete with such examples. The developments that have led to the ubiquitous use of DSP in modern electronic devices is itself often held up as the canonical example that justifies questioning the assumptions of the time.

Prior to the 1950s the signal processing used to obtain desired information was entirely analog or mechanical in nature. Over time, among many other developments, the invention of the transistor by William Schockley and the ensuing rapid developments in digital computing technology led to the development of the sophisticated tools of DSP which are now an integral component of modern transceiver design. Briefly put, the appeal of communicating short messages without wires has evolved into the insatiable demand for communicating audio, video, and the real-time extraction of statistical trends from enormous data sets.

The work in this thesis is motivated by this demand and specifically concerns the design of RF receivers with extremely large *effective instantaneous bandwidths* EIBW. The approach utilizes results from the relatively young field of compressed sensing (CS): whose existence can be interpreted as the outcome of reexamining the cornerstone of modern DSP, around which all modern transceiver sytems are designed, the Shannon-Nyquist sampling theorem [19]:

Theorem 1 (Shannon-Nyquist-Whittaker Sampling Theorem). *Let $x(t)$ be a band-limited signal with $X(j\omega) = 0$ for $|\omega| > |\omega_M|$. Then $x(t)$ is uniquely determined by its samples $x(nT)$, $n = 0, \pm 1, \pm 2, \ldots$, if*

$$\omega_s > 2\omega_M,$$

where

$$\omega_s = \frac{2\pi}{T}$$

Given these samples, we can reconstruct $x(t)$ by generating a periodic impulse train in which successive impulses have amplitudes that are successive sample values. This impulse train is then processed through an ideal lowpass filter with gain T and cutoff frequency greater than ω_M and less than $\omega_s - \omega_M$. The resulting output signal will exactly equal $x(t)$.

In short, a bandlimited signal B, can be perfectly reconstructed by sampling the signal uniformly in time at rate $2B$ and convolving the samples with a sinc function, i.e.,

$$x(t) = \sum_{n=-\infty}^{\infty} x(n\Delta T)\frac{\sin(2\pi f_s t)}{2\pi f_s}, \quad f_s = 2B \text{ where } \Delta T = \frac{1}{f_s} \tag{2.0.1}$$

In the rest of this chapter, we give the necessary technical background for understanding the operation of CS-based receivers and their potential benefits to addressing the current challenges to wideband receiver design mentioned in Ch. 1. We start by giving background on conventional RF front-ends and analog-to-digital converters (ADC) and give further discussion of the specific challenges in realizing receivers with greater EIBW. We conclude the chapter by briefly reviewing basic CS theory and answer the question of how and why it can be used to achieve the stated goals.

2.1 Wireless Receiver Background

In its simplest form, a wireless transceiver system consists of a transmitter, a signal path (often referred to as the channel), and a receiver. Ultimately, the goal of a wireless receiver is to acquire information about physical signal(s) lying within a large swath of bandwidth. B. A modern receiver carries out this function by partitioning the system into three sections:

- An analog/RF front-end—to provide a proper impedance match to the antenna and condition the physical signals via frequency translation and filtering to convert them into a form suitable for digitization.

- An analog-to-digital converter (ADC)—to perform the digitization necessary for analysis of the acquired on a digital computing platform.

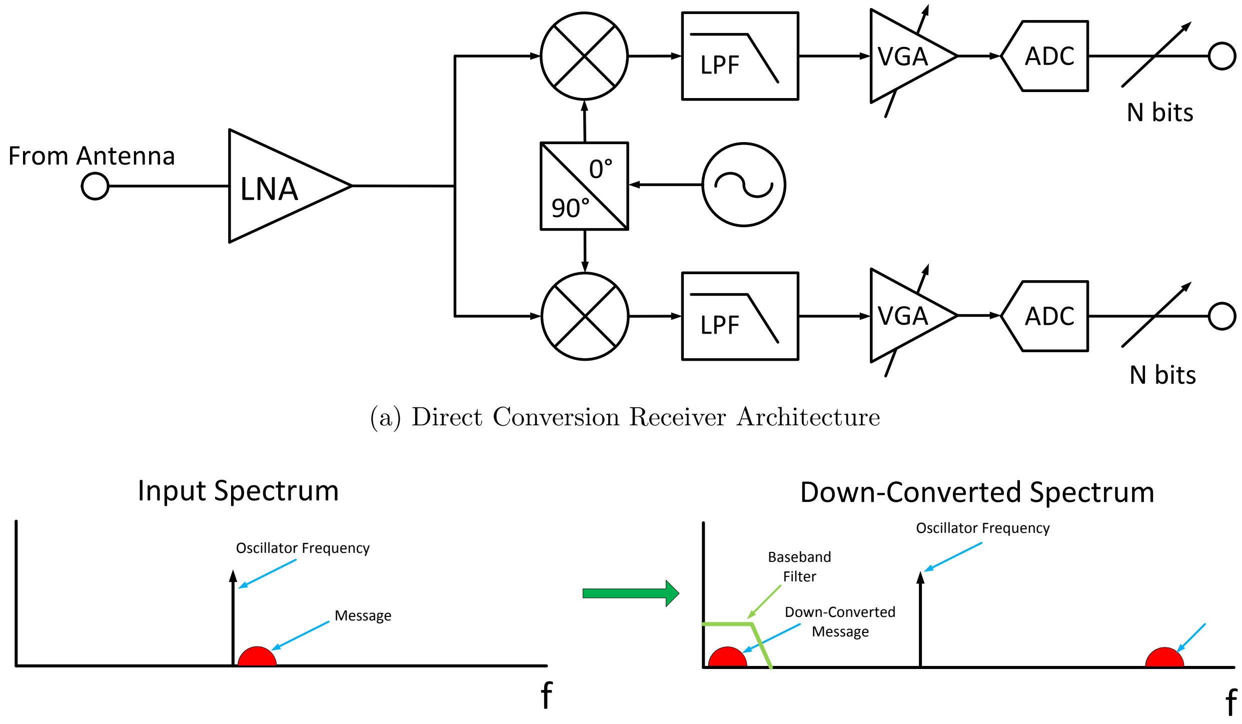
- A digital signal processing (DSP) back-end (implemented in either software or hardware) to extract the desired information from the obtained samples.

We describe the basic operating principles of the RF front-end and ADC below.

2.1.1 Radio Frequency Front-Ends

RF front-ends serve as the interface between the physical signals of interest and the digitizer. Its function is to condition the potentially high dynamic range, wide bandwidth, analog signals and convert them to signals that minimize the required digitization rate f_{adc} and facilitate conversion of information without degradation or loss to the digital domain. The first ancestor of today's RF front-ends trace back to the late 19th century. The first successful radio transmission was made by David Edward Hughes in 1879, but it would not be until the experiments of Heinrich Hertz in 1886 that the Hughes radio transmission would be recognized as the propagation of EM waves. The radio transmitter/receiver pair which

Heinrich Hertz used in his 1886 experiments consisted of discharging a high-voltage induction coil across a spark gap and connecting the output to an antenna. The receiver used to detect the emitted radio waves was another spark-gap formed from a piece of copper wire 1 *mm* thick bent into a 7.5 *cm* diameter circle. The “detection” consisted of adjusting the gap between two ends of the circle formed so that when it detected EM waves of a certain frequency a visible spark would appear. The spark-gap radio served as the basic architecture employed by wireless telegraphs up through the 1930s.



(a) Direct Conversion Receiver Architecture

(b) Frequncy Domain Operation of Direct Conversion Receiver

Figure 2.1: (a) Depiction of direct-conversion receiver architecture. (b) Illustration of the operation of the direct-conversion receiver in the frequency-domain. A low-noise-amplifier (LNA) amplifies the signal received at the antenna. The output of the LNA is then multiplied by a pure tone generated by a local oscillator (LO) with frequency f_{LO} which creates copies of the input spectrum shifted up (image) and down in frequency by f_{LO}. The downconverted copy is then filtered by a lowpass filter which attenuates the mirror spectrum as well as filters out excess noise. The baseband filter is also a part of the anti-aliasing filter to reduce SNR degradation from the noise-folding due to the impulse sampling operation of the ADC.

There have been many radio architectures developed since the spark gap radio. However, despite their relative sophistication, the basic principles of modern RF front-ends have remained largely the same since the superheterodyne vacuum-tube receiver was conceived of by Edwin Armstrong in 1924. While the superheterodyne architecture still sees wide use today, the architecture of choice for many monolithic receivers (such as those found in cellphones) is the direct-conversion (homodyne) receiver due to its relative simplicity. Indeed, the direct-conversion receiver was a topic of heavy research during the 1990s and early 2000s as part of the pursuit to realize fully-integrated monolithic cellphone receivers [20–24]. The essential operation of both architectures are the same and can be understood by examining the operation of the direct-conversion receiver architecture shown in Fig. 2.1.

Fig. 2.1a shows a direct-conversion receiver and depicts its operation in the frequency domain in Fig. 2.1b. In the context of digital communications, an RF transmitter will send a message $m(t) = a(t) \cdot \mathrm{e}^{j2\pi f_0 t}$ where $a(t) = a_I(t) + a_Q(t)$ is the complex baseband representation of the message $m(t)$ [25,26]. The purpose of the RF front-end is to downconvert the received message $m(t)$ back into its complex-baseband representation so that the desired information can be extracted. In general, the spectral occupancy of the message is known a priori and the signal is down-converted to minimize the digitization rate of the back-end ADCs. There are many figures of merit by which RF front-ends are evaluated and we refer the reader to [23, 24, 27] for more details. In this thesis we are primarily concerned with the design of wideband receivers, Table 2.1 lists the performance of state-of-the art receivers up to 2011.

RF BW	IF BW	NF	SFDR	Power	Area(mm^2)	Technology	Ref.
$0.2 - 2$ GHz	25 MHz	6.5 dB	79 dB	67 mW	0.13	65 nm CMOS	[28]
$0.8 - 6$ GHz	20 MHz	5 dB	70 dB	60 mW	3.8	90 nm CMOS	[29]
$0.4 - 0.9$ GHz	22 MHz	4 dB	79 dB	67 mW	1.0	65 nm CMOS	[30]
$0.1 - 3$ GHz	65 MHz	6 dB	55 dB	48.5 mW	2.4	130 nm CMOS	[31]
$0.05 - 2.4$ GHz	20 MHz	5.5 dB	—	60 mW	2	65 nm CMOS	[32]
$0.4 - 6$ GHz	20 MHz	3 dB	—	100 mW	2	40 nm CMOS	[33]

Table 2.1: List of RF front-ends reported in major journals or conferences during the years of 2006-2011.

.

2.1.2 Analog-to-Digital Converters

The function of an ADC is to translate analog quantities into the digital representation needed for storage and processing on a computer. The basic operation of an ADC is comprised of 4 operations:

- A continuous-time anti-aliasing filter.
- A sample-and-hold (S/H).
- A stage which quantizes the value held by the (S/H).
- A codifier to apply an encoding to the quantized value.

Metrics for Comparison

There are a large number of specifications which are used to classify and compare ADC performance [2, 13]. The three parameters of interest most commonly used for comparisons are the total power dissipation (P_{diss}) of the ADC, number of bits N_b of resolution (often referred to as the effective number of bits ENOB), and speed f_{adc}. ENOB is defined as

$$\text{ENOB} = \frac{\text{SINAD}_{dB} - 1.76}{6.02} \tag{2.1.1}$$

where SINAD and SNDR both refer to the signal-to-noise-and-distortion ratio. ENOB is generally favored to other metrics like signal-to-noise ratio (SNR), spurious-free-dynamic range (SFDR), and SINAD to characterize resolution. This is due to the fact that it takes into account the degradation in the stated resolution due to noise, distortion, as well as quantization noise unlike the other metrics listed.

In terms of how P_{diss}, f_{adc}, and N_b scale with one another, as a general rule of thumb, P_{diss} is proportional to f_{adc} and each additional bit of resolution requires an increase in P_{diss} by a factor of 2 (in practice this is between a factor of $2 - 4\times$ [2]). For many comparisons, the relationship of these three parameters are combined into a single figure-of-merit (FOM):

$$\text{FOM} = \frac{2^{\text{ENOB}} \cdot f_{\text{adc}}}{P_{\text{diss}}} \tag{2.1.2}$$

In terms of broad classification, ADCs are separated into two main types: Nyquist-rate and oversampling. These types differentiate ADCs based on the ratio of the input bandwidth BW_{ADC} to the digitization rate f_{adc}. For Nyquist-rate converters this ratio is high, typically in between $1/4 - 1/2$. In contrast, oversampling converters utilize the excess bandwidth by employing digital filters that remove noise and reduce quantization noise power. In general, every increase in the oversampling ratio by a factor of 4 leads to an improvement of the converter resolution by 1-*bit*. There are many specific ADC architectures that have been developed for both types, we refer the reader to [34–37] for thorough treatments. We simply mention here that ADCs that achieve the highest sampling rates (f_{adc}) are Nyquist-rate converters that employ a time-interleaving or pipelined architecture [2, 10, 38].

Trends and Challenges

Although there is no general consensus on when and in what form the ADC was first realized, there is a substantial amount of documentation on their development from the 1940s onward. The creation of the digital computer, starting with ENIAC (1942), was the key impetus behind commercial ADC development. Prior to the 1950s, ADCs were developed for application specific purposes such as the message encryption systems used during World War II (ADCs had virtually no commercial application at the time). The first ADC to be sold commercially was in 1954. It was produced by a company named Epsco, built using vacuum tubes, and digitized data at a rate of 50 *ksps* with a resolution of 11-*bits*. The ADC was named the DATRAC, it dissipated 500 *W*, occupied a volume of approximately 4ft.^{3}, and sold for a mere \$8000 [37]. Needless to say, much progress has been made since that time; Table 2.2 lists the state-of-the art for ADCs implemented in either CMOS or SiGe BiCMOS technologies as of 2011.

ADC Type	Speed	Bits	Power	Technology	Area(mm^2)	Ref.
Interleaved	2.6 $Gsps$	10	480 mW	65 nm CMOS	5.1	[12]
Interleaved	1.0 $Gsps$	12	575 mW	180 nm SiGe BiCMOS	2.35	[39]
Folding	1.0 $Gsps$	10	2.52 W	180 nm CMOS	49	[40]
Pipelined	800 $Msps$	12	105 mW	40 nm CMOS	0.49	[41]
Pipelined	125 $Msps$	16	385 mW	180 nm CMOS	15	[42]
Pipelined	100 $Msps$	12	130 mW	90 nm CMOS	0.4	[43]
Pipelined	50 $Msps$	12	4.5 mW	90 nm CMOS	0.3	[44]
Pipelined	50 $Msps$	10	9.9 mW	180 nm CMOS	1.4	[45]

Table 2.2: List of ADCs reported at the IEEE International Solid-State Circuits Conference (ISSCC) between 2009-2011.

.

The role of ADCs in modern electronics systems is in sharp contrast to that of the 1950s. The ever-increasing reliance on sophisticated DSP techniques in modern electronic systems have made the ADC of principle concern. This is particularly true in many emerging applications where common demands include higher bandwidth in a portable form-factor. At present, the primary limitations in realizing increased EIBW systems is the power dissipation associated with the necessary digitization rates. While ADCs with extremely high conversion rates have been realized [46,47], current implementations draw amounts of power that make their use impractical in power-constrained applications. This limitation is such that the design of present systems derives choices for various parameters area derived based upon the fraction of the power budget allocated to the operation of digitization. While remarkable advancements in ADC design continue to be made, several observed trends in the rates of improvement in certain ADC metrics bode poorly for creating higher EIBW systems.

One trend of concern is the historic relative rates of improvement of ADCs and digital computing devices. This is illustrated in Fig. 2.2 where a factor of *100* difference in improvement of the relevant figures of merit during a 15 year time span is observed. The growing disparity in performance only emphasizes the ADC's role as a major bottleneck in achieving superior performance.

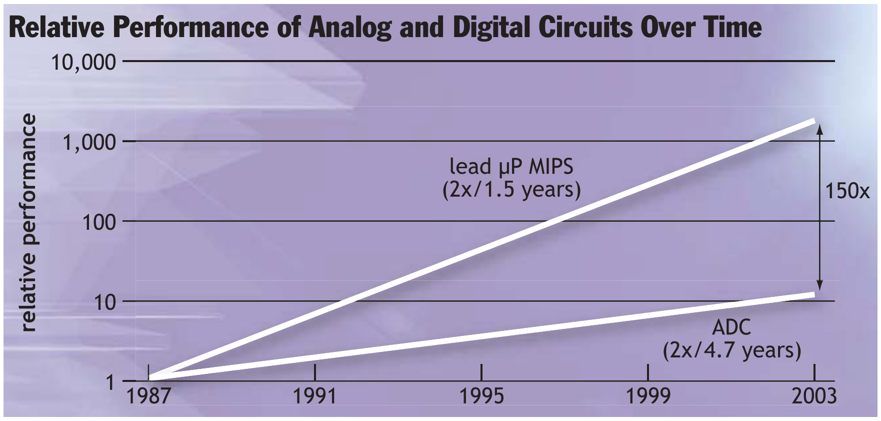
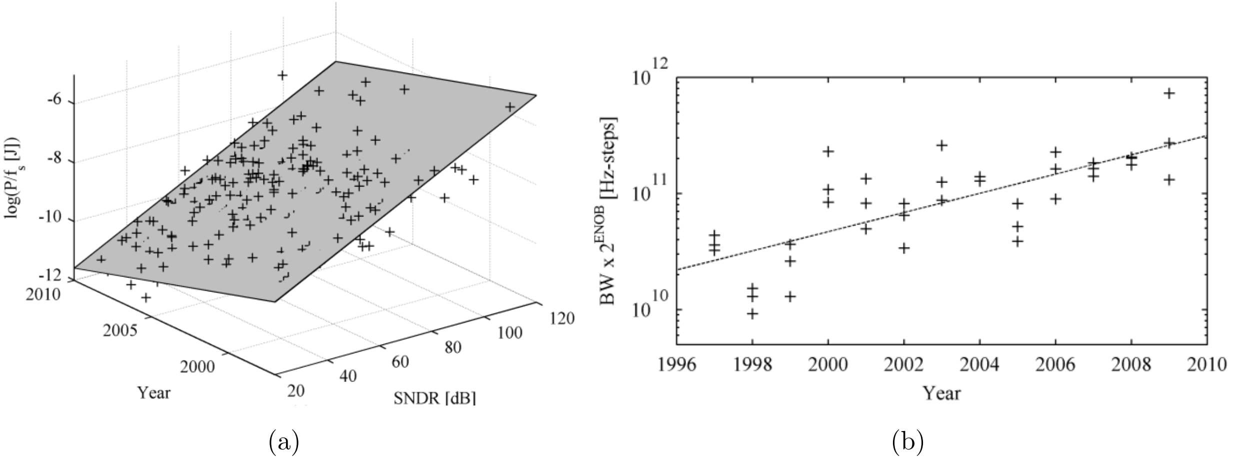


Figure 2.2: The relative improvements in performance of microprocessors and ADCs in terms of their respective figures of merit. While the performance of both systems have improved at an exponential rate over time, ADCs have only improved by a factor of 10 in sharp contrast to microprocessors which have improved by a factor of 1000 in the period between 1987 and 2003. The depicted data is from [10].



(a)

(b)

Figure 2.3: Trends in ADC performance (Figures from [2]): (a) 3-D fit to conversion energy. The fit plane has a slope of 0.5 × /1.9 years along the time axis. (b) Fit to speed-resolution product of top 3 designs in each year. The slope of the fit line is 2 × /3.6 years. The data plotted is based on survey data collected from a period of 12 years ending in the year 2010. Data points are designs presented at the IEEE International Solid-State Circuits Conference (ISSCC) and the VLSI Circuit Symposium from this time period.

Another trend of concern is the rates of improvement in converter power efficiency and speed-resolution product. Fig. 2.3b and 2.3a (taken from [2]) show that energy conversion efficiency halves every two years while speed-resolution product doubles every four years.

Based on empirical evidence, it has also been suggested that aggressively pushing the sampling speed of ADCs in a given process technology sacrifices enormous power efficiency [2]. In light of the slowing of process technology scaling, which is highly correlated with the gains in conversion efficiency, it seems that relying solely on efforts in realizing superior ADCs is an approach with rapidly diminishing returns. Thus, it appears that an increase in system bandwidths will have to be achieved by means beyond advances in ADC technology.

The approach taken in this thesis is to utilize results from the field of CS which takes advantage of the mathematical structure, specifically *sparsity* and *compressibility*, found in many signals of interest to reduce the required *digitization* rate for observation of a given EIBW. This approach is complementary to existing approaches and has two distinct benefits: the first is that for many applications of interest CS enables the realization of higher EIBW systems at any underlying level of mixed-signal technology, and the second is that CS reduces the dependence of system bandwidth on available ADC performance and ties it to the amount of desired information. A review of CS theory and how it can be applied to designing wideband receivers is the subject of the next section.

2.2 Compressed Sensing

In this section we briefly review CS theory. CS is a relatively new signal processing technique which enables sub-Nyquist rate signal acquisition from what was previously thought to be insufficient data. For an excellent introduction to the topic, see [16]. Before we review the detailed mathematical results, we discuss the central concepts in an intuitive style.

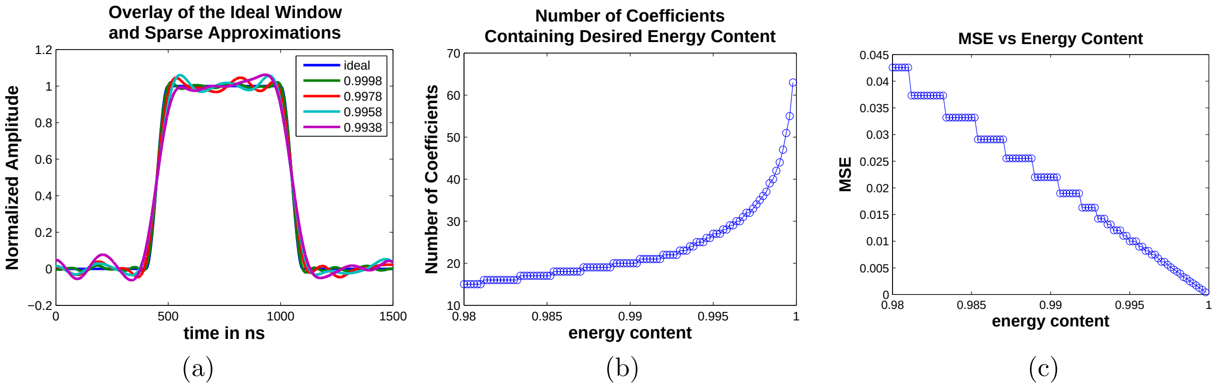
The Shannon-Nyquist sampling theorem is implicitly or explicitly used in many applications to set the minimum digitization rate to greater than twice the worst-case spectral occupancy of the input (B), i.e., $f_{\text{nyq}} > 2B$. As Shannon-Nyquist sampling is optimal in the absence of a priori knowledge other than the signal being bandlimited [48], it is natural to ask how this is possible? In short, bandlimitedness of signals is an accurate but not very sharp assumption. In many applications one usually knows much more about the targets signals. In this sense, CS exploits prior information just as Bayesian methods exploit prior belief of distributions (although CS results are independent and often much stronger than Bayesian results). The results of CS are due to essentially two insights:

1. The first insight is that many signals (or features of the signal) we are interested in acquiring are structured and depend on far fewer degrees of freedom than the bandwidth suggests. That is, when expressed with the right dictionary, the signal has a concise representation. This is an insight routinely exploited by image compression algorithms which are known to be compressible in wavelet domains. This "compressibility" motivates the idea that undersampling the prescribed Nyquist rate is possible.

2. The second is that the assumed compressibility of the signal has bearing on the data acquisition process itself. The assumption of compressibility of the signal naturally lends itself to recasting the signal recovery operation from the conventional framework of interpolating sampled data to estimating the signal from acquired measurements under constraints imposed by the assumed compressibility. In undersampled situations, this leads to rethinking the data acquisition process to prevent information loss during sampling. A natural idea is to restructure the measurements so that they capture more "global" information.

The two insights above directly motivate the two concepts central to CS theory: *sparsity* and *incoherence*. Sparsity captures the idea that many high-dimensional signals can be represented, without perceptual loss, using a relatively small set of coefficients when expressed using an appropriate signal dictionary. For example, Fig. 2.4 shows an ideal trapezoidal radar pulse-envelope as well as sparse approximations to it. The approximations are made by performing a discrete cosine transform (DCT) on the ideal pulse and only keeping the minimum number of coefficients that retain a specified percentage of the total signal energy[1]. We see that capturing 99.9% of the signal energy would require greater than 60 coefficients, whereas retaining even just 98% of the signal energy would require only 15 coefficients. What is notable is the factor of 4× reduction in data realized with minimal (if any) perceptual loss. The way in which sparsity is exploited by CS to achieve sub-Nyquist rate signal acquisition is

[1] This is commonly referred to as thresholding.



(a) (b) (c)

Figure 2.4: An example of the effect of approximating an ideal trapezoidal radar-pulse window by thresholding. The approximations are made by taking the DCT of the input and retaining only those coefficients that are above a certain value (thresholding). The threshold is set by choosing a percentage of the total signal energy to retain. (a) shows an overlay of the ideal window as well as approximations of the window with the threshold set to various percentages of the original energy level. (b) shows the number of coefficients that were needed to retain $98 - 99.98\%$ of the total signal energy. (c) shows the mean-squared error (MSE) of the approximation normalized to the total signal energy. In radar applications, a typical boundary for acceptable error in the recovered signal is MSE < 0.1.

based on the concept of incoherence. Incoherence captures the idea of dissimilarity between any two representations; two bases are said to be incoherent if any signal having a sparse expansion in one of them must be dense in the other. An example of an incoherent pair comes from the classical time-frequency duality. A signal sparse in time, e.g., a Dirac-delta function has a dense spectrum. Similarly, a single tone is sparse in the Fourier domain but dense in time. The remarkable result of CS is that when a signal is well-approximated by a sparse-representation in some known basis, it can be acquired by taking a small number of measurements that are *incoherent* with the basis in which the signal has a sparse representation [49]. The principal result of CS is that using random measurements is a good choice for implementing incoherent measurements and that the signal can be recovered by solving a ℓ_1-norm minimization problem. The idea that random measurements are efficient can be intuitively explained by the observation that noise has a dense representation in any basis. Thus, a random measurement would acquire more global as opposed to specific information. The intuition behind recovering the signal via ℓ_1-norm minimization is motivated by the fact that the ℓ_1-norm is known to produce sparse solutions [50, 51] which is a natural goal given the assumption of compressibility.

We now give a more quantitative treatment of the theoretical results. The goal of both Shannon-Nyquist and CS sampling is to obtain information about a continuous-time signal $x(t)$ by recording the correlations of $x(t)$ with a set of linear functionals $\phi_i(t)$

$$y_i = \langle x, \phi_i \rangle, \quad i = 1, \dots, M \tag{2.2.1}$$

This is the standard setup. If the $\phi_i(t)$ are a set of uniformly time-shifted Dirac deltas $\delta(t)$ then $\mathbf{y} = \sum_i y_i$ is a vector of uniformly sampled time points of $x(t)$ which is the case of Nyquist sampling. In the case that the $\phi_i(t)$ are a set of sinusoids uniformly spaced in frequency, then $\mathbf{y}$ represents a vector of Fourier coefficients.

For the purposes of this work, CS can be viewed as a novel signal acquisition technique which enables the recovery of signals that have a *sparse* representation in some basis $\Psi = [\psi_1(t), \dots, \psi_N(t)]$ at a sub-Nyquist rate. An s-sparse vector $\boldsymbol{\alpha} \in \mathbb{R}^N$ is one in which all but s of the entries are equal to zero ($\|\boldsymbol{\alpha}\|_0 = s$). We say that $x(t) = \sum_{i=1}^{N} \alpha_i \psi_i(t)$ has an s-sparse representation if $\boldsymbol{\alpha}$ is s-sparse. In this case we call $\Psi = [\psi_1(t), \dots, \psi_N(t)]$ the sparsifying dictionary. We deal mainly with discrete, finite-length samples, so we are primarily concerned with $\mathbf{x} \in \mathbb{R}^N$ ($\mathbf{x}$ is the Nyquist-rate sampled version of $x(t)$), and $\Psi \in \mathbb{R}^{N \times N}$ is a matrix whose columns $\boldsymbol{\psi}_i$[2]. While one could develop a continuous-time/space analog of CS [16], we limit our attention to discrete signals $\mathbf{x} \in \mathbb{R}^N$. The reason for this is that CS specifically treats the case of *undersampled* situations in which the number of measurements (digitized samples) $M = |\mathbf{y}|,\ \mathbf{y} \in \mathbb{R}^M$ is much smaller than the dimension $N = |\mathbf{x}|,\ \mathbf{x} \in \mathbb{R}^N$, the Nyquist sampled version of $x(t)$. One of the initial results of CS is described in the theorem below:

Theorem 2 (From [54]). *Fix* $\mathbf{x} \in \mathbb{R}^N$ *and let* $\mathbf{x} = \Psi\boldsymbol{\alpha}$. *Suppose that the coefficient sequence* $\boldsymbol{\alpha}$ *is* s*-sparse. Select a set* $\Sigma \subset \{1, \dots, N\}$ *of* $|\Sigma| = M < N$ *measurements in the* Φ *domain*

[2] $\mathbf{x}$ can be a discrete representation of $x(t)$ sampled at any resolution, however, in this thesis we only consider $\mathbf{x}$ sampled on the discrete Nyquist grid. In addition Ψ does not necessarily have to be $N \times N$. Moreover, Ψ does not have to be a basis. The Ψ used in this thesis is the Gabor dictionary [52] which is highly overcomplete $\Phi \in \mathbb{R}^{L \times N},\ L \gg N$ and is a sparsifying dictionary for signals that exhibit time-frequency sparsity [53].

uniformly at random. Then if

$$M \geq C \cdot \mu^2(\Phi, \Psi) \cdot s \cdot \log(N) \tag{2.2.2}$$

for some positive constant C, the solution to

$$\min_{\hat{\boldsymbol{\alpha}} \in \mathbb{R}^N} \|\hat{\boldsymbol{\alpha}}\|_1 \quad \textit{subject to} \quad y_i = \langle \Psi \hat{\boldsymbol{\alpha}}, \boldsymbol{\phi}_i \rangle, \quad \forall i \in \Sigma \tag{2.2.3}$$

exactly recovers $\boldsymbol{\alpha}$ with overwhelming probability[3].

In other words, signal recovery is performed by finding the signal $\hat{\mathbf{x}} = \Psi\hat{\boldsymbol{\alpha}}$ among all candidate signals that is both consistent with the acquired measurements $\mathbf{y}$ and has coefficients $\hat{\boldsymbol{\alpha}}$ with minimum ℓ_1-norm ($\|\hat{\boldsymbol{\alpha}}\|_1 = \sum_{i=1}^{N} |\alpha_i|$). The accuracy of recovery is guaranteed provided that the number of measurements M is greater than specified by Eq. 2.2.2. The constant (C, Eq. 2.2.2) is bounded (but pessimistic); in practice, it is small, i.e., 4 [55]. We see that the minimum required M is a function of the coherence $\mu(\Phi, \Psi)$ between the sensing matrix Φ and the sparsifying basis Ψ where

$$\mu(\Phi, \Psi) = \sqrt{N} \cdot \max_{1 \leq i,j \leq N} |\langle \boldsymbol{\phi}_i, \boldsymbol{\psi}_j \rangle|. \tag{2.2.4}$$

is the largest value obtained by correlation between any one element (column) from Φ and one element from Ψ. In order to minimize $\mu(\Phi, \Psi)$ random measurement matrices Φ are prescribed by CS since they are largely incoherent with any fixed Ψ. Note, the Φ are chosen non-adaptively.

The above results treat the highly artificial situation of exactly sparse signals with measurements uncorrupted by noise. In general, objects of interest are not sparse but can be well approximated by sparse signals. The specifics of how the theory is extended to deal with real signals is beyond the scope of this thesis. However, for the sake of completeness, we quickly state the basic results in the literature which extend CS to more practical scenarios.

[3] To be precise, this result assumes the signs and support of the signal are random; such assumptions will not be needed in Theorem 3.

The following treats the case of measurements $y = \Phi\mathbf{x} + \mathbf{z} = \Phi\Psi\boldsymbol{\alpha} + \mathbf{z} = A\boldsymbol{\alpha} + \mathbf{z}$ (where $A = \Phi\Psi$) corrupted with noise $\mathbf{z} \sim \mathcal{N}(0, 1)$. For such cases the signal recovery operation is modified from Eq. 2.2.3 to:

$$\min_{\boldsymbol{\alpha}} \; \|\hat{\boldsymbol{\alpha}}\|_1 \quad \text{such that} \quad \|A\hat{\boldsymbol{\alpha}} - \mathbf{y}\|_2 \leq \varepsilon \tag{2.2.5}$$

This is known as basis pursuit denoising [56–58] (BPDN). The modification amounts to finding the $\hat{\mathbf{x}} = \Psi\hat{\boldsymbol{\alpha}}$ most consistent with the measurements $\mathbf{y}$ with a maximum deviation specified by ε. The standard model of CS utilizes a tool called the restricted isometry property (RIP) [59] to evaluate the accuracy of the signal recoveries given by Eq. 2.2.5 for a given A.

Definition 1 (RIP (Restricted Isometry Property))**.** *The restricted isometry constant δ_s of order s for a matrix A is the smallest number such that*

$$(1 - \delta_s)\|\boldsymbol{\alpha}\|_2^2 \leq \|A\boldsymbol{\alpha}\|_2^2 \leq (1 + \delta_s)\|\boldsymbol{\alpha}\|_2^2 \tag{2.2.6}$$

holds for all s-sparse vectors $\boldsymbol{\alpha}$.

Matrices A that possess smaller δ_s give better signal recovery guarantees. Consequently, a natural goal for using CS is to find A which minimize δ_s. The desirability of smaller δ_s can be understood by considering several different types of A. For example, if $A \in \mathbb{R}^{N \times N}$ is an orthogonal matrix, then $\delta_s = 0 \; \forall k$ and is the best possible constant. In cases where $A \in \mathbb{R}^{M \times N}$ ($M < N$), A will have a non-trivial nullspace. In this case if $s \geq M$ then $\delta_s \geq 1$. In the case that $s < M$ it is possible to find a matrix such that $\delta_s < 1$. In general, adding rows to a matrix A will improve (decrease) its RIP constant (δ_s). A matrix is said to satisfy RIP whenever it belongs to a class of matrices that, with high probability, have $\delta_{\gamma s} \ll 1$ whenever $M \simeq s\log(N)$, where γ is typically 2 or 3. There are many different versions of RIP [60–62] that have been used to show an even greater number of results. A concise version is stated below.

Theorem 3 (From [59]). *Let $\delta_{2s} < \sqrt{2} - 1$ and $\mathbf{y} = A\boldsymbol{\alpha} + \mathbf{z}$ for any (possibly deterministic) $\mathbf{z}$, with $\|\mathbf{z}\|_2 \leq \varepsilon$. Denote the best s-term approximation to $\boldsymbol{\alpha}$ with $\boldsymbol{\alpha}_s$. Then basis pursuit denoising* (2.2.5) *with this ε gives an estimator $\hat{\boldsymbol{\alpha}}$ satisfying*

$$\|\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}\| \leq C_0 \frac{\|\boldsymbol{\alpha} - \boldsymbol{\alpha}_s\|_1}{\sqrt{s}} + C_1 \varepsilon$$

for some constants C_0 and C_1.

There are many possibilities for implementing incoherent measurements [63]; the RIP is a sufficient but not *necessary* criteria to certify the suitability of a matrix. In general, for a given M, random matrices give the best δ_s. The initial results of CS theory were proven using randomly generated Gaussian and Bernoulli matrices [61]. In this thesis we restrict our attention to random Bernoulli measurement matrices. This is because Bernoulli measurement matrices are highly amenable to hardware implementation as multiplication by a random sequence of ±1 values amounts to modulation of the signal with a pseudo-random bit sequence PRBS. This is particularly convenient because PRBSs can be generated with a linear feedback shift-register (LFSR); LFSRs have well-known and efficient implementations, see [64] for details. So now that the theory has been established, the question that remains: how is the sampling procedure corresponding to the linear inverse problem $\mathbf{y} = \Phi\mathbf{x}$ implemented in hardware? The basic idea is to implement incoherent random sampling by correlating the signal of interest with a PRBS and is the subject of the next chapter, Ch. 3.

Chapter 3

The Random Modulation Pre-Integrator (RMPI)

Almost immediately after its inception in the seminal papers of [3–5], the field of CS inspired a fundamental reconception of many physical signal acquisition and processing platforms. The beginning of this renaissance has already seen the redesign of cameras [65], medical imaging devices [66], and RF receivers [9, 16, 18]. The *initial* allure of CS for researchers in the field of RF systems design, came from the assertion that signals with spectral occupany $< B$ (B is the observation bandwidth) could be acquired (without information loss) by implementing incoherent sampling at a rate proportional to the occupied degrees of freedom. This allure spurred a number of research efforts at fullfilling that goal including those that resulted in the work that is presented in this thesis.

The benefits and feasibility of applying any new theoretical result in designing a robust physical device is, of course, never certain and almost always difficult. With respect to research aimed at implementing CS-based receivers, this is evidenced by the relatively long period between the seminal papers of CS (versions of which were available on the arXiv as early as 2004 [3]) to the first reported high-speed implementation in this work. The content of this chapter chronicles that process with two specific aims:

1. The first is to answer the general question of how the theoretical sampling framework of CS is mapped to physical sampling hardware that can utilize the results. Specifically, CS prescribes incoherent sampling that results in the generation of samples compatible

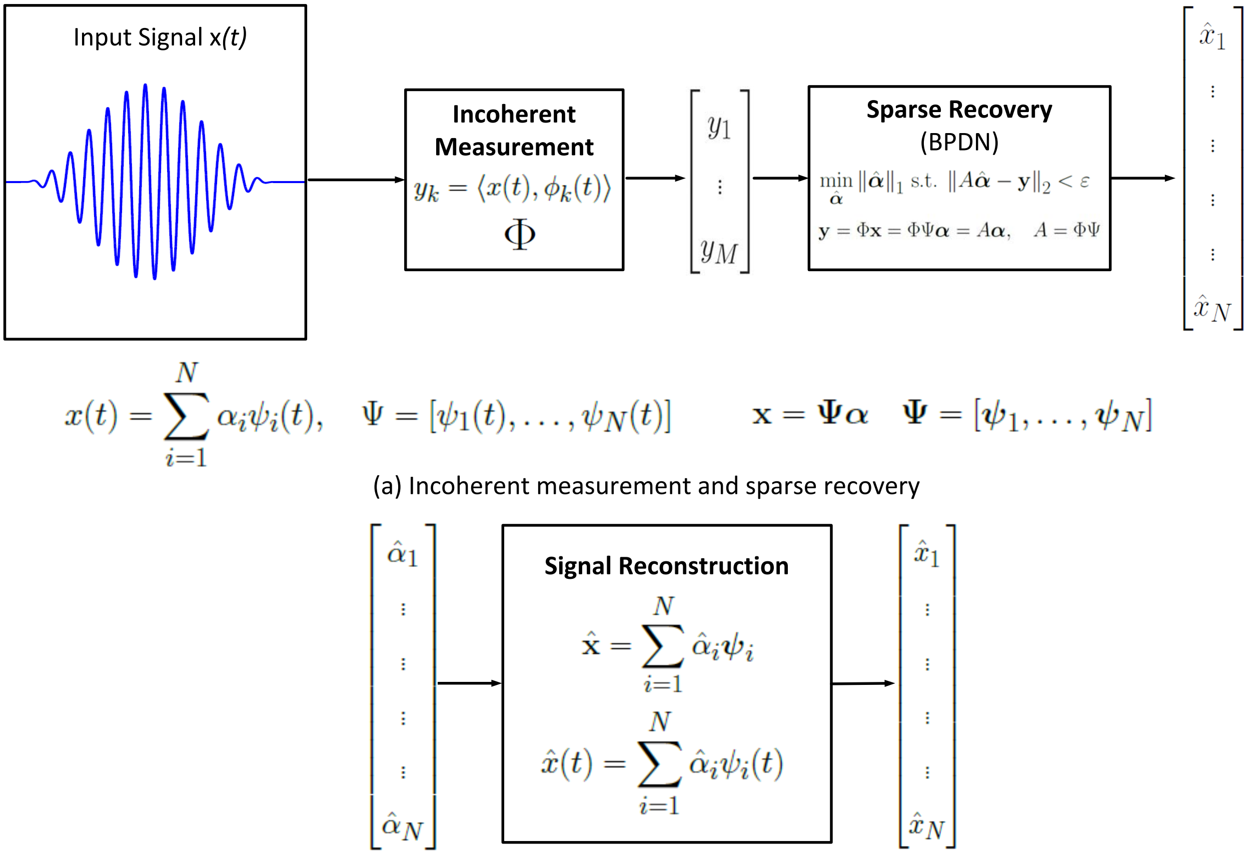
with CS-recovery methods.

2. The second is to describe the operation of the RMPI analytically in preparation for presentation of the material in Ch. 4 which addresses the question of how to design the RMPI.

We fulfill these aims by taking a chronological approach to the exposition. We begin by describing how the CS-framework was utilized in the first description of a CS-based receiver known as the random demodulator (RD) [8,18]. We then proceed to describe the theoretical framework of the RMPI which closely follows a similar discussion in [9]. Following this discussion,we describe the general class of signals that can be acquired by the RMPI. We conclude with a short review of other RMPI-like systems and sub-Nyquist proposals.

3.1 Implementing Compressed Sensing in Hardware

There are several immediate questions that arise when undertaking the task of mapping the theoretical framework of CS to the design of physical sampling hardware. For instance, how is the discrete-time linear formulation of CS theory adapted to acquiring a bandlimited continuous-time input $x(t)$? Before we answer this question, we define some notation commonly employed in descriptions of CS theory. Although the notation was defined in Ch. 2, we restate it here for the convenience of the reader. Let $x(t) = \sum_{i=1}^{N} \alpha_i \psi_i(t)$ represent the signal of interest that henceforth will be assumed to have an s-sparse representation $\boldsymbol{\alpha} \in \mathbb{R}^N$ in signal dictionary $\Psi = [\psi_1(t), \ldots, \psi_N(t)]$. We further define $\mathbf{x} \in \mathbb{R}^N$ to be the discrete representation of $x(t)$ sampled on the Nyquist-grid thus allowing us to define $\mathbf{x} = \Psi\boldsymbol{\alpha}$ where Ψ is composed of columns $\boldsymbol{\psi}_i \in \mathbb{R}^N$ that represent the corresponding $\psi_i(t)$ sampled on the Nyquist grid. Consider, the goal of our system is to recover a finite-length discrete representation of the time-domain of input $x(t)$ during a time-interval of duration T_{win}. Thus, the input to hypothetical system is a continuous-time signal $x(t)$, and the output is a discrete-time approximation of this signal. The basic model of CS is formulated as solving a linear


Input Signal x(t)
Incoherent Measurement
$y_k = \langle x(t), \phi_k(t) \rangle$
Φ
y_1
y_M
Sparse Recovery (BPDN)
$\min_{\hat{\boldsymbol{\alpha}}} \|\hat{\boldsymbol{\alpha}}\|_1$ s.t. $\|A\hat{\boldsymbol{\alpha}} - \mathbf{y}\|_2 < \varepsilon$
$\mathbf{y} = \Phi\mathbf{x} = \Phi\Psi\boldsymbol{\alpha} = A\boldsymbol{\alpha}, \quad A = \Phi\Psi$
$\hat{x}_1$
$\hat{x}_N$
$x(t) = \sum_{i=1}^{N} \alpha_i \psi_i(t), \quad \Psi = [\psi_1(t), \ldots, \psi_N(t)]$
$\mathbf{x} = \boldsymbol{\Psi\alpha} \quad \boldsymbol{\Psi} = [\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_N]$
$\hat{\alpha}_1$
$\hat{\alpha}_N$
Signal Reconstruction
$\hat{\mathbf{x}} = \sum_{i=1}^{N} \hat{\alpha}_i \boldsymbol{\psi}_i$
$\hat{x}(t) = \sum_{i=1}^{N} \hat{\alpha}_i \psi_i(t)$
$\hat{x}_1$
$\hat{x}_N$

(a) Incoherent measurement and sparse recovery

(b) Full signal reconstruction

Figure 3.1: Illustration of the 3 principle operations of a CS-based receiver. (a) Φ represents the operation of incoherent measurement. The incoherent measurement is done in the continuous-time domain. The output samples y_i represent correlations between elements of a set of predetermined test functions with the input. The samples $\mathbf{y}$ are then used in a sparse recovery operation to determine an estimate $\hat{\boldsymbol{\alpha}}$ of $\boldsymbol{\alpha}$. (b) Once the $\hat{\boldsymbol{\alpha}}$ are obtained, they can be used to obtain an estimate $\hat{\mathbf{x}}$ of $\mathbf{x}$.

inverse problem

$$\mathbf{y} = \Phi\mathbf{x} = \Phi\Psi\boldsymbol{\alpha} = A\boldsymbol{\alpha}, \quad A = \Phi\Psi, \tag{3.1.1}$$

where $\mathbf{y}$ are the measurements (the output of our device), and $\Phi \in \mathbb{R}^{M\times N}$, $(M < N)$ is the underdetermined matrix representing the incoherent measurement operation. The "robust" version of the recovery procedure implements ℓ_1-norm minimization via basis pursuit denoising described by Eq. 2.2.5 to recover an estimate of the coefficients of $\boldsymbol{\alpha}$ which we denote $\hat{\boldsymbol{\alpha}}$. The estimate $\hat{\boldsymbol{\alpha}}$ can then be used to generate an estimate of the full-signal $\hat{\mathbf{x}}$.

Thus, The three operations that are performed by a CS-based receiver chain consists of three parts:

1. A measurement device which implements incoherent measurement via correlation of $f(t)$ with elements of a signal dictionary $\Phi = [\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_M]$ that is incoherent to the sparsifying dictionary $\boldsymbol{\Psi}$. Thus we write
$$y_i = \langle x(t), \phi_i(t) \rangle. \tag{3.1.2}$$
2. A sparse-recovery operation which returns $\hat{\boldsymbol{\alpha}}$.
3. Full signal reconstruction using the known Ψ to obtain $\hat{\mathbf{x}}$.

In this work, we are primarily concerned with designing hardware the produces the incoherent measurements prescribed by Eq. 3.1.2. An obvious choice for implementation of incoherent measurement in hardware is time-domain correlation of $x(t)$ with PRBSs. This corresponds to the use of Bernoulli matrices which, as discussed at the end of Ch. 2, is both an admissible (satisfies RIP, Def. 1) and convenient choice for Φ from both the perspective of theory and implementation in hardware. It is natural that this was the approach taken by the RD.

Before moving on to discussion of the RMPI architecture, we point out that the RMPI is an architecture that employs a parallel bank of RDs with a common input. Therefore, much of the operation of the RMPI can be understood through discussion of the RD. The mathematical analysis of the two systems is essentially the same. Thus, results proven for the RD require little effort to extend to the RMPI. There are however, a few significant differences between the RMPI and the RD in terms of practical use and the signal processing required at the back end. In addition, the signal model of the RMPI in this work is subtly different from the one used in [9] and will be given some attention in Ch 3.3. For a more detailed treatment we refer the reader to [55].

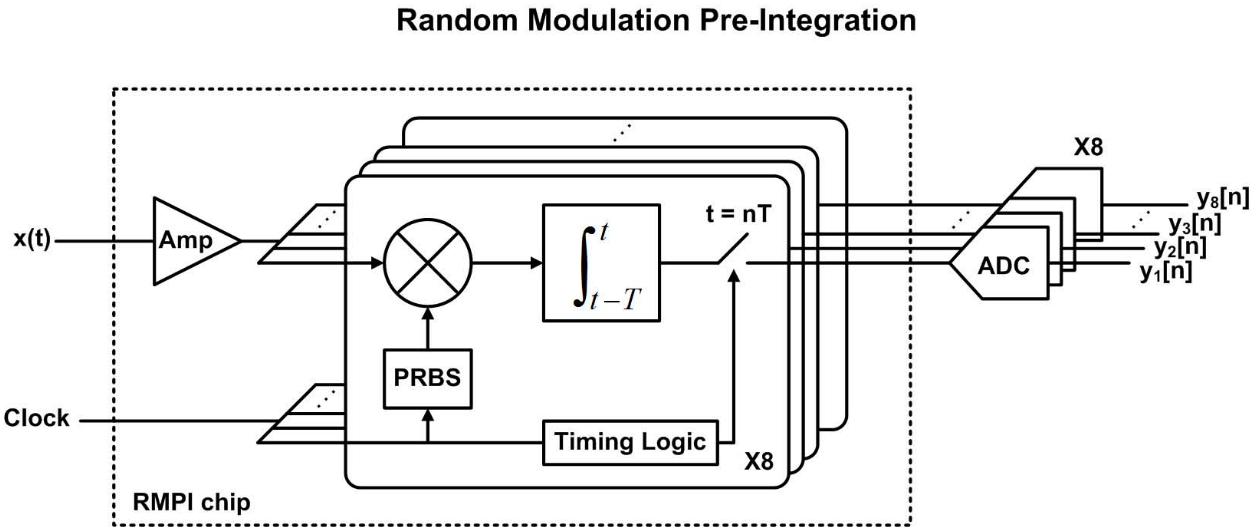


Figure 3.2: A block diagram of the random-modulation pre integrator architecture.

3.2 RMPI Architecture

A general system diagram of the RMPI is shown in Fig. 3.2. The RMPI is composed of a bank of $\mathrm{N_{ch}}$ parallel channels. Each channel essentially implements the random demodulator (RD) [9] and generates incoherent samples through time-domain correlation of input signal $x(t)$ with a distinct PRBS, which we denote $c_i(t)$ (we also refer to this as the "chipping" sequence), $i \in \{1, \ldots, \mathrm{N_{ch}}\}$, over a fixed integration time T_{int}. The correlation is implemented by modulating $x(t)$ with $c_i(t)$ followed by performing an integrate-and-dump[1] operation on the modulated product $x(t) \cdot c_i(t)$, i.e., random modulation of the input prior to integration, hence the name RMPI. The output of the integrator is reset every T_{int}. In general, denoting T_{win} the duration of the signal window of interest, $T_{\mathrm{win}} = LT_{\mathrm{int}}$ where $L \in \mathbb{Z}^+$ and typically $L > 1$. The terminal value of the integrator after each integration window is digitized by an ADC, so each channel produces several digitized outputs for each window T_{win} of interest.

[1] In an integrate-and-dump operation, the output of the integrator is reset to a predefined value immediately following sampling of the output.

3.2.1 Analytical Framework

We now present the mathematical model of the ideal RMPI. As the operation of all channels in the RMPI are essentially that of the RD, it is sufficient to describe the operation of one channel and extrapolate the operation of parallel channels. The description of the ideal RD in this section closely follows a similar discussion in [9]. To model the RMPI, we focus on the two basic functions of each channel: modulating the input $x(t)$ with $c_i(t)$ and sampling/resetting successive windowed integrations of the product over fixed windows of duration T_{int}. The digitized output samples of channel i can then be written

$$y_i[m] = \int_{(m-1)T_{\text{int}}}^{mT_{\text{int}}} x(t)c_i(t)dt. \tag{3.2.1}$$

Although it is not the only choice [67], for the purposes of this thesis we assume $c_i(t)$ is generated with a PRBS. Denoting the rate at which the PRBS is toggled as f_{chip}, we can write the chipping sequence for channel i

$$c_i(t) = \epsilon_{i,n},\ t \in [t_{n-1}, t_n),\ t_n = n\Delta T,\ n \in \mathbb{Z}^+, \tag{3.2.2}$$

where $\Delta T = 1/f_{\text{chip}}$. The coefficients of the PRBS are $\epsilon_{i,n} \in \{\pm 1\}$; in practice, these are a fixed sequence, but they are picked to mimic an independent and uniformly random $\{\pm 1\}$ sequence. For robust operation of the system in acquiring information lying in a bandwidth B, it is preferable for $f_{\text{chip}} \geq f_{\text{nyq}} = 2B$, see Ch. 4.4.2 for details. The important point to note here is that the back-end digitization rate f_{adc} occurs at rate $f_{\text{adc}} = 1/T_{\text{int}} \ll f_{\text{nyq}}$.

We now formulate Eq. (3.2.1) in the framework of Eq. (3.1.1) in order to relate it to the digital reconstruction process. We follow the description of the RD in [9] and incorporate our modifications where necessary. Our first step is choosing an appropriate discrete-representation $\mathbf{x}$ of continuous-time input $x(t)$. Substituting Eq. (3.2.2) into Eq. (3.2.1), we get the following expression for the sample produced by channel i after during the m^{th}

integration window

$$y_i[m] = \sum_{n \in \Omega_m} \epsilon_{i,n} \int_{t_{n-1}}^{t_n} x(t)dt, \tag{3.2.3}$$

where $\Omega_m = \{n : (m-1)T_{\text{int}} < (n-1/2)\Delta T < mT_{\text{int}}\}$ ($|\Omega_m| = T_{\text{int}}/\Delta T = N_{\text{int}} \in \mathbb{Z}^+$). Let $\boldsymbol{\epsilon}_i = [\epsilon_{i,1}, \ldots, \epsilon_{i,N}]$ and choose a discretization of $\mathbf{x} = [x_1, \ldots, x_N]$ where

$$x_n = \int_{t_{n-1}}^{t_n} x(t)dt,\ n \in \{1, \ldots, N\}, \tag{3.2.4}$$

then we can write

$$y_i[m] = \sum_{n \in \Omega_m} \epsilon_{i,n} x_n = \langle \boldsymbol{\epsilon}_i, \mathbf{x} \rangle_{\Omega_m} = \langle \boldsymbol{\phi}_m, \mathbf{x} \rangle. \tag{3.2.5}$$

Thus, in the general case where the recovery window spans several integration windows $N/N_{\text{int}} = L > 1$ ($N = T_{\text{win}}/\Delta T \in \mathbb{Z}^+$), we can model the action of channel i for recovery window of duration T_{win} in matrix form as the composition of two matrices. The first matrix $D = \text{diag}\{\boldsymbol{\epsilon}_i\} \in \mathbb{R}^{N \times N}$ represents the random modulation operation

$$D = \begin{bmatrix} \epsilon_1 & & & \\ & \epsilon_2 & & \\ & & \ddots & \\ & & & \epsilon_N \end{bmatrix}, \tag{3.2.6}$$

and the second matrix (H) represents the accumulate and dump sampling. H can be modelled by a matrix where each row corresponds to one integration window. Within each row, the entries of the row corresponding to times within the integration window are set to 1 and 0 otherwise. For example, if we have an $x \in \mathbb{R}^{N \times 1}$ with $N = 4N_{\text{int}}$, e.g., $N = 16,\ N_{\text{int}} = 4$, the matrix that represents the continuous time integration for $\mathbf{x}$ of the form given in 3.2.4 can be written

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}. \tag{3.2.7}$$

In general, $H \in \mathbb{R}^{M \times N}$ with $M = N/N_{\text{int}} = L$. Thus, if we use the discretization of $x(t)$ given in 3.2.4, then we can express the linear transfer function Φ as:

$$\begin{aligned} &\Phi\mathbf{x} = HD\mathbf{x} \\ &H \in \mathbb{R}^{M\times N}, \quad D \in \mathbb{R}^{N\times N}, \quad \text{and } \mathbf{x} \in \mathbb{R}^{N\times 1} \end{aligned} \tag{3.2.8}$$

Extending the example given in 3.2.6 and further assuming that $\boldsymbol{\epsilon}_i = [\epsilon_1, \ldots, \epsilon_5]$ repeats every 5 values, the Φ matrix would be written

$$\Phi = \begin{bmatrix} \epsilon_1 & \epsilon_2 & \epsilon_3 & \epsilon_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \epsilon_5 & \epsilon_1 & \epsilon_2 & \epsilon_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \epsilon_4 & \epsilon_5 & \epsilon_1 & \epsilon_2 \end{bmatrix}. \tag{3.2.9}$$

For systems with $\mathrm{N}_{\text{ch}} > 1$, if Φ_j represents the operation of channel j during T_{win}, then $\Phi \in \mathbb{R}^{\mathrm{N}_{\text{ch}} N_{\text{int}} \times N}$ for the total system can be constructed by appending the $\{\Phi_j\}$ together. While any permutation of the collection of all rows in $\{\Phi_j\}$ is allowed, we group the rows acting over the same T_{int} together. This will result in a block diagonal Φ. An example plot of such a matrix is given in Fig. 3.3. It is also possible to model x_n as *point* samples of the signal $x(t)$, instead of the discretization of Eq. (3.2.4), without qualitatively changing much. We adopt this approach in the rest of the paper, since the exact representation is no longer true when we model nonidealities, see Ch. 4.3.1.

3.2.2 Some Issues

The block-diagonal ±1 model is the simplest approximation of the system which is exact provided that x_n is as given in Eq. (3.2.4). In practice, however, it is unclear how to obtain certain important parameters such as f_{carrier} from Eq. (3.2.4); it is often more convenient to

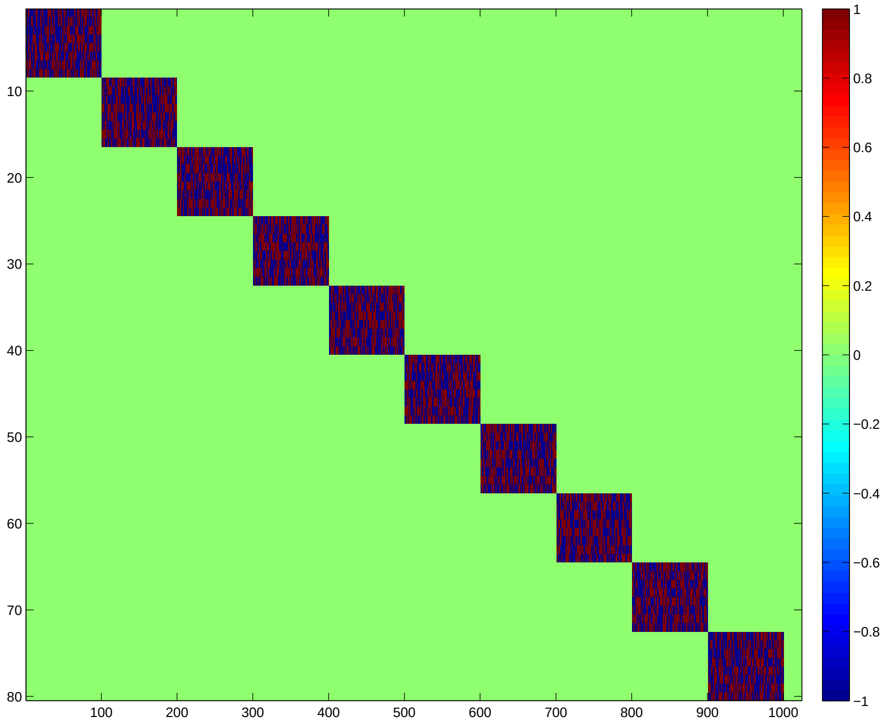


Figure 3.3: A plot of an $\mathrm{N_{ch}} = 8$ channel RMPI matrix for a time window corresponding to $T_{\text{win}} = 1024\Delta T$. Each block represents the output of the 8 channels during a single integration window of duration $T_{\text{int}} = 100\Delta T$. The red/blue colors represent normalized values of $+1/-1$ respectively. The green values indicate a value of zero.

use a quadrature approximation to the integral as shown in Eq. 3.2.10:

$$\begin{gathered} y_i[m] = \int_{(m-1)T_{\text{int}}}^{mT_{\text{int}}} c_i(t)x(t)dt = \sum_{n\in\Omega_m} \epsilon_{i,n} x_n \\ \text{where } \epsilon_{i,n} = c(t_i + \frac{\Delta T}{2}), \quad \text{and } x_i = x(t_i + \frac{\Delta T}{2}) \end{gathered} \tag{3.2.10}$$

The use of Eq. 3.2.10 in favor of Eq. 3.2.6 allows us to use the machinery of the DFT as $\mathbf{x} = \sum_i x_i$ now represents the input $x(t)$ sampled on the Nyquist grid.

3.3 Signal Model and Computational Aspects

In this section we comment on some computational aspects and considerations of the reconstruction process employed by the RMPI. The RMPI is a 'universal' encoder which, unlike other architectures, can be adapted to work with signals that are sparse in any *fixed* domain.

The general signal model targeted by the RMPI is the class of bandlimited functions, but for computational reasons, the signals are finite in length and can be represented by their Nyquist-rate samples over a finite period T. Thus, signals are just vectors $\mathbf{x}$ of length N.

We put special emphases on the fact that, unlike this work, all alternative CS proposals focus on the acquisition of signals that exhibit *spectral* sparsity. The strengths and flexibility of the RMPI are highlighted by the targeted signal class of bandlimited pulses which are not sparse in either the time or frequency domains. This is because they occur for brief periods of time which can add significant bandwidth around the carrier frequency. We know however that radar pulses have concise representation when expressed with respect to the Gabor dictionary (Fig. 3.4 depicts a radar pulse and elements from the Gabor dictionary). The way we utilize this knowledge is explained below.

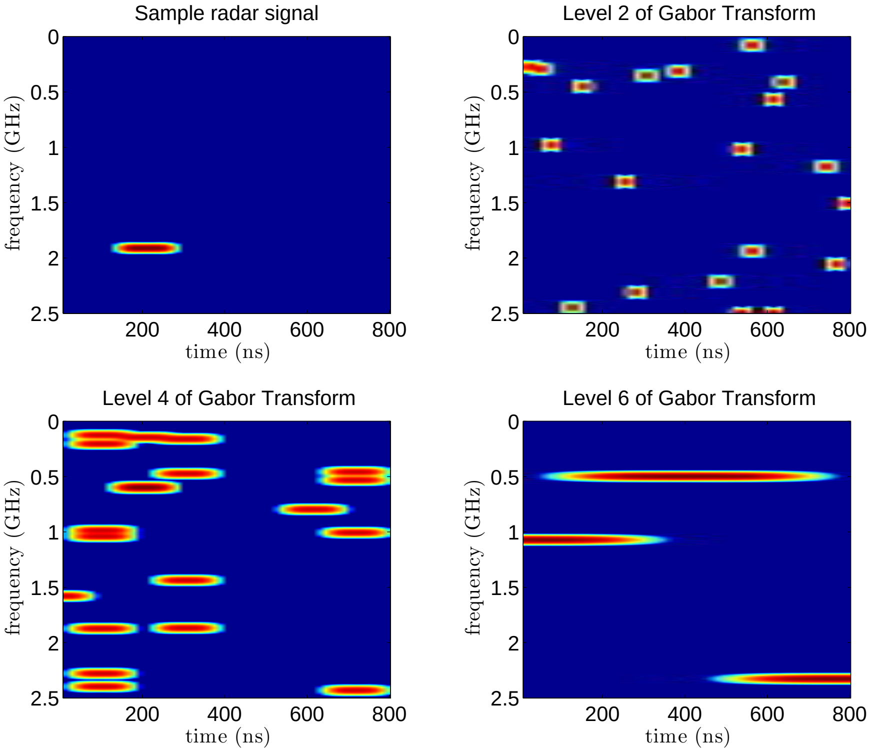


Figure 3.4: The plot in the upper-left hand corner shows a spectrogram of a trapezoidal pulse of 200ns in width with a carrier frequency $f_{\text{carrier}} = 1907.4$MHz. The other plots show elements that have been sampled from a multilevel Gabor dictionary that is 14× over-complete.

Because we are undersampling, which means $M \ll N$, the matrix Φ is very underdetermined so there are infinitely many $\hat{x}$ satisfying $\|\Phi\hat{\mathbf{x}} - \mathbf{y}\|_2 \leq \varepsilon$. CS suggests using the ℓ_1 norm to determine which solution is best, and gives conditions under which this ℓ_1 solution $\hat{\mathbf{x}}$ is close to $\mathbf{x}$. To apply this, we need to minimize the ℓ_1 norm of a sparse object. In the case of radar pulses, $\mathbf{x}$ is not sparse in the time or frequency domains, so we must transform it. In order to accomplish this, we use an overcomplete time-frequency transformation W, known as the multilevel Gabor transform. The parameters of the transform are chosen so that Wx is highly compressible, meaning that energy is concentrated in a few coefficients, so that it is well approximated by a sparse vector. Thus, the BPDN problem is modified from Eq. 2.2.5 to:

$$\min_{\hat{x}} \|W\hat{x}\|_1 \quad \text{such that} \quad \|\Phi\hat{x} - b\|_2 \leq \varepsilon \tag{3.3.1}$$

Criticisms of the RMPI

It has been noted in the literature [68], that the setup of the RMPI is deficient in the sense that it will capture signals with frequencies off the DFT grid with limited accuracy because the dictionary is limited to a discrete set of sinusoids. This is due to the fact that reconstruction is done by trying to construct a closest-approximation to the signal of interest given the samples collected from a finite dictionary. Thus when the RMPI attempts to recover an off-grid pure-tone, it will employ a weighted sum of tones from the library. While this will result in some frequency error, the measurement results presented in Ch. 6 as well as in [69] seem to provide evidence that the RMPI is robust to these frequency errors.

3.4 Related Systems

The number of research efforts aimed at designing CS-based sub-Nyquist receivers has been rapidly growing since 2006. These efforts are often referred to as analog-to-information (AIC or A2I) systems in the literature. We briefly discuss some of the more well-known proposals below. This review focuses primarily on CS-proposals with physical implementations (not necessarily integrated) and those proposals that have gained a wide following in the research community. We begin our review with a discussion of other RMPI implementations.

Other RMPI Systems

While there are no other high-speed RMPI implementations that have been reported at the time of this writing, there has been one notable IC implementation of the RMPI [70, 71] for capturing bandwidths that are orders of magnitude lower than the one reported in this work. The work is specifically focused on processing biophysical signals such as electrocardiogram (ECG) and electromyogram (EMG) signals. The broad motivation behind the work is to address energy and telemetry bandwidth constraints common to wireless sensor nodes in the biomedical electronics sector. What makes this a particularly appealing application space for CS is the fact that common biophysical signals exhibit extreme time sparsity [72] and allow considerable reductions in the number of measurements needed to acquire data of interest with high fidelity. While very few measurements are shown, the performance in terms of power and undersampling reported are impressive. The reported system was implemented in 90nm CMOS and achieves an order of magnitude compression while sampling signals at sub-20ksps while consuming only 1.9μW of power. Two architectures for implementing the RMPI are proposed, a standard analog approach and a purely digital architecture based on multiplexing the output of an ADC between several channels. This use of a "high-rate" ADC at the front-end is possible due to the vastly different constraints in this bandwidth regime. The authors aptly point out that ADCs are not the dominant consumer of power in wireless sensor nodes, but rather, the power consumed by data transmission is which enables reductions in measurements to still lead to substantial savings in power. While the architecture at an abstract level used is the same, the considerations and goals were vastly different. Unlike in this work, the blocks employed in terms of desired transfer function were, for the frequency ranges involved, close to ideal. No calibration was discussed, and specs like dynamic range weren't of principal concern.

Non-Uniform Sampler (NUS)

The NUS was first proposed in [16]; the first IC implementation of the NUS was reported in [11]. A block diagram depicting the conceptual operation of the non-uniform sampler

(NUS) is shown in Fig. 3.5. The NUS can be thought of as a Nyquist-rate ADC which randomly discards samples taken uniformly in time. The NUS takes advantage of the incoherence between time and frequency domains and uses the random time-domain sampling protocol to recover signals with sparse frequency domain representations. Suppose the Nyquist rate is f_{nyq} and the signal of interest is $x(t)$. The ADC digitizes samples taken uniformly in time $\{t_1, t_2, \ldots, t_i\}$ where $t_i - t_{i-1} = T_{\text{nyq}} = 1/f_{\text{nyq}}$. We can then write the samples digitized by the ADC as $x[n] = x(n \cdot T_{\text{nyq}})$. For a time window of time duration $T_{\text{win}} = NT_{\text{nyq}}$ the output samples can be written as:

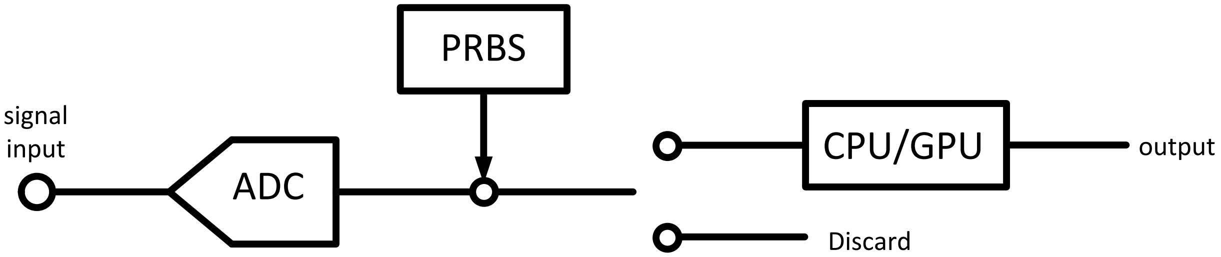


Figure 3.5: The NUS takes Nyquist-rate samples of the input signal and randomly discards most of the samples. For example, the implementation reported in [11] uses a 4.4GHz clock and keeps only one of every 19 samples (average) for a mean back-end sample rate of 236MHz.

$$\mathbf{y}_{\text{nyq}} = \mathrm{I}\mathbf{x} + \mathbf{z} \tag{3.4.1}$$

where $\mathbf{z}$ represents any corruption of the sample and I is the $N \times N$ identity matrix. If we let $\Omega = \{1, \ldots, N\}$ be the set indexing the Nyquist samples of the window, the NUS only collects a subset Ω_M of cardinality $M = |\Omega_M|$. Thus, the NUS can be modeled by a $M \times N$ matrix S which is simply the M rows of I indexed by the Ω_M.

$$\mathbf{y}_{NUS} = S\mathbf{x} + \mathbf{z} \tag{3.4.2}$$

In practice, the NUS uses a PRBS to generate the set Ω_M. For more details on NUS see [11]

Xampling

The "xampling" methodology [68,73–77] utilizes a topology named the modulated wideband converter (MWC) which looks very similar to the RMPI. Both ideas utilize the spread-spectrum downconversion implemented by random demodulation. Despite the similarity, however, there are significant but subtle differences. The biggest difference comes from the fact that the baseband filter used in the MWC architecture is not an integrator, but rather a low-pass filter in the conventional sense. The signal model relies not only on frequency sparsity, but also on a type of block-sparsity, namely the fact that only a few continuous bands of the entire bandwidth are active and that the spectral occupancy of any single band does not exceed that of the back-end low-pass filter. The reconstruction method is based on a combination of CS and blind multiband sampling theory [78]. CS is used to solve an ℓ_1 block sparsity problem and find the blocks of spectrum containing non-zero power content. Once the support is known, the MWC simply inverts the known filterbank implemented by the MWC and, provided the spectral support doesn't change, achieve real-time signal acquisition. While the computational costs are lower, the MWC has to periodically solve the ℓ_1 block sparsity problem and is not robust for tracking quick changes in the spectrum, e.g., a frequency hopping signal.

A discrete PCB prototype implementing the amplifiers, mixers, and baseband filters was reported in [68]. Several other research groups have joined the effort in exploring the potential application of the MWC [79–81] and are exploring further applications and modifications of the hardware.

A Comment on Terminology

Now that several research efforts in CS hardware implementation have been reviewed, there are a few comments that are in order. Many papers on CS sampling hardware, at least at the time of this writing, have been described as implementing a novel form of analog-to-digital converter (ADC). To the best of the author's knowledge, this is a misnomer; while CS systems implement a novel *abstract* form of sampling, it is misleading to classify them as an ADC. The primary function of an ADC is to *digitize* physical voltage levels, which represent desired information, a function that all currently reported CS samplers perform utilizing a standard ADC. The confusion in terminology is probably due to the fact that the term ADC is often used interchangeably with the term sampler. In general, ADCs often utilize sample-and-hold (S/H) circuitry in order to present a single constant voltage value during the entire digitization operation, see [34, 35, 37] for more details. Most proposed CS samplers bear a much stronger resemblance to conventional RF/base-band architectures and are more accurately classified as RF front-ends, or alternatively, as analog pre-processors.

The important thing to note is that while the physical form of the *digitization* operation, mainly digitizing a physical voltage level is identical in both CS and Nyquist signal chains, the information which the voltage level represents are very different. In the convention Nyquist case; the physical voltage level represents correlation with a single tone (ideally) whereas in the case of CS the sample represents correlation with a incoherent basis element. This is an observation whose importance will be made more lucid in the discussion of the recovery of two pulses overlapping in time in Ch. 6.2.4.

Chapter 4

RMPI Analysis and Design

When work on this thesis began, there were few proposals *describing* CS receivers let alone providing a comprehensive mathematical analysis. The first formal theoretical arguments establishing the mathematical *feasibility* of the RD (and by extension the RMPI) were reported in [9], however, as of yet there is little if any work addressing the question of how to *design* an RMPI. Previous work assumed the use of ideal blocks, measurements corrupted only by additive white Gaussian noise (AWGN), and a finite-dimensional space of input signals [8, 9, 18]. The currently available literature that discusses practical implementations or design issues (unrelated to the work in this thesis) focus on the use of CS for either remote-sensing wireless networks in substantially lower-bandwidth applications where the constraints are considerably different [70, 82].

In this chapter we bridge the gap between theory and practice. We address those questions that arise when designing and building a physical RMPI such as: the amenability of the RMPI to implementation with nonideal blocks that is both robust to physical sources of corruption and realized with nonideal blocks. The chapter presents the results of extensive design and feasibility studies that were conducted. Important considerations in reducing performance degradation due to physical constraints and picking system parameters during the design process are discussed. Although the results presented in this chapter often use numbers specific to the physical implementation [6, 7] implementation, the discussion in this chapter is relevant to the design of any RMPI architecture. The transistor-level details of the physical implementation are presented in Ch. 5.

Realization of the RMPI required the work of a large number of people on both the hardware and theory to realize. At this juncture, we especially want to mention the efforts of Dr. Stephen Becker. He developed the recovery algorithms for this project and collaborated on the system design presented in this chapter.

4.1 Goals and Performance Criterion

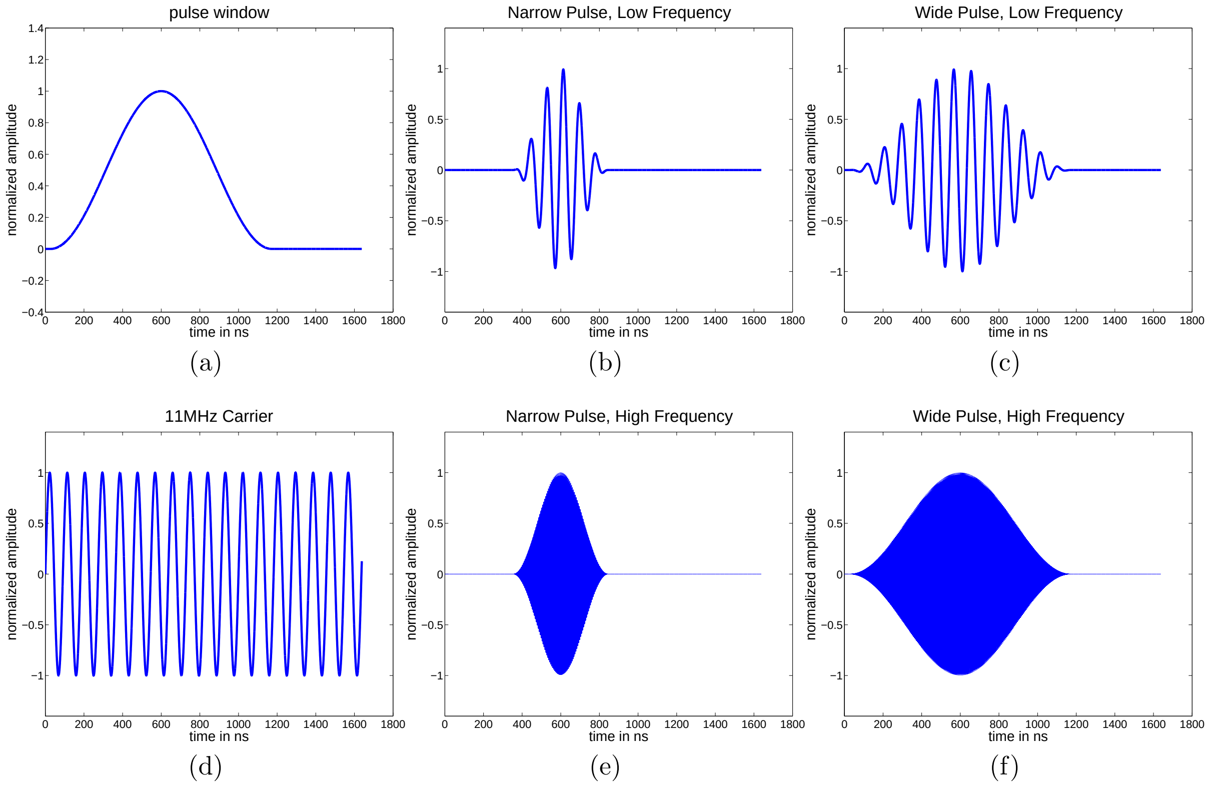
In anticipation of the discussion of various design-related issues in subsequent sections, this section: outlines the goals of the system, defines the criteria used to evaluate the system, and also describes the general strategy used to answer questions pertaining to its design.

4.1.1 Goals

The end-goal is high-fidelity reconstruction/approximation of the original target-signal. The RMPI produces samples that are in a universal format and can be adapted to take advantage of sparsity (or other forms of structure) in any fixed domain. The work in this thesis focused on designing an RMPI based on capturing radar pulses. A radar pulse consists of a pulse envelope modulated by a carrier frequency f_{carrier}. The relevant feature of the radar pulse, with respect to this work, is that it is sparse in the time-frequency plane. This is a distinguishing feature of this work since all prior work only examines situations exhibiting exhibiting frequency-domain sparsity. The condition of sparsity is quite prevalent in radar signals as they typically repeat on the order of 10 *kHz*. In general, the time-separation between any two pulses is much greater than the length of the pulses themselves. This makes CS particularly appealing to applications involving radar.

4.1.2 Performance Criterion

Test Signals To test the range of effective input pulses, a wide range of: amplitude (A_{pulse}), pulse duration (T_{pulse}), and carrier frequencies (f_{carrier}) were used as inputs to evaluate the performance of the system. Several types of simulations were performed using extreme (boundary) values of each of the three parameters while either holding the other parameters fixed or selecting them randomly. Other simulations performed include a Monte Carlo approach where T_{pulse}, A_{pulse}, f_{carrier} were drawn at random many times, or Monte Carlo simulations with a single value fixed. For each type of simulation conducted, both smooth



(a)

(b)

(c)

(d)

(e)

(f)

Figure 4.1: Illustration of several types of input pulses used to test the RMPI design. A pulse consists of a baseband pulse window (a) modulated by a sinusoidal tone with frequency f_{carrier} (b). Permutations of narrow and wide pulses modulated by low and high f_{carrier}. Tukey pulse windows (not shown) were also tested as well as pure tones on and off the DFT grid.

Gaussian-like pulse windows (shown in Fig. 4.1) and sharper trapezoidal pulse (Tukey) windows were used. In the case of trapezoidal pulses, the rise and fall times ($T_{\text{rise}}/T_{\text{fall}}$) were also varied. In simulations attempting exact time-domain recovery, trapezoidal pulses are more difficult to recover than Gaussian pulse windows due to the increased bandwidth required to accurately represent the sharper corners (this is often called spectral leakage) of the trapezoidal window. In the case of reconstruction using a compressive matched filter [55, 69, 83], the type of pulse window has little effect due to its lower reliance on sparsity.

Error Metrics As computer simulations of the system made the test input signal $\mathbf{x}$ known precisely, $\mathbf{x}$ was used as a reference for comparison. A typical simulation involved inputting a radar pulse (baseband pulse window modulated by a specified f_{carrier}) into the RMPI, collecting the generated output samples, and performing exact reconstruction using BPDN. The metric most commonly employed in applications involving radar is the normalized root-mean-squared error (RMSE) as well as the normalized mean-squared-error (MSE) which is just the square of the normalized RMSE

$$\text{RMSE}(\hat{\mathbf{x}}, \mathbf{x}) \triangleq \frac{\|\hat{\mathbf{x}} - \mathbf{x}\|_2}{\|\mathbf{x}_2\|}, \quad \text{MSE}(\hat{\mathbf{x}}, \mathbf{x}) \triangleq \frac{\|\hat{\mathbf{x}} - \mathbf{x}\|_2^2}{\|\mathbf{x}\|_2^2}. \tag{4.1.1}$$

While other metrics are considered such as the maximum absolute deviation (captured by $\|\hat{\mathbf{x}} - \mathbf{x}\|_\infty$), the MSE proves to be convenient for most cases. Ultimately, the choice of error metric is determined by the specific application. We note here that while the original recovery methods used for samples generated by the RMPI were primarily done via basis pursuit (BP/BPDN) methods, other CS-based estimation techniques were developed which implemented matched-filtering to do parameter estimation directly in the domain of CS samples [83], see [55,69] for descriptions of the various implementations of CS matched-filter algorithms developed specifically for the RMPI.

Design Strategy In order to design the system, a critical step is to identify and understand the effect of those system parameters with the most significant effect on overall performance (Ch. 4.4). In terms of requirements for sampling hardware, this translates into producing RMPI samples of a minimum quality that ensures adequate reconstruction/approximation of the target signal for a given set of specifications with respect to an appropriate metric. In this work, the relevant metric is the normalized MSE (Eq. (4.1.1). Successful reconstruction and parameter estimation require a minimum SNR and, in the case of optimization based recovery methods such as BPDN (Eq. (2.2.5)), has a normalized MSE of reconstruction proportional to the SNR of the samples. In general, whatever the information of interest, the accuracy of the result can be quantified with respect to the SNR, which can be tied to physical design parameters. Once the significant system parameters are identified, their effect on overall performance can be obtained through quantifying their effect on the sample SNR.

4.2 Design Principles and Considerations

Upon initial inspection, the RMPI bears a strong resemblance to a spread-spectrum transceiver. It has an obvious decomposition into building blocks that have well-known and efficient implementations. The goal of the RMPI is the reconstruction (or estimation of parameters) of the signal of interest $x(t)$ during a finite window of time of duration T_{win}. The key function implemented by the RMPI is correlation of the input $x(t)$ with a random measurement vector. This operation amounts to randomly modulating the input signal $x(t)$ with a PRBS (the terms PRBS and chipping sequence are used interchangeably throughout this thesis) $c(t)$, integrating the modulated signal $x(t) \cdot c(t)$ over a fixed integration time T_{int}, and dumping the results (sampling/digitizing the output and resetting the integrator to pre-determined state). As the implementation of electronic correlators is well-studied[1], the RMPI appears deceptively simple to realize. However, despite the similarity of the RMPI to conventional RF receiver systems, the use of CS sampling and signal reconstruction algorithms present a set of unique challenges that necessitate a novel approach to the design process. While the matrix representation(Φ) of the ideal RMPI was discussed in Ch. 3, it is fruitful to reason about the RMPI from a more physical point of view to gain insight into how one might go about designing it. Thus, before we we begin, we consider the operation of the RMPI from a frequency-domain stand-point. This viewpoint will elucidate the different considerations in the RMPI design and motivate several design principles that guided the this work.

[1] The first digital correlator was reported by Dr. Sander Weinreb in a technical report in 1963 [84].

4.2.1 Frequency Domain Description of RMPI Operation

The idea that a signal lying anywhere in some bandwidth B (or even occupying the full band B) can be recovered by samples collected at a sub-Nyquist rate $f_s \ll 2B$ seems to, by definition (and naming convention), violate the Shannon-Nyquist sampling theorem. We point out however that the Shannon-Nyquist sampling theorem is a statement about acquisition of a signal with bandlimited but unknown spectral support within bandwidth B; the RMPI targets a more specific class of signals that possess structure beyond being bandlimited. The frequency-domain operation of the RMPI is illustrated in Fig. 4.2. A PRBS with period T_{prbs} will have power concentrated at harmonics of the repetition rate $f_{\text{prbs}} = 1/T_{\text{prbs}}$ (denoted by the vertical arrows in Fig. 4.2. By mixing input $x(t)$ with a PRBS, a frequency-shifted copy of the entire input spectrum is made by each harmonic of the PRBS. Consequently, signal energy from the entire spectrum is down-converted into the passband of the back-end digitizers. Thus, signal energy from the entire input spectrum is captured by the back-end of the ADCs. In the general case, the information converted to baseband is insufficient to

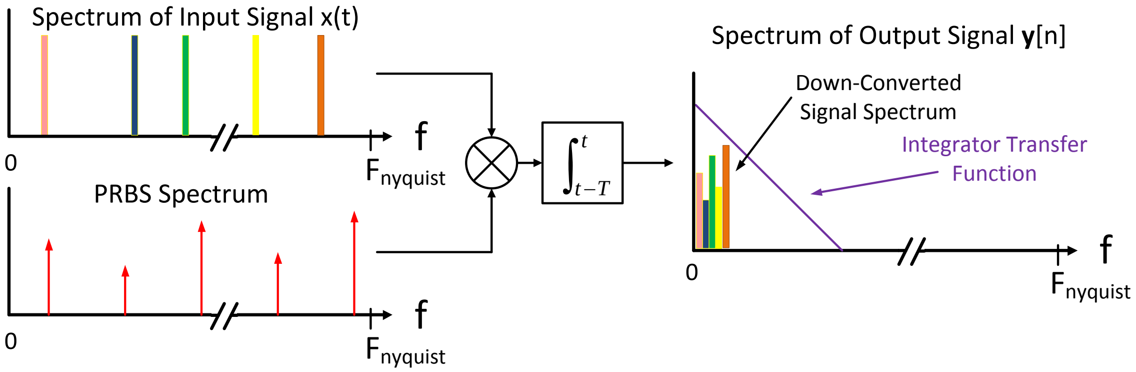


Figure 4.2: Operation of the RMPI depicted from the frequency domain. The left hand side of the figure depicts the power spectrum of both the input signal $x(t)$ (the colored vertical bars represent spectral occupancy of the input) and a PRBS $c(t)$ (red vertical arrows depict the PRBS harmonics). The power spectrum of the output of the windowed integration of $x(t) \cdot c(t)$ is shown on the right hand side of the figure. Signal energy from the entire input spectrum has been downconverted to baseband.

uniquely determine the original spectrum of an arbitrary bandlimited signal. However, because we assume extra prior information (namely sparsity) CS techniques allow for recovery via sparse approximation methods[2]. Thus, whereas a spread-spectrum transceiver uses a PRBS to prevent jamming by expanding the spectral occupancy of the message, the RMPI uses PRBS to compress information over a large spectrum into a smaller baseband to collect a summary of the entire bandwidth. We make the following observations based upon the frequency-domain description of RMPI operation:

- For a given PRBS sequence used, there will be wide variation in the distribution of power among the harmonics of the fundamental frequency (repetition rate f_{prbs}). This will lead to frequency dependent gain variation with specific behavior determined by the values of parameters such as the integrator unity-gain bandwidth $f_{\text{ug,int}}$ and f_{prbs}.
- On average, as the lowest frequency components (highest power gain at output) of the downconverted spectrum arise from PRBS harmonics closest to the frequency content of the input signal. Consider the case of a pure tone input with frequency f_{in}; the gain of the system will be lowest for tones withf_{in} lying exactly between 2 PRBS harmonics.

These observations intuitively motivate the following design principles.

4.2.2 Principles

Principle 1: Uniqueness All distinct signals in the signal class (k-sparse signals) should produce distinct measurements.

Principle 2: Consistency All equi-energy inputs should produce measurements of similar energy.

Principle 1 basically states that the relationship between input and output measurements should be bijective for the signal class in order to result in an unambiguous answer. Restriction to the signal class is required, since for the case of undersampling, it is not possible

[2] See [55, 69, 85–87] for details of algorithms used in this work. For a general overview of techniques employed to perform signal recovery from CS samples, see [51, 88] and the references therein.

to produce distinct measurements for all signals. Principle 2 asserts that noise will affect all measurements equally; given a minimum required output SNR for acceptable reconstruction, the sensitivity and dynamic range should not be (or at least should have minimal dependence) on the specific frequency content. This idea can be interpreted as the RMPI equivalent of the "minimum distance" concept of coding theory. These principles serve as general guidelines throughout the remainder of this chapter. Ch. 4.3 discusses the effects of using nonideal building blocks to realize the RMPI. Ch. 4.4 identifies and discusses the effect on performance of several of the dominant system parameters. Ch. 4.5 examines the robustness of the RMPI to physical nonidealities.

4.2.3 Challenges and Considerations

Before diving into discussion of any specific implementation details we outline some general challenges and considerations in the design of the RMPI.

Sensitivity to Accuracy of Φ As CS is equivalent to solving an underdetermined inverse problem, reconstruction accuracy is sensitive to the error between the transfer function Φ used for reconstruction and the actual Φ of the physical system. Indeed, empirical studies reveal that the performance degradation due to the introduction of parasitic transfer functions can be quite severe. One of the contributions of this thesis was the development of a calibration procedure that measures the Φ of the physical system. Calibration is critical to achieving high-fidelity results, this procedure is described in Ch. 4.3.2. In terms of the design, extra effort was expended to minimize the introduction of performance degrading parasitic transfer functions.

Computational Cost and System Simulation Nyquist reconstruction is based on linear interpolation of the acquired samples of the signal under the assumption of bandlimitedness. In contrast, CS reconstruction techniques are based on approximation/estimation of signal parameters under a sparsity or measurement error constraint. The reconstruction algorithms are nonlinear functions of the acquired samples and are costly, from a computational standpoint, compared to their Nyquist counterparts. In addition, the random-correlation operation of the RMPI is a time-variant function. The issues of computational cost and time variance make simulating the system, especially at the transistor-level, more cumbersome and difficult because of the increased reliance on transient simulation. These difficulties were addressed by creating detailed models of the blocks of the system by fitting parameters extracted from the simulation of the corresponding transistor-level schematic. Details of the simulation methods and nonideal effects modeled are the subject of Ch. 4.3 and App. A.

Differences in Use of Similar Circuit Blocks Despite the similarity in the block diagrams of both the RMPI and a conventional direct-conversion receiver architecture, the intended use of corresponding blocks is considerably different. For example, the baseband filter in a conventional direct-conversion receiver (DCR) is designed for memoryless operation in a specified bandwidth, i.e., constant group delay (linear phase), minimum passband gain ripple, and a certain amount of attenuation within a specified transition bandwidth. In contrast, the baseband filter in the RMPI is desired to behave as closely as possible to an ideal integrator and thus is focused on minimizing the filter f_{3dB} for a given unity-gain bandwidth f_{ug}. Another important consideration arises in the design of the mixer. Although there are many wideband receivers reported in the literature [28–30, 33, 89–91], the receivers generally do not operate with a large "effective instantaneous bandwidth" (EIBW); the IF bandwidth (usually set by the bandwidth of the baseband filters) in many receivers is set much lower than the bandwidth of the input LNA. This means that the receiver effectively only looks at a small portion of the total input spectrum at a given time. For example, in [28] the IF bandwidth of the receiver is 25MHz while the bandwidth of the front-end spans $0.2 - 2$GHz. Consequently the mixer for a DCR architecture only needs to provide flat (self-consistent) transfer characteristics within a comparatively small bandwidth (IF bandwidth). The mixer in the RMPI, on the other hand, needs to provide a known (ideally flat) transfer characteristic across the entire input bandwidth. This is due to the fact that the RMPI mixes input $x(t)$ with a wideband chipping sequence $c(t)$ as opposed to a pure-tone.

Physical Constraints While convenient for analytical modelling, these functions are not achievable in a physical form factor. For example, an ideal integrator has a single pole with infinite gain at DC; a physically implemented integrator (usually implemented in CMOS as either a switched capacitor, gm-C, or opamp based RC filter) has a finite gain bandwidth product and will not have a pole at DC, but rather a single dominant pole at a low frequency along with several (perhaps many) parasitic poles at much higher frequencies. This point ties back to the discussion above about the effects in inaccuracies of the reconstruction Φ.

4.3 RMPI Modeling

In this section we examine issues related to using physically realizable blocks in place of their ideal counterparts used throughout the literature. This is an important step in establishing the feasibility of the approach as ideal blocks have characteristics such as infinite gain, flat group-delay, and 0 dB gain ripple that are unachievable. Thus, it is important to verify that the RMPI does not rely upon physically unrealizable attributes for stable and uniform operation. For example, suppose the input signal were altered by the input amplifiers, e.g., dispersion and nonlinear distortion: a signal that was sparse in the chosen domain, could become significantly less sparse after passing through the amplifier resulting in failure of the sparse-recovery algorithm. The exposition will primarily focus on using continuous-time integrators in place of an ideal one. The effects of nonideal transfer functions can be analyzed in an analogous manner. For the purposes of the discussion in this section, we assume all blocks other than the integrator are ideal. In other words, we assume amplifiers and mixers with flat gain and linear phase response over bandwidths far in excess of the desired system input bandwidth.

4.3.1 Modeling General Transfer Functions

We begin by building upon the analytical model of the ideal RMPI presented in Ch. 3. Let $m_i(t) = x(t)p_i(t)$ and $h(t)$ represent the general transfer function of the overall system, we adopt the more general model

$$y_i(t) = \int_{-\infty}^{\infty} h(t-\tau) m_i(\tau) d\tau. \tag{4.3.1}$$

where we write the Laplace transform of $h(t)$ as

$$H(s) = \frac{\prod_{i=1}^{N_z}(s-z_i)}{\prod_{j=1}^{N_p}(s-p_i)}. \tag{4.3.2}$$

Though the RMPI is a time-variant system due to mixing and windowed integration, LTI analysis tools can still be used to describe the operation within a single integration window. We then rewrite Eq. (3.2.1) to consider the effect of a non-ideal integrator

$$y_i[m] = \int_{(m-1)T_{\text{int}}}^{mT_{\text{int}}} h(mT_{\text{int}} - t) m_i(t) dt. \tag{4.3.3}$$

The chipping sequence within the m^{th} window is not necessarily equal (but can be) to the chipping sequence in any other window. However $h(t)$ is a constant through every integration window. We see that $m_i(t)$ is windowed by the time-reversed transfer function $h(mT_{\text{int}} - t)$ of the integrator. In the ideal case we have:

$$h(t) = u(t) = \begin{cases} 1 & t \geq 0 \\ 0 & \text{else} \end{cases}, \text{ and } H(s) = 1/s. \tag{4.3.4}$$

where $u(t)$ is the unit step function which, when substituted into Eq. (4.3.3) results gives Eq. (3.2.1). In practice, the transfer function of Eq. 4.3.4 cannot be realized: physical integrators cannot be built with infinite gain-bandwidth product. For continuous-time (CT) integrators, the transfer function will deviate from the ideal at low-frequencies due to finite dc gain and at high frequencies due to parasitic poles and zeros. A reasonable approximation is a single "dominant" low-frequency pole p_1 followed by several higher frequency poles/zeros ($p_2, \ldots, p_n/z_1, \ldots, z_n$ with the p_i, z_j numbered lowest to highest in magnitude) [92]. We first examine the effect of the "dominant" low-frequency pole p_1 and examine the effects of the

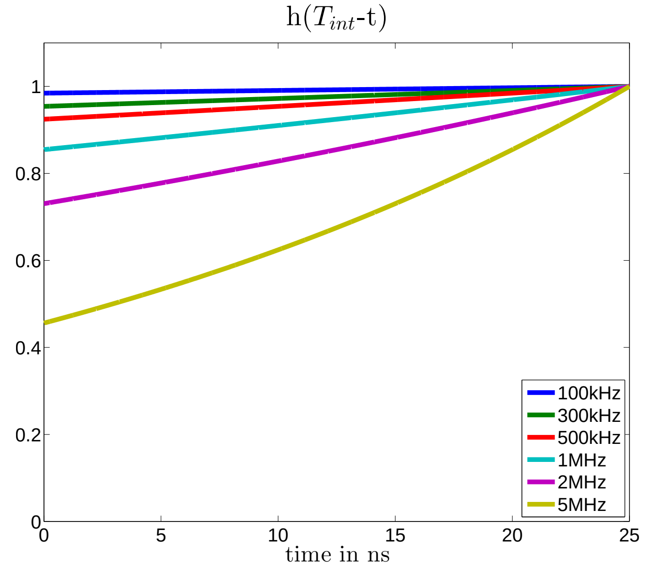


Figure 4.3: Impulse responses for various dominant-pole locations.

other poles/zeros at higher frequencies later. Suppose we have a single-pole system

$$H(s) = \frac{1}{s + p_1} \quad \text{with} \quad h(t) = \mathrm{e}^{-p_1 t} u(t). \tag{4.3.5}$$

In this case, the $f_{3\mathrm{dB}} = p_1/2\pi$. In contrast to the case of the ideal unit-step window, the samples will be windowed by the exponential function $\mathrm{e}^{-p_1(mT_{\mathrm{int}}-t)}$. Windows for several different values of p_1 are shown in Fig. 4.3. We see for high-values of p_1, the window attenuates the parts of the modulated signal from earlier in the integration window. Thus, a natural requirement is that $1/p_1 \gg T_{\mathrm{int}}$ in order to prevent loss of information. In order to see the effect of higher frequency poles $p_2, \ldots, p_n$, the general model can be decomposed via a partial fraction expansion into a sum of first and second-order responses[3] which will result in a time-domain impulse response that is a sum of exponentials and/or exponentials modulated by an oscillating term. As we are interested only in the higher-gain portions of the integrator, in practice, many of the higher-frequency poles (the first of which occurs two or more decades away from the dominant pole) can be ignored because they occur in a portion of the transfer function that is significantly attenuated. Consider a two-pole system

[3] Note, over the field of real numbers, the highest irreducible polynomial is of degree 2, for complex numbers it is of degree 1.

$p_1 \ll p_2$

$$H(s) = \frac{K}{(s+p_1)(s+p_2)} = \frac{A_1}{s+p_1} + \frac{A_2}{s+p_2} \tag{4.3.6}$$

$A_1 = -A_2$ where $A_1 = K/(p_2 - p_1)$. Thus, the impulse-response will be a fast decaying exponential with time-constant $\tau_2 = 1/p_2$ with a slowly decaying response $\tau_1 = 1/p_1$. The effect of p_2 will be to cause information loss in last part of the integration where as the effect of p_1 will be to cause information in the earliest portion of the integration window. We see that it is also beneficial to make sure that $p_2 \gg p_1$. Design issues pertinent to the integrator are further discussed in Ch. 4.4.

4.3.2 Calibration

We delay discussion of how other system design considerations and their affect how the integrator parameters are chosen until the relevant blocks are discussed. For now we state that while careful parameter selection can mitigate the more dominant deleterious effects, many of the other ones can be accounted for by characterizing the impulse-response, i.e., $\Phi \in \mathbb{C}^{M \times N}$ by sending in a series of N linearly independent inputs and collecting/processing the resulting outputs. From an abstract point of view, measuring the column-vector outputs $\mathbf{y}_i$ resulting from unit impulse functions $\mathbf{e}_i$ (canonical basis elements) is easiest since they can be appended together to assemble

$$\Phi = [\mathbf{y}_1, \ldots, \mathbf{y}_N]. \tag{4.3.7}$$

This approach is potentially problematic when $\mathbf{e}_i$ represent time-domain impulses, as these signals are difficult to generate. The approach is more amenable to implementation in the Fourier-domain. The frequency response

$$\Phi_f = \Phi F^{-1}, \tag{4.3.8}$$

where F is the DFT matrix, can be assembled from measurements

$$\mathbf{b}_k = \Phi_f \mathbf{e}_k = \Phi F^{-1} F \boldsymbol{\alpha}_i, \tag{4.3.9}$$

where $\mathbf{e}_k = F\boldsymbol{\alpha}_i$ ($\boldsymbol{\alpha}_i$ are time-domain representations of elements from an N-point DFT grid). Once Φ_f is known, Φ can be obtained from the simple computation

$$\Phi = (F^{-1}(\Phi_f)^*)^*, \tag{4.3.10}$$

where $*$ denotes the conjugate transpose.

4.4 Parameter Selection

This section discusses some important system-level design choices and how to choose values for critical system-level parameters. Various performance trade-offs and important relationships between highly interdependent parameters are discussed. A list of some of the questions this section aims to answer is provided below:

- At what rate does the chipping sequence need to operate?
- What periodicity does the chipping sequence (length) need to have?
- How are the operating specifications for different blocks determined?

In addition, there are many questions that arise from the use of parallelization: what are the considerations for choosing a certain number of channels, what marginal benefit does it provide from a CS reconstruction perspective, what is the real marginal cost per channel, and will channel-to-channel become a serious issue?

Due to the highly interdependent nature of several parameters of the system, most notably the gain-bandwidth product of the integrator and PRBS parameters, it is not possible to study each parameter in complete isolation from the other parameters. Therefore, initial studies estimated a conservative value for different parameters that were based on reasonable physical constraints.

4.4.1 Parameter Selection Models

While the order in which the blocks are discussed can be introduced in many ways, we choose to start by examining simplified models and adding complexity as necessary. At its heart, the sole purpose of the RMPI is to collect random projections acquired physically in the time domain. It is thus natural to first focus the discussion of parameter selection on the bare minimum of blocks necessary to implement this operation. Indeed, many of the other blocks inserted that are discussed later in Ch. 5 are added to enhance performance or enable physical compatibility, e.g., the addition of a 50 Ω impedance-matched low-noise amplifier (LNA). The three principal components of the random projection operation require, at a minimum, the implementation of the following blocks:

1. A block which modulates the input $x(t)$ with PRBS $c(t)$.

2. A PRBS generator to produce waveform $c(t)$.

3. A block that implements integration.

Using this model, the signal ($P_{\text{sig, out}}$) and noise ($P_{\text{noise, out}}$) power at the output of a single RD for $x(t) = Ae^{j2\pi f_{\text{in}} t}$, $A \in \mathbb{C}$ and input noise power spectral density (PSD) $P_{\text{noise, in}}$ is:

$$\begin{aligned} P_{\text{sig, out}} &= \int_{-\infty}^{\infty} |A|^2 |H_{\text{sys}}(f)|^2 \Upsilon(f - f_{\text{in}}) df \\ P_{\text{noise, out}} &= \int_{-\infty}^{\infty} P_{\text{noise, in}} |H_{\text{sys}}(f)|^2 \Upsilon(f) df, \end{aligned} \tag{4.4.1}$$

where $\Upsilon(f)$ is the PSD of the PRBS and $H_{\text{sys}}(f)$ is the transfer function of the signal chain. Thus, we see that the significant parameters of our system will be related to the PRBS and the integrator.

4.4.2 PRBS Parameters

Perhaps the two most important blocks in the design of the system are the PRBS (we also refer to this as the chipping sequence) and the integrator. The RMPI PRBS is periodic, as the requirement of retaining knowledge of the measurement matrix Φ prohibits the use of infinite length chipping sequences. The two quantities that characterize $\Upsilon(t)$ is $f_{\text{chip}} = 1/\Delta T$, where f_{chip} is the rate at which the PRBS is toggled, and the length of the PRBS sequence N_{chip} which with f_{chip} defines the period of the PRBS $T_{\text{prbs}} = N_{\text{chip}}\Delta T$. We defined the ideal chipping sequence in Eq. (3.2.2). For notational convenience, we drop the subscript i from the expression $c_i(t)$. Below, to facilitate future discussion, we also rewrite Eq. (3.2.2) as a sum of weighted and uniformly time-shifted pulse windows $g(t)$

$$c(t) = \sum_{n=-\infty}^{\infty} \epsilon_n g(t - n\Delta T), \quad \epsilon_n \in \{\pm 1\}, \tag{4.4.2}$$

where in the ideal case, $g(t)$ is a rectangular window that is 1 inside $[-\Delta T/2, \Delta T/2]$ and 0 elsewhere. The spectral properties of $g(t)$ ultimately set the properties of the PRBS PSD $\Upsilon(f)$ and thus affect how the PRBS parameters are chosen and the achievable performance. Whenever $N_{chip} < \infty$, $\epsilon_n = \epsilon_m$ whenever $n \equiv m \bmod N_{chip}$[4]. So how do we pick f_{chip} and N_{chip}? We answer these questions by quantifying the qualitative description of the RMPI from Ch. 4.2.1. Before we describe the effects and trade-offs in picking these parameters, it is instructive to review the spectral properties of both finite and infinite length $c(t)$ below.

Spectral Properties of the PRBS

In the following we compute both the PSD of both the infinite and finite-length PRBS as well as the fourier series coefficients of the finite-length PRBS. We start by computing the PSD for an infinite-length PRBS $\Upsilon(f)_\infty$. Recall that the autocorrelation function of a random signal $x(t)$ is defined

$$R_x(t, \tau) = \mathbb{E}[x(t)\bar{x}(t - \tau)], \tag{4.4.3}$$

[4] For the implementation reported in [7], $N_{chip} = 128$ was chosen empirically and is not optimal.

where if $x(t)$ is wide-sense station (WSS), then the autocorrelation $R_x(t,\tau) = R_x(\tau)$. In the case of $x(t)$ that is WSS, the Wiener-Khinchin theorem states that the PSD of $x(t)$ is merely the fourier transform of the autocorrelation

$$\Upsilon_x(f) = \mathcal{F}\{R_x(\tau)\}. \tag{4.4.4}$$

So we begin computing the PSD by first computing the autocorrelation of $c(t)$

$$\begin{aligned} R_{c,\infty}(t,\tau) = \mathbb{E}[c(t)\bar{c}(t+\tau)] &= \sum_{j=-\infty}^{\infty}\sum_{k=-\infty}^{\infty} \mathbb{E}[\epsilon_j\epsilon_k] g(t-k\Delta T)g(t+\tau-j\Delta T) \\ &= \sum_{k=-\infty}^{\infty} g(t-k\Delta T)g(t+\tau-k\Delta T), \end{aligned} \tag{4.4.5}$$

as ϵ_j and ϵ_k are independent. Note that the autocorrelation is not WSS, but cyclostationary. It can be made WSS by observing that a uniform time-shift drawn from $U[0,\Delta T]$ will have the same PSD. Thus, we can take an additional expectation over a time $t \sim U[0,\Delta T]$ of Eq. (4.4.5) and take the fourier transform of the result giving us

$$\Upsilon_\infty(f) = \frac{|G(f)|^2}{\Delta T}, \tag{4.4.6}$$

where $G(f) = \mathcal{F}\{g(t)\}$. For the rectangular window $g(t)$, the Fourier transform $\mathcal{F}(g(t)) = G(f)$ can be expressed

$$\mathcal{F}(g(t)) \overset{\text{def}}{=} G(f) = \frac{\sin \pi f \Delta T}{\pi f} = \Delta T \operatorname{sinc}(f\Delta T), \tag{4.4.7}$$

where sinc $\overset{\text{def}}{=} \sin(\pi x)/(\pi x)$ at $x \neq 0$ and $\operatorname{sinc}(0) = 1$ is the normalized sinc function.

PSD of infinite length PRBS The power spectral density (PSD) of an *infinite sequence* of data modulated at rate $f_{\text{chip}} = 1/\Delta T$ is given by [64] substituting Eq. (4.4.7) into Eq. (4.4.6)

$$\Upsilon_\infty(f) = \Delta T \left(\frac{\sin \pi f/f_{\text{chip}}}{\pi f/f_{\text{chip}}}\right)^2 \tag{4.4.8}$$

PSD of finite length PRBS For a random bit sequence with finite repetition rate $f_{\mathrm{prbs}} = f_{\mathrm{chip}}/N_{\mathrm{chip}}$, the power spectral density is given by

$$\Upsilon(f) = \frac{N_{\mathrm{chip}}}{f_{\mathrm{chip}}} \left(\frac{\sin \pi f N_{\mathrm{chip}}/f_{\mathrm{chip}}}{\pi f N_{\mathrm{chip}}/f_{\mathrm{chip}}} \right)^2 . \tag{4.4.9}$$

The computation of the finite-length case is slightly more involved and a partial derivation is given in the next subsection.

Fourier Series Representation of the PRBS We now derive the Fourier series representation for the finite length PRBS chipping sequence. Let N_{chip} denote the number of bits in the PRBS sequence resulting in a period $T_{chip} = N_{chip}\Delta T$. The periodicity of $c(t)$ naturally admits a Fourier series representation

$$\begin{aligned}
\hat{c}[k] &\overset{\mathrm{def}}{=} \frac{1}{T_{chip}} \int_0^{T_{chip}} c(t) e^{-j2\pi kt/T_{chip}} dt \\
&= \frac{1}{T_{chip}} \sum_{n=0}^{N_{chip}-1} \int_{n\Delta T_{chip}}^{(n+1)\Delta T} c_n e^{-j2\pi kt/T} dt \\
&= \frac{1}{T_{chip}} \sum_{n=0}^{N_{chip}-1} c_n \frac{T_{chip}}{-j2\pi k} \left(e^{-j2\pi k(n+1)/N_{chip}} - e^{-j2\pi k(n)/N_{chip}} \right) \\
&= \frac{\sin(\pi k/N_{chip})}{\pi k} \sum_{n=0}^{N_{chip}-1} c_n e^{-j2\pi k(n+\frac{1}{2})/N_{chip}} .
\end{aligned} \tag{4.4.10}$$

With the convention $\sin(0)/0 = 1$, this holds for any integer k. Note that this would be periodic in N_{chip} if it were not for the sinc term. If we calculate $\hat{c}[k + lN_{chip}]$ where $l \in \mathbb{Z}$ we get:

$$\begin{aligned}\hat{c}[k+lN_{chip}] &= \frac{\sin(\pi k/N_{chip}+\pi l)}{\pi(k+lN_{chip})}\sum_{n=0}^{N_{chip}-1} c_n e^{-j2\pi k(n+1/2)/N_{chip}}(-1)^l \\ &= \frac{k}{k+lN_{chip}}\hat{c}[k] \\ &\text{where for } k=0,\ \hat{c}[lN_{chip}]=0.\end{aligned} \tag{4.4.11}$$

Then the fourier transform $(\mathcal{F}\{c(t)\} = \hat{c}(f)$ can be written as:

$$\hat{c}(f) = \sum_{k=-\infty}^{\infty} \hat{c}[k]\delta(f - \frac{k}{T_{chip}}) \tag{4.4.12}$$

What is important to note is that the periodicity of $\hat{c}[k]$ is windowed by an attenuating factor of $\frac{k}{k+lN_{chip}}$. A computation of the power spectral density of a periodic chipping sequence ultimately gives:

$$\begin{aligned}\Upsilon(f) &= \frac{|G(f)|^2}{\Delta T}\sum_{j'=-\infty}^{\infty} e^{-i2\pi f j' T_{chip}} \\ &= \frac{|G(f)|^2}{\Delta T}\frac{1}{T_{chip}}\sum_{k=-\infty}^{\infty}\delta(f-\frac{k}{T_{chip}}) = \frac{1}{N_{chip}}\left(\frac{\sin\pi f/f_s}{\pi f/f_s}\right)^2\sum_{k=-\infty}^{\infty}\delta(f-\frac{k}{T_{chip}})\end{aligned} \tag{4.4.13}$$

where the poisson summation formula was used [53]:

$$\sum_{k=-\infty}^{\infty} e^{-j2\pi kt/T} = T\sum_{k=-\infty}^{\infty}\delta(t-k'T). \tag{4.4.14}$$

When the PRBS sequence is generated with an LFSR as described in [64], the statistics are slightly different due to the odd length of the sequence generated $(L = 2^r - 1,\ r \in Z)$. The sequences generated by the LFSRs have $(L+1)/2$ 1's and $(L-1)/2$ -1's; the number of consecutive runs of $\{\pm 1\}$ of length j is proportional to $1/2^j$. The power spectral density of an LFSR generated sequence of length L is [64]:

$$\begin{aligned} &\Upsilon_{LFSR}(f) = \left[\sum_{m=-\infty}^{\infty} \delta(f - mf_0)\right] \frac{L+1}{L^2} \left(\frac{\sin(\pi f/f_c)}{\pi f/f_c}\right)^2 + \frac{1}{L^2}\delta(f) \\ &\text{where} \quad f_0 = \frac{f_c}{L}, \; L = 2^r - 1, \quad \text{and} \quad \Upsilon_{LFSR}(0) = \frac{1}{L^2}\delta(0). \end{aligned} \tag{4.4.15}$$

Note, the spectra in the case of the PRBS generator concentrates the signal power at harmonics of the PRBS fundamental frequency $f_{\text{prbs}} = 1/(N_{\text{chip}}\Delta T)$; if N_{chip} is very large, f_{prbs} decreases moving the spectral lines closer together. For practical purposes, the spectrum may be viewed as continuous and similar to that of an infinite length PRBS.

PRBS Toggling Rate

Once it is decided upon to use a PRBS, an obvious set of questions is how fast does the LFSR which generates the PRBS need to be toggled, and how long should the sequence be before it repeats? This answers to these questions are important as they determine the number of logic elements needed to generate the LFSR. As the logic elements used in the LFSR are the circuits which operate at the highest frequency in the entire receiver, they can become a dominant consumer of power. The dynamic power consumption of any logic gate is typically dominated by driving the input and output capacitances from 0 to V_{DD} at rate f_{chip} which is expressed as

$$P_{dynamic} = 1/2 C_{total} V_{DD}^2 f_{\text{chip}}, \tag{4.4.16}$$

assuming the use of static logic (readily available in CMOS but not necessarily available in a process such as InP, for example, CML is employed for all logic in [11, 69]).

A chipping sequence rate of $f_{\text{chip}} = f_{\text{nyq}}$ is prescribed in [9] and used in most, if not all, reported analyses of the RD and RMPI architectures [8, 18, 93]. This requirement can be understood by examining the PSD of the chipping sequence used. The PSD of both the finite and infinite length PRBS is proportional to $|G(f)|^2 \propto |\operatorname{sinc}(f/f_{chip})|^2$ which drops off monotonically for $0 < f/f_{chip} < 1/2$ where it achieves its first null at $f = f_{chip}$, shown in Fig. 4.4.

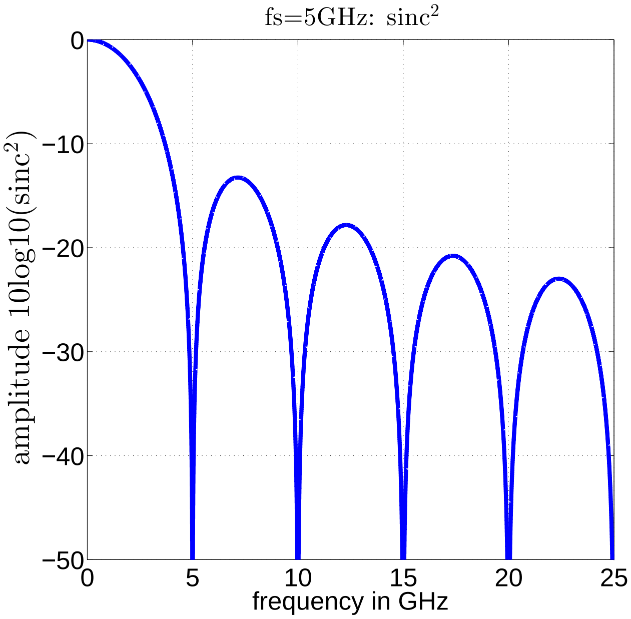


Figure 4.4: Window of $\mathrm{sinc}(\pi f/f_s)^2)$.

Eq 4.4.1 indicates that $P_{\text{sig,out}}$ is the proportional to the input signal power $|A|^2$ multiplied by the $\Upsilon(f)$ shifted by f_{in} and weighted by $|H_{sys}(f)|^2$ (the squared magnitude of the low-pass-filter) transfer function response. The effect of using different values of f_s (f_{chip}) is shown in Fig. 4.5. Consider the case in which $f_s \le f_{nyq}$, the first null of the $\mathrm{sinc}(f/f_s)^2$, which occurs at $f = f_s$ can potentially be shifted into a portion of $|H_{sys}(f)|^2$ with relatively high gain. In this case, $P_{\text{sig, out}}$ would be extremely low. Although $f_{\text{chip}} > f_{\text{in},max}$ is enough to ensure that a null is not downconverted directly to DC, the magnitude of $\mathrm{sinc}(f/f_{\text{chip}})^2$ drops off precipitously past $f = f_{\text{chip}}/2$ making any shifts of that region into even the center of $|H_{sys}(f)|^2$ negligibly small. Thus, the condition $f_s \ge 2f_{\text{in},max}$ is appropriate to maximize average downconverted power per channel while minimizing gain roll-off with f_{in}. While increasing f_s will minimize average gain variation with f_{in}, it should be noted that as the frequencies of interest lie at a fixed boundary (fixed input bandwidth), the average down-converted power from the spectrum inside the bandwidth also falls off with f_s, shown in Fig. 4.6(b) and that more noise power outside of the input bandwidth will be folded into the high-gain portion of $|H_{sys}(f)|^2$ reducing the overall SNR. The situation in which $f_{\text{chip}} = f_{\text{nyq}} = 2f_{\text{in},max}$ is shown in the center column of Fig. 4.5. Even compared to the

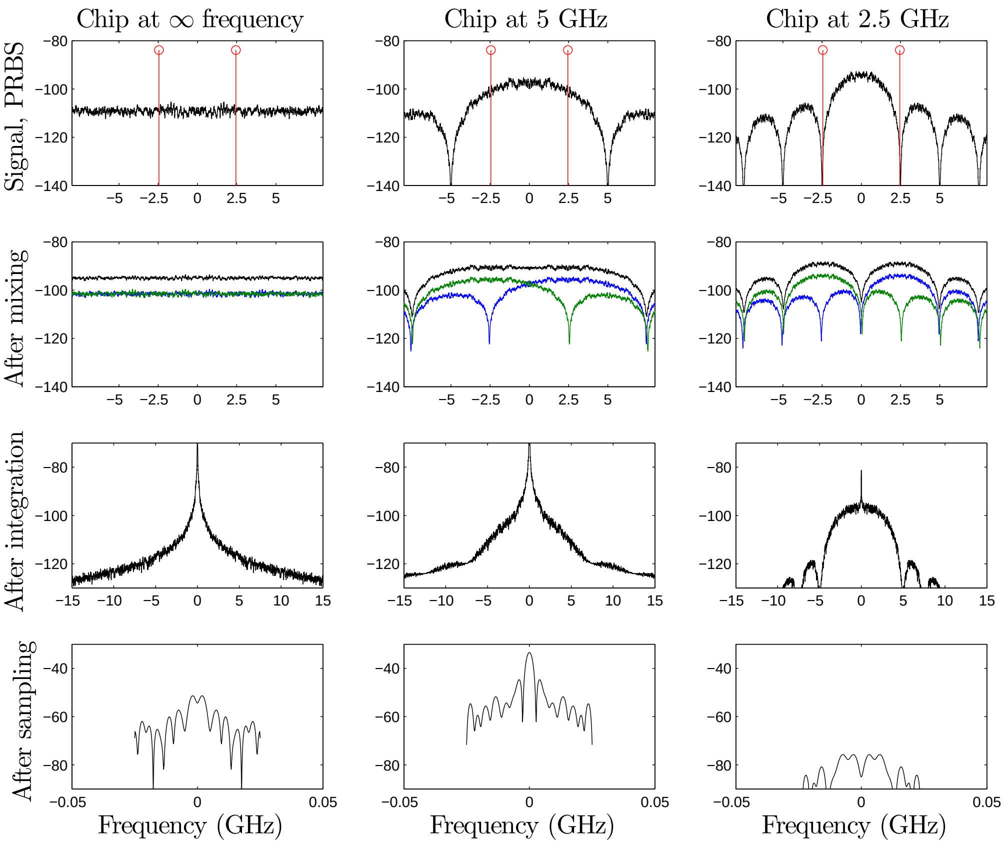
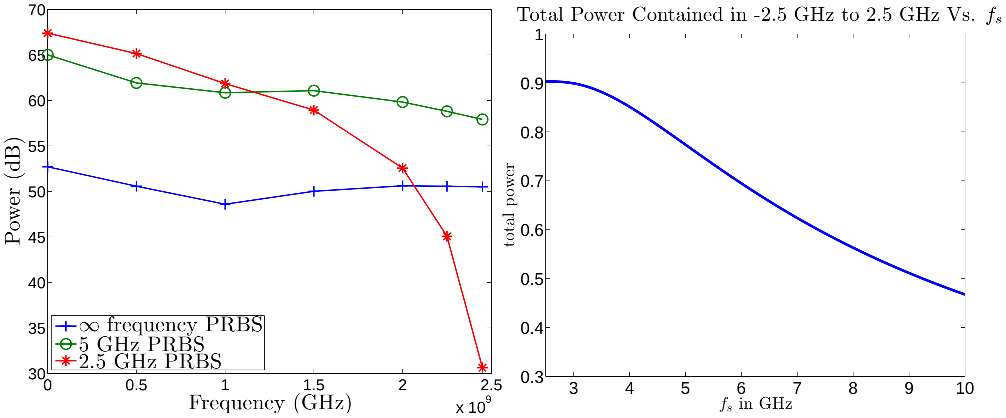


Figure 4.5: Study of the effects of different chipping rates f_{chip} on the downconversion of a tone with $f_{\text{in}} = f_{\text{nyq}}/2 = 2.5\ GHz$ (the spectrum of the tones is depicted by the red vertical lines). The first row depicts the PSD $\Upsilon(f)$, the second row the spectrum after mixing of the PRBS with the input tone (the frequency translated copies of $\Upsilon(f)$ are shown by the blue and green lines), the third row depicts the spectrum of the output of the integrator, and the fourth after sampling the output of the integrator at a fixed rate. This was done for chipping rates corresponding (from left-most column to right-most column) to infinite, 5 GHz, and 2.5 GHz frequencies. This figure illustrates why it is desirable to set $f_{\text{chip}} = f_{\text{nyq}}$. If $f_{\text{chip}} = f_{\text{nyq}}/2$ (right-hand column), then a null of the PSD (sinc) downconverts negligible power into the center of the lowpass filter. If f_{chip} is set high, the majority of the energy contained in $\Upsilon(f)$ is filtered out by the low-pass filter and results in a net downconverted power lower than that in the situation where $f_{\text{chip}} = f_{\text{nyq}}$.

situation in which $f_s = \infty$ (shown in the left-hand column of Fig. 4.5 which would result in a completely flat PSD over all frequency), we see that the total integrated power in the case of $f_{in} = f_{\text{nyq}}/2$ that there is more signal power at the output when $f_{\text{chip}} = f_{\text{nyq}}$ (third/fourth row Fig. 4.5) then the case of $f_{\text{chip}} = \infty$. The gain roll-off with f_{in} and the absolute magnitude of the downconverted power is illustrated in Fig. 4.6(a).

We point out that the average difference in downconverted power from f_{in} close to DC versus $f_{\text{in}} \approx f_{\text{chip}}/2$ is approximately $10 * log10((2/\pi)^2) \approx -3.92dB$. While it is desirable



(a) $P_{\text{sig,out}}$ Vs. f_{in} (b) P_{total} from $-2.5GHz < f < 2.5GHz$

Figure 4.6: The effect of $f_s = f_{\text{chip}}$ on output signal power from a single RMPI channel: (a) Power output from a single channel ($P_{\text{sig,out}}$) for several values of chosen chipping rate f_{chip}. (b) Total power contained in PSD $\Upsilon(f)$ as a function of f_{chip}, note that at higher f_{chip} values, a fixed amount of power is spread across a greater range of frequencies lowering the effective average "gain" for a given f_{in}..

to minimize f_{chip}, reducing frequency dependent variation of downconverted power is also desirable and can be achieved by increasing f_{chip}. The cost in increasing f_{chip} is of course increased power consumption of the PRBS generators used as well as a reduction in total power gain across the entire bandwidth of the channel as depicted in Fig. 4.6(b).

PRBS Length

Once f_{chip} is fixed ($f_{\text{chip}} = f_{\text{nyq}}$), it is natural to ask what length (N_{chip}) of PRBS is necessary. This is particularly important in light of the fact that a multi-pole/multi-zero low pass filter is being used in place of an ideal filter, see Ch. 4.3. As explained in Ch. 4.2.1 and quantified in Ch. 4.4.2, we see that the power spectrum of a PRBS is concentrated in the harmonics of the fundamental repetition rate $f_{\text{prbs}} = f_{\text{chip}}/N_{\text{chip}} = f_{\text{nyq}}/N_{\text{chip}}$. Let us assume for the moment that $|H_{\text{sys}}(f)|^2 \propto C/(f_c^2 + f^2)$, $C \in \mathbb{R}$ where f_c is the 3 dB frequency of the single-pole system. Assuming that we are operating with N_{chip} and f_{chip} such that $T_{\text{sig}} \leq N_{\text{chip}}\Delta T$, we see that as $|H_{\text{sys}}(f)|^2$ attenuates signals as $\approx 1/f^2$ that on average, the lowest average $\text{P}_{\text{sig,out}}(f_{\text{in}})$ will be the result of $f_{\text{in}} = (k+1/2)f_{\text{prbs}},\ k \in \{1, \ldots, N_{\text{chip}}/2\}$ lying in between two of the PRBS harmonics. Thus, f_{in} falling between two harmonics will result in the minimum frequency content of the downconverted spectra lying at $f \approx f_{\text{prbs}}/2$ and will therefore result

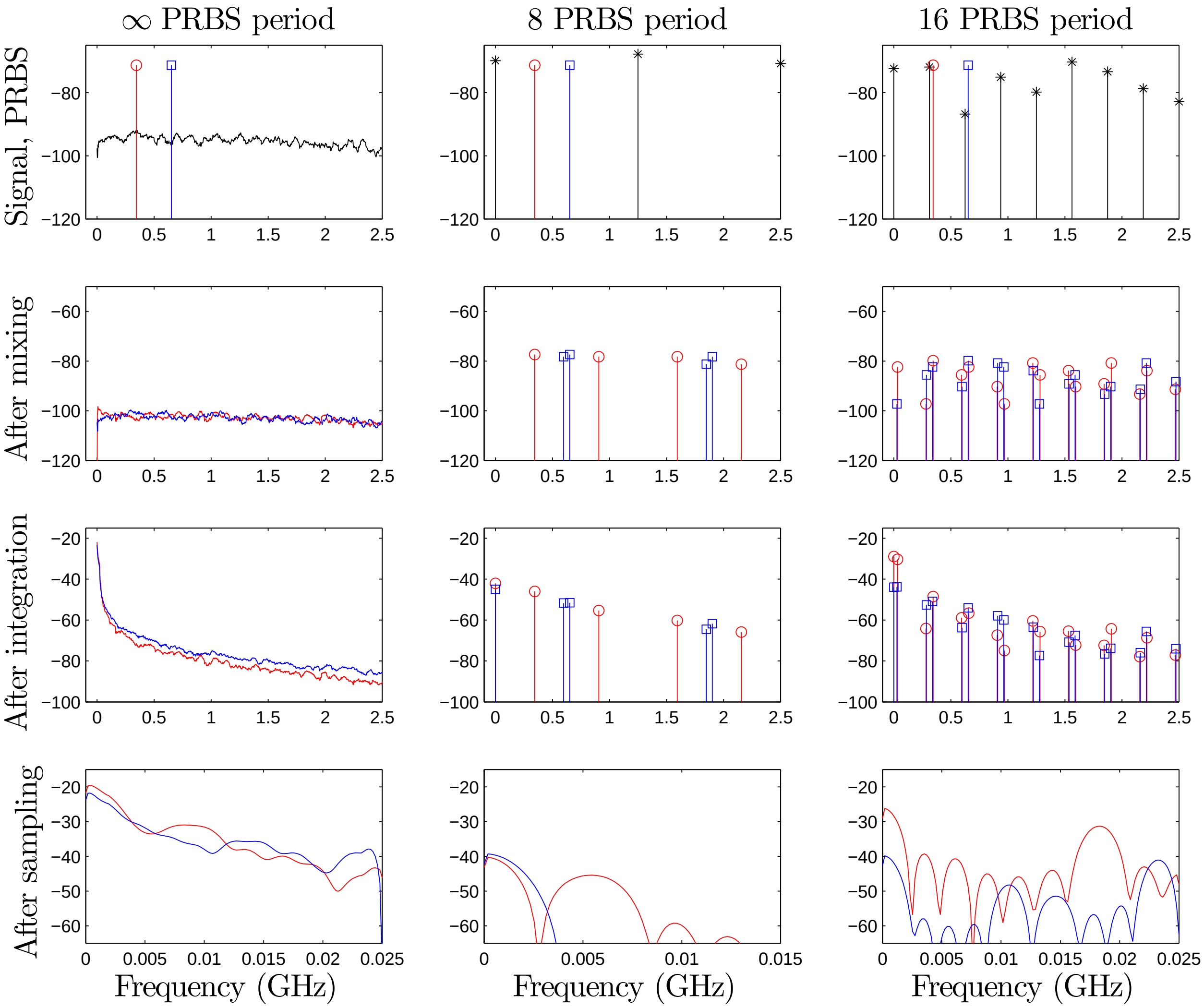


Figure 4.7: Example of the effect of PRBS length N_{chip} at a fixed toggling rate f_{chip} on the output power spectrum. The situation depicted is for signal windows $T_{\text{win}} = N\Delta T$ with $N = 2048$. Shown are two possible input tones (red and blue lines). The columns show the situation of $N_{\text{chip}} = \infty$ ($N_{\text{chip}} = 2048$ effectively), 8, and 16 respectively. The rows show (from top to bottom) an overlay of the two potential tone inputs and the PRBS spectrum, the spectrum after the input and PRBS are mixed, the spectrum of the output of the integrator, and the spectrum after sampling the output of the integrator. The power contained in the depicted bandwidth is identical for both tones for the $N_{\text{chip}} = \infty$ case whereas they are quite dissimilar for $N_{\text{chip}} = 8$. For the case of $N_{\text{chip}} = 16$, this variation is reduced by the reduction in distance between a PRBS tone and either of the inputs.

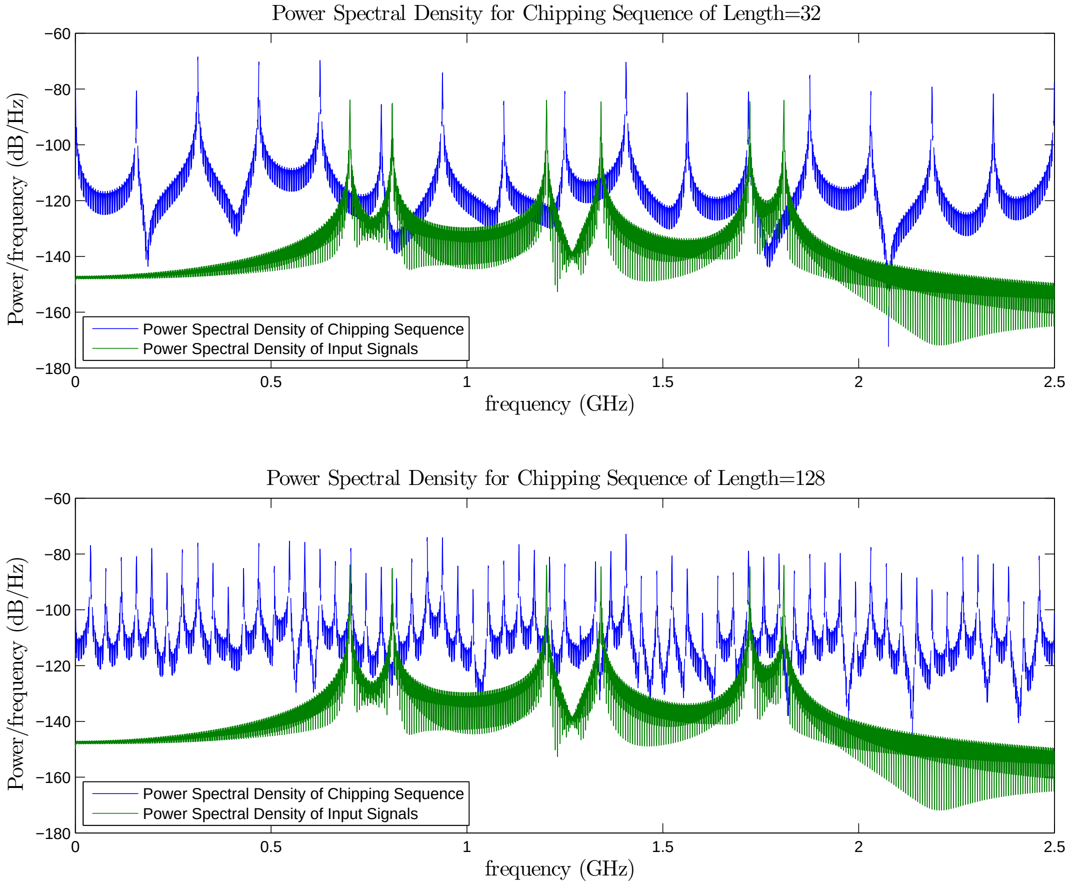


Figure 4.8: Spectrum of PRBS for two different lengths.

in the lowest $P_{\text{sig,out}}$ as it will be the most heavily attenuated by $H_{\text{sys}}(f)$; this is illustrated graphically in Fig. 4.7. The reduction in average SNR in these regions will make signals from this region the most difficult to recover/estimate the parameters of; this is demonstrated numerically in Fig. 4.9. Intuitively, the basic trade-offs can be understood by considering Fig. 4.8. Fig. 4.8 shows the superposition of the power spectra of a frequency sparse input (6 tones) and a PRBS PSD for $N_{\text{chip}} = 32$ (top) and $N_{\text{chip}} = 128$ (bottom). Note that in both situations the total integrated power contained in the spectrum of $N_{\text{chip}} = 32, 128$ is the same. What is different is that the number of discrete tones for the $N_{\text{chip}} = 128$ spectrum is 4 times that in the $N_{\text{chip}} = 32$ spectrum. Given the fixed power content, this also means that the average power contained in each tone of the $N_{\text{chip}} = 128$ spectrum is $1/4$ that contained in a tone of the $N_{\text{chip}} = 32$ spectrum. As a result, the number of downconverted tones within a fixed bandwidth will be larger for higher N_{chip}; although the tones will be on average at a lower power, the power contained within a fixed bandwidth at the output of the integrator will be greater than or equal to that in the case of lower N_{chip} and will also reduce the downconverted power variation as a function of input frequency—this trend is desirable in light of principle 1. Numerical validation of the reasoning behind Fig. 4.7 is provided in the simulations of Fig. 4.10(a)(b). While larger values of N_{chip} result in increased $\text{P}_{\text{sig,out}}$ and reduced output power variation for a fixed input power, it is desirable to minimize N_{chip} as the power requirement at a given f_{chip} grows as $\log_2(N_{\text{chip}})$ if using an LFSR. We also point out that this benefit is limited by the duration T_{int}. The maximum effective sequence length for a given T_{int} is $N_{\text{int}} = T_{\text{int}}/\Delta T = T_{\text{int}}/T_{\text{nyq}} \in \mathbb{Z}$; this is essentially a consequence of digital windowing.

4.4.3 Number of Channels

For a fixed overall sampling rate (f_{agg}) it is possible to decrease the ADC sampling rate of each channel (f_{adc}) by increasing the number of channels N_{ch}. For an aggregate back-end sampling rate ($f_{agg} = \text{N}_{\text{ch}} f_{\text{nyq}}/(\text{N}_{\text{usamp}})$), where N_{usamp} is the factor under Nyquist at which the back-end ADC of a single channel digitizes the channel, some questions that arise when considering a parallelization strategy include:

1. How does performance change (if at all) with an increased number of channels?
2. What is the marginal cost in terms of resources such as power and die area?
3. What are the constraints on achievable scaling?

When we answer the above questions, we fix f_{agg} and the time duration of the signal to recover T_{sig}. The answer to the question (1) turns out to be that there is indeed a benefit to employing parallelization. This benefit is illustrated by simulations shown in Fig. 4.11. Although there is no absolute increase in the number of measurements M acquired for a given time window, there are several distinct benefits to employing a parallelization strategy which we explain below.

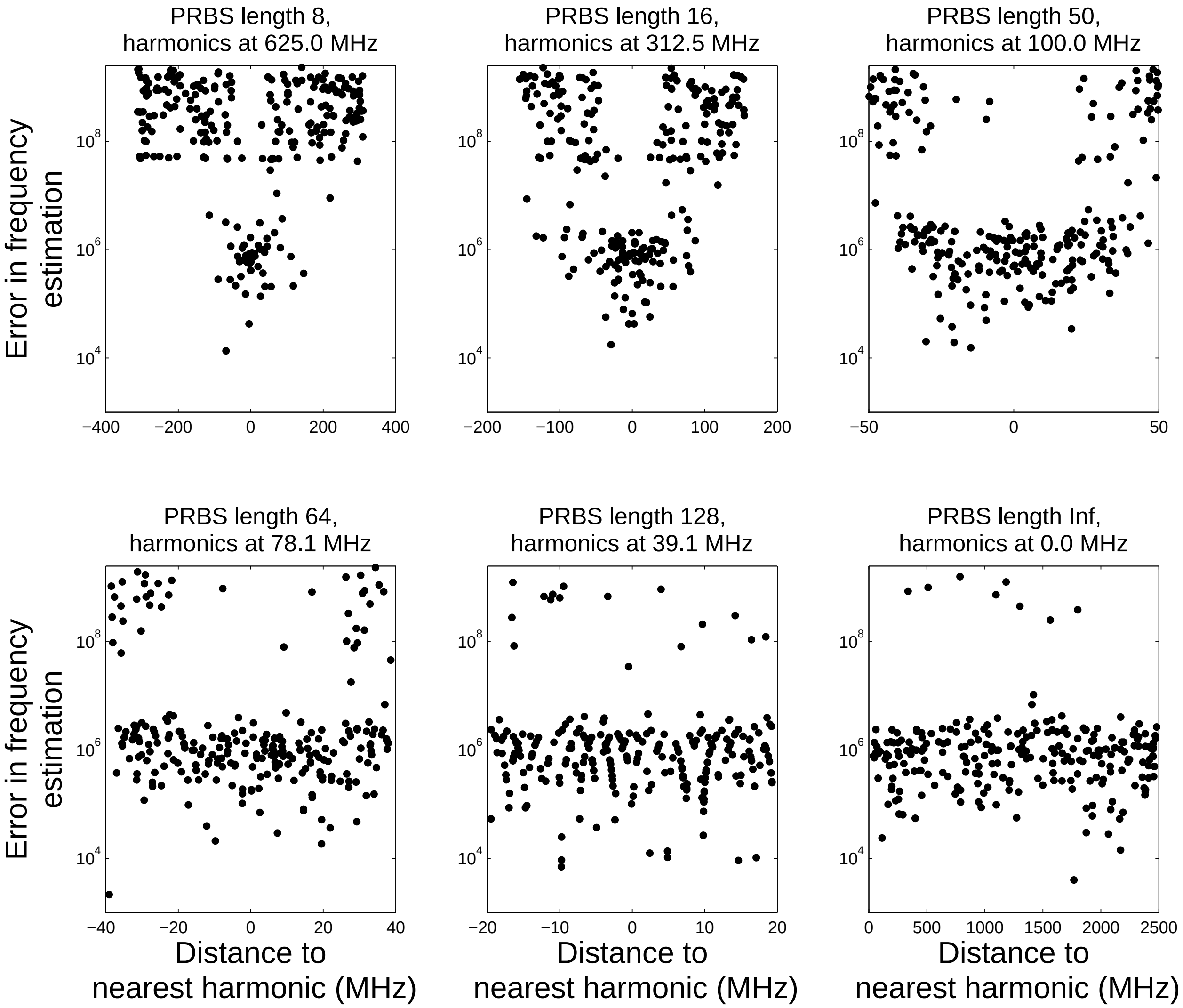
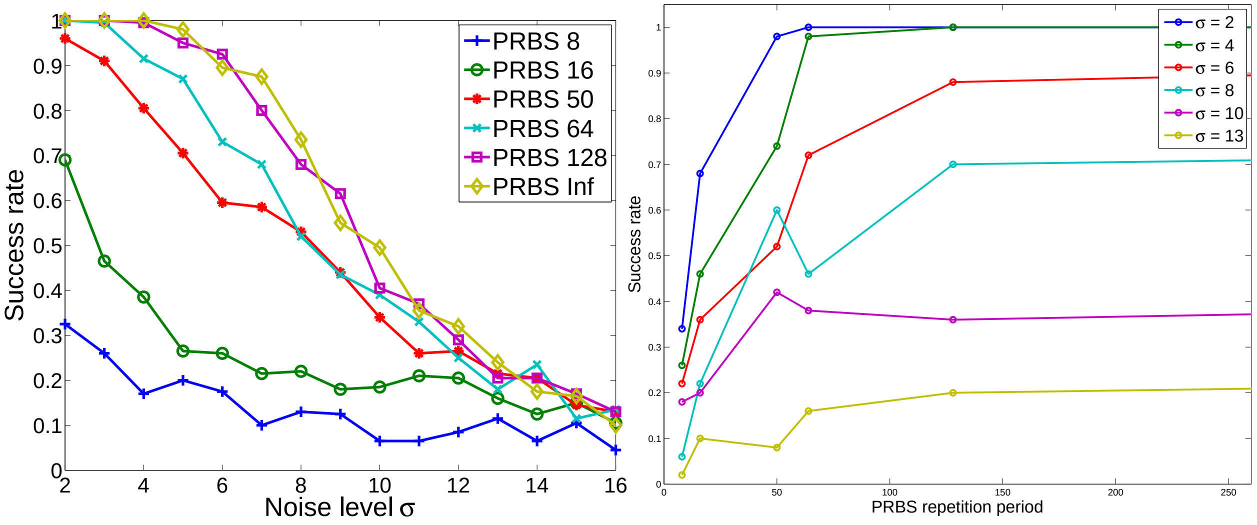


Figure 4.9: Error in carrier frequency estimation vs. distance from nearest harmonic. The distance to the nearest harmonic decreases with increasing values of the PRBS length N_{chip}. The sequence of plots in the figure show average error in frequency estimation reduces as N_{chip} is increased.



(a) Success Rate of Recovery Vs. Noise level σ (b) Success Rate of Recovery Vs N_{chip}

Figure 4.10: Success rate of frequency estimation across noise, for various PRBS repetition lengths, via simulation. The noise level is unphysical, and for comparison purposes only. A PRBS of length 128 or higher is most robust to noise. A "success" is defined as estimation error of less than 5MHz (each data point is the outcome of 200 independent trials)

Improved Φ Matrix Coherence Properties From a purely theoretical standpoint, more channels is better, since the matrix representation of the RMPI Φ better approximates a signed Bernoulli matrix. This results in lower coherence and isometry constants δ_s which translate to better theoretical recovery guarantees. Intuitively, there is more randomization involved, and less chance that a signal passes through the sieve. In general, as N_{ch} increases, the classes of signals we can efficiently sample becomes larger. With a single channel, we can handle spectrally sparse signals (or any other type of signal that is generally diffuse in time). As N_{ch} increases, we can handle signals that are more localized in time.

Robustness to Power Variation for a Given Chipping Sequence The reasoning used in Ch. 4.4.2 to explain the relationship between N_{chip} and successful signal recovery/parameter extraction all relied on arguments based on average power of a given PRBS harmonic, or the use of the PSD to compute $P_{\text{sig, out}}$. A given instance of a PRBS will have power variation from one harmonic to another as illustrated in Fig. 4.8. This power variation will make receiver performance characteristics such as gain, dynamic range, and sensitivity highly frequency dependent. The use of several distinct PRBS via parallel channels reduces this variation in an average sense and is consistent with the spirit of design principle 2.

Integration Window Length Increasing the number of channels for a fixed back-end sampling rate has the effect of increasing the length of time each integration occurs. For a given integration window size, the effective PSD corresponds to a sequence with an $N_{\text{chip}} = N_{\text{int}}$. There is no benefit in increasing the N_{chip} beyond N_{int} as the spectral content is constrained by the window size.

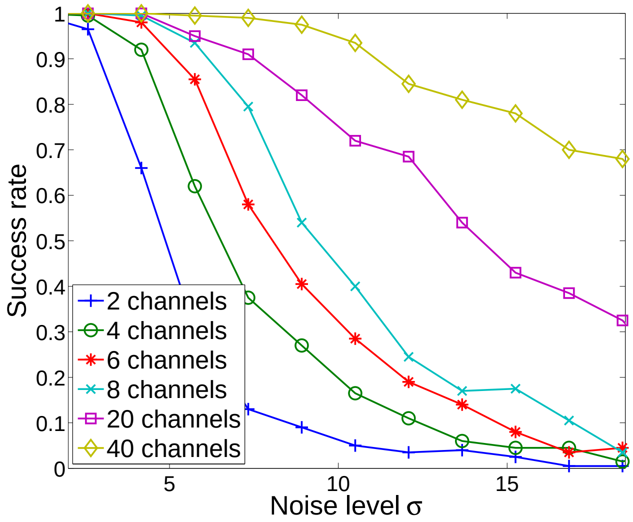


Figure 4.11: Same setup as Fig. 4.10. The overall data output rate is fixed, and the number of channels is varied (as the number of channels increases, the sampling rate of each channel decreases). Designs with more channels are more robust to noise. estimation error of less than 5MHz (each data point is the outcome of 200 independent trials).

Difficulties which may discourage the implementation of the maximum number of channels (for a given window size) consist principally of issues due to spatial constraints, complexity of signal routing, and differences in the performance of each channel due to statistical process variation.

An increased number of channels will require larger amounts of die area. This increased area will require increased complexity of design and power for distributing both the input signal and the Nyquist-rate clock. In addition, the integrator time constant has to be adjusted to accommodate the increased time of integration.

Consequently, the larger area over which the circuit is distributed along with the increased complexity in the routing will result in differences in signal path delays and other statistical variation related effects on the transfer function. This will increase the reliance of the system on calibration as it is sensitive to inaccuracies in Φ and may also make calibration more difficult.

Even with 8 channels, timing differences can be significant, and differences in time and phase delays hurt system performance (although this can be compensated for by calibration). The design of the CMOS RMPI intentionally sought to minimize the number of channels while retaining the parallelization required to ensure robust performance. Given these and other limitations imposed by pragmatic considerations such as available testing equipment, the final design uses 8 channels.

4.5 Evaluation of Robustness

In order to validate the final design, the effects of several nonidealities were empirically studied through numerical simulation. In the context of this section, a nonideality is any source of corruption that would lead to a deviation from the calibrated linear model. The nonidealities discussed in this section include: thermal noise, clock jitter, nonlinearities, cross-talk, and clipping. All of these sources of corruption were modeled and investigated using various simulation techniques including the simulink simulations described in App. A. Unless otherwise stated, the overall system parameters such as N_{chip}, N_{ch}, f_{agg}, and p_1 were fixed. In addition, while the effects of nonidealities were studied by varying their values one at a time, all simulations presented used a set of fixed nominal values for all other sources of nonideality. The input stimulus used was a radar-pulse with a Tukey window with $T_{\text{pulse}} = 100\ ns$. In the case of reconstructing the time-domain baseband pulse window, the criterion for declaring a success was reconstructing the input window to an MSE < 0.1.

4.5.1 Modeling Sources of Sample Corruption

Although the RMPI is not an LTI system, it is well approximated by a linear model. With the exception of compensating for transistor nonlinearity (which requires a slight extension), a suitable model to study the effects of different types of sample corruption is to aggregate all the various effects and treat them as additive noise:

$$\mathbf{y} = \Phi(\mathbf{x} + \sigma_1 \mathbf{z}_1) + \sigma_2 \mathbf{z}_2 + \mathbf{z}_3, \tag{4.5.1}$$

where $\mathbf{z}_1$ is iid $N(0, \sigma_1^2)$, $\mathbf{z}_2$ is iid $N(0, \sigma_2^2)$, and $\mathbf{z}_3$ encompasses the non-white corruption terms. The three types of additive errors in Eq. (4.5.1) represent corruptions that: occur prior to modulation ($\sigma_1 \mathbf{z}_1$), after the operation of the RMPI ($\sigma_2 \mathbf{z}_2$), and a term $\mathbf{z}_3$ to express the additive errors which cannot be modeled as Gaussian in nature. It is possible to use a *weighted* norm to incorporate the effects of $\Phi \mathbf{z}_1$ and $\mathbf{z}_2$; see [55] for details. For now, we use the regular ℓ_2 norm, so that we wish our candidate reconstructed signal $\hat{\mathbf{x}}$ to satisfy

$$\|\Phi \hat{\mathbf{x}} - \mathbf{y}\|_2 \leq \varepsilon \tag{4.5.2}$$

for an appropriate value of ε.

4.5.2 Reconstruction vs. Input SNR

Assuming the use of an ℓ_1-based reconstruction method and using the model of noise in Eq. (4.5.1), excluding $\mathbf{z}_3$, we can predict the reconstruction performance by considering an input-referred noise model

$$\mathbf{y} = \Phi \mathbf{x} + \sigma_{\text{in}} \mathbf{z}_{\text{in}}, \quad \Sigma = \sigma_1^2 \Phi \Phi^T + \sigma_2^2 I, \tag{4.5.3}$$

where Σ is the covariance matrix of $\sigma_{\text{in}} \mathbf{z}_{\text{in}}$. Fig. 4.12 shows the relationship between input SNR and MSE of the reconstructed baseband pulse envelope for a fixed input signal amplitude $A = 100\ \mu V$. The blue line indicates an MSE = 0.1. The SNR at which the MSE

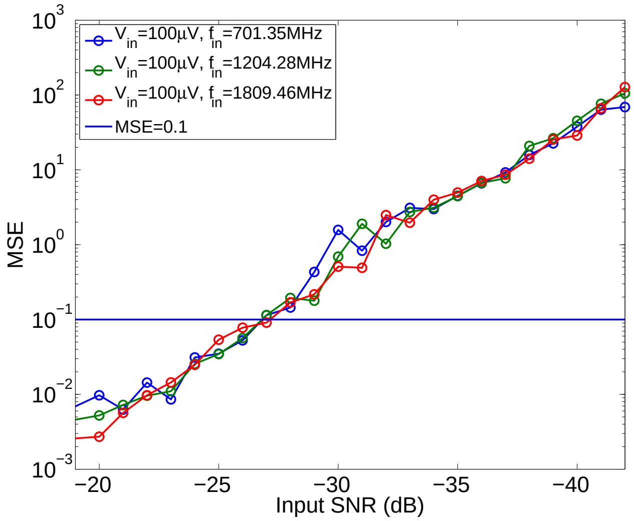


Figure 4.12: Reconstruction of 100 ns pulse vs. Input SNR

curves intersect the MSE line (SNR $= -33$ dB) indicates the minimum SNR at which the RMPI successfully recovers the pulse.

The approximately linear relationship between SNR and MSE can be attributed to the use of Eq. (2.2.5) for reconstruction. For BPDN, we have a bound on the discrepancy between the collected observations and recovered signal denoted by ϵ in Eq. (4.5.2). The use of this bound will result in a reconstruction error bounded by $C\epsilon$ for some constant C. The recovery program employed in this paper [86] solves Eq. (2.2.5) for successively lower values of ϵ until the lowest ϵ value is found. ϵ will be bounded below by the uncertainty introduced by noise during the measurement process, resulting in the linear relationship observed in Fig. 4.12.

4.5.3 Reconstruction vs. Jitter

Another nonideality of key concern was timing jitter. The effects of jitter were investigated at two different locations within the system: jitter of the sampling clock on the back-end ADCs and jitter on the Nyquist-rate clock that toggles the PRBS generators. At both locations, we modeled the effect of jitter on the obtained measurements as $y_m = y(t_m + \delta_m)$ for $m = \{1, \ldots, M\}$, where δ_m are i.i.d. Gaussian with variance σ_j. The sample corruption introduced by timing jitter (this will contribute to the $\sigma_2 z_2$ term in Eq. (4.5.1)) in the back-end sampler will be the same as in the Nyquist case. The resulting output SNR, ignoring all other sources of error, is

$$\mathrm{SNR}_{\sigma_j} \propto \frac{1}{(2\pi)^2 f^2 \sigma_j^2}. \tag{4.5.4}$$

In the case of the RMPI, however, jitter on the Nyquist-rate clock that toggles the PRBS generator also has to be considered. For acquisition of a signal $x(t)$, we are interested in the effect of jitter on each of the output samples

$$y[m] = \sum_{\ell=1}^{N_{\mathrm{int}}} \epsilon_\ell \int_{t_{\ell-1}}^{t_\ell} x(t)dt, \tag{4.5.5}$$

where $t_\ell = \ell T_{\mathrm{nyq}}$ represents the ideal set of time points at which the PRBS would toggle and $\epsilon_\ell \in \{\pm 1\}$ is the value of the PRBS from $[t_{\ell-1}, t_\ell)$. We now compute the variance of the error due to nonideal toggle points $t'_\ell = t_\ell + \delta_\ell$, where $\delta_\ell = \sigma z_\ell$ and $z_\ell \sim \mathcal{N}(0,1)$. We write the ideal measurement with unjittered time points as $y[m]$ and the measurements with jittered time points as

$$y_\sigma[m] = \sum_{\ell=1}^{N_{\mathrm{int}}} \epsilon_\ell \int_{t'_{\ell-1}}^{t'_\ell} x(t)dt. \tag{4.5.6}$$

We can then write the error caused by jitter:

$$e[m] = y[m] - y_\sigma[m] = \sum_{\ell=1}^{N_{\mathrm{int}}} \epsilon_\ell \left[\int_{t'_{\ell-1}}^{t_{\ell-1}} x(t)dt - \int_{t'_\ell}^{t_\ell} x(t)dt. \right] \tag{4.5.7}$$

If we consider a pure tone input $x(t) = Ae^{j2\pi f_{\mathrm{in}} t}$ and compute the variance of $e[m]$, we get

$$\mathbb{E}[|e[m]|^2] = 2|A|^2 N_{\mathrm{int}} \sigma^2. \tag{4.5.8}$$

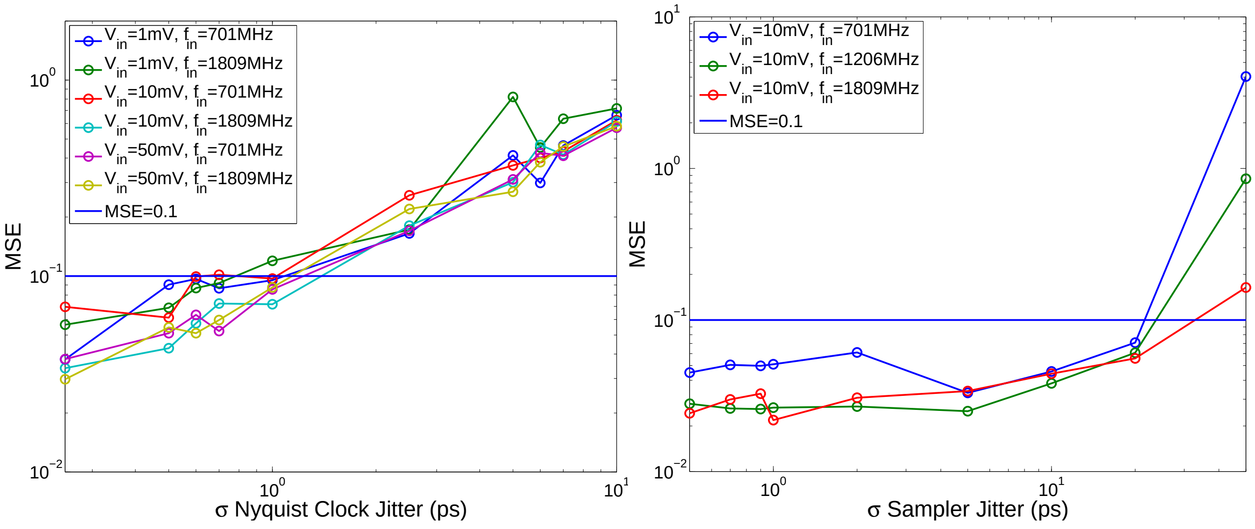
Similarly, the output sample power for a pure tone input will be

$$P_{\mathrm{in}} = \frac{|A|^2 N_{\mathrm{int}}}{f_{\mathrm{nyq}}^2} \operatorname{sinc}^2(f_{\mathrm{in}}/f_{\mathrm{nyq}}). \tag{4.5.9}$$

Thus, the SNR due to jitter in the Nyquist-rate clock is

$$\mathrm{SNR}_\sigma = \frac{\mathrm{sinc}^2(f_{\mathrm{in}}/f_{\mathrm{nyq}})}{2f_{\mathrm{nyq}}^2\sigma^2}. \tag{4.5.10}$$

The above calculations are verified by figures 4.13(a) and 4.13(b) which show the effect of jitter on the Nyquist-rate PRB clock ($\sigma_{\mathrm{Nyq,Clk}}$) and the effect of jitter on the ADC sampling clock ($\sigma_{\mathrm{ADC,Clk}}$), respectively. We see in Fig. 4.13 that there is an approximately linear relationship between $\sigma_{\mathrm{Nyq,Clk}}$ and the reconstruction MSE. As indicated by Eq. (4.5.10), the sample corruption due to jitter variance behaves similarly to AWGN, and should therefore have a relationship similar to that observed in Fig. 4.12. Note that the error variance due to jitter in the Nyquist-rate clock is independent of f_{in}, in contrast to errors caused by jitter in the sampling clock of an ADC. We also observe that the signal recovery process is much less sensitive to clock jitter in the sampling clock of the back-end digitizers. This is expected, as the back-end ADCs sample the outputs of integrators and therefore have signal power confined to a much smaller bandwidth.



(a) MSE vs. PRBS Clk Jitter $\sigma_{Nyq,Clk}$ (b) MSE vs. Digitizer Jitter $\sigma_{\mathrm{ADC,Clk}}$.

Figure 4.13: Reconstruction of 100 ns pulse vs. (a) $\sigma_{\mathrm{PRBS,clk}}$ and (b) $\sigma_{sampler}$

4.5.4 Nonlinearity

While some work [94] has been done to extend CS to nonlinear measurement, the theory for addressing nonlinearities remains considerably less developed than for dealing with AWGN. In this section we describe a correction technique that was developed to compensate for measurements corrupted by LNA dominated nonlinearity. Unlike other sources of sample corruption discussed in this section, nonlinearity-induced signal distortion cannot be modeled as AWGN. In addition, there are several types of nonlinearity which occur in practice and are difficult to both model and to measure. In a conventional receiver, the concerns when considering the effect of nonlinearity on receiver performance are the desensitization of the LNA and the generation and/or inadequate suppression of nonlinear distortion products within the receive bandwidth. In terms of CS, there are additional considerations, e.g., the de-sparsification of the input signal which would lead to failure of the reconstruction program altogether. Another motivation for examining the effects of nonlinearity comes from the fact that we are applying CS for extremely wide-band receivers. This is in stark contrast to narrow-band receivers, where many of the deleterious effects of nonlinearity are mitigated through the use of high rejection-ratio input filters. Thus, one question of considerable interest that we explored was whether or not nonlinear distortion could be modeled and compensated for via post-processing to improve the reconstruction dynamic range. We present here a technique that was devised to compensate for a known polynomial nonlinearity of the form

$$\mathbf{y} = \Phi p(\mathbf{x}), \quad p(\mathbf{x}) = a_0 + a_1\mathbf{x} + a_2\mathbf{x^2} + \dots. \tag{4.5.11}$$

In general, nonlinearities come from several sources, including the mixer and integrator. In many contemporary receiver architectures, however, the receiver nonlinearity is dominated by the LNA [95]. Eq. 4.5.11 models the LNA-dominated nonlinearity scenario. We now outline the basic technique below. Start with the assumption that $\mathbf{y} = \Phi\mathbf{x}$ and solve Eq. 2.2.5 to generate an initial guess for the solution $\hat{\mathbf{x}}_1$. Linearize $p(\mathbf{x})$ about $\hat{\mathbf{x}}_1$ and update the model to be

$$\mathbf{y} = \Phi p(\mathbf{x}) + (\Phi p'(\mathbf{x}))h, \quad \tilde{\mathbf{y}} = \mathbf{y} - \Phi p(\mathbf{x}),\ \tilde{\Phi} = \Phi p'(\mathbf{x}). \tag{4.5.12}$$

The updated model is then solved for $\tilde{\mathbf{y}}$, $\tilde{\Phi}$ is solved for h, and the reconstruction $\hat{\mathbf{x}}$ is updated via

$$\hat{\mathbf{x}}_{i+1} \leftarrow \hat{\mathbf{x}}_i + h. \tag{4.5.13}$$

This process is repeated until the difference between successive iterates is below a specified threshold. Fig. 4.14 shows the results of numerical experiments simulating the effectiveness of the developed technique. The reconstruction MSE is plotted as a function of input amplitude. At lower amplitudes, compensation makes little difference due to the relative lack of generated nonlinear products. As the amplitude is increased, we observe considerable improvement in the quality of reconstruction. At very high amplitudes, the solver does not converge and the technique makes little difference.

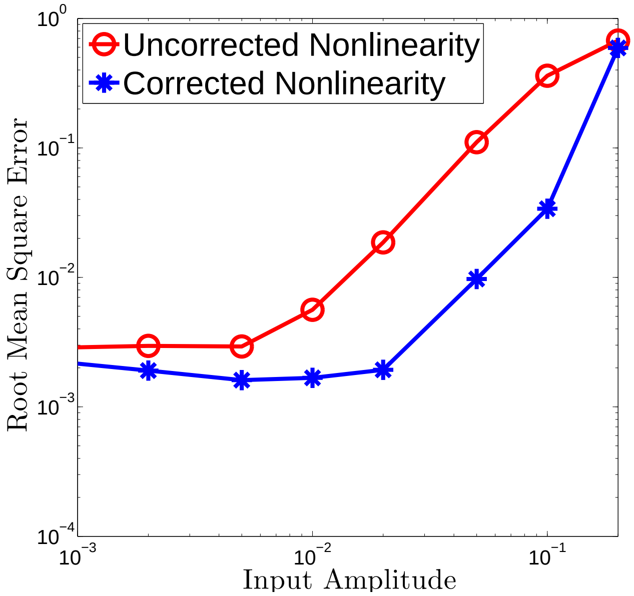


Figure 4.14: Error of reconstruction as a function of input amplitude, with and without nonlinearity correction. Data points were generated for pulses with $f_{\text{carrier}} \approx 700\ MHz$ and duration 100 ns .

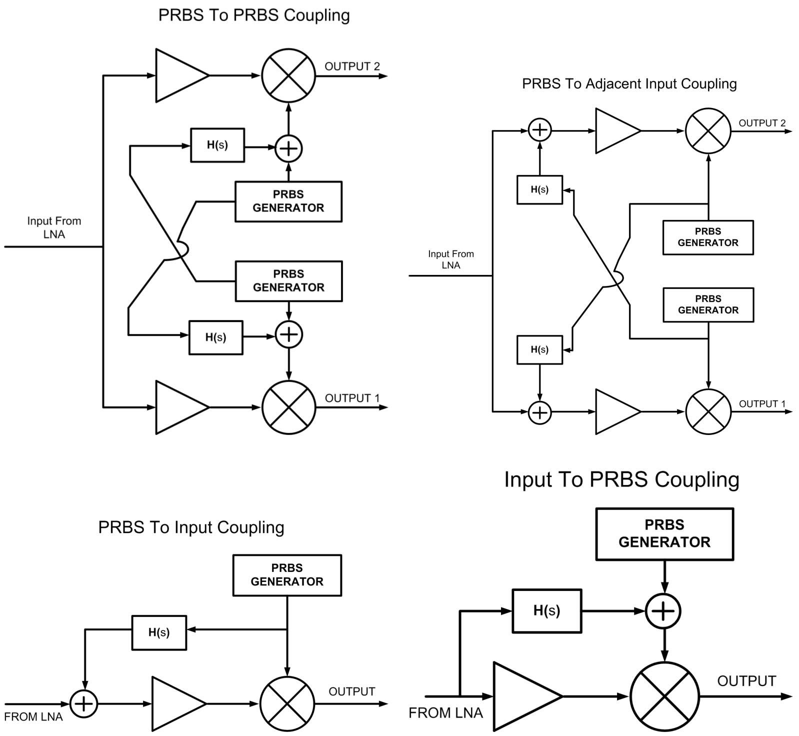


Figure 4.15: The types of cross-talk mechanisms considered in the testing show in Figure 4.16. The $H(s)$ block represents the unwanted cross-talk that might occur.

4.5.5 Cross-talk

Due to the large number of channels in the RMPI, a chief concern during its design was the potentially deleterious effect of several forms of cross-talk including both channel-to-channel coupling as well as coupling between critical nodes within a given channel. The dominant sources of cross-talk are shown in Fig. 4.15. Using values for several worst-case coupling mechanisms reported in the literature [96] as a guide, extensive numerical simulations were conducted to investigate the robustness of the RMPI to these mechanisms. Simulation results studying the quality of signal reconstruction as a function of the level of coupling were studied for the four mechanisms depicted in Fig. 4.16. The studies revealed that the RMPI was most vulnerable to PRBS-to-input cross-talk. While the simulations ultimately indicated acceptable performance for the simulated values, several safeguards were incorporated in the physical implementation to mitigate risk and ensure desired performance: each channel's analog and digital supplies were isolated from one another; the supplies and references for the clock distribution was also isolated from supplies for the signal bearing circuits. In addition, a triple-well design strategy along with extensive use of guard rings was made around all blocks, sub-blocks, and channels.

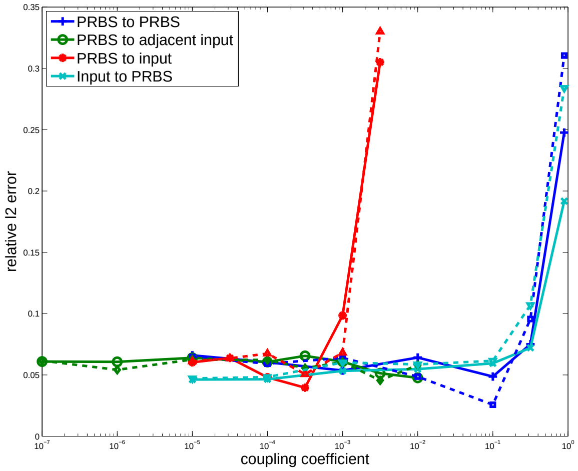


Figure 4.16: Results of the cross-talk simulations. Solid line is a 700 MHz input; dashed line is a 2.1 GHz input. Both inputs were pulses 100 ns long and amplitude .01 V (-20 dBFS).

4.5.6 Clipping

Another concern during the design of the system was the limitations on dynamic range imposed by clipping from the integration of input signals with both large amplitude and temporal extent. Since distinct PRBS lead to frequency and PRBS sequence dependent gain, we provide an analysis of clipping in an average sense. We analyze clipping using the ideal Φ with ± 1 entries and work with discrete signals. From inspection we know the worst-case input, in terms of causing the output to rail, is a signal x_n that exactly mimics the PRBS $\boldsymbol{\epsilon}_i \subset \{-1, 1\}^N$. This situation, however, is extremely unlikely to arise in practice. Since the PRBS $\epsilon_{i,n}$ is essentially random, then a "typical worst-case" signal whose analysis imparts insight is any arbitrary but slowly-varying signal with amplitude A_{max} (a signal approximately constant over several Nyquist periods). Therefore, we analyze the case where x_n is a DC signal with amplitude A_{max}. If we let $S_k = \sum_{n=1}^{k} \epsilon_{i,n}$, we can define the event that clipping occurred as the probability that the magnitude of the integrator output

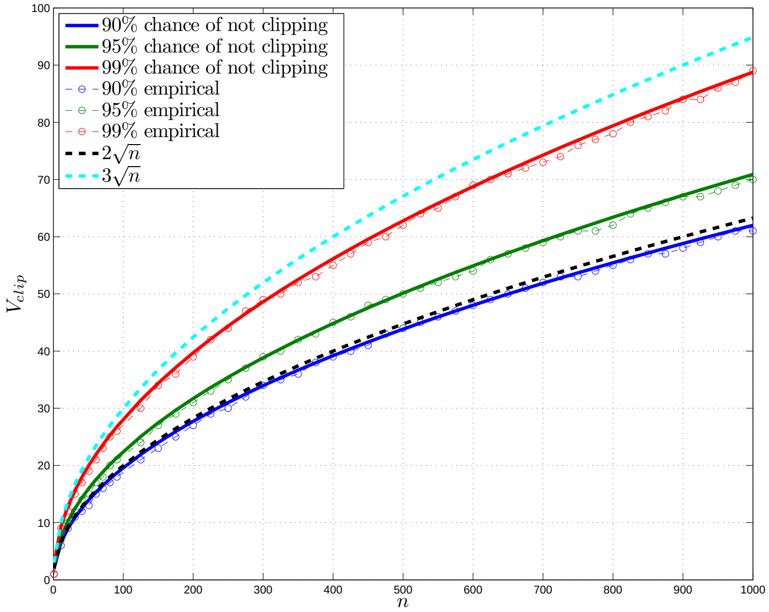


Figure 4.17: Simulation to verify the chance of a random walk clipping. The solid lines are the predictions from using the inverse CDF of a normal distribution, and the circles are the empirical values from 10,000 Monte Carlo simulations, which show excellent agreement, thus validation the $2p$ argument. Furthermore, the lines are bounded between $2\sqrt{n}$ and $3\sqrt{n}$, which provides a useful heuristic. To use the chart: suppose $V_{clip} = 70$ and we want to be 99% sure that a maximum amplitude signal will not clip. Then we must reset the integrator every $n = 600\Delta T$.

exceeds $|V_{clip}|$ at any time $k \leq N$ within the integration window of length $T_{\text{int}} = N_{\text{int}}\Delta T$: $|S_k| \geq \frac{V_{clip}\Delta T}{A_{max}} \equiv v$. Below, we compute $\mathbb{P}(S_k \geq v,\, k \leq N)$ in three distinct parts.

First, consider the sum at time $(k = N)$ and let $p \equiv \mathbb{P}(S_N \geq v)$. The sum S_N is a shifted and scaled sum of Bernoulli random variables. For large N, S_N is known to be well-approximated by the CDF of a Gaussian variable with $\mu = N/2$ and $\sigma = \sqrt{N/4}$. Next, consider the situation where the intermediate sum $S_k = v$ and look at the set of all length $N - k$ random walks that start at this point. The expected value at time N given $S_k = v$ is v since a random walk has mean 0. Moreover, half of the set will end up with value greater than v, and half will end up with value less than v (this reasoning requires that $N - k$ is odd to prevent a final value of exactly v at $k = N$). Therefore, the number of sums S_N which exceed v accounts for half of the sums S_k that had value v at any time $k \leq N$. Thus $\mathbb{P}(S_k \geq v,\, k \leq N) = 2p$. Finally, accounting for negative amplitudes, $p_{\text{clip}} = \mathbb{P}(|S_k| \geq v,\, l \leq N) = 4p$. This reasoning is somewhat imprecise since it depends on the respective parities of v (assumed to be an integer) and N; for large N and values of v that are not too close to N, the error is small. Fig. 4.17 verifies these arguments numerically.

4.6 Recovery Algorithms

In this section we briefly discuss several methods that have been developed for either recovering the full time-domain signal and/or estimating parameters of the signal directly from CS samples. It is emphasized that the content of this section is due to the work others, in particular, several other members of the Caltech/Northrop Grumman A2I team [55,69,83,97,98]. This section is included for the sake of completeness as well as the convenience of the reader.

Algorithms for Time-Domain Signal Recovery

CS theory suggests basis pursuit to recover the signal, but there are many variations. We refer the reader to [88] for a review, and to [55] for techniques used specifically for RMPI samples. Table 4.1 shows a representative sample of state-of-the-art solvers, and their performance on a sample problem generated from the RMPI simulations. The table plots relative reconstruction error $\|\mathbf{x}-\hat{\mathbf{x}}\|_2^2/\|\mathbf{x}\|_2^2$ where $\hat{\mathbf{x}}$ is the estimate produced by the solver. The input $\mathbf{x}$ was a radar pulse, and measurements were recorded from the Simulink model using realistic values for the non-ideal blocks and noise sources. Reconstruction requires knowledge of some sparsifying dictionary Ψ, and the three columns of numbers represent three choices of Ψ: an 8× oversampled multi-scale Gabor time-frequency dictionary, a 32× oversampled Gabor dictionary, and a discrete cosine basis. The discretized version $\mathbf{x}$ of the input $x(t)$ was of size $N = 2048$, and all solvers took on the order of 1 minute or less.

Since the RMPI produces samples which are in a "universal" format, it can be processed in a plethora of ways. Depending on the type of desired data, different algorithms outperform others. As an example, Table 1 lists the time-domain reconstruction performance of several different sparse solvers on a *single* dataset generated by the RMPI Simulink setup. The results suggest that many solvers do quite well and achieve less than 10% relative error.

It also appears that the reweighted techniques used in [55] are among the best. Since this was a single trial, and algorithm parameters need to be tailored for any application, one should not view the results as a comparison. Rather, these give an idea that there are many alternatives to basis pursuit (basis pursuit is equivalent to ℓ_1 synthesis in the table) that display competitive performance.

Solver	Reconstruction Error		
	8× Gabor	32× Gabor	DCT
OMP [99]	1.7e-2	2.6e-2	9.9e-3
OMP (SPAMS [100])	1.0e-2	3.8e-3	6.9e-3
CoSaMP [101]	2.3e+1	DNC	2.3e-2
CoSaMP (modified)	1.7e-2	8.5e-3	4.0e-2
ℓ_1 synthesis [85]	8.4e-2	1.2e-1	5.7e-2
“ ” with reweighting	1.7e-2	4.4e-2	4.0e-2
ℓ_1 analysis [85]	1.2e-2	2.0e-2	7.8e-2
“ ” with reweighting	4.4e-3	2.9e-3	4.4e-2
LARS (SPAMS [100])	1.7e-2	9.0e-3	2.9e-2
ALPS [102]	2.0e-2	1.2e-2	5.2e-2
SL0 [103]	6.7e-2	1.8e-1	1.4e-2
AMP [104]	DNC	DNC	DNC

Table 4.1: State-of-the-art solvers on a realistic sparse recovery problem. The problem uses realistic measurements, and the signal is compressible but not exactly sparse. When the algorithm diverged or failed to converge, we report DNC.

Chapter 5

Hardware Design

In this chapter we present the physical implementation of the RMPI. Whereas previous chapters described the theoretical setup and addressed system-level design concerns, in this chapter we discuss the transistor-level design/simulations performed. The RMPI was fabricated in the IBM CMOS9SF 90 *nm* digital process. The process features 9 metal layers: 6 thin metal layers, 2 thick copper layers and the top LB metal layer. In terms of devices, the process features thin/thick oxide transistors with zero/low/regular/high V_{T} devices. Beyond considering blocks to implement the random-demodulator channels, this section discusses implementation issues specific to the parallelization strategy of the RMPI and describes in detail the clock distribution system. We also highlight implementation decisions made purely to answer research questions and conclude with design suggestions for future designers of RMPI-like systems.

5.1 Physical Relaxation

Constraints on choosing a physical relaxation of the system were discussed in detail in Ch. 4. In the case of most blocks found in ideal models such as presented in [9], the functionality of the ideal blocks were replaced with their physical equivalents of the same name; in most cases, several variants were attempted. For example, implementation of the mixing functionality in the RD was considered with both a passive switching mixer as well as a gilbert-cell mixer. Several additional blocks were also added to enable physical compatibility, such as the addition of a 50 Ω input-matched low-noise amplifier (LNA) and the use of output buffers to drive off-chip digitizers. Further considerations were the result of using parallel channels which required the distribution of a global clock and synchronization scheme.

In general, the strategy employed in the block-level design was to create minimum deviation of the transfer function of an implemented block from its ideal equivalent. Considerations for creating a transfer function with minimum departure from the ideal model were traded off against more practical constraints such as power consumption, linearity, process variation effects, and limited die area.

5.2 Physical Architecture Description

A block diagram of the CMOS RMPI is shown in Fig. 5.1. The RMPI uses, 2 primary supplies: 1.5 V for the RF/analog signal path, and 1.2 V for the purely digital circuits. A 2.5 V supply is required to power the I/O pads to make it compatible with off-chip testing equipment such as an FPGA and logic analyzers. The core system consists of 8 parallel channels with a common input node driven by the input LNA. Each channel modulates the input signal $x(t)$ with a distinct PRBS sequence $c_i(t),\ i \in \{1, \ldots, 8\}$, where i is the channel index, toggling at the Nyquist rate. The output of the mixer $m(t) = x(t)c_i(t)$ is then

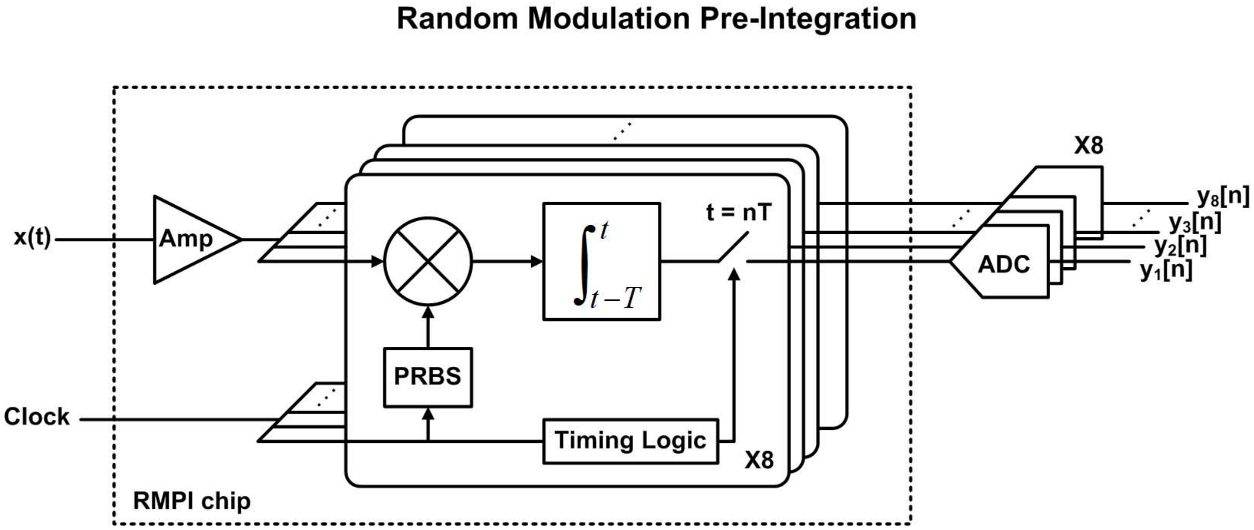


Figure 5.1: Simplified block diagram of the 8-channel CMOS RMPI.

integrated over a fixed time interval T_{int} and digitized at rate $f_{\text{adc}} = 1/T_{\text{int}} \ll f_{\text{nyq}}$. We write the digitized output samples $y_i[m],\ i \in \{1, \ldots, 8\}$ as $y_i[m] = \int_{t-T_{\text{int}}}^{t} x(t)c_i(t)dt,\ t = mT_{\text{int}}$. In our system, we set $T_{\text{int}} = 100/f_{\text{Nyq}}$, where $f_{\text{nyq}} = 4\ GHz$, for a digitization rate of $f_{\text{ADC}} = 40$ *Msps*, leading to an overall system back-end sampling-rate of $f_{agg} = 320\ Msps$.

The general goal was to maximize performance in terms of EIBW, dynamic range, and undersampling ratio $f_{\text{nyq}}/f_{\text{adc}}$. The most challenging performance metric to maximize, in the case of the CMOS RMPI, was the dynamic range. This can be attributed to the fact that unlike conventional receivers, in which the mixing signal at the local-oscillator port (known to RF designers the LO port) is a pure-tone or square wave, the mixing signal in the RMPI is a PRBS. Since the total power in the mixing waveforms in both the PRBS and the LO are constant, in the case of very narrow-band signals, the average signal power downconverted by the PRBS is less that that of a pure-tone because the signal energy of a PRBS is spread over a much larger bandwidth. In the CS literature, this effect has been studied to some degree [105, 106] and is often inaccurately referred to as the "noise-folding" property of CS receivers which is a natural consequence of undersampling. Reductions in dynamic range from undersampling were further exacerbated in early versions of the CMOS RMPI design due to the use of a gilbert-cell mixer. It is well established in the literature that the nonlinearity and $1/f$ noise of the gilbert-cell mixer are the dominant effects which constraint maximum input amplitude A_{max} in conventional receiver chains. These limitations were addressed in the final version of the CMOS RMPI by utilizing a current-mode strategy similar to that used in [29, 32, 95, 107, 108]. The amplitude limiting effects and how they are circumvented by the current-mode receiver architecture is described in detail below.

Current Mode Architecture

A block diagram illustrating the current-mode receiver design strategy along with the more traditional voltage-mode (which we denote Irx and Vrx respectively) strategy is shown in Fig. 5.2[1]. A direct-conversion RF-receiver consists primarily of 4 blocks: LNA, mixer, fre-

[1] The terms voltage-mode and current-mode are initially confusing because the voltage and current of a signal are linearly related by some impedance (Z). In the context of this discussion, the conventions dictating

quency synthesizer (LO), and baseband filter. In a conventional Vrx strategy, the LNA acts as a voltage amplifier which drives a gilbert-cell mixer. The LO-port of the mixer is driven by the frequency synthesizer and creates both an upconverted and downconverted copy of the input signal at its output. The frequency translated copies are then filtered by the baseband filter to prevent aliasing prior to digitization by an off-chip ADC. The dynamic range of the Vrx strategy, shown in the bottom path of Fig. 5.2, is limited by the *active*[2] voltage amplification stage used in the gilbert-cell mixer. This limitation is a result of nonlinear behavior of the mixer switches that result from the large terminal-to-terminal voltages applied to the mixing switch during normal operation.

In Vrx receivers, the employed gilbert-cell mixer consists of 3 parts: an input transconductor, switching-pair, and high-impedance load. The input transconductor pulls current through the switching pair which generates the mixing action. This mixed current is then converted back into a voltage signal by the high-impedance load. In this situation, the voltage between the gates of the switches (driven by a frequency synthesizer) and the drain/source terminals (signal-path) varies widely. This causes fluctuations in the channel-impedance and results in the generation of nonlinear distortion products.

The Irx strategy circumvents this limitation through use of a passive mixer. The basic premise is that if $V_{\mathrm{gs}}/V_{\mathrm{ds}}$ are kept relatively low (for high-signal inputs) to minimize switch-impedance variation, the generation of distortion products can be avoided. This is accomplished by use of a passive-mixer which is loaded with a very low-impedance at its outputs. The signal is converted from a large voltage to a large current through the use of a high gain transconductance amplifier. The large current is passed through the passive mixer and then converted back into a large voltage by a transimpedance amplifier (TIA) at the back-end (which also usually serves as the first stage of the baseband filter). This strategy

whether a system is operating in the voltage or current domains (mode) is determined by in which domain, the relative magnitude of the signal amplitude is larger than in other parts of the system. For example a very high current when passed through a very low impedance node will result in a relatively small voltage signal. The term is subjective as many situations in which the magnitude of both voltage and current representations of the signal can be considered to be moderate. In the case of Irx and Vrx receivers the relative magnitudes are extreme and thus more clear.

[2] The term "active" is used by circuit engineers to refer to the use transistors which amplify signals.

is shown in the upper path of of Fig. 5.2. The price paid for the increased dynamic range is a significant increase in power consumption. However, the marginal benefit to system dynamic range is well worth the power cost as evidenced by the current popularity of the approach in both academia and industry. Even larger input signals can be handled by the receiver by performing the V-to-I conversion in the LNA itself through the use of a low-noise transconductance amplifier (LNTA) [109], however, this strategy was not employed in the CMOS RMPI and has not been investigated.

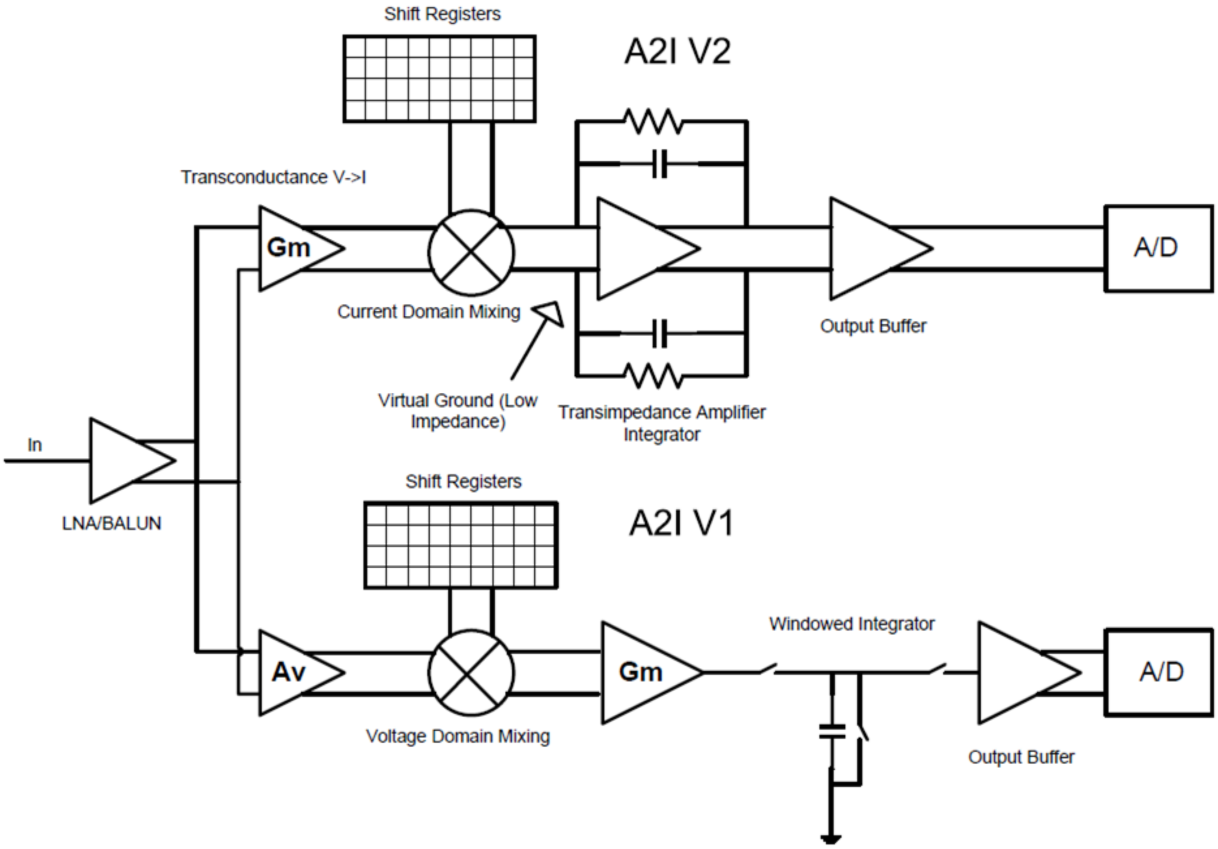


Figure 5.2: Block diagram depicting the difference between Vrx and Irx receiver design strategies. The Vrx and Irx receiver strategies are depicted on the top and bottom respectively.

5.3 RMPI Circuit Blocks

5.3.1 Input LNA and Signal Distribution

The input stage of the RMPI (LNA) was chosen under the constraints of achieving the required wideband 50 Ω input match as well as distribution of the LNA output to each of the 8 RD channels. The input LNA employs a modified version of the circuit topologies reported in [110,111]. The LNA is inductorless and also performs single-ended to differential conversion of the input by using a common-gate stage (non-inverting signal path) in parallel with common-source stage (inverting signal path) with replica biasing, a schematic is shown in Fig. 5.3. The gain of the LNA is 18 *dB* and has $f_{3dB} \approx 3\ GHz$. Fig. 5.4 shows an overlay of simulations for the designed (orange curve) and extracted (purple) AC gain of the LNA driving a load of 3 *pF* at each output. Simulations of large signal transients indicated $P_{1dBm} = -9.1\ dBm$ and IIP3 $\geq 0\ dBm$ along with IIP2 $\geq 10\ dBm$ across the entire 3 *GHz* input bandwidth. An overlay of simulated S_{11} curves of the designed (red curve) and extracted (blue) schematics are shown in Fig. 5.5. The extracted simulations indicate $S_{11} < -10\ dB$ up to $f_{\text{in}} < 7\ GHz$. A plot of the LNA noise figure (NF) (Fig. 5.6) reveals approximately 3 *dB* NF over the entire input bandwidth. The total power consumed by the

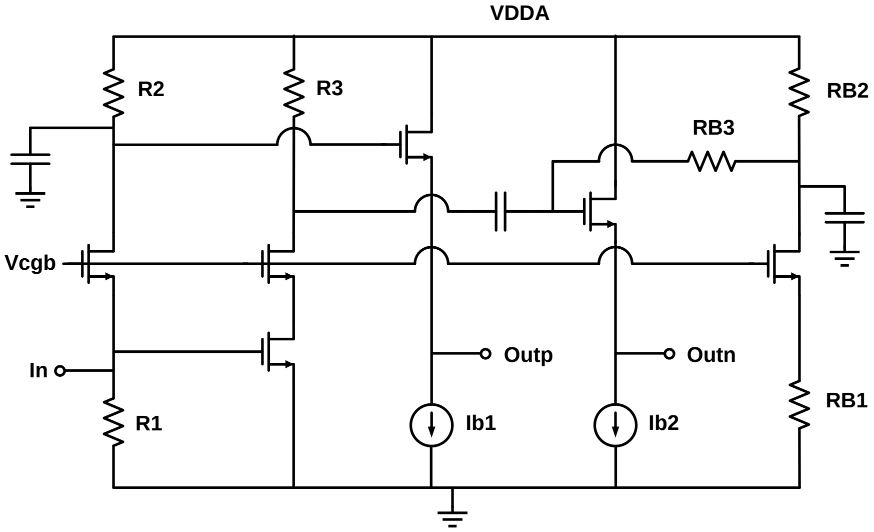


Figure 5.3: Schematic of the low-noise amplifier (LNA).

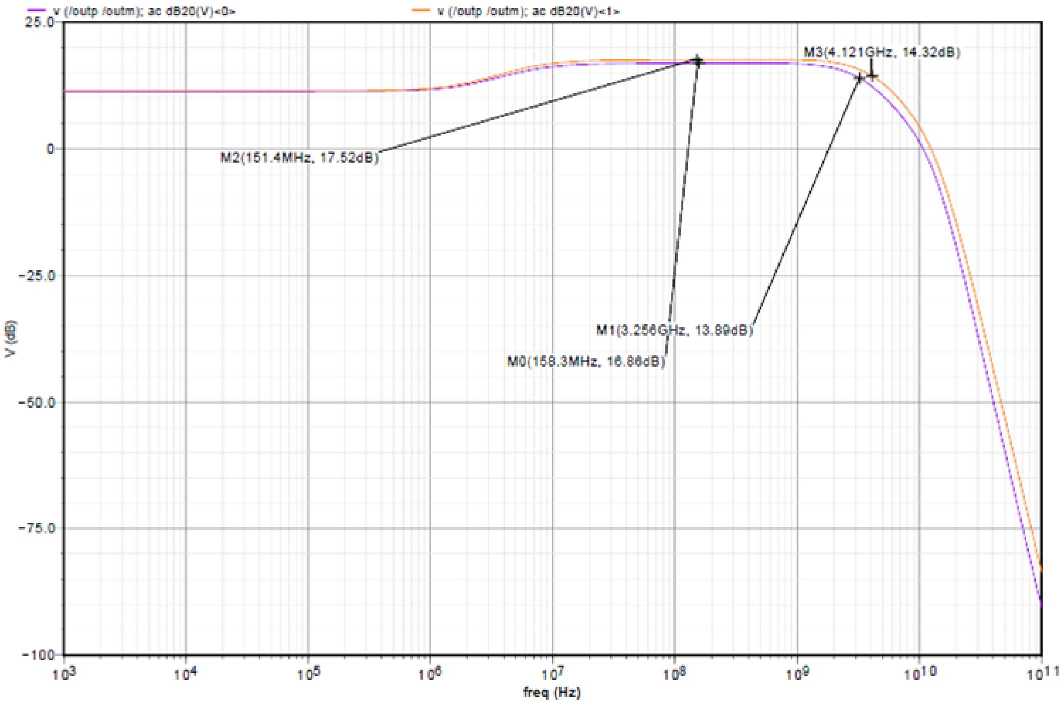


Figure 5.4: Low-noise amplifier AC gain.

LNA with 3 pF single-ended loads is 20 mW.

Distribution to 8 Channels

The maximum target bandwidth of the system was $f_{\max} = 2\ GHz$, at which the wavelength in silicon, $\lambda_{f_{\max}} \approx 15\ cm$, is far greater than the dimensions of the chip. We thus modeled the node connecting the output of the LNA to the inputs of each of the correlator channels with a simple RC wire model. The distribution node was surrounded with a ground shield and the source-follower outputs were designed to drive the capacitive load of the input distribution. As a worst case, the aggregate capacitive load (including that from the 8 transconductor inputs as well as the input distribution) was 3 pF.

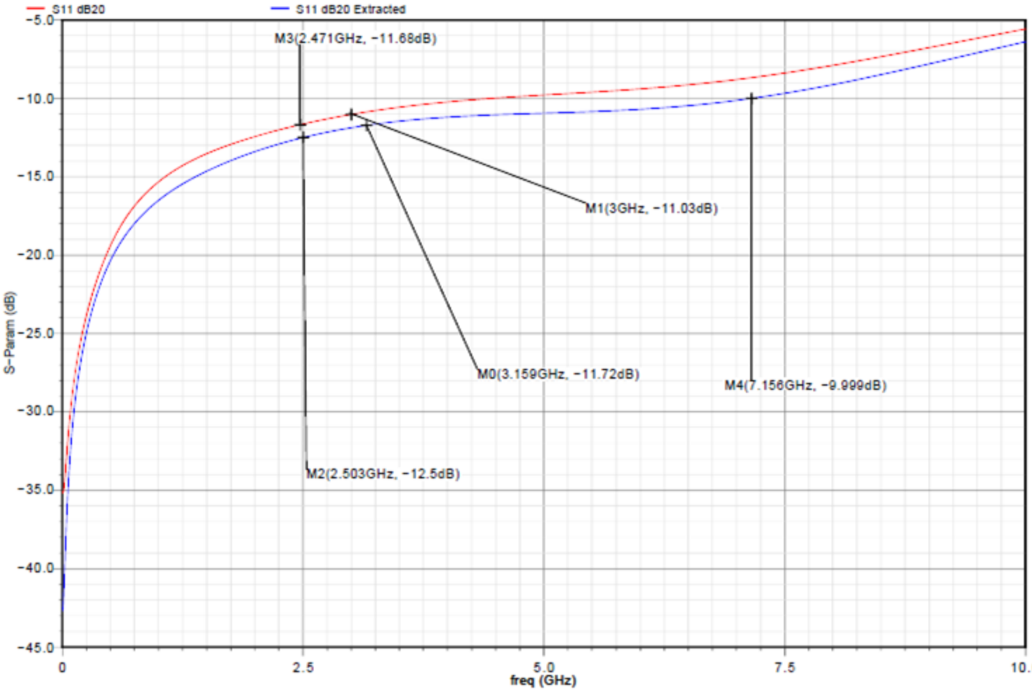


Figure 5.5: Simulation of LNA S_{11}.

5.3.2 Random Demodulator Channel Circuit Blocks

A circuit-block diagram of the RMPI is shown in Fig. 5.7. Each channel implements the specified random correlation operation and is implemented similarly to an Irx direct-conversion receiver. As described in Ch. 5.2, the large voltage-amplitude output of the LNA is converted by the channel input transconductor to a large-current/low-voltage signal. The transconductor is capacitively coupled to a passive mixer and pulls current through a series combination of the mixer switch impedance Z_{switch} and the input impedance of the following current-buffer stage $Z_{\text{in, buffer}}$, shown in Fig. 5.11. This combination is designed to present a low-impedance to the output of the transconductor, so that even at large signal swings the passive-mixer switches remain biased in the linear regime.

The LO port of the mixer is driven by a 128 *bit* digitally programmable shift register which is programmed with a desired PRBS. A small resistor is used to prevent interaction between the common-mode feedback circuits in the current-buffer stage and the integrator. The integrator drives an output buffer designed to drive the load presented by the combination

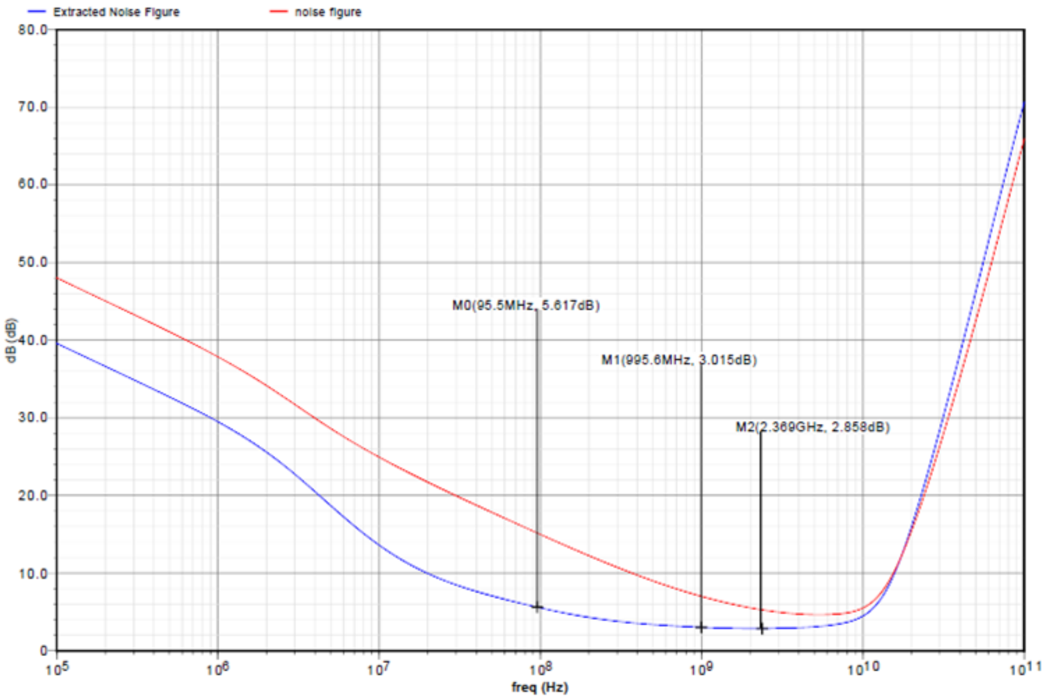


Figure 5.6: LNA noise figure simulation.

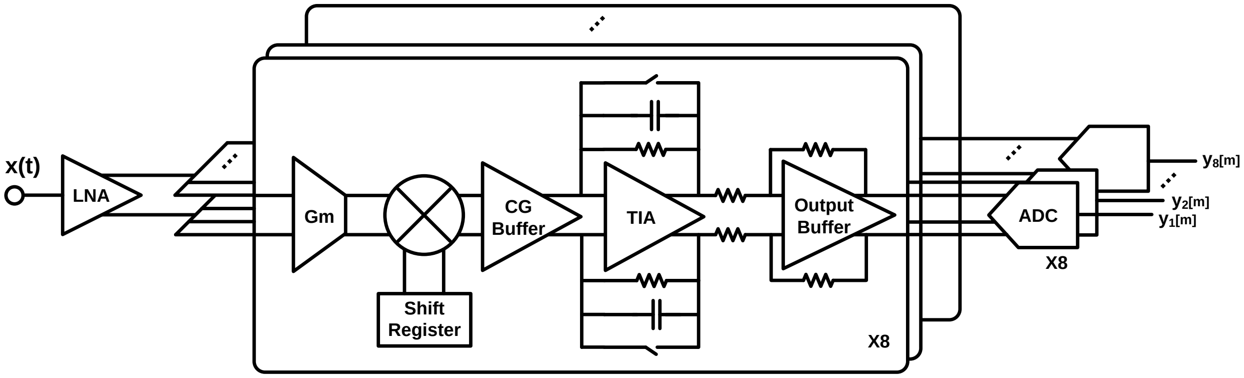


Figure 5.7: Simplified circuit block diagram of a single channel (RD) of the RMPI..

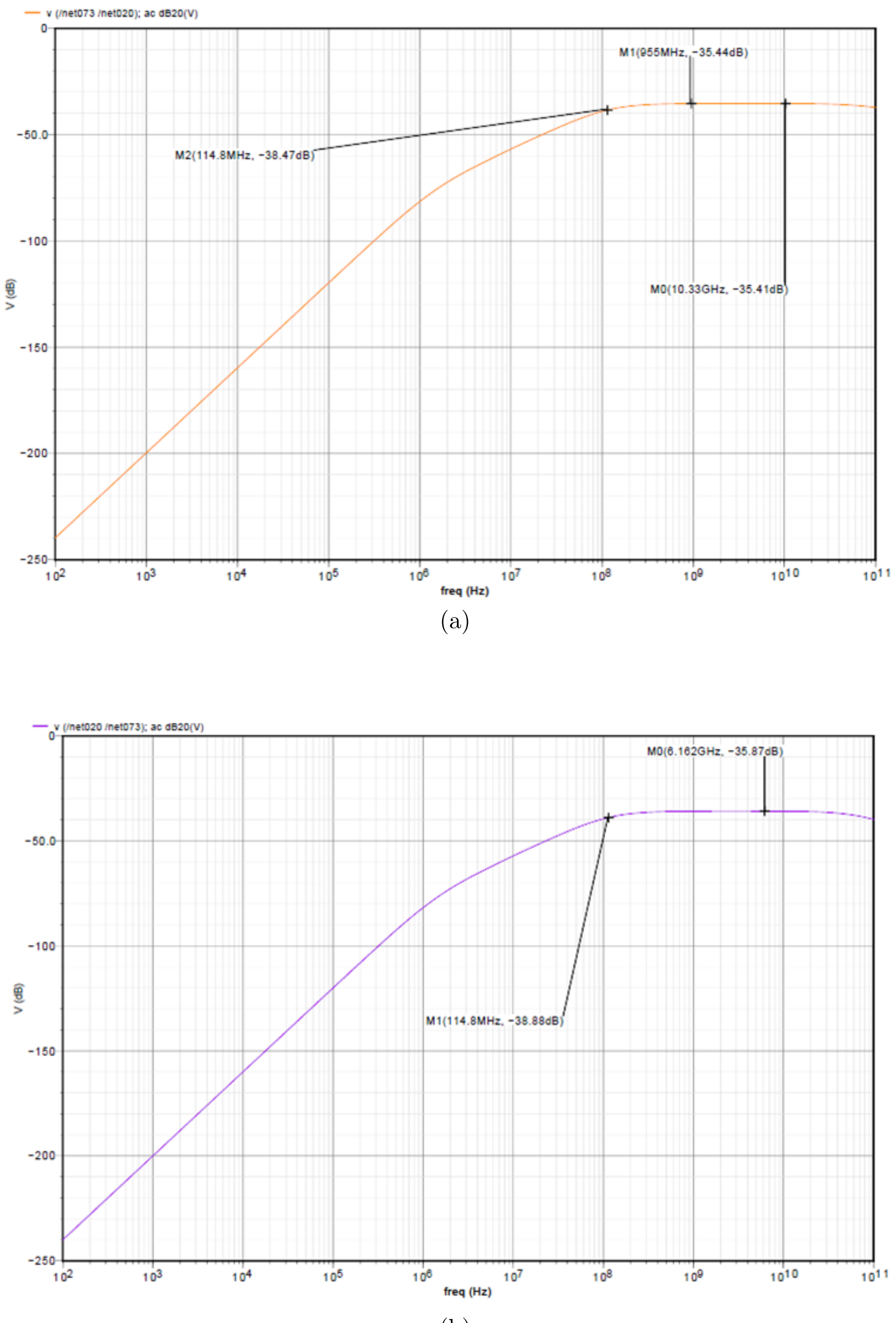
of the output pads, the bondwire from chip to PCB, the input capacitance of the off-chip digitizer, as well as the traces connecting bondwire pads on the PCB and the digitizer (up to 30 pF). Integration of the digitizers with the RMPI onto a single die would eliminate the power consumption as well as area requirements of the output-buffer. Details of each of the blocks as well as the global clock distribution and other testing features designed into the chip are given below.

Input Transconductor

The design of the input transconductor, passive mixer, and common-gate current buffer are tightly coupled.The RD channel input transconductor is realized as a pseudo-differential source-degenerated NMOS differential pair, shown in Fig. 5.9. The pseudo-differential configuration was chosen in favor of the fully differential configuration to avoid the loss in headroom from the IR drop of the source-degeneration resistors in a fully differential configuration. The $\mathrm{P_{1dBm}} = 8\ dBm$ if the outputs are unloaded. This number increases substantially when the transconductor is loaded with the low impedance, which we denote $Z_{\mathrm{mix,L}}$, seen looking into the mixer loaded by the input-impedance of the common-gate current buffer. An AC simulation of the designed and layout extracted transfer function is shown in Fig. 5.8a and Fig. 5.8b respectively. The bandwidth of the transconductor is in considerable excess of the bandwidth of the system at higher frequencies due to the low loading impedance ($\mathrm{Z_{gm,\ L}} = (\mathrm{Z_{mix,L}} \parallel \mathrm{R_1})$) presented by the parallel combination of the unloaded output impedance of the transconductor $\mathrm{R_1}$ and the net mixer impedance $\mathrm{Z_{mix,L}}$. The principal bandwidth limitation occurs due to the coupling used at both the input and output of the transconductor, which creates a lower f_{3dB} cutoff frequency around 114 MHz. This cutoff can be lowered by using more die area to realize larger coupling capacitors.

Mixer

Immediately following the RD channel input transconductor is the passive mixer shown in Fig. 5.10. Large coupling capacitors $\mathrm{C_1}$ and $\mathrm{C_2}$ set the lower frequency cutoff for signals that are input into the mixer. Beyond enabling the current-mode architecture, passive-mixers are convenient as their lack of DC current lowers the $1/f$ noise corner [112–117]. The switches consist of four transistors forming a standard double-balanced structure. The DC bias level at the gate of the switches is set so that they operate very close to the threshold of conduction; operation in this regime minimizes the on-resistance of the switches (R_{on}) which improves linearity, while minimizing the overlapping on-periods of the switches. This maximizes gain and minimizes noise [107, 113, 118, 119]. The bias scheme used to set the



(a)

(b)

Figure 5.8: Transconductor AC gain simulations: (a) designed schematic (b) extracted from layout.

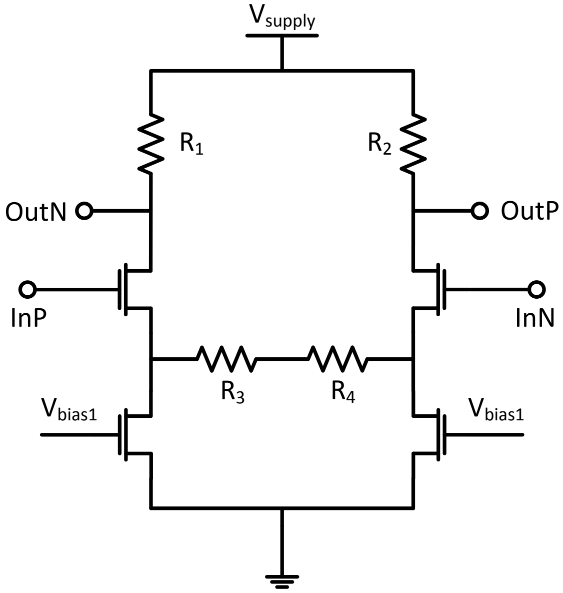


Figure 5.9: Schematic of the RD channel input transconductor.

switch gate voltages is similar to the scheme presented in [95, 107, 115]; a small current is run through a device matched to the switching transistors, which generates a replica V_{T} that serves as part of the reference in a servo biasing circuit. The common-mode voltage at the output of the mixer is set by the following common-gate current buffer stage. The V_{CM} is set to be lower than half of one V_{T} below V_{dig}. The switch devices in the mixer are sized large to minimize the $R_{\mathrm{on}} \approx 2\ \Omega$ of the switches and make their R_{on} variation (ΔR_{on}) minimal

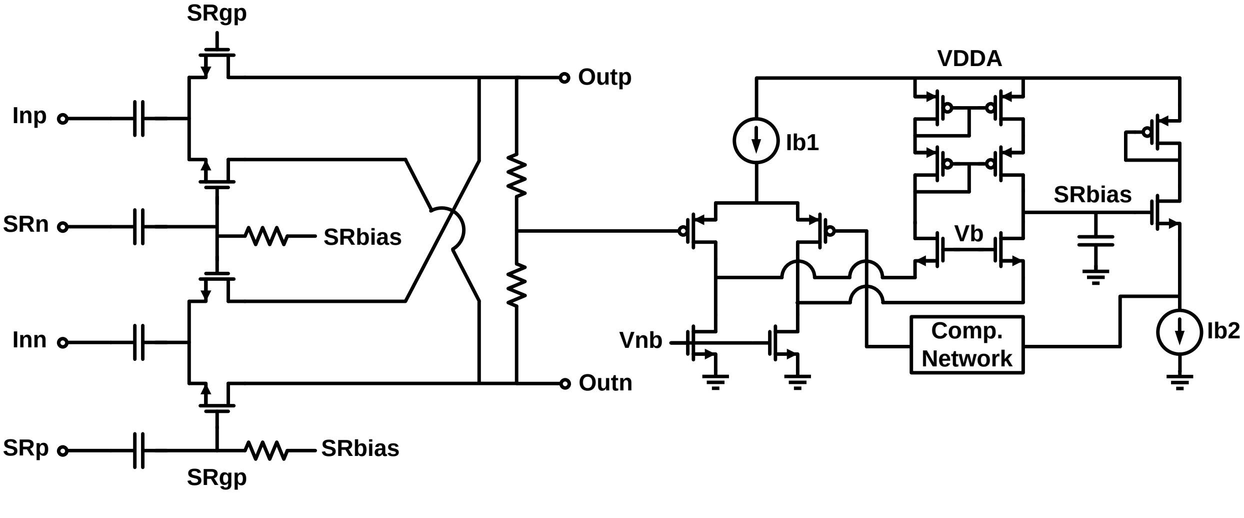


Figure 5.10: Schematic of passive mixer.

compared to $R_{\text{on}} + \mathrm{Z}_{in,CG}$. The upper limits on the size of the switches is constrained by the power consumption of the LO port drivers needed to maintain large LO port drive and short rise/fall times. In addition, another significant constraint on the maximum switch device size is the desire to maintain a flat mixing transfer function to avoid attenuating the highest harmonics of the PRBS. An additional constraint that exists to the wideband spectral occupancy of the PRBS is the high-pass filter created by the combination of the resistors and coupling capacitors used at the LO port. The low-frequency cutoff must be set such that the lower-order PRBS harmonics are not attenuated, this cutoff along with the lower-frequency cutoffs imposed by the use of AC coupling capacitors in the primary signal path determine the lower limit of the receiver bandwidth.

Common-Gate Current Buffer

The purpose of the common-gate current buffer is two-fold: the first is to reduce the input-referred noise contribution of the transimpedance amplifier (TIA)-based integrator [95, 107], and the second is to allow more flexibility in choosing the parallel RC load used in the feedback loop of the operational transconductance amplifier (OTA) that is used to realize the TIA-based integrator. A schematic of the modified current-gate buffer [95, 117] is shown in Fig. 5.11. It is important to note that the input bias to the buffer is set in an open-loop fashion by the same mechanism which generates the mixer switch gate biases. In addition, the output common-mode of the buffer is set by an independent loop shown on the right hand side of Fig. 5.11.

In a standard receiver employing current-mode passive mixers, the passive mixer is loaded with either a low input-impedance common-gate stage or an op-amp with RC feedback [29, 95, 120, 121]. One well-known issue with this design approach is the relative noise contribution of the TIA, which is exacerbated by the relatively low impedance seen looking into the outputs of the mixer driven by the transconductor [107]. In addition, there are competing constraints in terms of the RC load used in the feedback loop of the integrator, as well as the desire to minimize the impedance seen by the output of the passive mixer in order to

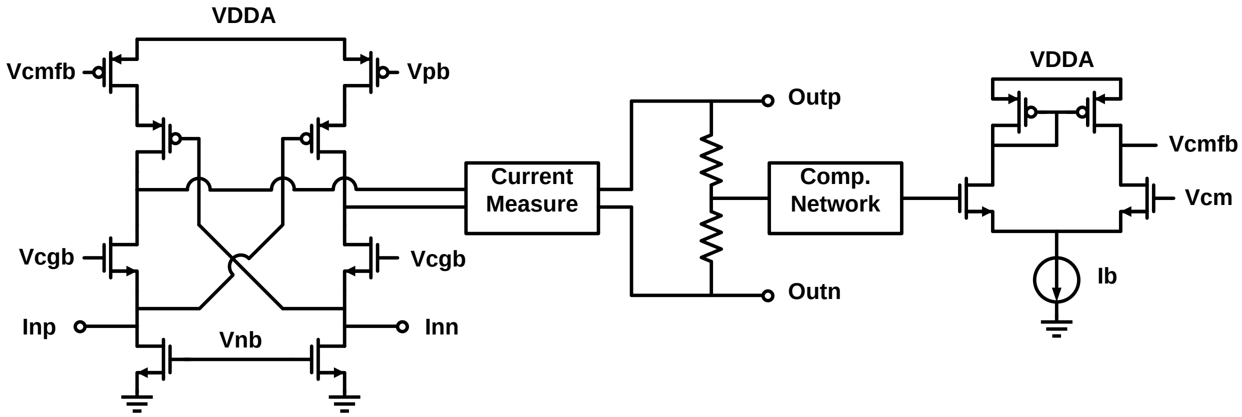


Figure 5.11: Schematic of the modified common-gate current buffer.

maximize current-gain. The buffer mitigates these constraints by serving as an impedance transformation stage that decouples the input impedance of the integrator and the impedance seen by the output impedance of the mixer.

The bandwidth requirements of this buffer are set by two requirements. The first is that the bandwidth of the buffer should be sufficiently larger than the baseband bandwidth over which the integrator must mimic the action of an ideal integrator. Thus, by association, the bandwidth requirement is also directly tied to both the length of the PRBS sequence chosen as well as the duration of integration T_{int}. These relationships are discussed in detail in Ch. 3. The second consideration, which suggests that the bandwidth should be kept at the minimum, comes from both the difficulty in design and power consumption associated with the necessary common-mode feedback loop which must achieve high loop gain in excess of the buffer. A large capacitor is placed at the input of the buffer to filter out the RF frequencies output by the mixing operation is placed across the inputs (no shown in Fig. 5.11). The buffer achieves a current gain of 12 *dB*.

Integrator

The integrators were implemented using standard class-A operational amplifiers (OTA) in a standard RC-integrator configuration [122]. Programmable reset switches were placed in parallel with the integration capacitors. The RC-integrator configuration is depicted in

Fig. 5.7 and a schematic of the class-A op-amp used is shown in Fig. 5.12.

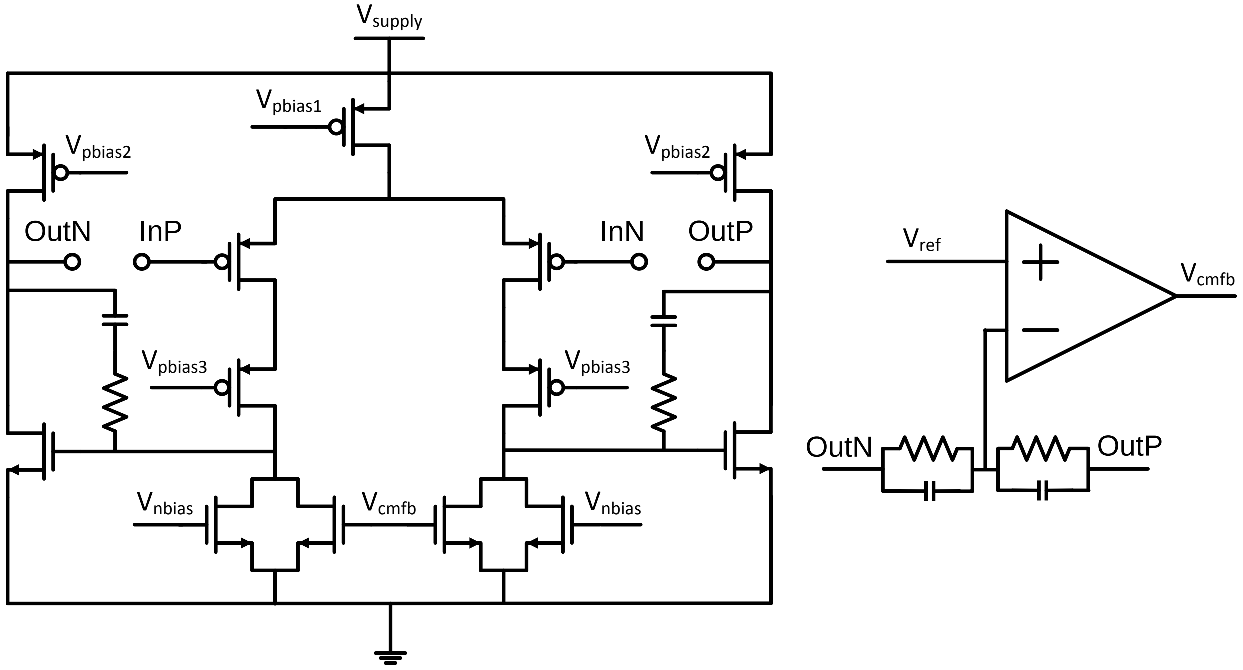
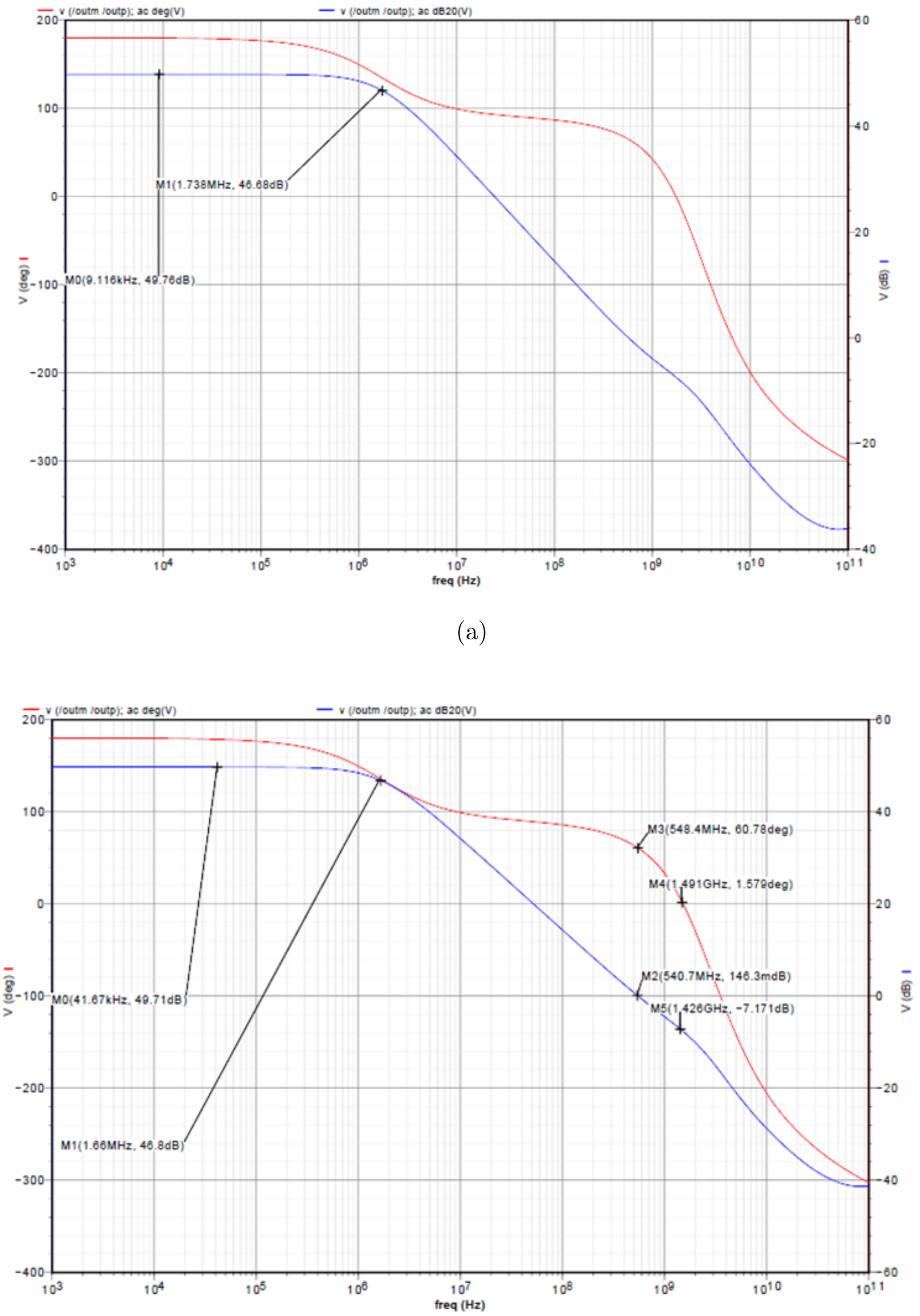


Figure 5.12: Schematic of a class-A op-amp which is used in both the TIA-integrator and output buffer.

An OTA TIA type integrator, as opposed to a single passive capacitor, was chosen for its linearity as well as its greater ability to define its input impedance (via an impedance placed in the feedback loop in combination with the high loop gain of the OTA). The AC transfer characteristics of the integrator OTA are shown in Figures 5.13a and 5.13b. For testing purposes, the reset switches were made programmable via a 128 *bit* programmable shift register and designed with extremely high bandwidth to enable fast reset times. The OTA used in the integrator was designed with a unity-gain bandwidth of approximately 1 *GHz*. The common-mode feedback loop was also made to have in excess of 20 *dB* of loop gain across over half the unity-gain bandwidth $f_{\text{ug,int}}$. Extensive corner simulations were performed to ensure all feedback had sufficient gain and phase-margin ($> 75°$). The parallel combination of R and C placed in the feedback loop allowed us to set the low-order pole while also setting the DC gain and traded off with the input-referred noise contribution.



(a)

(b)

Figure 5.13: Integrator AC simulations

Output Buffer

The output buffer was also realized through the use of an OTA. A worst case output capacitive load of 30 pF was calculated during design (from simulations of the trace lengths of the test board, the wirebond pads, and the input capacitance of the off-chip digitizers). The output buffer used a standard resistive feedback configuration to set unity-gain operation as shown in Fig. 5.7. The common-mode feedback loop used in the output buffer OTA had requirements similar to those of the integrator OTA.

Offset Compensation

As the downconverted frequency spectrum in a direct-conversion receiver extends to DC, extraneous offset voltages (especially prior to large gain stages) can saturate the output, or reduce the dynamic range of the signal chain by altering the DC operating point leading to corruption of the signal [20]. In order to address this issue, a 5 *bit* (-200-200 mV) offset compensation scheme was incorporated. Monte Carlo simulations of the entire RF/baseband analog signal path suggested that the standard deviation of the output DC offset was $\sigma_{\text{offset}} \approx 80$ mV, hence the offset compensation scheme could correct for about $5\sigma_{\text{offset}}$ of mismatch. The offset-compensation scheme injected currents into the virtual ground nodes of the TIA integrator and was set by a programmable shift-register. The functionality of this feature was validated in the fabricated hardware.

Shift Registers

PRBS generators were realized using 128 *bit* shift registers composed of positive-edge triggered static D-type flipflops. A functional block diagram of the flipflop is shown in Fig. 5.14. An extracted schematic from the layout of the shift register was simulated with both the Cadence UltraSim and SpectreRF tools. The extracted simulation predicted a maximum toggling rate of 7 *GHz*, well above what was required for the target bandwidth. Each of the shift registers has a clock input port which feeds into a symmetric clock distribution tree internal to the shift register designed to accommodate not only high frequency operation

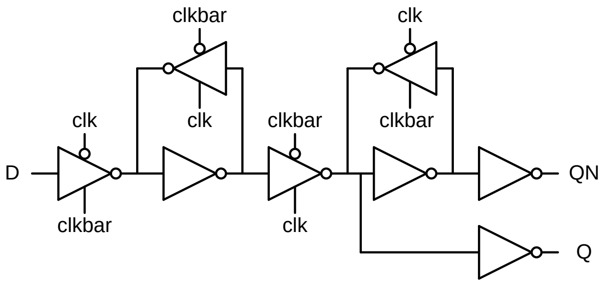


Figure 5.14: Simplified schematic of a standard D-flipflop used to realize the programmable shift-registers.

but also prevent race conditions. Delays from the input clock node to flip-flops at extremal points at the extremes of the layout were calculated via the Elmore delay method [123].

5.3.3 Global Clock Distribution

Channel-to-channel timing accuracy and minimum duty-cycle distortion is crucial in producing compressed-samples that allow high-fidelity reconstruction. Empirical simulations reveal that time-domain signal reconstruction is sensitive to the σ_{jitter} of the Nyquist-rate clock distributed to each channel. Detailed numerical simulations of the effect of jitter is presented in Ch. 4. Simulations specific to this RMPI suggested that achieving > 60 *dB* dynamic range requires $\sigma_{\text{jitter}} \leq 0.5$ *ps*.

Architecture Description

A functional block diagram of the CMOS RMPI global (Nyquist rate) clock distribution is shown in Fig. 5.15. The clock distribution employs a symmetric binary tree topology that distributes an input to the 4 pairs of channels(8 in total) as well as the ninth test channel. The clock distribution is also overlaid over a die photo of the CMOS RMPI shown in Fig. 5.20. The current-mode distribution consists of 3 basic sub-blocks: an open-drain driver, a 100 Ω differential transmission-line, and a 100 Ω differential input TIA. The composition of these three blocks is used to realize repeaters. Repeaters are necessary as the transmission-lines (T-lines) are lossy and would attenuate the clock without them (the loss of the transmission line was simulated to be 1 *dB/mm*. In addition, whenever the clock comes to a node where

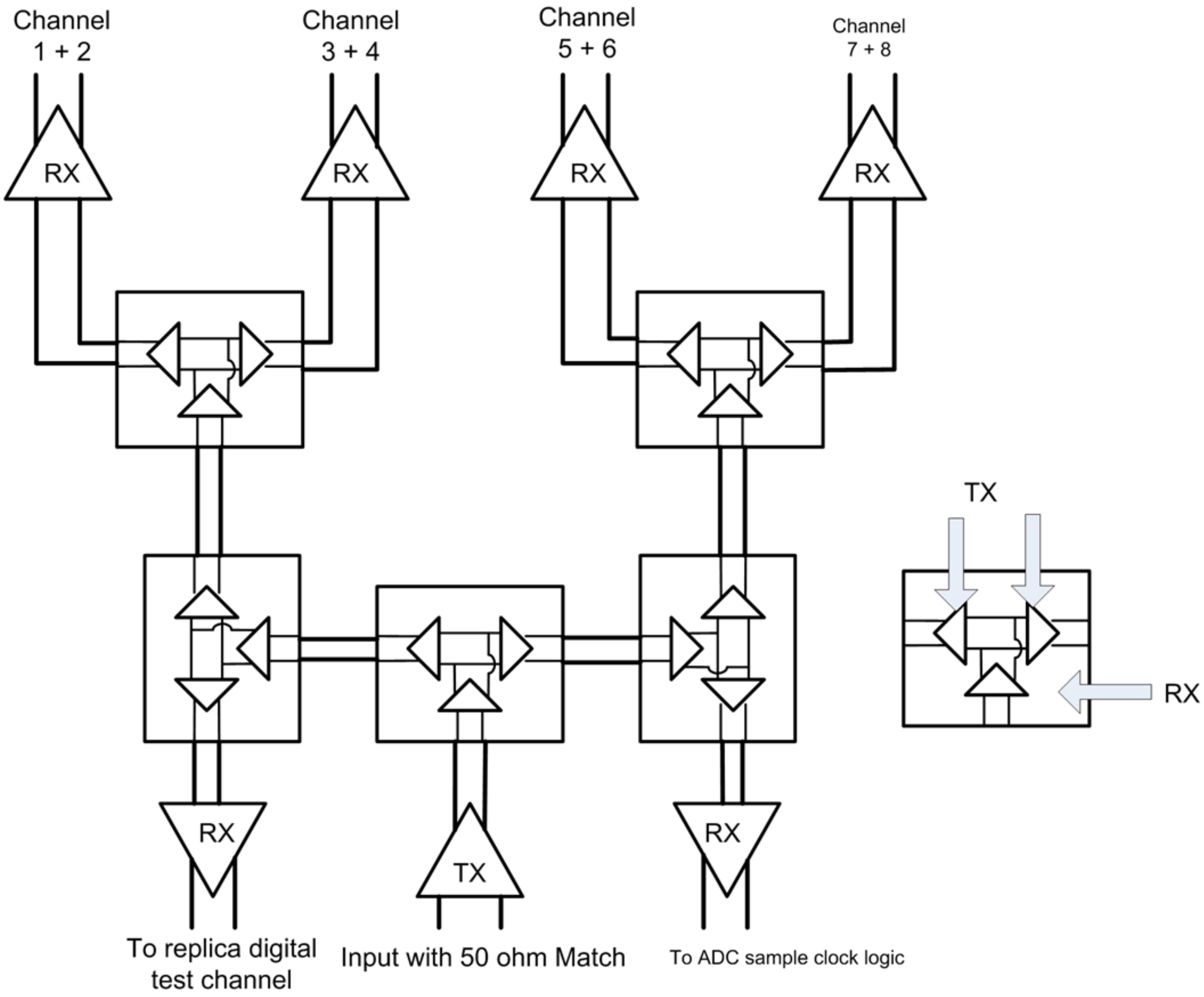
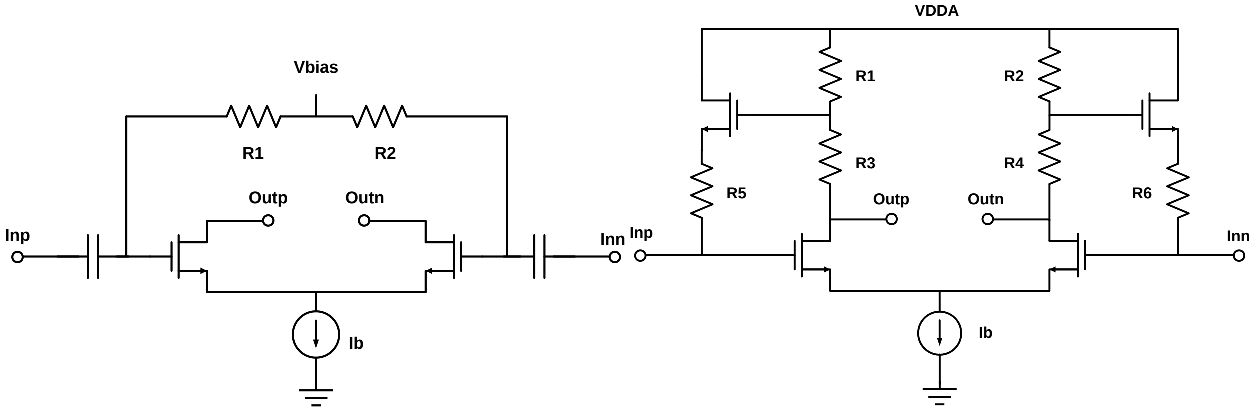


Figure 5.15: Block diagram of global clock distribution

the clock signal is split, the repeaters serve to isolate the different T-line segments from one another removing the need to match them. The input to the overall clock-distribution is an open-drain driver whose inputs have been terminated with 50 Ω resistors to provide an input match to the external signal source which feeds in the Nyquist-rate clock.

Current-mode Clock Distribution

A schematic of the open-drain driver which which converts the large-amplitude clock signal into a large amplitude current signal is shown in Fig. 5.16b. The clock signal bearing current is then pulled through the 100 Ω differential transmission line whose physical dimensions and



(a) Clk TX.

(b) Clk RX.

Figure 5.16: Simplified schematics of the clock transmitter and receiver blocks used in the global clock distribution. (a) Clock Transmitter (b) Clock Receiver.

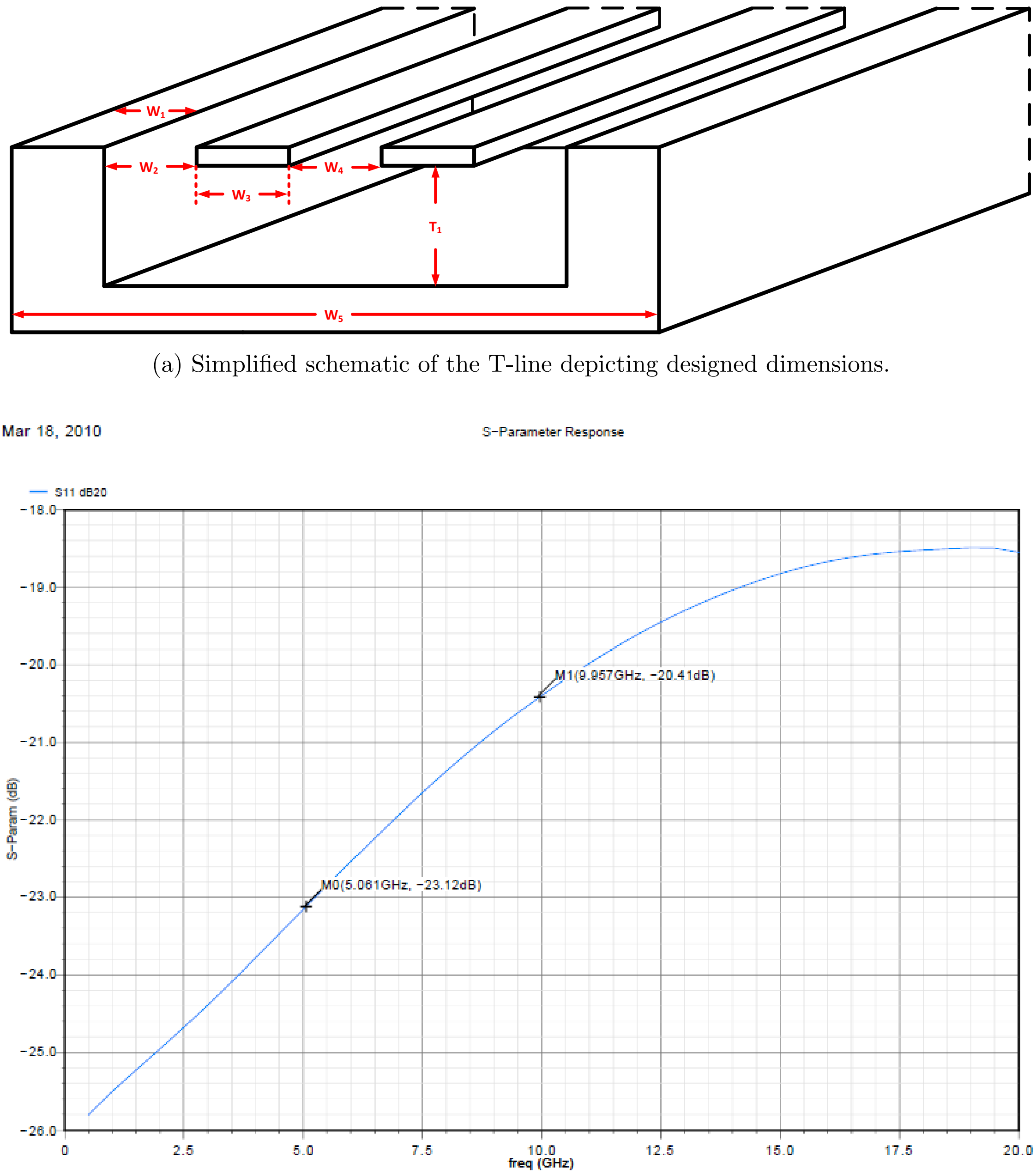
S_{11} characteristics are shown in Fig. 5.17a and Fig. 5.17b respectively.

The AC simulation was performed using and S-parameter file, generated by simulation of the structure in the Ansoft HFSS environment. The transmission line is terminated by the input-impedance of the TIA receiver stage shown in Fig. 5.16b. The purpose of the TIA is to allow driving of high-impedance/capacitive nodes. A repeater node, shown in the bottom-right corner of Fig. 5.15, consists of an input TIA receiver which drives 2 transmitter blocks. The total path length from the port named **Clk Diff Input** (shown in the bottom-right corner of Fig. 5.20) to all terminal nodes of the analog portion of the clock distribution is $\approx 4\ mm$.

Local Channel Clock Distribution

The voltage across the terminal nodes of the analog clock distribution is converted to a digital clock via the CML-to-CMOS converter shown in the right-half of Fig. 5.17. In between the TIA receiver block and the CML-to-CMOS converter is a PMOS differential pair, which acts as an analog level shifter between the two stages.

The output of the CML-to-CMOS block is then fed into a chain of weakly coupled inverters, shown in Fig. 5.18, to correct the duty cycles of the CMOS clocks [124]. The output of the duty-cycle distortion (DCD) correcting circuit is of critical importance. Any



(a) Simplified schematic of the T-line depicting designed dimensions.

(b) AC simulation of transmission line S_{11}.

timing mismatch between the RD channels leads to the introduction of a linear-phase filter that must be accounted for to obtain high-fidelity reconstructions.

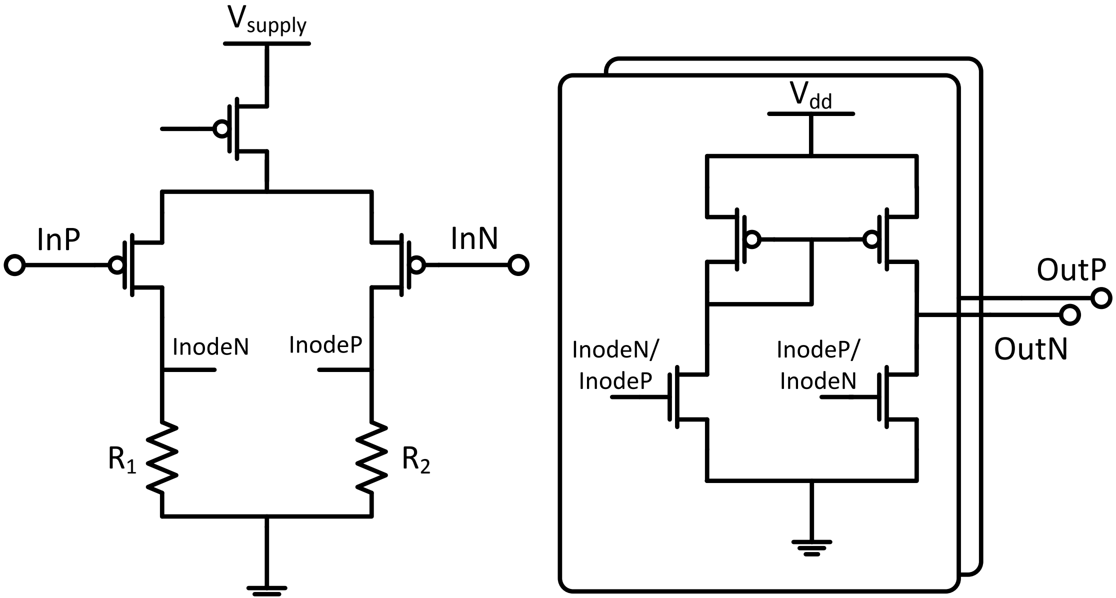


Figure 5.17: Schematic of the CML-to-CMOS Converter.

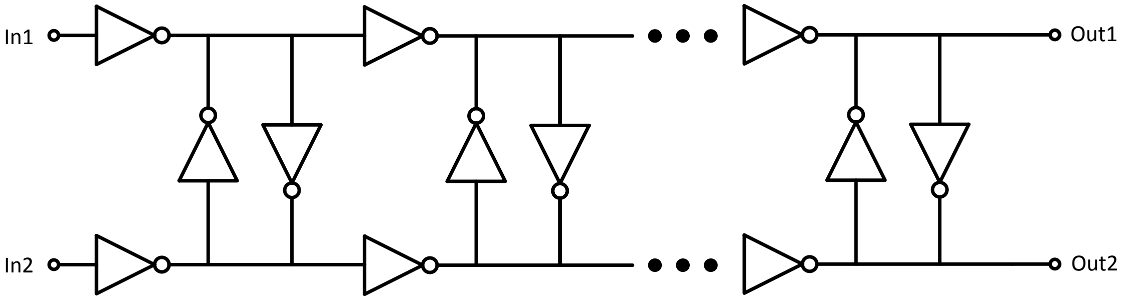
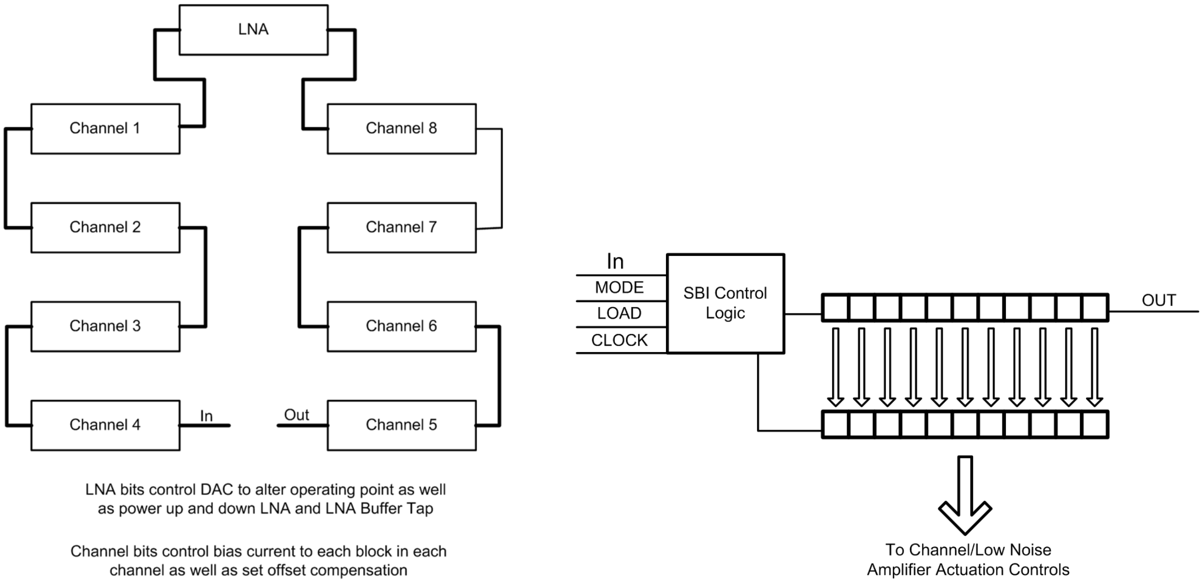


Figure 5.18: Schematic of the DCD array.

5.3.4 Supporting Circuits

The CMOS RMPI incorporated many supporting circuits for the purposes of facilitating testing. The major additions included a ninth random demodulator channel which was clocked off the same clock distribution network and was an exact duplicate of the RF/analog signal path without the LNA. Several of the internal critical nodes such as the output of the mixer were also connected to pads to allow examination of the signal being fed into the baseband filters as well as facilitate direct measurement of the filters if desired.

A replica of several critical nodes in the Nyquist-rate clock distribution path were connected to 50 Ω drivers to allow direct measurement of the clock distribution network. Each channel included a programmable clock divider which could be set to /1, /2 /4 /8 /16, and /32 modes to maximize testing flexibility. Each of the blocks in the RF/analog signal path including the LNA could be independently powered up and down The chip was designed with


LNA
Channel 1
Channel 2
Channel 3
Channel 4
In
Out
Channel 5
Channel 6
Channel 7
Channel 8
LNA bits control DAC to alter operating point as well as power up and down LNA and LNA Buffer Tap
Channel bits control bias current to each block in each channel as well as set offset compensation
In
MODE
LOAD
CLOCK
SBI Control Logic
OUT
To Channel/Low Noise Amplifier Actuation Controls

(a) Block diagram of the SBI

(b) SBI unit cell

Figure 5.19: Architecture of the serial bus interface used to enable/disable functionality as well as calibrate offsets/mismatches.

two independent sets of shift registers: the first was a chain of of all shift registers used to program the PRBS as well as the reset-switch across the integration capacitor (shown in Fig. 5.19a). When in the programming mode, all of the shift registers were configured in a linear daisy-chain configuration. Once all the values were loaded, the shift registers were reconfigured in a circular array. The second set of shift-registers, shown in Fig. 5.19b holds bit-values which control the offset-compensation circuits as well as the enabling/disabling of power to each block in the analog signal path. The chip also implements an industry-standard prescribed 2 kV HBM ESD protection strategy.

5.4 Power Consumption Breakdown and Die Photo

Table 5.1 summarizes the measured performance of the CMOS RMPI. Table 5.2gives a breakdown by block of the total power consumption of the chip. A die photo is shown in

Fig. 5.20.

RMPI Performance	
Technology	IBM 90 nm CMOS9SF
Die Area	8.85 mm^2
Supply Voltage	1.5 V Analog 1.2 V Digital 2.5 V Digital I/O
PRBS Clk Freq	4 GHz
Gain	41 dB
Noise Figure	7 dB
S_{11}	< -15 dB, 10 MHz-5 GHz
P_{1dBm} (input-referred)	-25 dBm
P_{1dBm} (CS mode)	-10 dBm
Rx Chain BW	2.7 GHz
Bandwidth	100 MHz–2 GHz
Dynamic Range (CS Recover)	54 dB
"Undersampling"	12.5×
$\sigma_{\text{system jitter}}$ (rms)	< 300 fs
Channels (units)	8
Power Consumption (w/o output buffers)	506.4 mW

Table 5.1: Summary of measured performance of the CMOS RMPI.

Power Consumption of 8-Channel CMOS RMPI		
Circuit Block	$\mathrm{P_{diss}}$ (mW)/Channel	$\mathrm{P_{diss}}$ (mW) Total
LNA/Balun	—	20
Transconductor	18	144
CG TIA	3	24
Integrator	12	96
Mixer	0.3	2.4
Shift Register	20	160
Output Buffer	30	240
Clk Distribution	—	60
Total Power Consumption		746.4 mW
Total Power Consumption Excluding Output Buffers		506.4 mW

Table 5.2: Breakdown of power consumption of the different blocks of the 8-channel CMOS RMPI.

.

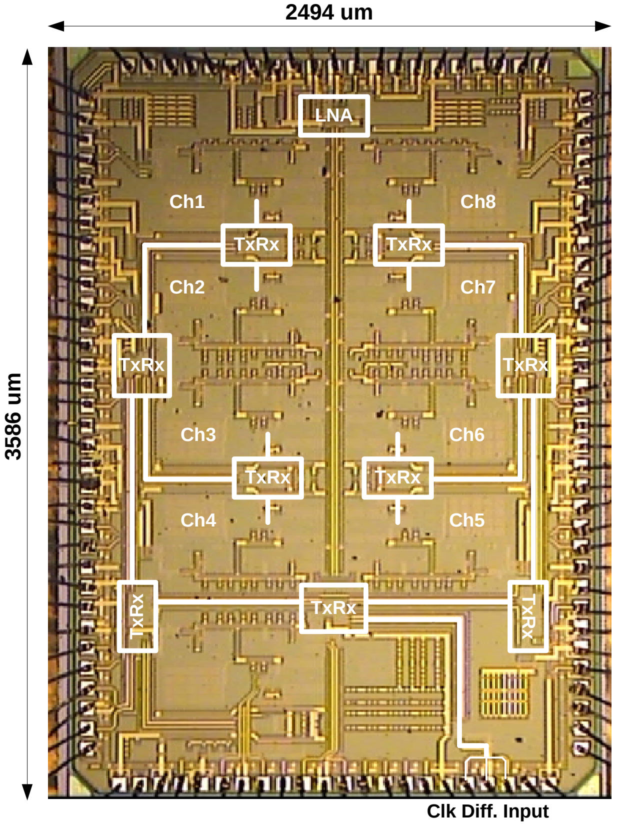


Figure 5.20: Die photo of the implemented 8-channel RMPI

Chapter 6

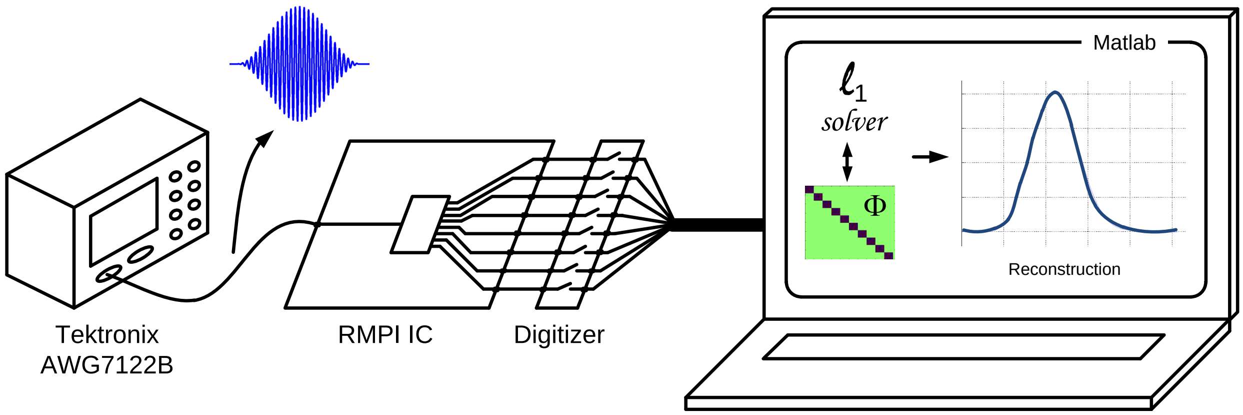
Measurement Results

In this chapter we present signal recoveries obtained from *physical* measurements generated by the 8-channel CMOS RMPI chip. We begin by describing the test setup used to generate the measurements including how the test stimuli were generated. We then conclude by presenting several representative cases of full time-domain signal recoveries. The obtained results establish the physical feasibility of the RMPI and CS-based receivers in general. For all presented measurement results, the aggregate system digitization rate was 12.5× lower than the f_{nyq} of the EIBW. For details and further specifics on signal recovery methods used, see [55, 69, 86].

6.1 Measurement Setup

A diagram of the test setup used to validate the RMPI is shown in Fig. 6.1. The Nyquist-rate clock that toggles the shift registers was provided by an external frequency source. The RMPI IC was mounted on a custom PCB which served as a means to interface the chip to an FPGA and numerous test inputs/outputs that enabled observation of critical nodes. The FPGA was programmed with a custom CPU that was used to program the on-chip shift-registers (which allows powering specific blocks on/off, loading desired PRBS sequences, adjusting offset compensation, and routing test outputs etc..) as well as handle the logistics of acquiring, storing, and exporting the digitized output samples to a PC for analysis.

At this juncture, we point out that the RMPI is a 'universal' encoder which outputs



Φ represents the RMPI System Transfer Function

Figure 6.1: Diagram of the test setup used to collect RMPI measurements.

data that can be used to recover signals that are sparse in *any* fixed domain. However, for the purposes of concreteness, the testing performed specialized in trains of short radar pulses embedded in an ultra-wideband. Test stimuli fed into the RMPI consisted of both pure-tones and pulses; the carrier frequencies f_{carrier} (which we use interchangeably with f_{in}) were randomly drawn from an interval spanning 85 *MHz*–to–2 *GHz* and the initial phases ϕ_{init} were likewise similarly random. In the case of pulse stimuli, the tested pulse durations T_{pulse} ranged from 25 ns –to–2 μs . For both types of signals, amplitudes (A) tested ranged from 400 μV_{pp} to 200 $m V_{pp}$. Pulse stimuli were synthesized via an 8-*bit* 12 *Gsps* arbitrary waveform generator. For the synthesis of pure-tone inputs, both the AWG and a 10 *MHz*–to–40 *GHz* signal generator were used. The signal generator was occasionally used as it allowed superior control over ϕ_{init} and had greater output amplitude range than the AWG.

Once the digitized CS-samples were loaded onto a PC, the MATLAB computational environment was used to perform signal recovery of the time-domain (discrete Nyquist-grid) representation, which we denote $\mathbf{x}$, of the original input $x(t)$-we denote the recovered signal $\hat{\mathbf{x}}$. The signal recovery was performed via a numerical optimization procedure [55] that employed a modified form of basis-pursuit with reweighting [86]; for further details, see [55]. In the case of pulses, the quality of recovery (the MSE of the recovered baseband window and the original baseband window) of $\hat{\mathbf{x}}$ was evaluated by estimating the carrier frequencies present in $\mathbf{x}$ from $\hat{\mathbf{x}}$ and demodulating $\hat{\mathbf{x}}$ to obtain an estimate of the original ideal baseband pulse window(s). In the case of pure-tones, the MSE of the recovery was directly compared to the known ideal input. For pulses, the signals programmed into AWG served as the ideal reference for comparison. In the case of tones, knowledge and control of ϕ_{init} enabled synthesis of an appropriate reference.

6.2 Case Studies

In this section we present several case studies representative of the overall performance of the chip. The first case study examines reconstructions of single pulses modulated by carrier frequencies lying in the receiver bandwidth. Other measurement results presented include recovery of two pulses with distinct carrier frequencies overlapping in the time-domain, a low-amplitude tone recovery, recovery of short time-duration (narrow) pulses comparable to a single RMPI integration period ($T_{\text{pulse}} \approx T_{\text{int}}$), and the error in frequency estimation for pulses randomly chosen from within the input bandwidth. It should be noted that for the results presented, there were no changes made in operating conditions of the circuit (such as the tuning of any clocks or alterations in programmed PRBS).

6.2.1 Single Pulse Reconstructions

Fig. 6.2 shows reconstructions of two 400 ns pulses with carrier frequencies of about 87 *MHz* and 1947 *MHz*. This demonstrates the EIBW of the RMPI. There was no reconfiguration, change in operating conditions, e.g., tuning of the Nyquist-rate clock fed into the chip, in order to recover these signals. The RMPI is capable of capturing signals from the entirety of the input bandwidth and not just the IF bandwidth of the baseband filter with downconversion at a fixed LO frequency.

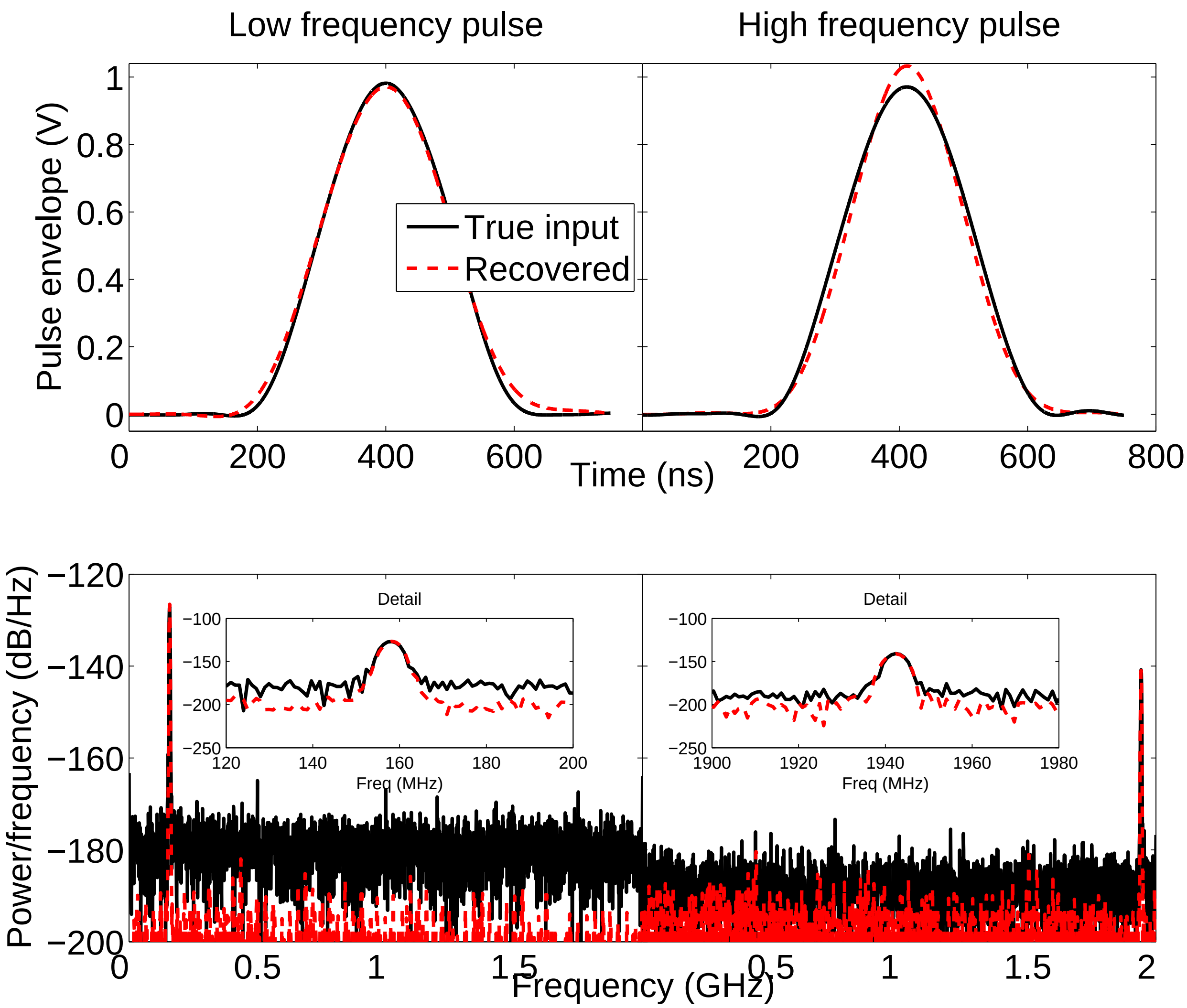


Figure 6.2: Recoveries of pulses at both low and high carrier frequencies. The upper two boxes show an overlay of the time domains of the programmed input signal and the recovered baseband window. The lower two boxes show an overlay of the frequency domains of the signals, the normalized MSE < 0.1 for all recoveries. The amplitude of the signal has been normalized in the plots; the input peak-to-peak amplitude was 1 mV.

6.2.2 Frequency Error Versus Carrier Frequency

Fig. 6.3 shows the absolute error in the estimation of f_{carrier} produced from $\hat{\mathbf{x}}$ while reconstructing the baseband pulse envelope; all obtained reconstructions had a normalized MSE < 0.1. The median f_{carrier} estimation error was < 69 kHz, demonstrating the RMPI's potential in spectral-sensing applications.

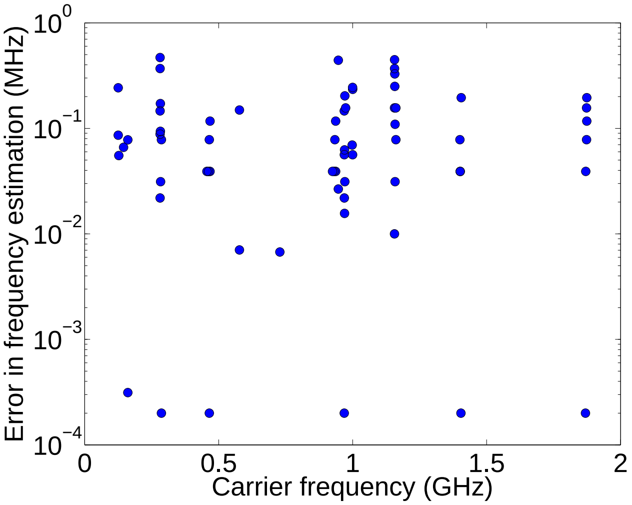


Figure 6.3: Error in estimation of carrier frequency for reconstructions lying in the input bandwidth.

The points shown on the graph are of varying amplitudes, pulse-widths and input phases generated as described in Ch. 6.1. This suggests that cognitive radio applications could benefit from the use of RMPI/CS ideas to quickly generate an estimate of the spectral support of the bandwidth of interest.

6.2.3 Low-Amplitude Tone Reconstruction

The single-tone dynamic range was tested by sending in and reconstructing low-amplitude tones. Fig. 6.4 shows the reconstruction of a tone of 400 μV_{pp}, which is 54 dB below the full-scale input (if we conservatively define the full-scale (FS) input as being set by the P_{1dBm} compression point of the receiver) of the receiver.

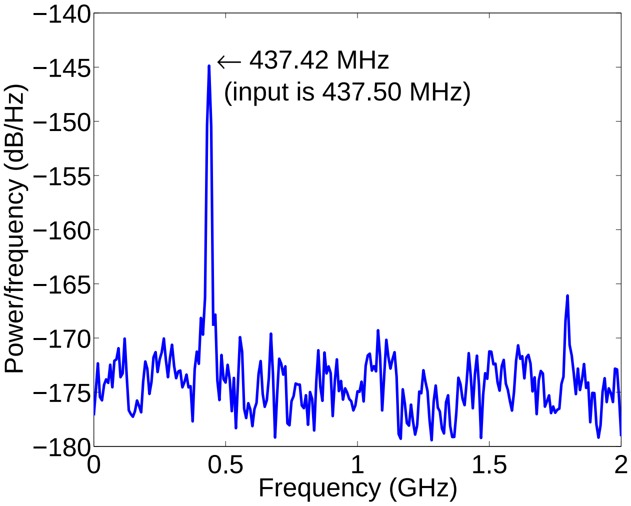
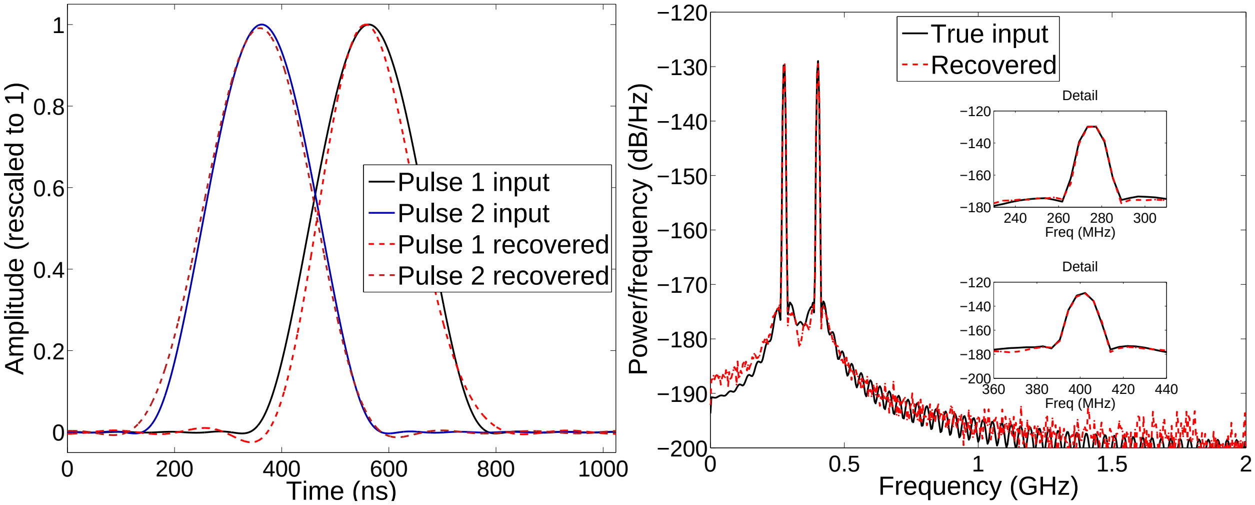


Figure 6.4: Reconstruction of a low-amplitude tone.

We point out that the P_{1dBm} of the receiver is not necessarily the upper-bound of the reconstruction capabilities of the receiver. Due to the variation in concentration of power at different harmonics for different frequencies, larger amplitude signals can still be recovered. Channels which produce either minimal or "railed" outputs provide information which can be used in conjunction from channels which produced detectable "unrailed" outputs to generate accurate signal reconstructions. In more conventional receivers, the upper-bound on the input power of the signal is defined by the largest input amplitude from which the output signal-to-noise and distortion (SNDR) guarantees better than a certain probability of error in estimating the desired information, e.g., bit-error-rate (BER). Other considerations also involve the desire to prevent the generation of excessive amounts of interfering signal power (or degradation of desired information) in channels of interest from nonlinear effects such as intermodulation distortion (IMD), cross modulation, and gain desensitization due to large blockers in nearby channels.

6.2.4 Pulse-on-Pulse Reconstruction

A signal that is difficult for even conventional radar receivers to handle is that of two pulses overlapping in time—pulse-on-pulse—with distinct f_{carrier}. Figures 6.5a and 6.5b show the time and frequency domain reconstruction, respectively, of two pulses overlapping in the time-domain. Both pulses are 400 ns in width and have a center-to-center time separation of 200 ns . The normalized MSE of the pulse envelope reconstruction is well below the threshold of 0.1.



(a) Recovery in the time domain.

(b) Recovery in the frequency domain.

Figure 6.5: Pulse-on-pulse recovery. Two pulses overlapping in time, with $f_{\text{carrier}} = 275\ MHz/401\ MHz$, are recovered form hardware data. The f_{carrier} of both pulses is estimated to within .234 MHz.

Current electronic intelligence (ELINT) systems [125, 126] use analog signal processing to extract pulse descriptor words (PDW) which are digitally post-processed in order to deinterleave and identify pulse trains. One of the shortcomings of current analog ELINT systems, which the RMPI was designed to address, is the fact that PDW cannot be extracted in the presence of interferers—also referred to as "pulse-on-pulse" and "pulse-on-CW" (either time or frequency domain overlap).

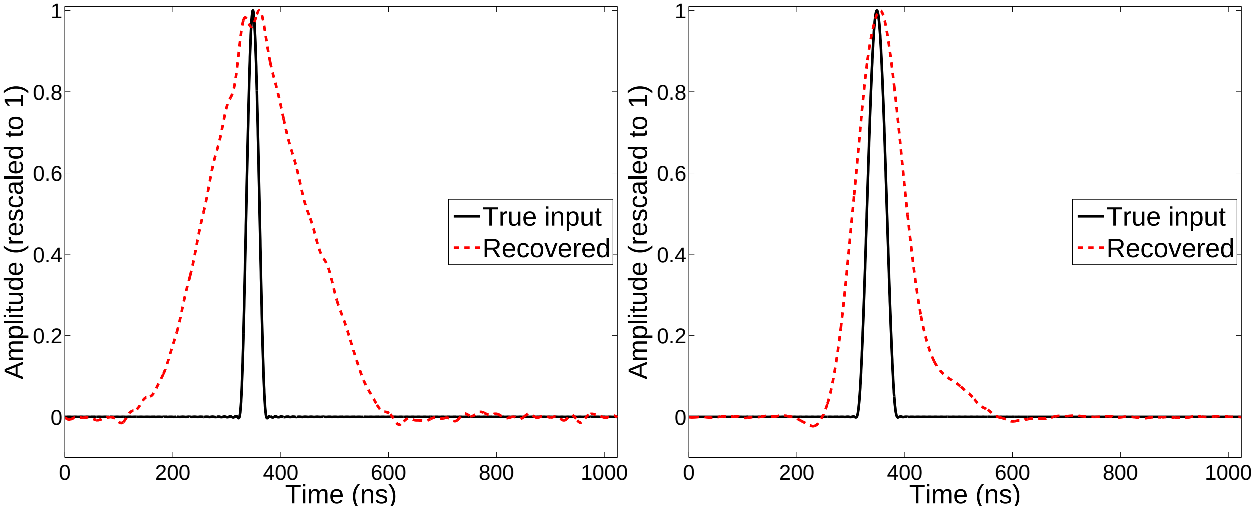
These shortcomings arise from the limitations of the relatively simple analog signal processing techniques used such as amplitude or phase monopulse detection. These methods are known to be inferior in tasks such as direction-finding (DF)/direction-of-arrival (DOA) estimation to more modern digital methods such as MUSIC [127–131] which enable separating, isolating, and recovering signals that simultaneously occupy multiple channels at once—a capability which the analog counterparts lack. At present, analog techniques are employed simply because the required bandwidth of current ELINT systems makes such processing difficult in the digital-domain. In terms of hardware, a typical setup is to use a front-end that implements a narrowband search in conjunction with the combination of either a large baseband filter-bank or variable video bandwidth filter followed by energy-detection/thresholding for parameter estimation. The narrowband scanning front-end typically employs a combination of variable length dwell-times and a complicated scanning plan to optimize search revisit time. The use of scanning precludes the possibility of getting a snapshot of the majority of the entire bandwidth at any given time; this prohibits reliable detection of overlapping pulses or even two pulses that do not have some minimum separation in time. Other difficulties arise from the fact that the use of large filter banks or variable video bandwidth filters limits the resolution of estimation of parameters such as T_{arrival} and amplitude as well as setting a hard limit on the usable minimum pulse-width respectively.

Current efforts in designing ELINT systems are focused on implementing digital channelized receiver strategies. While channelization *potentially* allows the entire bandwidth to be observed and provides benefits such as increasing sensitivity (through SNR enhancement via reduction of noise bandwidth) and mitigating the use of high-speed digitizers, the limitations to resolution that arise from the narrowband baseband filters still apply. Also, while theoretically possible [132], it is not an altogether trivial matter to “stitch” the information from several channels back together to recover the input. In particular, the cases of signals occupying either a large bandwidth (relative to that of a single baseband filter) or portions of the spectrum lying in the overlapping passbands of 2 adjacent channels present considerable challenges. In general, the bookkeeping is complicated and may still necessitate the use of large amounts of analog hardware.

In contrast, while the RMPI does employ multiple channels, it does so not for the purposes of channelizing the input bandwidth but to produce distinct measurements of the entire bandwidth; see Ch. 4.2. Although the baseband filters (the integrators) are extremely narrowband (when viewed as a lowpass filter), the resolution limitations due to the limited bandwidth do not apply as CS-samples represent correlations as opposed to point-samples. Specifically, the RMPI relies on the good memory properties of the baseband filter unlike conventional counterparts which rely on high out-of-band rejection and flat group-delay in the passband.

6.2.5 Narrow Pulse Reconstruction

The last case-study we examine is the reconstruction of extremely narrow pulses. Figures 6.6a and 6.6b depict time domain reconstructions of a 50 ns and 75 ns pulse respectively. Although the pulse-envelope reconstructions are of low-quality, what is notable is that the accurate frequency estimation of the pulses is possible from 16 and 24 RMPI samples respectively versus the ≥ 200 required by Nyquist. These results are not isolated incidents and demonstrate the data-compressing aspects of the RMPI.



(a) 50ns pulse

(b) 75ns pulse

Figure 6.6: Reconstructed baseband windows of pulses of widths (a) 50 ns and (b) 75 ns .

These reconstructions, in conjunction with the other results presented in this section, suggest that using the RMPI may have benefits/advantages beyond reducing required digitization rate or compressing data. The ability of the RMPI to estimate the carrier frequencies from across the entire input (with no adjustment of operating conditions) simultaneously from a comparatively small amount of samples suggests that applications that require quick spectral support estimation such as cognitive radio [15, 133–136] could benefit from application of the ideas at the core of the RMPI. The change in data format from impulse-sampling to correlation is shown to be helpful in circumventing limitations in more conventional counterparts, e.g., estimation resolution limitations imposed by the bandwidth of the baseband filters. We reserve further discussion for the conclusion in Ch. 7.

Chapter 7

Conclusion

7.1 Comparison to Similar Conventional Systems

While a fair and meaningful comparison between a CS system and a Nyquist-rate system is heavily application dependent, we make a brief comparison between the RMPI and a high-speed ADC to provide a context for interpreting the results. Examination of state-of-the-art ADCs reported in ISSCC from 2009-2011 (Table 2.2) gives several possible points of comparison. For example, a 10 *bit* interleaved ADC implemented in 65 *nm* CMOS with $f_{\text{adc}} = 2.6$ *Gsps*, consumes 480 *mW*, and 5.1 mm^2 [12]. Implementation of digital filtering necessitates oversampling the Nyquist-rate by a factor ≥ 2. This means that the 2.6 *Gsps* ADC would be used for a bandwidth of ≤ 600 *MHz*. In addition, two of these ADCs would be required to perform coherent detection via I/Q demodulation consuming 2 *W*, excluding any RF front-end that was required. Assuming a roughly linear scaling between sampling-rate and power consumption, this would equate to > 6 *W* of power consumption to realize an EIBW ≈ 2 *GHz*.

In contrast, the prototype RMPI consumes only 506.4 *mW* of power. We do not include the power consumed by the computational platform needed for signal recovery. CS-based signal reconstruction is a non-linear function of the acquired samples and more computationally expensive than Nyquist-rate reconstruction. As a result, real-time *time-domain* reconstruction is not currently practical in portable (low-power) applications and is the subject of extensive on-going research. To give a rough idea of the computational costs, a typical

CS-recovery algorithm requires about $20 - 1000$ FFTs.

This cost only applies when a complete time-domain reconstruction is needed. Often—if not almost always—the desired information is not the complete time-domain waveform but rather a small set of parameters (e.g. pulse width, carrier frequency, initial phase, etc.) which are extracted from the acquired waveform. It is possible to estimate parameters of the desired signal directly without first reconstructing the time-domain [55, 69, 83]. This procedure is less computationally expensive and potentially competitive with traditional Nyquist-rate approaches. The exact cost depends on the complexity of the signal model; for example, if the signal contains exactly one frequency, then frequency and phase information can be recovered at the cost of only one FFT.

7.2 Potential Applications

Despite the currently involved computational costs, there are still many benefits to using RMPI-like systems. We point out that from a pragmatic standpoint, *real-time* signal processing in the conventional sense is still limited for bandwidths in excess of several *GHz*. Many applications rely not on full time-domain reconstruction but perform feature extraction on the acquired data. In the discussion about the reconstruction of two pulses overlapping in time in Ch. 6, it was demonstrated that the RMPI may have several advantages in producing superior measurements as well as excellent potential in spectrum sensing applications.

We emphasize that CS-techniques are valuable tools that can be used to achieve both substantial power savings in current applications (a very crucial issue in many remote sensing applications) and realize systems with EIBWs currently unobtainable by working in conjunction with state of the art ADCs. This is in addition to the data traffic reduction. In fact, CS principles have already been employed in the Herschel satellite telescope: it has limited on-board processing capability and a highly constrained down-link channel to the planet surface and sends CS samples to a central processing station located on the planet.

CS is a promising solution for many applications such as UAV, where recovery is done off-line. A depiction of the UAV application is shown in Fig. 7.1. In such applications,

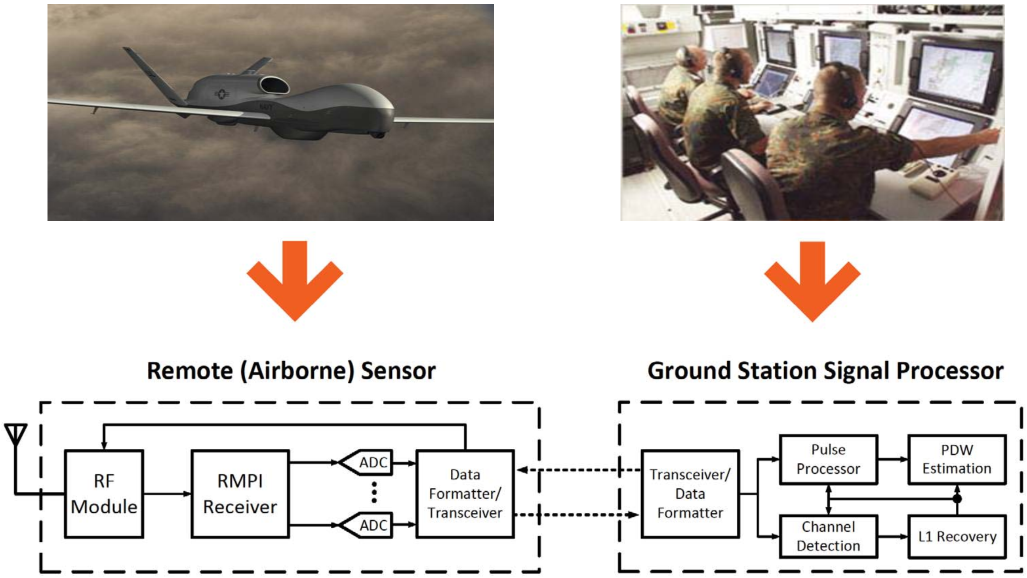


Figure 7.1: An example application: unmanned aerial vehicles (UAV). The current link bandwidth for airborne signal/electronic intelligence (SIGINT/ELINT) applications ranges in the $1 - 10$ *Mbps* range. This data-link has to be shared between the many functions that the UAV has to perform which includes sending and receiving commands from the base-station, telemetry for navigation, as well as relaying image data. The acquired data is sent to a base-station for analysis which is already computationally intensive. The potential benefits of CS in UAV are clear.

the RMPI not only can reduce power consumption and/or provide observation of superior bandwidths, but it can also reduce the strain on the precious downlink budget.

So in what scenarios is use of the RMPI beneficial? We give a non-comprehensive list of scenarios below:

- When the input signals are known to be sparse in some domain.
- In applications in which it is desired to observe instantaneous bandwidths far in excess of ADCs or DSPs allowed by the specified power budget.
- For systems that transmit signal information to a centralized location for sophisticated and thus computationally costly analysis.
- For noisy environments containing signal with low-information content. This is due to the fact that the estimation/approximation algorithms already incorporate digital denoising, a function which conventional Nyquist-based recovery does not perform.

Despite the *present* computational costs involved, we note that there have already been remarkable improvements in the speed and efficiency of CS recovery algorithms with new information recovery methods being reported all the time [137–140]. In addition, research into exploitation of specialized computational hardware such as GPUs [11] is still in its infancy. Further developments in hardware specialized to solving sparse-approximation problems is just beginning [141–143] and there is little doubt that we are far from exhausting the many avenues towards addressing this problem. Some potential future directions are discussed in the next section.

7.3 Future Directions

The RMPI is in many ways the most literal application of CS theory to receiver design. It is not unreasonable to expect that there is considerable room for improvement. By demonstrating feasibility of implementation, this work motivates continued investigation of both further refinements of the RMPI architecture, as well as the investigation of new applications of CS-type approaches to the design of physical devices.

On the back-end, a considerable amount of skepticism regarding the CS approach concerns the cost of signal reconstruction. While the speed of these methods continues to improve, their computational cost can still be appreciably greater than their more conventional Nyquist sample processing counterparts. The limitations imposed by the computational cost may prevent more wide-spread adoption if not addressed. An interesting approach investigated during the course of the RMPI project was the development of algorithms which performed parameter estimation of radar pulse characteristics directly form CS samples [69] whose underlying theory is described in [83]. Further investigation in the direct processing of CS-type samples to extract information of interest is a direction that merits further investigation. On the hardware side, this effort could potentially be assisted with hardware architectures that efficiently implement correlation of signals in more mathematically abstract domains. On the theory side, research in the design of structured measurement matrices [63, 144] could also play a significant role in the direct processing of CS samples.

7.4 Summary

The design and implementation of the first fully-integrated, high-speed, CS-based receiver capable of recovering structured signals at a sub-Nyquist rate has been demonstrated in 90 *nm* CMOS. It can recover signals from an EIBW of 100 *MHz*–2 *GHz* with > 54 *dB* dynamic range while undersampling the Nyquist-rate by 12.5×. The chip implements an 8-channel RMPI, occupies a die area of 8.85 mm^2 and consumes 506.4 mW.

A detailed study/analysis of the RMPI was presented which elucidated what the significant parameters of the system were and their respective effects on overall system performance. These studies led to the development of a principled design procedure as well as a novel calibration procedure that enables an accurate approximation of Φ from measurements of pure-tone inputs. The approach taken to designing the RMPI was described and the effects of various forms of physical corruption were investigated. The results of numerical simulations that were used during validation of the design was also presented.

The measurements establish the physical feasibility of the approach and provide ample motivation for continued research. The work in this thesis presents a comprehensive overview of the first foray into the hitherto uncharted territory of designing high-speed CS-based receivers. The information provided in this thesis should provide considerable insight into any future research-efforts in the area of RMPI/CS-based wideband receivers.

Appendix A

System Numerical Simulation Models

The lack of previous work in the area of CS-based receiver design necessitated answering design feasibility questions at many different levels of abstraction. In order to answer these questions, several difficulties arose from the nature of CS-sampling and recovery algorithms themselves. For example, CS prescribes random projection in order to obtain measurements. Thus, even for relatively simple test inputs, the output samples are seemingly random and are difficult to directly interpret to assess the efficacy of any candidate design. Moreover, signal processing on CS samples for recovery/parameter estimaion is a nonlinear operation which is considerably more costly in terms of required time and computational resources. Consequently, models with varying levels of abstraction were created in order to study questions that arose during the design process. This section describes the techniques/models used to simulate the system for the purposes of: establishing feasibility, investigating effects of different parameters, optimizing performance during the final stages of the design process, and establishing the robustness of the designed system to different sources of nonideality. The numerical results presented throughout this thesis present many different types of simulations based on models of widely varying complexity that incorporated different assumptions. The models were improved over time by gradually adding additional effects that had either been initially neglected or deemed to be of interest based upon the application. The most accurate, and hence highest complexity models were based on simulink simulations; in contrast, the lowest complexity models were simply implemented by randomly generating a ± 1 matrix (which we denote Φ_{ideal} and using highly oversampled versions of the rows of Φ_{ideal} as well as the input $x(t)$ and employing various reconstruction algorithms on samples generated from these models.

There were several purposes behind modeling the system. The first is due to the sensitivity of CS-based reconstruction to the accuracy of the matrix Φ. In the initial stages of design, Φ_{ideal} was used to both synthesize RMPI samples to test the performance of recovery algorithms as well as serve as the reference matrix Φ_{ref} for use in the chosen recovery algorithm approximation. While this approach could be used to validate a given recovery algorithm's robustness to noise, it was inadequate for dealing with departures introduced by the designed hardware to Φ_{ideal}. In addition, studying and quickly changing characteristics of the system such as the bandwidth and specific aspects of the transfer function of each block could only be done by altering the schematics. It was advantageous, from both the standpoint of simulation time and potential for insight to create models that enabled studying the effect of different parameters quickly and in relative isolation from one another. The most detailed of the models used, created in the simulink simulation environment, in the design of the system is described below.

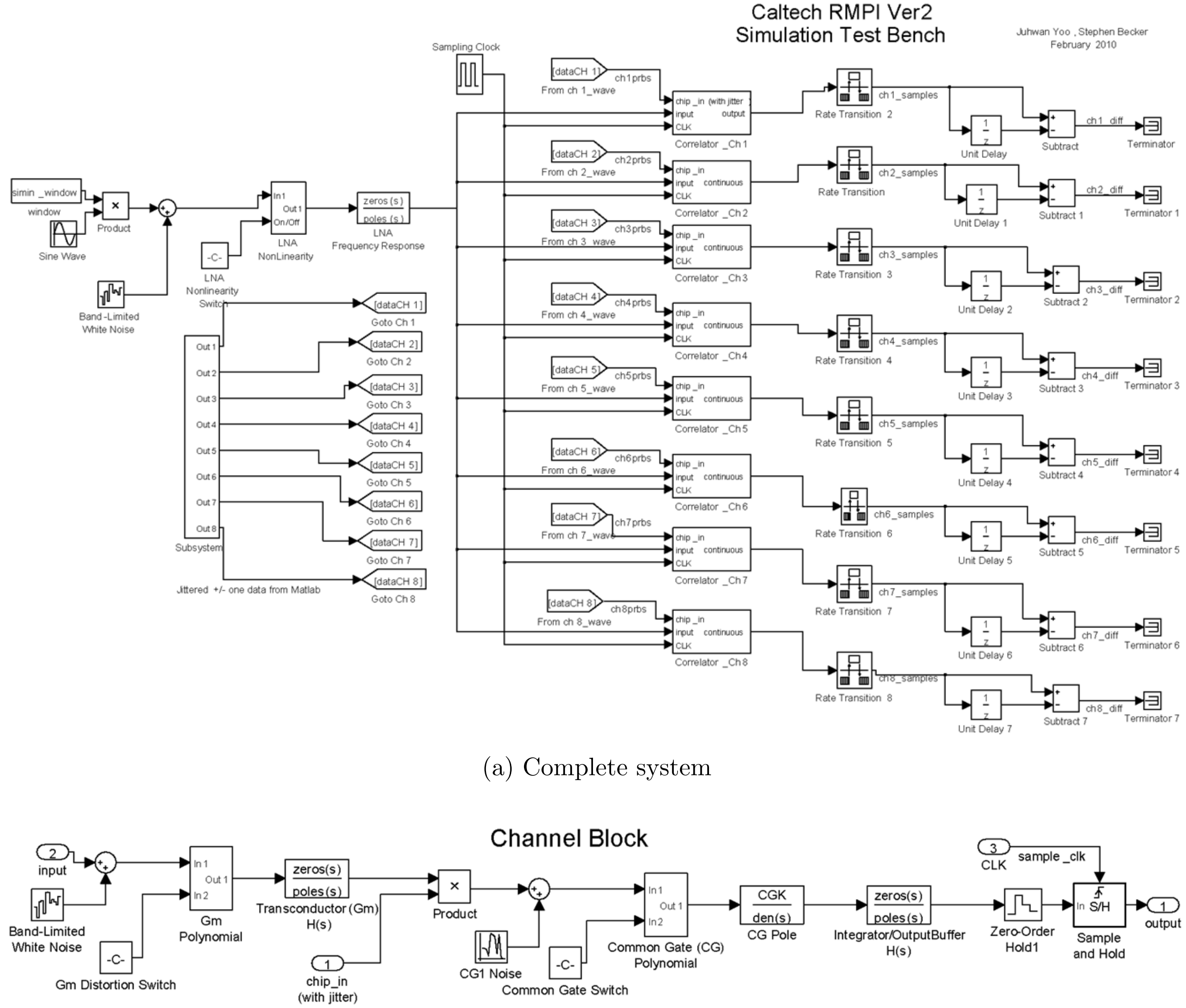
Simulink Model

Due to the cost in terms of design time of the schematic-level of the system, a behavioral model of the system was created in simulink. The purpose of the simulink model was two-fold:

1. Generate test samples quickly that could be input into reconstruction algorithms to test performance.

2. To generate a more accurate impulse response (in the form of the measurement matrix Φ) based upon the effects modeled by the system to be used within the chosen signal-recovery algorithm. This process is referred to as calibration and is described in Ch. 4.3.1.

An added benefit of this model was its pre-existing integration with the MATLAB work environment as well as the visual 'block-level' representation of the system provided. The simulink model was used to determine block-level specifications of each major circuit block to be used in the system. A detailed discussion of the implementation of each block is given in Ch. 5.3.



(a) Complete system

(b) Single RD channel

Figure A.1: Simulink model of the CMOS RMPI. (a) shows the entire system model. (b) depicts the internal blocks of a single RD channel circled in red in (a).The direction of signal flow in both (a) and (b) is form left to right.

1. Low-noise amplifier (LNA)—input to the complete system

2. Input transconductor (Gm)—input amplifier to each RD channel
3. Passive mixer—used to modulate (multiply) the input $x(t)$ with the chipping sequence
4. Current-buffer—used to interface the output of the mixer with the input to the integrator
5. Integrator—Analog correlator used
6. Sample-and-hold—used to model the windowed integration
7. Quantization

A diagram of the complete simulink model is shown in Fig. A.1.

Analog Signal Path Modeling Although the specific behavior modeled for each block varied by the function performed, in general, the blocks in the RF signal path were characterized via their input-referred noise-spectrum, a polynomial representing the nonlinearity of the block, and a linear transfer function consisting of an appropriate number of poles and zeros that mimicked their simulated small signal behavior. The general modeling approach consisted of designing and simulating the transistor-level schematic in Cadence and then fitting the simulated transient/AC results output by Cadence simulations. For example, for the LNA, the AC transfer function was fitted with a set number of poles and zeros to approximate the behavior of the bode plot produced by an AC simulation. The input-referred noise (which summarizes the effect of all noise sources within the block and then calculates the effective input noise source necessary to generate the observed output) is approximated with a white-noise source. The non-linear behavior is modeled by a fifth-order polynomial with coefficients calculated by the results of transient simulations that tested the input-intercept-points of various orders.

Modeling of Digital Components and Clock Distribution The PRBS was simulated by generating the waveform of the PRBS in advance of the simulink simulation and storing the generated signals for playback. The waveforms were generated on an appropriate resolution time grid for compatibility with the simulation step-size and solver used. The action of the sample-and-hold was approximated by generating the sampling clock in a similar fashion to the PRBS. The principal nonideal effects that were modeled were those introduced by clock jitter and bandlimitedness. Bandlimitedness was modeled by inserting a lowpass filter in between the block which output the PRBS and the mixer block. Jitter was modeled by varying the exact location in time of the zero-crossings with a zero-mean gaussian distribution. In order to model this effect accurately, the simulation time-step was set to as low as 0.01 *ps*. All simulations were performed using Matlab's ode3 solver with a fixed time-step to facilitate high time-resolution modeling of jitter. The total simulation time is a function of the time-duration of the signal window T_{win} and the resolution of the time-step. A typical simulation varied in duration from 10 *s* to a few minutes for a single window. Quantization was performed on the final samples generated and truncated to a predetermined output range.

Additional Capabilities The simulink model was designed so that different effects such as input-referred noise or nonlinearity could be switched on and off to allow study of the sensitivity of the reconstruction process to various effects in isolation. In addition, several variants of the simulink model were created to facilitate the study of channel-to-channel timing mismatch (insertion of delay elements at appropriate places) as well as the effects of cross-talk (insertion of a coupling mechanism between two or more different types of nodes within a single channel as well as channel-to-channel).

Bibliography

[1] G. E. Moore, "Cramming more components onto integrated circuits," *Electronics*, vol. 38, no. 8, pp. 114–117, April 19, 1965.

[2] B. Murmann, "Trends in low-power, digitally assisted A/D conversion," *IEICE Trans. Electron.*, vol. E93-C, no. 6, pp. 718–727, 2010.

[3] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inform. Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[4] D. Donoho, "Compressed sensing," *IEEE Trans. Inform. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[5] E. J. Candès, "Compressive sampling," *Int. Cong. Math.*, vol. 3, pp. 1433–1452, 2006.

[6] J. Yoo, S. Becker, M. Loh, M. Monge, E. Candès, and A. Emami-Neyestanak, "Design and implementation of a fully integrated compressed-sensing signal acquisition system," in *Proc. Int. Conf. on Acoustics Speech and Sig. Processing*, Kyoto, Japan, 2012.

[7] J. Yoo, S. Becker, M. Loh, M. Monge, E. Candès, and A. Emami-Neyestanak, "A 100MHz-2GHz 12.5x sub-Nyquist rate receiver in 90nm CMOS," Proc. of IEEE Radio Freq. Integrated Cir. Conf. (RFIC), *accepted for presentation*, 2012.

[8] J. N. Laska, S. Kirolos, M. F. Duarte, T. Ragheb, R. G. Baraniuk, and Y. Massoud, "Theory and implementation of an analog-to-information converter using random demodulation," in *Proc. IEEE ISCAS, New Orleans*, May 2007, pp. 1959–1962.

[9] J. A. Tropp, J. N. Laska, M. F. Duarte, J. Romberg, and R. G. Baraniuk, "Beyond Nyquist: Efficient sampling of sparse bandlimited signals," *IEEE Trans. Inform. Theory*, vol. 56, no. 1, pp. 520–544, 2010.

[10] B. Murmann and B. Boser, "Digitally assisted analog integrated circuits," *ACM Queue*, vol. 2, no. 1, pp. 64–71, 2004.

[11] S. Becker, E. Candès, D. Ching, A. Emami-Neyestanak, M. Grant, E. Nakamura, J. Romberg, E. Sovero, M. Wakin, and J. Yoo, "A non-uniform sampler for wideband spectrally-sparse environments," *submitted to IEEE J. on Emerging and Sel. Topics in Cir. and Sys.*, 2011.

[12] K. Doris, E. Janssen, C. Nani, A. Zanikopoulos, and G. Van der Weide, "A 480mW 2.6GS/s 10b 65nm CMOS time-interleaved ADC with 48.5dB SNDR up to nyquist," in *Proc. Int. Solid-State Circ. Conf.*, 2011, pp. 180–182.

[13] R. H. Walden, "Analog-to-digital converter survey analyis," *IEEE J. on Selected Areas in Communications*, vol. 17, no. 4, pp. 539–550, 1999.

[14] B. Le, T. W. Rondeau, J. H. Reed, and C. W. Bostian, "Analog-to-digital converters: A review of the past, present, and future," *IEEE Sig. Proc. Mag.*, vol. 22, no. 6, pp. 69–77, 2005.

[15] I. F. Akylidiz, W. Lee, M. C. Vuran, and S. Mohanty, "NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey," *Computer Networks*, vol. 50, no. 13, pp. 2127–2159, 2006.

[16] E. J. Candès and M. Wakin, "An introduction to compressive sampling," *IEEE Sig. Proc. Mag.*, vol. 25, no. 2, pp. 21–30, 2008.

[17] W. Skones, B. Oyama, S. Stearns, J. Romberg, and E. Candès, "Analog to information (A-to-I), technical and management proposal," Tech. Rep., (in response to DARPA BAA 05-35), 2005.

[18] S. Kirolos, J. N. Laska, M. Wakin, M. F. Duarte, D. Baron, T. Ragheb, Y. Massoud, and R. G. Baraniuk, "Analog-to-information conversion via random demodulation," in *Proc. IEEE DCAS*, Sept. 2006.

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

[20] B. Razavi, "Architectures and circuits for RF CMOS receivers," in *Proc. Cust. Integrated Circ. Conf.*, 1998, pp. 393–400.

[21] B. Razavi, "Design considerations for direct-conversion receivers," *IEEE Trans. on Circ. and Sys. II*, vol. 44, no. 6, pp. 428–435, 1997.

[22] A. Abidi, "Direct-conversion radio transceivers for digital communications," *IEEE J. of Solid-State Circ.*, vol. 30, no. 12, pp. 1399–1410, 1995.

[23] J. Laskar, B. Martinpour, and S. Chakraborty, *Modern Receiver Front-Ends: Systems, Circuits, and Integration*, Wiley-Interscience, 2004.

[24] T. H. Lee, *The Design of CMOS Radio-Frequency Integrated Cicruits*, Cambridge, U.K.: Cambridge Univ. Press, 1998.

[25] J. G. Proakis, *Digital Communications*, New York: McGraw-Hill, 2001.

[26] S. Haykin, *Communication Systems*, John Wiley & Sons: 4th Edition, 2001.

[27] B. Razavi, *RF Microelectronics*, Prentice Hall, 2nd Ed., 2011.

[28] M. C. M. Soer, E. A. M. Klumperink, Z. Ru, F. E. van Vliet, and B. Nauta, "A 0.2-to-2.0GHz CMOS receiver without LNA achieving >11dBm IIP3 and <6.5dB NF," in *Proc. IEEE Int'l Solid-State Circ. Conf.*, 2009.

[29] R. Bagheri, A. Mirzaei, S. Chehrazi, M. Heidari, M. Lee, M. Mikhemar, W. Tang, and A. Abidi, "A 800-MHz-6-GHz software-defined wireless receiver in 90-nm CMOS," *IEEE J. of Solid-State Cir.*, vol. 41, no. 12, pp. 2860–2876, 2006.

[30] N. A. Moseley, Z. Ru, E. A. M. Klumperink, and B. Nauta, "A 400-to-900MHz receiver with dual-domain harmonic rejection exploiting adaptive interference cancellation," in *Proc. IEEE Int'l Solid-State Circ. Conf.*, 2009.

[31] M. Cao, B. Chi, and C. Zhang, "A 1.2V 0.1-3GHz software-defined radio receiver front-end in 130nm CMOS," in *Proc. IEEE Radio-Freq. Integr. Circ. Conf.*, 2011.

[32] B. Cook, A. Berny, A. Molnar, S. Lanzisera, and K. Pister, "Low-power 2.4-ghz transceiver with passive rx front-end and 400mv supply," *IEEE J. of Solid-State Circ.*, vol. 41, no. 12, pp. 2757–2766, 2006.

[33] J. Borremans, G. Mandal, V. Giannini, T. Sano, M. Ingels, B. Verbruggen, and J. Craninckx, "A 40nm CMOS highly linear 0.4-to-6GHz receiver resilient to 0dBm out-of-band blockers," in *Proc. IEEE Int'l Solid-State Circ. Conf.*, 2011.

[34] B. Razavi, *Principles of Data Conversion System Design*, Wiley-IEEE Press, 1st Ed., 1994.

[35] F. Maloberti, *Data Converters*, Springer, 1st Ed., 2007.

[36] R. Schreier and G. C. Temes, *Understanding Delta-Sigma Converters*, Wiley-IEEE Press, 1st Ed., 2004.

[37] W. Kester, *The Data Conversion Handbook*, Elsevier, 2005.

[38] B. Murmann, "A/D converter trends: power dissipation, scaling and digitally assisted architectures," in *IEEE Int. Circ. Conf.*, 2008.

[39] R. Payne, C. Sestok, W. Bright, M. El-Chammas, M. Corsi, D. Smith, and N. Tal, "A 12b 1GS/s SiGe BiCMOS two-way time-interleaved pipeline ADC," in *Proc. IEEE Int'l Solid-State Circ. Conf.*, 2011.

[40] R. C. Taft, P. A. Francese, M. R. Tursi, O. Hidri, A. Mackenzie, T. Hoehn, H. Werker, and A. Glenny, "A 1.8V 1.0GS/s 10b self-calibrating unified-folding-interpolating ADC with 9.1 ENOB at nyquist frequency," in *Proc. IEEE Int'l Solid-State Circ. Conf.*, 2009.

[41] J. Mulder, F. M. L. van der Goes, Vecchi D., J. R. Westa, E. Ayranci, C. M. Ward, J. Wan, and K. Bult, "An 800MS/s dual-residue pipeline ADC in 40nm CMOS," in *Proc. IEEE Int'l Solid-State Circ. Conf.*, 2011.

[42] S. Devarajan, L. Singer, D. Kelley, S. Decker A. Kamanth, and P. Wilkins, "A 16b 125MS/s 385mW 78.7dB SNR CMOS pipeline ADC," in *Proc. IEEE Int'l Solid-State Circ. Conf.*, 2009.

[43] A. Panigada and I. Galton, "A 130mW 100MS/s pipelined ADC with 69dB SNDR enabled by digital harmonic distortion correction," in *Proc. IEEE Int'l Solid-State Circ. Conf.*, 2009.

[44] L. Brooks and H.-S. Lee, "A 12b 50MS/s fully differential zero-crossing-based ADC without CMFB," in *Proc. Int'l Solid-State Circ. Conf.*, 2009.

[45] I. Ahmed, J. Mulder, and D. A. Johns, "A 50MS/s 9.9mW pipelined ADC with 58dB SNDR in 0.18μm CMOS using capacitive charge-pumps," in *Proc. IEEE Int'l Solid-State Circ. Conf.*, 2009.

[46] N. Ben-Hamida, D. Pollex, J. Sitch, S. Wang, and J. Wolczanski, "A 24GS/s 6b ADC in 90nm CMOS," in *Proc. IEEE Int'l Solid-State Circ. Conf.*, 2008.

[47] K.Poulton, R. Neff, B. Setterberg, B. Wuppermann, T. Kopley, R. Jewett, J. Pernillo, C. Tan, and A. Montjio, "A 20GS/s 8b ADC with a 1MB memory in 0.18μm CMOS," in *Proc. IEEE Int'l Solid-State Circ. Conf.*, 2003.

[48] H. J. Landau, "Sampling, data transmission, and the nyquist rate," *Proc. of the IEEE*, vol. 65, no. 10, pp. 1701–1706, 1967.

[49] E. J. Candès and Y. Plan, "A probabilistic and RIPless theory of compressed sensing," *Preprint: arXiv:1011.3854*, 2010.

[50] D. L. Donoho and X. Huo, "Uncertainty principles and ideal atomic decomposition," *IEEE Trans. Info. Theory*, vol. 47, pp. 2845–2862, 2001.

[51] J. A. Tropp, *Topics in Sparse Approximation*, Ph.D. thesis, U. of Texas at Austin, 2004.

[52] D. Gabor, "Theory of communication: part 1. the analysis of information," *J. of the Instit. of Elec.Engineers - Part III: Radio and Commun. Engineering*, vol. 93, no. 26, pp. 429–441, 1946.

[53] S. Mallat, *A Wavelet Tour of Signal Processing: The Sparse Way*, Elsevier Academic Press 3rd Ed., 2009.

[54] E. J. Candès and J. K. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Problems*, vol. 23, no. 3, pp. 969–988, 2007.

[55] S. R. Becker, *Practical Compressed Sensing: modern data acquisition and signal processing*, Ph.D. thesis, Caltech, 2011.

[56] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge Univ. Press, 2004.

[57] S. Scott, *Basis Pursuit*, Ph.D. thesis, Stanford University, 1995.

[58] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Statist. Soc., Ser. B*, vol. 58, no. 1, pp. 267–288, 1996.

[59] E. Candès, "The restricted isometry property and its implications for compressed sensing," *Comptes Rendus Mathematique*, vol. 346, pp. 589–592, 2008.

[60] G. E. Pfander, H. Rauhut, and J. A. Tropp, "The restricted isometry property for time-frequency structured random matrices," *arXiv*, 2011, 1106.3184.

[61] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, "A simple proof of the resitrcted isometry property for random matrices," *Constructive Approximation*, vol. 28, no. 3, pp. 253–263, 2008.

[62] R. Berinde, A. Gilbert, P. Indyk, H. Karloff, and M. Strauss, "Combining geometry and combinatorics: a unified approach to sparse signal recovery," in *Allerton*, 2008.

[63] H. Rauhut, "Compressive sensing and structured random matrices," *Theoretical Foundations and Numerical Methods for Sparse Recovery*, vol. 9, pp. 1–92, 2010.

[64] R. L. Pickholtz, D. L. Schilling, and L. B. Milstein, "Theory of spread-spectrum communications—a tutorial," *IEEE Trans. on Comm.*, vol. 30, no. 5, pp. 855–884, 1982.

[65] M. Wakin, J. Laska, M. Duarte, D. Baron, S. Sarvothan, D. Takhar, K. Kelly, and R. Baraniuk, "An architecture for compressive imaging," in *Proc. Int'l Conf. Image Process.*, Atlanta, 2006.

[66] M. Lustig, D. L. Donoho, and J. M. Pauly, "Sparse mri: The application of compressed sensing for rapid mr imaging," *Magn. Reson. Med.*, vol. 58, no. 6, pp. 1192–1195, 2007.

[67] A. Harmsy, W. U. Bajwa, and R. Calderbank, "Beating nyquist through correlations: a contrained random demodulator for sampling of sparse bandlimited signals," in *Proc. Int. Conf. on Acoustics, Speech, and Sig. Processing*, Prague, 2011.

[68] M. Mishali and Y. C. Eldar, "From theory to practice: Sub-Nyquist sampling of sparse wideband analog signals," *IEEE J. Sel. Top. Signal Process.*, vol. 4, no. 2, pp. 375–391, 2010.

[69] J. Yoo, C. Turnes, E. Nakamura, C. Le, S. Becker, E. Sovero, M. Wakin, M. Grant, J. Romberg, A. Emami-Neyestanak, and E. Candès, "A compressed sensing parameter extraction platform for radar pulse signal acquisition," submitted to JETCAS, 2012.

[70] F. Chen, A. P. Chandrakasan, and V. M. Stojanovic, "Design and analysis of a hardware-efficient compressed sensing architecture for data compression in wireless sensors," *IEEE J. of Solid-State Circ.*, vol. 47, no. 3, pp. 744–756, 2012.

[71] F. Chen, A. Chandrakasan, and V. Stojanovic, "A signal-agnostic compressed sensing acquisition system for wireless and implantable sensors," in *Proc. IEEE Cust. Integr. Circ. Conf.*, 2010.

[72] B. Gosselin and M. Sawan, "Circuit techniques and microsystems assembly for intracortical multichannel ENG recording," in *Proc. IEEE Cust. Integr. Circ. Conf.*, 2009.

[73] M. Mishali, Y. C. Eldar, O. Dounaevsky, and E. Shoshan, "Xampling: Analog to digital at sub-Nyquist rates," *IET Cir. Dev. and Systems*, vol. 5, no. 1, pp. 8–20, Jan. 2011.

[74] M. Mishali, Y. C. Eldar, and A. Elron, "Xampling–part i: Practice," Tech. Rep. 0911.0519, arXiv, 2009.

[75] Y. C. Eldar, "Compressed sensing of analog signals in shift-invariant spaces," *IEEE Trans. Sig. Proc.*, vol. 57, no. 8, pp. 2986–2997, Aug. 2009.

[76] M. Mishali and Y. C. Eldar, "Expected RIP: Conditioning of the modulated wideband converter," in *IEEE Info. Theory Workshop*, October 2009, pp. 343–347.

[77] Y. Chen, M. Mishali, Y. C. Eldar, and A. O. Hero, "Modulated wideband converter with non-ideal lowpass filters," in *ICASSP*, Dallas, March 2010, pp. 3630–3633.

[78] M. Mishali and Y. C. Eldar, "Blind multiband signal reconstruction: Compressed sensing for analog signals," *IEEE Trans. Signal Process.*, vol. 57, no. 3, pp. 993–1009, 2009.

[79] D. Adams, C. S. Park, Y. C. Eldar, and B. Murmann, "Towards an integrated circuit design of a compressed sampling wireless receiver," in *Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Sig. Processing*, Kyoto, Japan, 2012.

[80] C. Choudhuri, A. Ghosh, U. Mitra, and S. Pamarti, "Robustness of xampling-based RF receivers against analog mismatches," in *Proc. IEEE Int'l Conf. on Acous. Speech and Sig. Process.*, 2012.

[81] N. Wagner, Y. C. Eldar, A. Feuer, G. Danin, and Z. Friedman, "Xampling in ultrasound imaging," arXiv:1104.5327v1, 2011.

[82] F. Chen, *Energy-efficient Wireless Sensors: Fewer Bits, Moore MEMS*, Ph.D. thesis, Massachusetts Institute of Technology, 2011.

[83] M. A. Davenport, M. F. Duarte, M. B. Wakin, J. N. Laska, D. Takhar, K. F. Kelly, and R. G. Baraniuk, "The smashed filter for compressive classification and target recognition," in *Proc. Comp. Imag. V at SPIE Elect. Imag.*, San Jose, 2007.

[84] S. Weinreb, "A digital spectral analysis technique and its application to radio astronomy," Tech. Rep., Massachusetts Institute of Technology Cambridge Research Lab of Electronics, 1963, Accession Number: AD0418413.

[85] S. R. Becker, E. Candeès, and M. Grant, "Templates for convex cone problems with applications to sparse signal recovery," *Mathematical Programming Computation*, vol. 3, no. 3, pp. 165–218, 2011.

[86] S. R. Becker, J. Bobin, and E. Candès, "NESTA: a fast and accurate first-order method for sparse recovery," *SIAM J. Imag. Sci.*, vol. 4, no. 1, 2011.

[87] ℓ_1 *Magic*, `http://www.acm.caltech.edu/l1magic`, 2006.

[88] J. A. Tropp and S. J. Wright, "Computational methods for sparse solution of linear inverse problems," *Proc. of the IEEE*, vol. 98, no. 6, pp. 948–958, 2010.

[89] M. C. M. Soer, E. A. M. Klumperink, B. Nauta, and F. E. van Vliet, "A 1.0GHz-to-4.0GHz CMOS four-element beamforming receiver using a switched-capacitor vector modulator with approximate sine weighting via charge redistribution," in *Proc. IEEE Int'l Solid-State Circ. Conf.*, 2011.

[90] C. Andrews and A. C. Molnar, "A passive-mixer-first receiver with baseband-controlled RF impedance matching, < 6dB NF, and > 27dBm wideband iip3," in *Proc. IEEE Int'l Solid-State Circ. Conf.*, 2010.

[91] Z. Ru, E. A. M. Klumperink, G. J. M. Wienk, and B. Nauta, "A software-defined radio receiver architecture robust to out-of-band interference," in *Proc. IEEE Int'l Solid-State Circ. Conf.*, 2009.

[92] Y. Tsividis, "Integrated continuous-time filter desing—an overview," *IEEE J. of Solid-State Circ.*, vol. 29, no. 3, pp. 166–176, 1994.

[93] X. Chen, Z. Yu, S. Hoyos, B. M. Sadler, and J. Silva-Martinez, "A sub-Nyquist rate sampling receiver exploiting compressive sensing," *IEEE Trans. Circuits and Systems I: Regular Papers*, vol. 58, no. 3, pp. 507–520, March 2011.

[94] T. Blumensath, "Compressed sensing with nonlinear observation," *IEEE Trans. Info. Theory,* under review, 2011, Available at `http://www.personal.soton.ac.uk/tb1m08/papers/B_Nonlinear.pdf`.

[95] E. Keehr and A. Hajimiri, "Successive regeneration and adaptive cancellation of higher order intermodulation products in rf receivers," *IEEE Tran. on Microwave Theory and Techniques*, vol. 59, no. 5, pp. 1379–1396, 2011.

[96] S. Chehrazi, A. Mirzaei, and A. A. Abidi, "Second-order intermodulation in current-commutating passive FET mixers," *IEEE Trans. on Circ. and Sys. I*, vol. 56, no. 12, pp. 2556–2568, 2009.

[97] A. Eftekhari, J. Romberg, and M. B. Wakin, "Matched filtering from limited frequency samples," Tech. Rep. 1101.2713, arXiv, 2011.

[98] M. A. Davenport, P. T. Boufounos, M. B. Wakin, and R. G. Baraniuk, “Signal Processing with Compressive Measurements,” *IEEE J. Sel. Top. Signal Proc.*, vol. 4, pp. 445–460, 2010.

[99] J. A. Tropp and A. C. Gilbert, “Signal recovery from random measurements via orthogonal matching pursuit,” *IEEE Trans. on Inform. Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[100] J. Mairal, *SPAMS: SPArse Modeling Software*, WILLOW, INRIA, 2.1 edition, June 2011.

[101] D. Needell and J. Tropp, “CoSaMP: Iterative signal recovery from incomplete and inaccurate samples,” *Appl. and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009.

[102] V. Cevher, “An ALPS view of sparse recovery,” in *IEEE Intl. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, Prague, may 2011, pp. 5808–5811.

[103] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, “A fast approach for overcomplete sparse decomposition based on smoothed l0 norm,” *IEEE Trans. Sig. Proc.*, vol. 57, no. 1, pp. 289–301, jan 2009.

[104] D. L. Donoho, A. Maleki, and A. Montanari, “Message-passing algorithms for compressed sensing,” *Proc. Nat. Acad. Sci.*, vol. 106, no. 45, pp. 18914–18919, 2009.

[105] M. Davenport, J. N. Laska, J. R. Treichler, and R. G. Baraniuk, “The pros and cons of compressive sensing for wideband signal acquisition: Noise folding vs. dynamic range,” arXiv:1104.4842, 2011.

[106] E. Arias-Castro and Y. C. Eldar, “Noise folding in compressed sensing,” arXiv:1104.3833v1, 2011.

[107] N. Poobuapheun, W. Chen, Z. Boos, and A. Niknejad, "A 1.5v 0.7-2.5-ghz CMOS quadrature demodulator for multiband direct-conversion receivers," *IEEE J. of Solid-State Circ.*, vol. 42, no. 8, pp. 1669–1677, 2007.

[108] E. Keehr and A. Hajimiri, "A rail-to-rail input receiver employing successive regeneration and adaptive cancellation of intermodulation products," in *Proc. IEEE Radio Freq. Int. Cir. Conf.*, 2010, pp. 47–50.

[109] E. Keehr and A. Hajimiri, "A wide-swing low-noise transconductance amplifier and the enabling f large-signal handling direct-conversion receivers," *IEEE Trans. on Circ. and Sys. I*, vol. 59, no. 1, 2012.

[110] S. Chehrazi, A. Mirzaei, R. Bagheri, and A. Abidi, "A 6.5 GHz wideband CMOS low noise amplifier for multi-band use," in *Proc. IEEE Custom Integrated Cir. Conf.*, 2005.

[111] S. Blaakmeer, E. Klumperink, and B. Nauta, "An inductorless wideband balun-lna in 65nm CMOS with balanced output," in *Proc. Eur. Solid State Cir. Conf.*, 2007.

[112] W. Redman-White and D. M. W. Leenaerts, "1/f noise in passive CMOS mixers for low and zero IF integrated receivers," in *Proc. of the 27th Eur. Solid-State Circ. Conf.*, 2001.

[113] H. Darabi and A. A. Abidi, "Noise in RF-CMOS mixers: A simple physical model," *IEEE J. of Solid-State Circ.*, vol. 35, no. 1, pp. 15–25, 2000.

[114] P.R. Gray, P. J. Hurst, S. H. Lewis, and R. G. Meyer, *Analysis and Design of Analog Integrated Circuits*, John Wiley & Sons Inc, fourth ed., 2001.

[115] E. Sacchi, I. Bietti, S. Erba, L. Lee, P. Vilmercati, and R. Castello, "A 15mW, 70kHz 1/f corner direct conversion CMOS receiver," in *Proc. of the IEEE Custom Int. Circ. Conf.*, 2003, pp. 459–462.

[116] M. Valla, G. Montagna, R. Castello, R. Tonietto, and I. Bietti, "A 72-mw CMOS 802.11a direct conversion front-end with 3.5dB NF and 200-kHz 1/f noise corner," *IEEE J. of Solid-State Circ.*, vol. 40, no. 4, pp. 970–977, 2005.

[117] N. Kim, V. Aparin, and L. E. Larson, "A resistively degenerated wide-band passive mixer with low noise figure and high IIP2," *IEEE Trans. on Micr. Theory and Techniques*, vol. 58, no. 4, pp. 820–830, 2010.

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

[119] H. Darabi, "A noise cancellation technique in active RF-CMOS mixers," *IEEE J. of Solid-State Circ.*, vol. 40, no. 12, pp. 2628–2632, 2005.

[120] A. Mirzaei, H. Darabi, J. C. Leete, X. Chen, K. Juan, and A. Yazdi, "Analysis and optimization of current-driven passive mixers in narrowband direct-conversion receivers," *IEEE J. of Solid-State Circ.*, vol. 44, no. 10, pp. 2678–2688, 2009.

[121] T. Nguyen, N. Oh, V. Le, and S. Lee, "A low-power CMOS direct conversion receiver with 3-dB NF and 30-kHz flicker-noise corner for 915-MHz band IEEE 802.15.4 zigbee standard," *IEEE Trans. Microwave Theory and Tech.*, vol. 54, no. 2, pp. 735–741, 2006.

[122] R. Gregorian, *Introduction to CMOS Op-Amps and Comparators*, New York: Wiley, 1999.

[123] N. Weste and D. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective*, Addison Wesley, 4th Ed., 2010.

[124] A. Emami-Neyestanak, A. Varzaghani, J. F. Bulzacchelli, A. Rylyakov, C. K. Yang, and D. J. Friedman, "A 6.0-mw 10.0-gb/s receiver with switched-capacitor summation dfe," *IEEE J. of Solid-State Circ.*, vol. 42, no. 4, pp. 889–896, 2007.

[125] R. Wiley, *Electronic Intelligence: The Analysis of Radar Signals*, Artech House, Inc., 1982.

[126] C. Knapp, "The generalized correlation method for estimation of time delay," in *Proc. IEEE Int'l Conf. on Acous. Speech and Sig. Process.*, 1976.

[127] R. O. Schmidt, "Multilinear array manifold interpolation," *IEEE Trans. on Sig. Processing*, vol. 40, no. 4, pp. 857–866, 1992.

[128] A. J. Barabell, "Improving the resolution performance of eigenstructure-based direction-finding algorithms," in *Proc. IEEE Int'l Conf. on Acous. Speech and Sig. Process.*, 1983.

[129] B. Poat and B. Friedlander, "Direction finding algorithms based on higher-order statistics," *IEEE Trans. on Sig. Process.*, vol. 39, no. 9, pp. 2016–2024, 1991.

[130] R. Roy, A. Paulraj, and T. Kailath, "ESPRIT-a subspace rotation approach to estimation of parameters of cisoids in noise," in *Proc. IEEE Int'l Conf. on Acous. Speech and Sig. Process.*, 1986.

[131] P. P. Vaidyanthan, S. Phoong, and Y. Lin, *Signal Processing and Optimization for Transceiver Systems*, Cambridge University Press, 1st Ed., 2010.

[132] S. K. Mitra and J. F. Kaiser, *Handbook for Digital Signal Processing*, Wiley-Interscience, 1993.

[133] Z. Tian and G. Giannakis, "Compressed sensing for wideband cognitive radios," in *Proc. IEEE Int'l Conf. on Acous. Speech and Sig. Process.*, 2007.

[134] S. M. Almalfouh and G. L. Stuber, "Interference-aware power allocation in cognitive radio networks with imperfect spectrum sensing," in *Proc. IEEE Int'l Conf. on Communications*, 2010.

[135] F. Ge, "Cognitive radio: From spectrum sharing to adaptive learning and reconfiguration," in *Proc. IEEE Aerospace Conf.*, 2008.

[136] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Comm. Surveys & Tutorials*, vol. 11, no. 1, pp. 116–130, 2009.

[137] H. Pham, W. Dai, and O. Milenkovic, "Sublinear compressive sensing reconstruction via belief propagation decoding," in *Proc. IEEE Int. Symp. on Inform. Th.*, 2009.

[138] A. Maleki, *Approximate Message Passing Algorithms for Compressed Sensing*, Ph.D. thesis, Stanford University, 2011.

[139] A. S. Charles, P. Garrigues, and C. J. Rozell, "Analog sparse approximation with applications to compressed sensing," arXiv:1111.4118, 2011.

[140] A. Charles, M. S. Asif, J. Romberg, and C. Rozell, "Sparsity penalties in dynamical system estimation," in *Proc. Inform. Sciences and Sys.*, 2011.

[141] P. Maechler, C. Studer, D. Bellasi, A. Maleki, A. Burg, N. Felber, H. Kaeslin, and R. G. Baraniuk, "VLSI implementation of approximate message passing for signal restoration and compressive sensing," submitted to JETCAS, 2012.

[142] J. Yoo, A. Khajehnejad, B. Hassibi, and A. Emami-Neyestanak, "An analog sub-linear time sparse signal acquisition framework based on structured matrices," in *Proc. Int. Conf. on Acoustics Speech and Sig. Processing*, Kyoto, Japan, 2012.

[143] A. Balavoine, C. J. Rozell, and J. Romberg, "Global convergence of the locally competitive algorithm," *Digital Signal Processing*, 2011.

[144] T. Do, T. D. Tran, and L. Gan, "Fast compressive sampling with structurally random matrices," in *Proc. Int. Conf. on Acous. Speech and Sig. Processing*, 2008.