Building Compressed Sensing Systems: Sensors and Analog-to-Information Converters

by

Omid Salehi-Abari

B.Eng., Carleton University, Ottawa (2010)

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

Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2012

Author . . . .
Department of Electrical Engineering and Computer Science
August 31, 2012

Certified by . . . .
Vladimir Stojanović
Associate Professor
Thesis Supervisor

Accepted by . . . .
Leslie A. Kolodziejski
Chairman, Department Committee on Graduate Theses

Building Compressed Sensing Systems: Sensors and Analog-to-Information Converters

by

Omid Salehi-Abari

Submitted to the Department of Electrical Engineering and Computer Science
on August 31, 2012, in partial fulfillment of the
requirements for the degree of
Master of Science in Electrical Engineering and Computer Science

Abstract

Compressed sensing (CS) is a promising method for recovering sparse signals from fewer measurements than ordinarily used in the Shannon's sampling theorem [14]. Introducing the CS theory has sparked interest in designing new hardware architectures which can be potential substitutions for traditional architectures in communication systems. CS-based wireless sensors and analog-to-information converters (AIC) are two examples of CS-based systems. It has been claimed that such systems can potentially provide higher performance and lower power consumption compared to traditional systems. However, since there is no end-to-end hardware implementation of these systems, it is difficult to make a fair hardware-to-hardware comparison with other implemented systems. This project aims to fill this gap by examining the energy-performance design space for CS in the context of both practical wireless sensors and AICs. One of the limitations of CS-based systems is that they employ iterative algorithms to recover the signal. Since these algorithms are slow, the hardware solution has become crucial for higher performance and speed. In this work, we also implement a suitable CS reconstruction algorithm in hardware.

Thesis Supervisor: Vladimir Stojanović
Title: Associate Professor

Acknowledgments

First and foremost, I would like to express my appreciation and gratitude to my thesis supervisor, Professor Vladimir Stojanivić. It is as a result of his excellent guidance and support that this research has been possible. I will not forget his guidance and advice. I thank my closest collaborators, Fabian Lam and Fred Chen. I have been lucky to have worked with them. I would also like to thank the rest of Integrated Systems Group team. It has been a real pleasure working with everyone.

Finally, I thank all of my amazing friends and family for their support. I am grateful that I have wonderful parents and I owe all of my success to their support and effort. I especially thank my brother, Amirali, who always cares about my education and success, and cheered me on along the way.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years, advances in sensor technologies and integrated electronics have revolutionized communications systems. However, high energy consumption and limited energy sources have rendered the proposed communication systems not practical. Often times, the energy consumption of such systems is dominated by the amount of energy required to transmit the data in wireless sensors/systems or by high-speed data converters such as high-speed analog-to-digital converters (ADC) [7–9].

1.1 Wireless Sensors

Figure 1-1 shows the energy cost of functional blocks in a wireless sensor node. As the figure shows, the amount of energy consumed to transmit data is orders of magnitude greater than any other functions common in most wireless sensors. Hence, to reduce the energy consumption of the system and consequently improving the battery life, it is necessary to minimize the amount of transmitted data [10,11]. Many promising compression schemes have been proposed to enable reduction in the transmitted data [11,12]. However, there is always a trade-off between the compression gain and complexity (implementation cost). The goal is to have a low-complex compression scheme which minimizes the total energy cost, while the required information can still

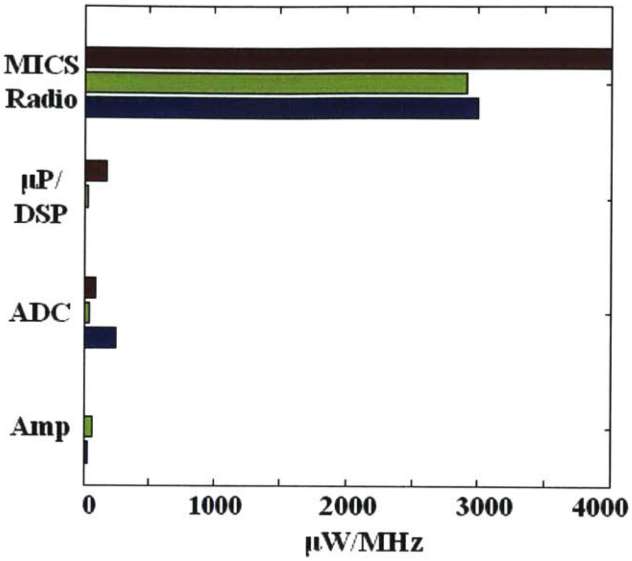

Figure 1-1: Energy costs of functional blocks in a wireless sensor [1].

be recovered. Recent innovations in wireless sensor networks, suggest the use of compressed sensing as a data compression scheme that achieve the aforementioned goals. CS is a promising method for recovering sparse signals from fewer measurements than ordinarily used in the Shannons sampling theorem [13, 14].

1.2 High-Speed Samplers

Figure 1-2 shows the energy cost of recently published ADCs. As their sampling rate increases, not only do their power consumption increase but also their performance worsen. Performance limitations of high-speed ADCs are mainly due to sampling jitter which causes a significant error at high sampling rate. Recently, in applications where signal frequencies are high, but information bandwidths are low, analog-to-information converters (AICs) have been proposed as a potential solution to overcome the resolution and performance limitations of sampling jitter in high-speed analog-to-digital converters (ADC). AICs can be used for any type of signals, which are sparse in some domain. Consequently, AICs can relax the frequency require-

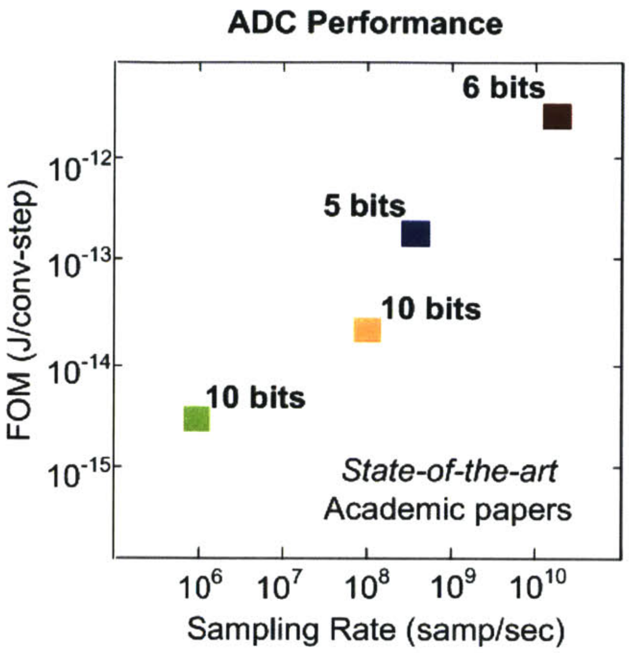

Figure 1-2: Energy cost and performance of recently published ADCs. [2–5]

ments of ADCs, potentially enabling higher resolution and/or lower power receiver front-ends.

1.3 Contributions of this Thesis

In this work, end-to-end system evaluation frameworks are designed to analyze CS performance under practical sensor settings and to evaluate the performance of AIC for high-bandwidth sparse signal applications. Furthermore, relative energy-efficiency of AIC versus ADCs is examined. Finally, different reconstruction algorithms are evaluated in terms of performance (reliability), implementation scalability, speed, and power-efficiency. Then, by considering these factors, a suitable reconstruction algorithm is chosen to implement in hardware.

1.4 Thesis Overview

Chapter 2: CS-Based Wireless Sensor

In this chapter, we examine the performance of CS-based source encoders in terms of both energy cost and compression ratio. We show that there is a trade-off between the compression ratio and the quality of recovered signal in CS-based encoders. We also explore how impairments, such as input noise and channel noise, can affect the performance of such systems. Finally, we propose some diversity schemes to improve the performance of these encoders in case of packet loss.

Chapter 3: Analog to Information Converters

This chapter provides some background on high-speed samplers and analog-to-information converters (AIC). Then, we explore how circuits non-idealities, such as jitter and aperture impairments, impact AIC performance. We show that similar to high speed ADCs, AIC systems also suffer in high-bandwidth applications. In this chapter, we also compare the energy cost of AIC systems versus high-speed ADCs for high-bandwidth sparse signal applications.

Chapter 4: Hardware Implementation of a CS Decoder

This chapter starts with some background on different reconstruction algorithms (Matching Pursuit, Orthogonal Matching Pursuit and Approximate Message Passing) which can be used in a CS Decoder. In particular, we examine the performance of these algorithms in terms of complexity and the quality of reconstructed signal. We show that there is a trade-off between hardware cost (i.e. complexity) and reconstruction performance. As our results show, Matching Pursuit algorithm is suitable for applications that require high speed, but can handle lower signal quality. This is a direct result of the low computational complexity of this algorithm. Finally, we present hardware architecture for Matching Pursuit Algorithm and evaluate the performance of this architecture in terms of both energy-cost and throughput.

Chapter 5: Conclusion

In this chapter, we review the results and briefly discuss advantages and disadvantages of using CS-based systems versus traditional systems.

Chapter 2

CS-Based Wireless Sensors

CS-based source encoders have been proposed as a potential data compression scheme for wireless sensors. Unlike the Shannon sampling theory, which requires the signal to be sampled at a rate proportional to bandwidth, a CS-based encoder captures the data at a rate proportional to the information rate [14] . Hence, this method enables fewer data samples than traditionally required when capturing a signal with relatively high bandwidth, but a low information rate. We designed an end-to-end system evaluation framework to analyze CS performance in a practical wireless sensor environment. We examined the trade-off between the required energy at the transmitter (i.e., compression performance) and the quality of the recovered signal at the receiver. In particular, we explored how impairments such as input signal noise, quantization noise and channel noise impact the transmitted signal energy required to meet a desired signal quality at the receiver.

2.1 Compressed Sensing Background

This section presents a brief overview of Compressed Sensing (CS). CS theory states that a signal can be sampled without any information loss at a rate close to its information content. CS relies on two fundamental properties: signal sparsity and incoherence [13, 14].

Signals are represented with varying levels of sparsity in different domains. For example, a single tone sine wave is either represented by a single frequency coefficient or by an infinite number of time-domain samples. In general, consider signal $\boldsymbol{f}$ represented as follows:

$$\boldsymbol{f} = \boldsymbol{\Psi x} \tag{2.1}$$

where $\boldsymbol{x}$ is the coefficient vector for $\boldsymbol{f}$, which is expanded in the basis $\Psi \in R^{N \times N}$ such that $\boldsymbol{x}$ is sparse. When a signal is sparse, most of its coefficients are zero, or they are small enough to be ignored without much perceptual loss. Fortunately, in many applications such as wireless sensors, cognitive radios or medical imagers, the signals of interests have sparse representations in some signal domain. For example, in a cognitive radio application, the signal of interest is sparse in the frequency domain since at any one time only a few percentage of total available channels are occupied by users. Similarly, in medical applications, bio-signals such as ECG signals are sparse in either Gabor or wavelet domain which makes them good candidates as a CS application.

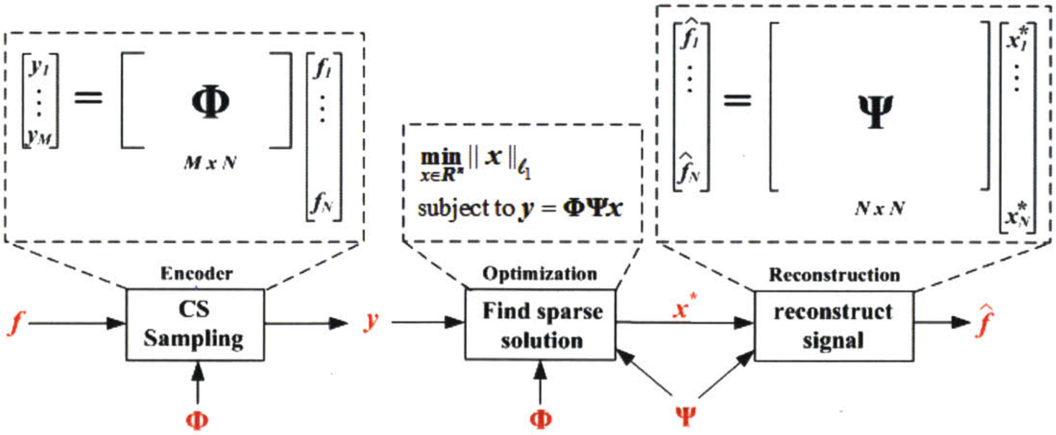

Figure 2-1: Compressed sensing (CS) framework.

The CS framework is shown in Figure 1, where an N dimensional input signal $\boldsymbol{f}$

is compressed to M measurements y, by taking M linear random projections, i.e.,

$$\boldsymbol{y} = \boldsymbol{\Phi f} \tag{2.2}$$

where $\boldsymbol{\Phi} \in R^{M \times N}$, $\boldsymbol{f} \in R^{N \times 1}$ and $M < N$. In this case the system is undetermined, which means there are an infinite number of solutions for $\boldsymbol{f}$. However, the signal is known a-priori to be sparse. Under certain conditions, the sparsest signal representation satisfying 2.2 can be shown to be unique. The sparse solution can be produced by solving the following convex program:

$$\min_{x \in R^N} \parallel x \parallel_{l_1} \; subject \quad to \quad \boldsymbol{y} = \boldsymbol{\Phi \Psi x} \tag{2.3}$$

where $\boldsymbol{\Psi}$ is the basis matrix and $\boldsymbol{x}$ is the coefficient vector from 2.1 [13, 14]. The recovered signal is then $\overset{\wedge}{\boldsymbol{f}} = \boldsymbol{\Psi} \overset{*}{\boldsymbol{x}}$, where $\overset{*}{\boldsymbol{x}}$ is the optimal solution to 2.3.

So far we have explained the first component of compressed sensing which is signal sparsity. The next important property of compressed sensing is incoherent sampling. The coherence, which measures the largest correlation between any two elements of $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ (i.e., any row of $\boldsymbol{\Phi}$ and column of $\boldsymbol{\Psi}$), can be defined as:

$$\mu(\Phi, \Psi) = \sqrt{n} \cdot \max_{1 \leq k,j \leq N} |\langle \varphi_k, \psi_j \rangle| \tag{2.4}$$

The coherence, μ, can range between 1 and $\sqrt{N}$ [13]. As it is shown in [15], the minimum number of measurements needed to recover the signal with overwhelming probability is as follow:

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

where C is a positive constant, S is the number of significant non-zero coefficients in $\boldsymbol{x}$, and N is the dimension of the signal to be recovered, $\boldsymbol{x}$. In other words, the less coherence between $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$the fewer number of measurements needed to recover the signal. Hence, to minimize the required number of measurements, it is required to have the minimum coherence between $\mathbf{\Psi}$ and $\mathbf{\Phi}$. Random matrices are a good candidate for sampling matrix as they have low coherence with any fixed basis, and, as a result, the signal basis $\mathbf{\Psi}$ is not required to be known in advance in order to determine a suitable sampling matrix, $\mathbf{\Phi}$ [15]. It should be noted that, in this work, to maintain the hardware simplicity described in [6], the measurement matrix, $\mathbf{\Phi}$, is chosen to be a random Bernoulli matrix (± 1 entries) generated from a randomly seeded pseudo-random bit sequence (PRBS) generator.

To summarize, compressed sensing requires the signal of interest to be sparse in some fix domain. If a signal has a sparse representation, then only $O(SlogN)$ measurements are required to recover the original signal. Furthermore, because random sampling matrices have good properties, they can be used to build generic and universal data acquisition systems. Hence, the universal CS-based system can be used for different application and signal types, assuming that they are sparse.

2.2 Evaluation Framework

To be able to evaluate the performance of CS-based wireless sensor, we introduce the system models shown in Figure 2-2. Figure 2-2(a) shows a baseline model. In this model, each sample is quantized into Q bits and is directly transmitted over the noisy channel. Figure 2-2 (b) shows a CS-based system where the quantized data is compressed and then measurements (i.e., compressed data) are sent over the noisy channel. In this model, each sample is quantized into Q bits, and each measurement is B bits. Our goal is to compare the performance of these two systems, considering the effect of quantization noise, channel noise, and signal noise.

To illustrate the performance comparison results, random 4-sparse input signals of length $N = 1000$ are constructed from an over-complete dictionary of Gaussian pulses. The dictionary consists of N sample-shifted unit-amplitude copies of a single Gaussian pulse. The signals are generated by drawing on a uniform random distribution over

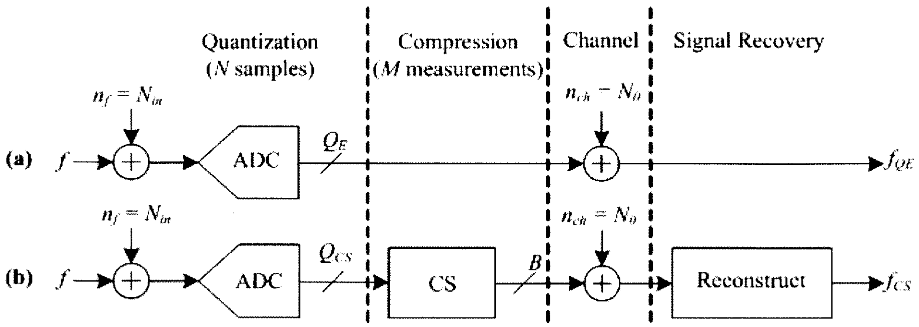

Figure 2-2: Evaluation system models for an input signal block of N samples: (a) a baseline system with input noise, quantization noise, and channel noise and (b) a CS system with input noise, quantization noise, compression error, and channel noise.

[-1,1] to assign the sign and magnitude and over [1,1000] to assign the position of each pulse in the 4-pulse signal. Although this signal was chosen as an example to illustrate the comparison results, the framework and results are directly extendable to alternative signals and dictionaries.

Finally, to compare the performance of the systems shown in Figure 2-2, we adopted the percent root-mean-square difference (PRD) metric [12], which is defined as:

$$PRD = 100\sqrt{\frac{\sum_{n=1}^{N}\left|f[n]-\tilde{f}[n]\right|^2}{\sum_{n=1}^{N}|f[n]|^2}}. \tag{2.6}$$

2.3 Effect of Channel Noise

Similar to other compression schemes, a CS system performance can also be affected by channel noise. However, these undesirable effects can be minimized by optimizing the number of measurements (M) required in a CS-based sensor and the quantization method chosen in the system. This section examines how these two factors can affect the performance of a CS based system.

2.3.1 Quantization Method

In most communication systems, each bit carries the same amount of information. However, in the case of CS-based systems, this is not true. In the bit presentation of measurements in the CS system, some bits are more important than others because they carry more information. One solution to this problem is to combine the CS system with an appropriate channel coding system to protect some of the bits that carry the highest amount of information. However, this approach requires extra hardware. One can choose an optimum quantization method for a specific application and signal to distribute the information over all the bits as evenly as possible. To illustrate the uneven impact of a single bit error on different quantization methods, the reconstruction SNR versus bit position error for the CS-based sensor is plotted in Figure 2-3. Uneven distribution of information in bits can be observed in both *sign and magnitude* quantization and *two's complement* quantization methods. As Figure 2-3 shows, the impact of channel noise can be minimized by using the *sign and magnitude* quantization. This observation is due to the fact that most measurements are small in amplitude. As a result, a single bit error in the first bit position can make a tremendous error since a small positive number has changed to a big negative number. In contrast, in the *sign and magnitude* quantization method, getting an error in the first bit will just change the sign of the small number.

2.3.2 Required Number of Measurements

In the previous section, we explained how different quantization methods can affect the performance of CS-based sensors. The other factor which can impact the performance of these systems is the number of measurements used in CS. In the case of an ideal (i.e. noiseless) channel, a higher number of measurements (M) means a better quality of reconstructed signal. However, it should be noted that increasing the number of measurements also increases the required transition power since more measurements and, consequently, more bits need to be transmitted. In contrast,

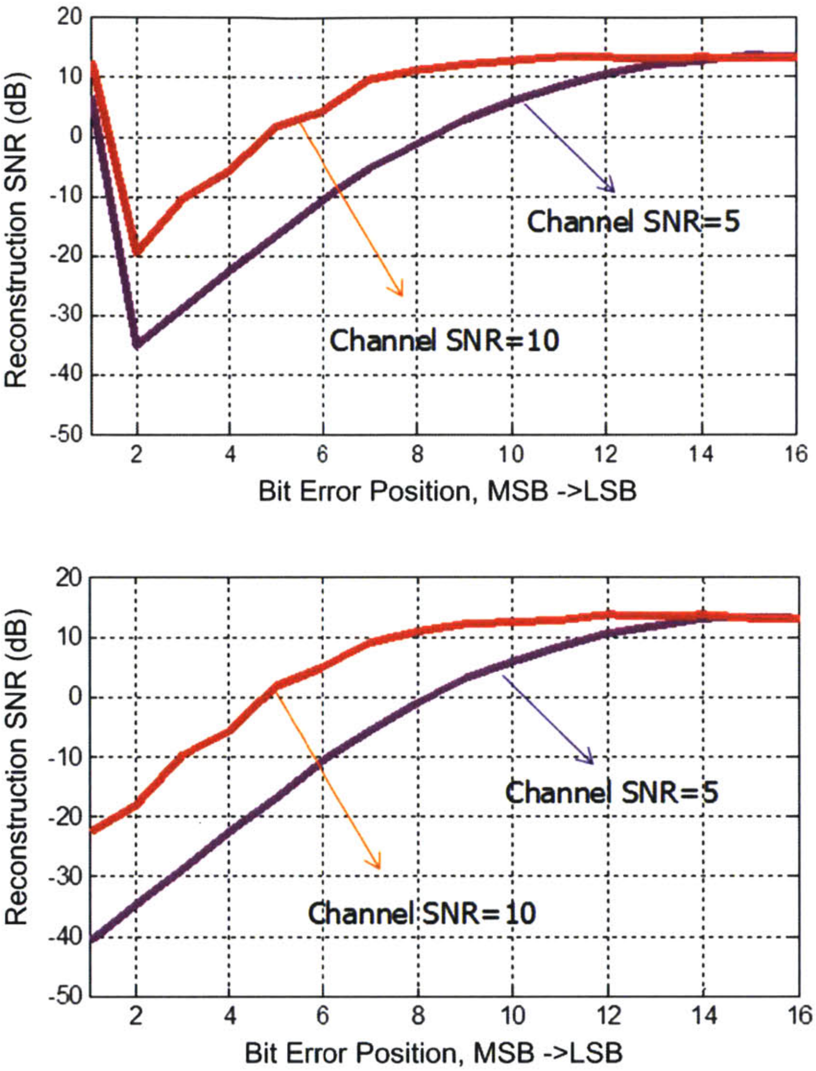

Figure 2-3: Bit position of single bit error in transmitted data versus reconstruction SNR for a 4-sparse signal quantized with (a) sign and magnitude method (b)twos complement method.

in a noisy channel, increasing the number of measurements (M) may worsen the performance of the system. Figure 2-4 show the performance (i.e. quality of the reconstructed signal) versus the position of bit error due to channel noise for CS-based sensors with different numbers of measurements ($M = 50$ and $M = 100$). To compare these two systems, we assume the total available transmission energy is equal in both systems, and as a result, the energy of each bit (Eb) in the system with

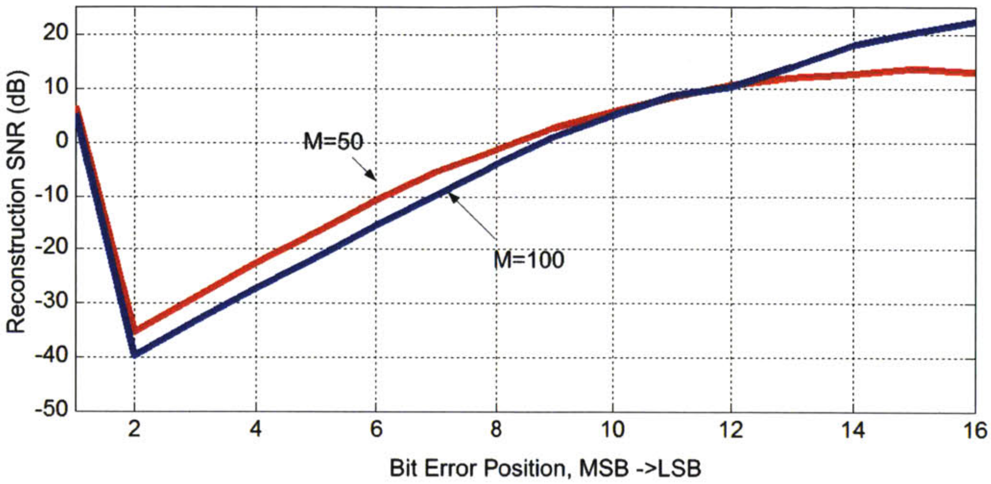

Figure 2-4: Bit position of single bit error in transmitted data versus reconstruction SNR for a 4-sparse signal in CS-based sensors with $M = 50$ and $M = 100$. Channel SNR of 5dB and 2dB are chosen for the system with $M = 50$ and $M = 100$, respectively. (i.e. Assuming that both systems have equal available transmission energy per signal block)

$M = 50$ is two times higher than the E_b in the system with $M = 100$. As figure 2-4 shows, increasing the number of measurements worsens the performance of the system when bit errors happen at higher bit positions (i.e. MSBs). This is due to the fact that the number of transmitted bit is higher and E_b is smaller for the case with higher M, and as a result, the probability of getting errors in measurement bits is higher. However, when the channel is not noisy or when the bit error happens in lower bit positions (LSBs), increasing the number of measurements improves the performance of the system (see figure 2-4). In general, a higher M can improve the quality of reconstructed signal when the channel is relatively noiseless. In conclusion, it is critical to choose a proper M when designing a CS-based sensor since using a small number of measurements worsens the performance of the system and using too many measurements can worsen the performance for noisy channels.

2.4 Enery Cost of Channel Errors

To better understand and compare the performance of CS-based and baseline systems, we first consider the noiseless signal ($n_f = 0$). Figure 2-5 plots the minimum required energy for a range of target PRD performances for both baseline and CS systems, where the energy is in units of channel noise (N_0). For the baseline system, the required energy per sample is simply $Q_E.N_0.SNR_{min,E}$, where $SNR_{min,E}$ is the minimum SNR required to meet the target PRD. In contrast to the baseline system, for the CS system, the minimum required energy per sample is $M.B.N_0.SNR_{min,cs}/N$, where M is the required number of measurements, B is the resolution of each measurement, and N is the total number of samples to be compressed ($N = 1000$ in our setup). It should be noted that there are many system specifications (M, B, Qcs, and SNR) that can achieve a targeted PRD performance. However, they are not all equally energy-efficient. There is a tradeoff between transmitting more bits at a lower SNR and transmitting fewer bits at a higher SNR. Our goal here is to find the system specification which enables the most energy-efficient CS system for a target PRD.

In Figure 2-5, the energy cost is plotted for both net PRD (PRD_{net}) and average PRD (PRD_{avg}) which are defined as:

$$PRD_{net} = 100\sqrt{\frac{\sum_{k=1}^{K}\sum_{n=1}^{N}\left|f_k[n] - \tilde{f}_k[n]\right|^2}{\sum_{k=1}^{K}\sum_{n=1}^{N}\left|f_k[n]\right|^2}}, and \tag{2.7}$$

$$PRD_{avg} = \frac{100}{K}\sum_{k=1}^{K}\sqrt{\frac{\sum_{n=1}^{N}\left|f_k[n] - \tilde{f}_k[n]\right|^2}{\sum_{n=1}^{N}\left|f_k[n]\right|^2}}, \tag{2.8}$$

where K is the total number of input signal blocks (10,000 in this example). Note that PRD_{net} is equivalent to the PRD if all K input blocks were considered as a single input signal. In general, PRD_{net} is the accumulated error performance, whereas PRD_{avg} is equivalent to the time-averaged error performance. Depending on the

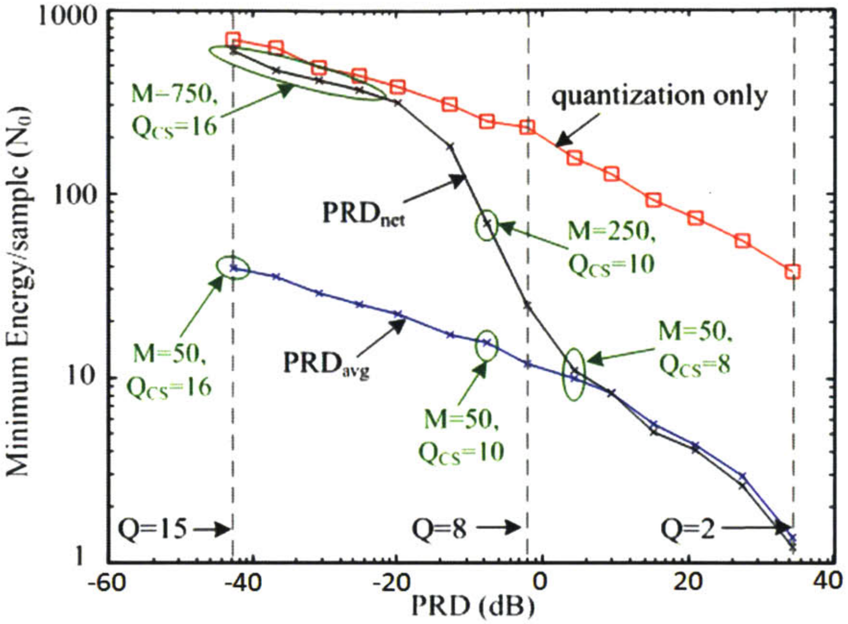

Figure 2-5: Minimum energy per sample (in units of channel noise N0) for each required *PRD* performance for both the baseline system (i.e. uncompressed ADC system) and the CS system for 4-sparse signals.

application, one metric may be more important than the other.

Figure 2-5 also shows that in both systems, the minimum required energy increases as the required system resolution increases (i.e., lower *PRD*). It also shows the order of magnitude energy savings that CS can provide through data compression. However, to achieve a low $PRD_{net}(< 1\%)$, an energy cost on par with the uncompressed quantized samples is required. The reason for this is because more measurements (larger M) are needed to improve the net reconstruction error and hence more energy is required. In contrast, if occasional performance degradation is acceptable over brief blocks of time, as with the time average $PRD(PRD_{avg})$, then CS can offer an order of magnitude energy savings over the entire range of *PRD* specifications when compared to transmitting the raw quantized samples.

2.5 Effect of Input Noise

In the previous sections, we evaluated the performance of CS-based sensor for a noiseless input signal. However, in reality, the input signals are somewhat noisy in any practical sensor. To be able to evaluate the performance of the CS-based system in the presence of input noise, white Gaussian noise is added to our input signal. The input noise variance is chosen to give signal SNR of 40dB and 20dB which are in the range of reasonable SNRs in a practical sensor. It should be noted that the SNR of 40dB and 20dB are equivalent to PRD of 1% and 10%, respectively. Figure 2-6) shows the minimum required energy versus target PRD for the case when input signal is noisy. In the baseline system (i.e. quantization only, see figure 2-2, the recovered signal quality is limited by the PRD of input signal. However, in the CS-based system, some of the input noise is filtered during the signal reconstruction and as a result this system is able to achieve a better PRD. The CS-based system is capable of filtering the input noise since the noise is not correlated with the signal basis and as a result it can be filtered during reconstruction. In contrast, the quantization error is highly correlated with both input signal and basis and as a result this error (noise) cannot be filtered during the reconstruction.

2.6 Diversity Schemes for Packet Loss

To protect against catastrophic error events such as packet loss, we borrow ideas from *multiple description coding* [16] and utilize a simple diversity scheme. In the proposed scheme, each transmitted packet includes the combination of two measurements (See Figure 2-7). In the absence of the proposed protocol, each packet includes only one measurement; in the case of packet loss, that individual measurement cannot be recovered. On the other hand, consider the "Twice Diversity" protocol shown in the figure. Here, the individual measurement can be recovered by adding or subtracting two succesive packets when there is no packet loss. In addition, the scheme

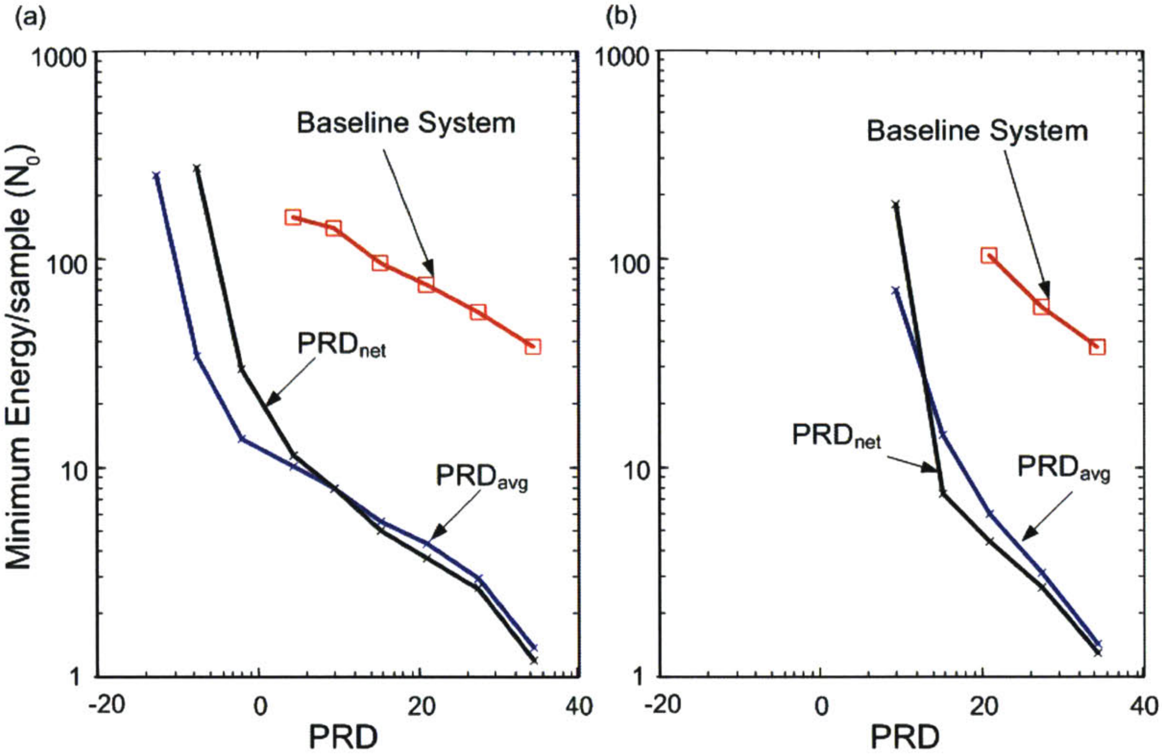

Figure 2-6: Minimum required energy per sample versus required PRD performance for both the baseline system and the CS system for PRD_{avg} and PRD_{net} 4-sparse signals corrupted with noise such that the input PRD equals (a) 1% and (b)10%.

eanables recovering individual measurement in the case of packet loss as long as two consecutive packets are not dropped. This recovery can be achieved by jointly recovering the measurements from the other received combination. Figure 2-7 shows an example where packet 3 is lost, which can cause the loss of measurement y_3. However, by using the proposed scheme, measurement y_3 can still be recovered using packet 4. Although the proposed scheme does not require the transmission of any extra bits, it can improve the performance of the CS-based sensor system over a lossy channel. The performance of the proposed scheme is shown in Figure 2-8 for packet loss probabilities of 0.1% and 1%. The proposed scheme can improve the quality of recoverd signal by more than 10X while it only requires limited hardware overhead as described in [9].

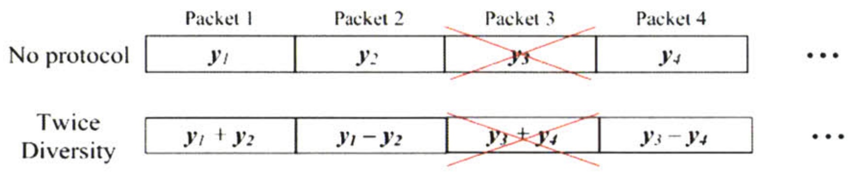

Figure 2-7: Diversity protocol for protection against burst errors and dropped packets.

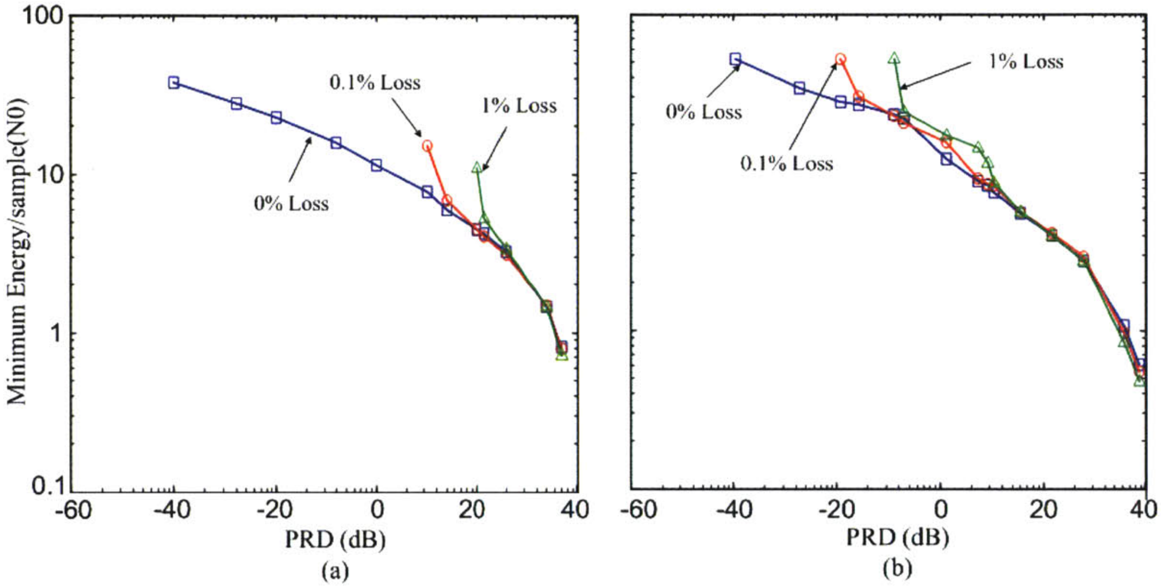

Figure 2-8: Minimum energy per sample vs. required PRD performance for 0%, 0.1% and 1% packet loss for a PRDavg signal with (a) no diversity protocol, and (b) with the diversity protocol.

2.7 ECG: An Example CS application

In the previous sections, we examined the performance of CS-based sensor system for a constructed 4-sparse signal. Here, we do the same analysis and evaluations on a real electrocardiogram (ECG) signal obtained from the MIT-BIH Arrhythmia database. Figure 2-9(a) shows a segment of the ECG signal used to evaluate the performance of the systems shown in figure 2-2. Figure 2-9(b) shows the reconstructed signal for the case when $Q = 12$, $M = 100$ and $N = 1000$, resulting in a PRD of 0.42%. The minimum required energy versus target PRD curves for the ECG signal

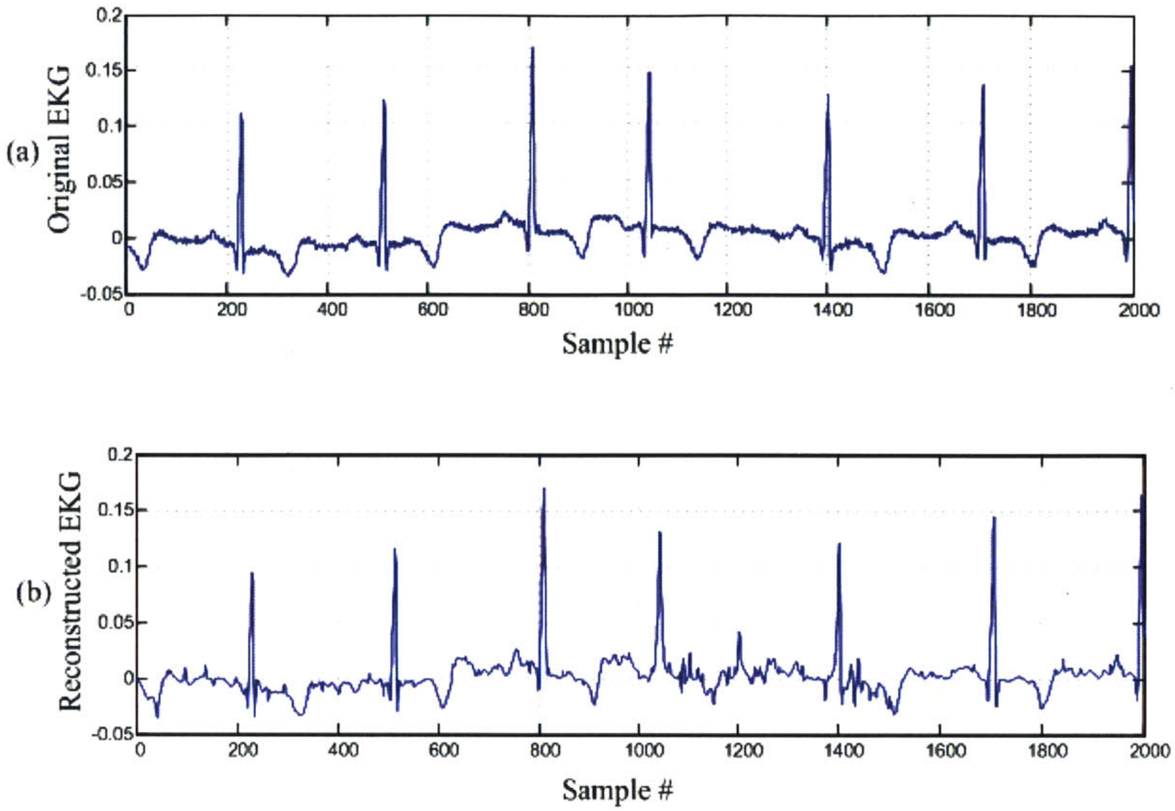

Figure 2-9: Original ECG (a) versus CS reconstruction (b) for $Q = 12$ and $M = 100$.

are plotted for both baseline and CS-based systems. This is a similar result to the case when the input is a constructed 4-sparse signal (see figure 2-5). For target PRD above 1.5% (3.5dB), CS enables about 10X reduction in required energy. It should be noted that in Figure 2-10, PRD_{avg} and PRD_{net} are almost identical. This is due to the fact that the used ECG signal is fairly noisy. Hence, both PRD_{avg} and PRD_{net} become limited by noise effects and as a result they will be identical.

2.8 Summary

In this chapter, we first looked at the effect of channel noise on the performance of CS-based sensors. We showed how this undesirable effect can be minimized by optimizing the number of measurements and quantization method used in the system.

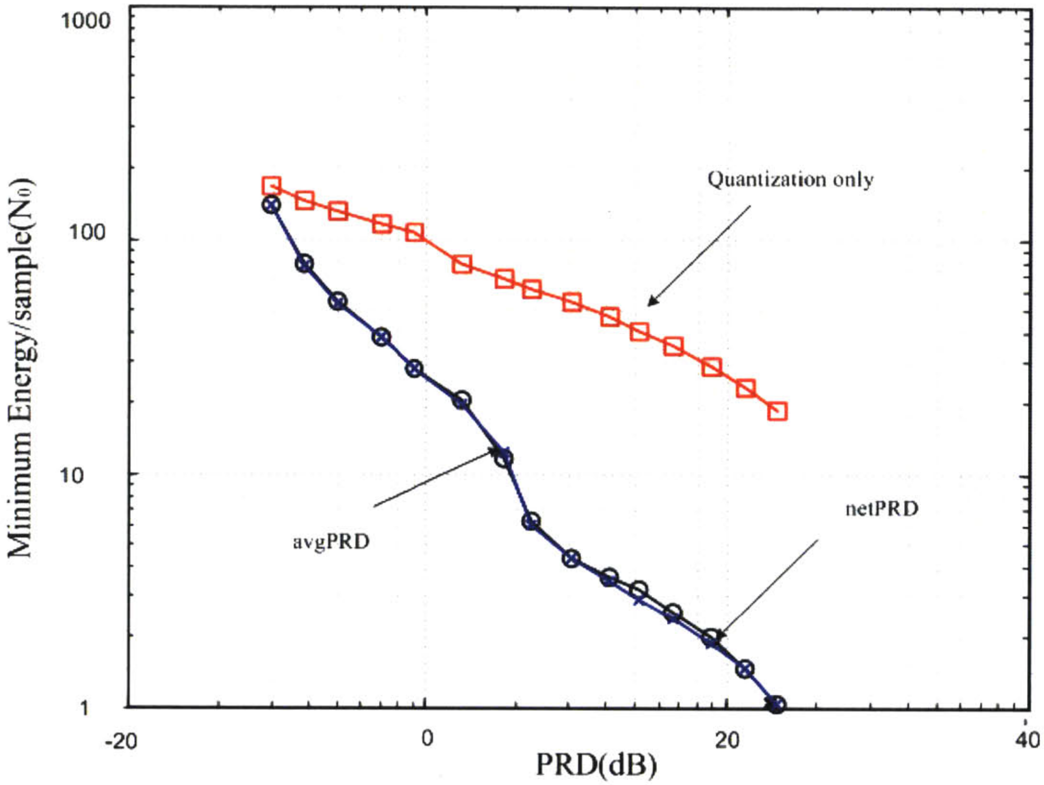

Figure 2-10: Minimum energy versus target PRD for an ECG signal from the MIT BIH Arrhythmia database.

In addition to showing robustness to channel noise, we also evaluate the performance of CS-based systems in the presence of input signal noise. We showed that CS-based system is capable of filtering the input noise while the performance of the baseline system (i.e quantization only) will be limited by the input noise. We have also shown that CS can enable on the order of 10X reduction in transmission energy when compared to the baseline system (i.e. transmitting raw quantized data). Furthermore, we proposed a diversity scheme for CS which requires no additional transmission costs and provides greater than 10X improvement in recovered signal quality. Finally, we showed that the design framework and analysis presented is applicable to real world signals, such as ECGs, with a similar order of magnitude reduction in transmission energy costs.

Chapter 3

Analog to Information Converters

Analog-to-information converters (AICs) have been proposed as a potential solution to overcome the resolution and performance limitations of sampling jitter in high-speed analog-to-digital converters (ADC). In this chapter, we compare both energy cost and performance limitations of AIC and high-speed ADC systems, in the context of cognitive radio applications where the input signal is sparse in the frequency domain. We explore how jitter and aperture impairments, which commonly limit ADC performance at high sampling frequencies, also impact AIC performance

3.1 Cognitive Radio: An example AIC application

Efficient, high-speed samplers are essential for building modern electronic systems. One such system is cognitive radio, which has been proposed as an intelligent wireless communication protocol for improving the utilization of un-used bandwidth in the radio spectrum [17]. To implement this protocol, the entire radio spectrum has to be simultaneously observed in order to determine the location of used channels. A straightforward approach is to utilize a wideband, Nyquist rate high speed analog-to-digital converter (ADC), however, a severe drawback is that ADCs operating at multi-Giga samples per second (GS/s) require high power and have limited bit resolution [7,8]. An alternative approach is to utilize an analog-to-information converter (AIC) based on compressed sensing (CS) techniques [18–26]. Consequently, AICs can relax the frequency requirements of ADCs, potentially enabling higher resolution and/or lower power receiver front-ends. In general, for applications where signal frequencies are high, but information rates are low, AICs have been proposed as a potential solution to overcome the resolution and performance limitations of traditional, Nyquist-rate high-speed ADCs.

3.1.1 Limitations in High-Speed Sampling

To date, high-speed samplers are used in most of the modern electronic systems [8]. These systems, which work on a variety of signals such as speech, medical imaging, radar, and telecommunications, require high-speed samplers, such as high-speed ADCs, to have high bandwidth and significant resolution while working at high frequencies (10s of GS/s). Unfortunately, with the current technology, designing high resolution ADCs is highly challenging at such high frequencies. This is mainly due to the fact that these samplers are required to sample at the Nyquist rate (i.e. at least twice the highest frequency component in the signal) to be able to recover the original signal without any loss. Ideally, each sampling event should result in the signal value at the specific sampling instant. However, in practice, there are two main factors that limit the ADC performance: i) uncertainty in the sampling instant, called jitter, and ii) the finite sampling bandwidth, manifested as a weighted integration over a small time interval around the sampling instant, called aperture [27].

As Figure 3-1 shows, the sampling process is really first multiplying with some signal known as sampler signal, and then low pass filtering. The ideal sampler signal would be a delta train with impulses evenly spaced apart at sampling intervals T_S. The non-ideal sampler signal takes into account jitter effects by allowing the interspacing of the impulses to be uneven. The n-th sampling error is given by the difference of two signal values, respectively taken at times $n \cdot Ts$ and $n \cdot Ts + \epsilon_n$, where ϵ_n is a

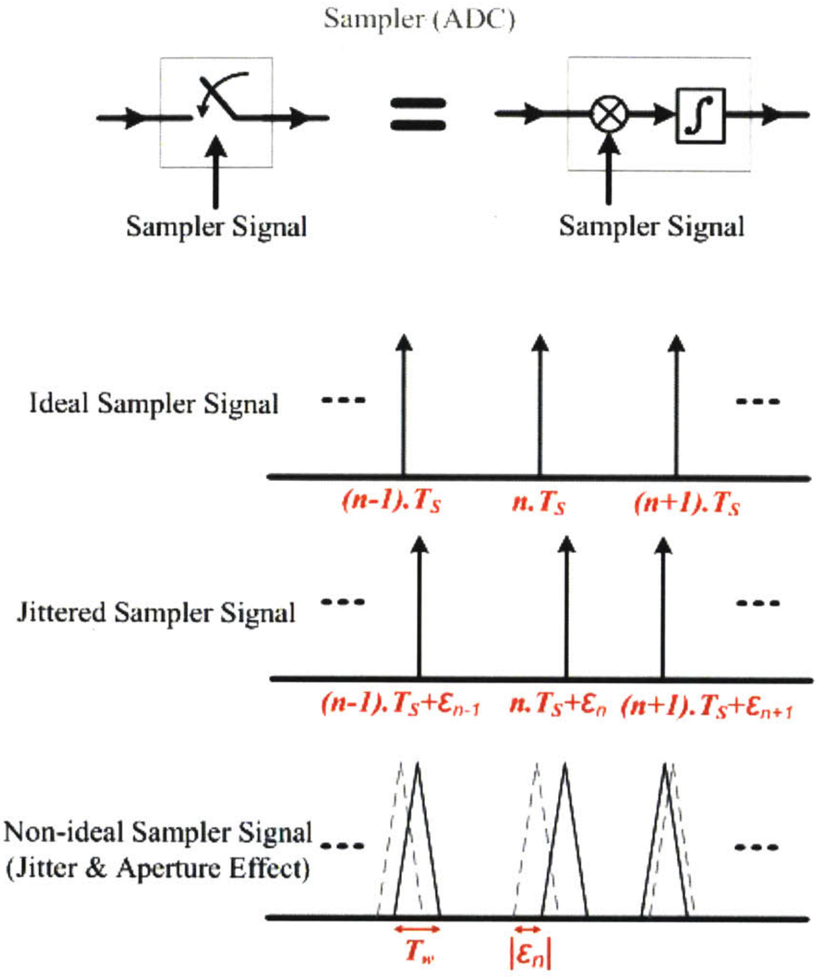

Figure 3-1: Ideal and non-ideal sampler, including jitter and aperture effects.

random variable that represents the n-th jitter value. The jitter effect becomes more serious at higher input signal frequencies, as the signal slew-rate (i.e. rate of change of a signal) is proportional to the signal frequency. Thus, a small jitter can cause a significant error in high-speed sampling. We go on to allow the non-ideal sampler signal to further incorporate aperture effects (in addition to the previously described jitter effects). This is also illustrated in Figure 3-1. We model the aperture effect by replacing the delta impulses in the sampler signal, with triangle pulses, where the area under the triangle is unity. In reality, the aperture in the sampler is caused by two circuit non-idealities: i) low-pass filtering of the sampler (i.e. limited sampler

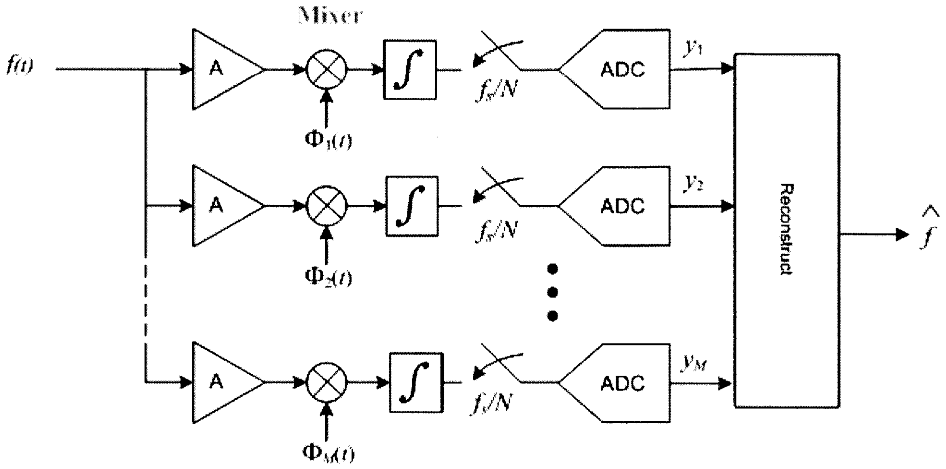

Figure 3-2: Block diagram of an AIC system.

bandwidth in the signal path), and ii) non-negligible rise/fall time of the clock signal (sampling signal). These non-idealities make the sampler band-limited and cause significant error at high frequencies [28].

As it was already mentioned, CS has enabled alternative solutions to high-speed ADCs. A well-known example is the AIC. It has been claimed that these AIC architectures enable high resolution at high frequencies while only using low frequency, sub-Nyquist ADCs [18–26]. In this work, we investigate whether or not AIC systems can indeed resolve both jitter and aperture issues in high-speed samplers, by examining their performance in the presence of these non-idealities.

3.1.2 Analog-to-Information Converter Architecture

While there have been many theoretical discussions on AIC systems in the literature [18–26], to our knowledge, an actual hardware implementation of an AIC system working for wide signal bandwidth (10s of GHz), is yet to be seen. Hence, it is difficult to make a fair hardware-to-hardware comparison with other already implemented high-speed ADCs. In this work, the AIC circuit architecture shown in Figure 3-2 is considered to be compared with a baseline high-speed ADC. In this architecture, the input signal $f(t)$ is amplified by using M number of amplifiers. Each signal branch is then individually multiplied with a different pseudorandom number (PN) waveform $\Phi_i(t)$ to perform CS-type random sampling. The multiplication with the PN waveform is at Nyquist rate to avoid aliasing in this stage, which we call the mixing stage. At each branch, the mixer output is then integrated over a window of N sampling periods T_S. Finally, the integrator outputs are sampled and quantized to form the measurements y_i which are then used to reconstruct the original input signal $f(t)$. Note that because we now sample at the rate f_s/N (see Figure 3-2), this AIC architecture employs sub-Nyquist rate ADCs, which are less affected by jitter noise and aperture. The actual advantage over standard ADCs is really unclear until experimentally justified. Also, it is important to point out that the mixing stage still works at the Nyquist frequency, and circuit non-idealities such as jitter and aperture can still be a potential problem in the mixing stage in a manner similar to the sampling circuit in high-speed ADCs. In the following section, we present our framework for investigating the impacts of mixer jitter and aperture on AIC performance.

3.2 Evaluation Framework

Figure 3-3(a) shows the block diagram of the AIC system indicating the location of injected noise due to the jitter and aperture. Figure 3-3(b) shows the same functionality of the AIC system implemented simply using an amplifier and an ADC operating at the Nyquist-rate (N times that of Figure 3-3(a)). This is the system referred to as the high-speed ADC system, which also suffers from jitter and aperture effects, as illustrated in Figure 3-3(b). The potential advantages of using AICs stem from having a different sensitivity to sources of aperture error and jitter introduced by different control signals in the AIC system. In the AIC system, the jitter error from sampling clocks on the slower ADCs, denoted $n_i'(t)$, is negligible, whereas the main source of error, denoted $n_i(t)$, comes from the mixer aperture and the jitter in the PN waveform mixed with the input signal at the Nyquist frequency. On the other

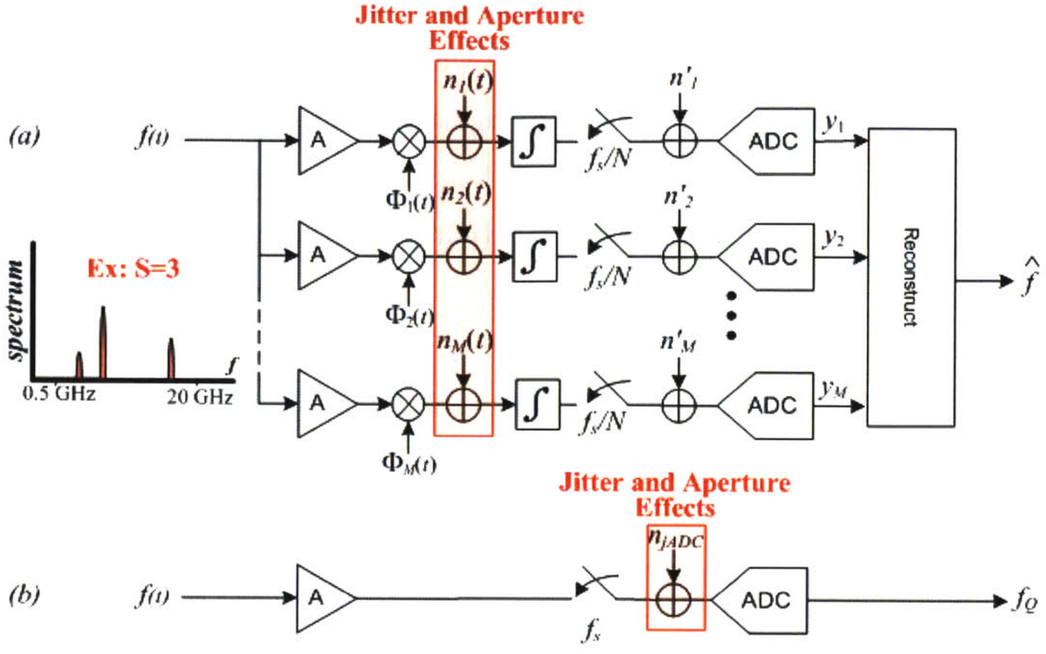

Figure 3-3: Jitter effects in sampling: block diagram of (a) an AIC system, and (b) a high-speed ADC system both with same functionality in the cognitive radio setting.

hand, in the high-speed ADC system, the main source of error is due to the sampling jitter in the high-speed clock. In this section, we provide signal and noise models used to evaluate the performance of these two systems.

3.2.1 Signal Model

The signal model

$$f(t) = \sum_{j=1}^{N_{ch}} x_j \sin(\omega_j t), \tag{3.1}$$

consists of user information coefficients, x_j, riding on the carriers with frequencies ω_j (chosen from N_{ch} available channel frequencies in the range of 500 MHz - 20 GHz). This model emulates sparse narrowband or banded orthogonal frequency-division multiplexing communication channels. Our sparsity assumption states that only coefficients x_j are non-zero, i.e. only $S \ll N_{ch}$ users are active at any one time.

3.2.2 Mixer Clocking Jitter

Figure 3-4 shows our jitter noise model where the noise is multiplied by the input signal and filtered in the integrator block. The i-th PN waveform$\Phi_i(t)$ satisfies:

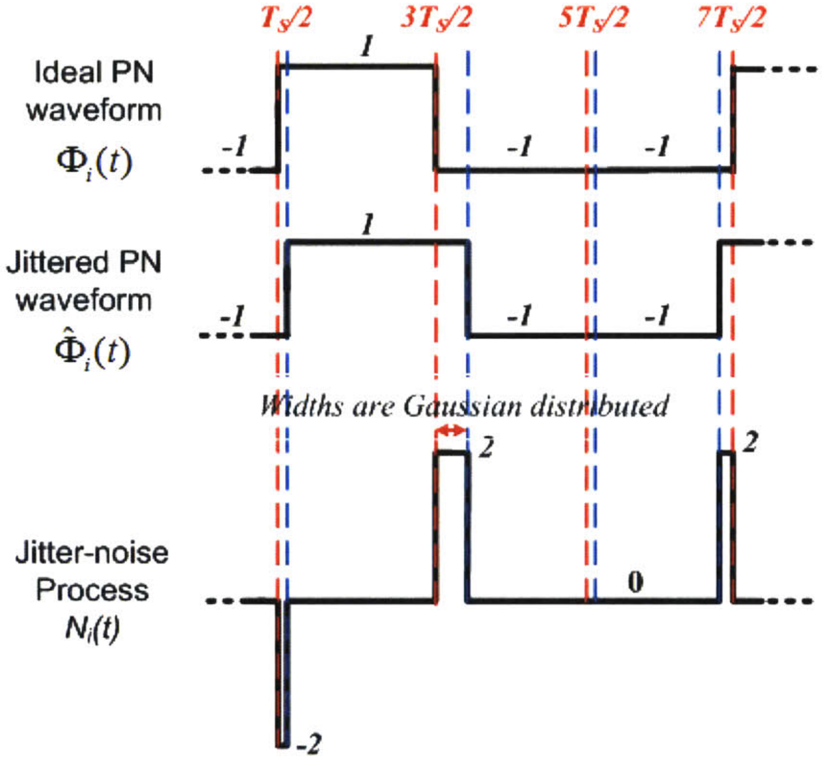

Figure 3-4: Ideal and jittered PN waveforms.

$$\Phi_i(t) = \sum_{j=1}^{N} \varphi_{ij} p(t - jT_s), \tag{3.2}$$

where ϕ_{ij} is the (i,j)-th PN element, and $p(t)$ is a unit height pulse supported on $Ts/2$ to $Ts/2$. Denoting the jittered PN waveform as $\hat{\Phi}(t)$, then: $\hat{\Phi}_i(t) = \Phi_i(t) + N_i(t)$. Here, $N_i(t)$ is the jitter noise affecting $\hat{\Phi}(t)$, described as :

$$N_i(t) = \sum_{j=1}^{N+1} (\varphi_{ij-1} - \varphi_{ij}) \mathrm{sgn}(\varepsilon_j) \hat{p}_j(t - jT_s + \frac{T_s}{2}, \varepsilon_j), \tag{3.3}$$

where the j-th jitter width is $\epsilon_j \sim N(0, \sigma)$ with σ equal to the jitter root-mean-square (rms), and $\hat{p_j}(t, \epsilon)$is a unit amplitude pulse supported over the interval $[min(0, \epsilon), max(0, \epsilon)]$. To verify 3.3, consider the first transition in the i-th PN waveform $\Phi_i(t)$ in Figure 3-4, where ϕ_{i0} and ϕ_{i0} are -1 and 1, respectively. As it is shown, the jitter value ϵ_1 at that transition happens to be positive (i.e. PN waveform is shifted to the right due to jitter). Hence, by using 3.3, the jitter noise $N_i(t)$ at that transition, is a pulse with a width of ϵ and an amplitude of minus two located at $T_S/2$.

As a side comment, note that in our model for $N_i(t)$, we assumed that the same phase-locked loop (PLL) is used across all signal paths, resulting in the exact same jitter sequence ϵ_j for all jittered PN waveforms $\hat{\Phi}_i(t)$, $1 \leq i \leq M$. This model can be extended to include the effect of a longer clock tree distribution, by adding an uncorrelated (or partially correlated) component to each branch, i.e., we would then have a different jitter sequence for each PN waveforms $\hat{\Phi}_i(t)$.

3.2.3 Aperture Models

In the AIC system, the aperture is caused by two circuit non-idealities: i) mixers do not operate instantaneously, and ii) the PN waveforms are not ideal. Figure 3-5 illustrates our aperture error model, whereby the aperture effects are captured by the limited rise and fall times in the PN waveform. The aperture error, $D_i(t)$, corresponding to the i-th non-ideal PN waveform $\tilde{\Phi}_i(t)$, is taken with respect to the i-th jittered PN waveform $\hat{\Phi}_i(t)$, i.e. $\tilde{\Phi}_i(t) = \hat{\Phi}_i(t) + D_i(t)$. We emphasize that the reference point for the aperture error is the jittered PN waveform, not the ideal waveform (as was for the jitter noise $N_i(t)$).

The formula for the i-th aperture error $D_i(t)$ is given as:

$$D_i(t) = \sum_{j=1}^{N+1} (\frac{\phi_{ij} - \phi_{ij-1}}{2}) q(t - jT_s + \frac{T_s}{2} + \varepsilon_j), \tag{3.4}$$

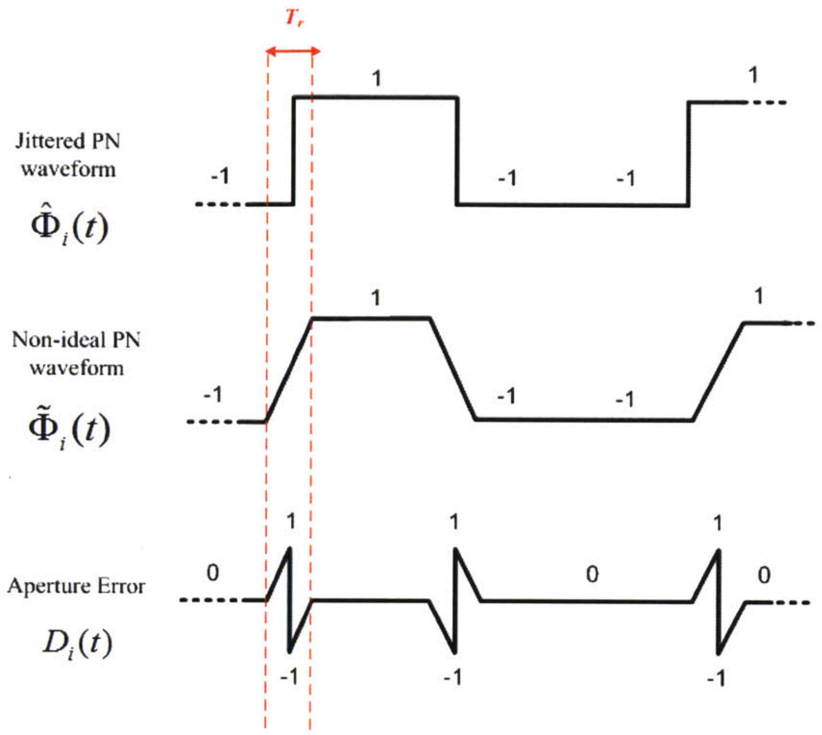

Figure 3-5: Aperture error caused by non-ideal PN waveform.

where ϕ_{ij} is the (i,j)-th PN element, and $q(t)$ can be described as:

$$q(t) = \begin{cases} \left(\frac{2t}{T_r} + 1\right) & -\frac{T_r}{2} < t \leq 0 \\ \left(\frac{2t}{T_r} - 1\right) & 0 < t < \frac{T_r}{2} \\ 0 & otherwise \end{cases} \tag{3.5}$$

where T_r is the parameter that dictates the rise/fall time of the PN waveform. Similar to the jitter noise, the aperture error $D_i(t)$ is also multiplied by the input signal and filtered in the integrator block.

3.2.4 Reconstruction of Frequency Sparse Signal

Using the described CS framework in chapter 2 , we now frame the reconstruction problem for the AIC. As Figure 3-3(a) shows, each measurement y_i is computed by integrating the noise,$n_i(t) = f(t) \cdot (N_i(t) + D_i(t))$, and the product of the signal $f(t)$ and the PN waveform $\Phi_i(t)$, as follow:

$$y_i = \int\limits_{T_s/2}^{N \cdot T_s + T_s/2} f(t) \cdot \Phi_i(t) dt + \int\limits_{T_s/2}^{N \cdot T_s + T_s/2} n_i(t) dt. \tag{3.6}$$

Substituting the signal model from 3.1, the measurements can be shown to satisfy $y = \Phi \Psi x + n^o$, where PN matrix Φ has entries ϕ_{ij} and

$$\Psi_{ij} = \int\limits_{(i-1).T_s + T_s/2}^{i.T_s + T_s/2} \sin(\omega_j t) dt \tag{3.7}$$

$$n^o_i = \int\limits_{T_s/2}^{N \cdot T_s + T_s/2} n_i(t) dt = \int\limits_{T_s/2}^{N \cdot T_s + T_s/2} f(t) \cdot (N_i(t) + D_i(t)) dt \tag{3.8}$$

where $n^o = (n^o_1, n^o_2, ..., n^o_M)^T$. Here, the noise n^o_i is merely the projection of $f(t)$ by the i-th jitter noise pulse process $N_i(t)$ and i-th aperture error pulse $D_i(t)$(see Figure 3-4 and Figure 3-5).

In the next section, we use our noise model and reconstruction framework to compare the performance of AIC versus high-speed ADC systems.

3.3 Evaluation Results

For our signal $f(t)$, refer to model 3.1, we assume 1000 possible subcarriers (i.e. $N_{ch} = 1000$). We test our system using a randomly generated signal $f(t)$, where S non-zero values are drawn from a uniform random distribution over [0, 1] to assign the information coefficients x_i, and S integer values are drawn from a uniform random distribution over [1, N_{ch}] to assign subcarrier (channel location) of S active users.

To compare the performance of the high-speed ADC and the AIC systems, we adopt the same ENOB metric from ADC literature, which is defined as:

$$ENOB = \log_2 \left(\frac{V_{swing}}{\left| f - \widehat{f} \right|_2 \cdot \sqrt{12}} \right), \tag{3.9}$$

where V_{swing} is the full-scale input voltage range of the ADCs and $\left| f - \hat{f} \right|_2$ is the rms signal distortion (use f_Q in place of $\hat{f}$ for the high-speed ADC system in Figure 3-3(b)). In order to illustrate the relative impact of jitter and aperture, we first ignore aperture effects, and limit our evaluation results to only jitter limited systems. We later add aperture effects to the jitter noise, and observe the differences.

3.3.1 Jitter-limited ENOB

The jitter-limited $ENOB$ for both systems is plotted in Figure 3-6. As the number of non-zero components of x, S, increases, we see that the AIC performance worsens while the high-speed ADC performance improves. The reasons for this are as follows. In the receiver, the input signal $f(t)$ peaks are always normalized to V_{swing}, the full-scale voltage range of the ADC. When S increases, this normalization causes the coefficient values $|x_j|$ to get smaller with respect to Vswing. In the high-speed ADC system, the jitter-error is dominated by the coefficient $|x_j|$ corresponding to the highest input frequency and the error drops if the coefficient value drops. Hence, ENOB increases since increases with S, see 3.9. On the other hand, the AIC system has a different behavior. As S increases, the reconstruction performs worse and as a result AIC distortion gets worse, resulting in poorer $ENOB$ performance. As shown in Figure 3-6, when we consider only the impact of jitter, the AIC system can improve the ENOB by 1 and 0.25 bits for S of 1 and 2, respectively. For signals with higher S, the high-speed ADC performs better than the AIC system. As a point of reference, the standard Walden curve [7] is also plotted in Figure 3-6, which depicts the ADC performance with input signal at Nyquist frequency. We see that compared to the Walden curve, the high-speed ADC can actually achieve a better resolution

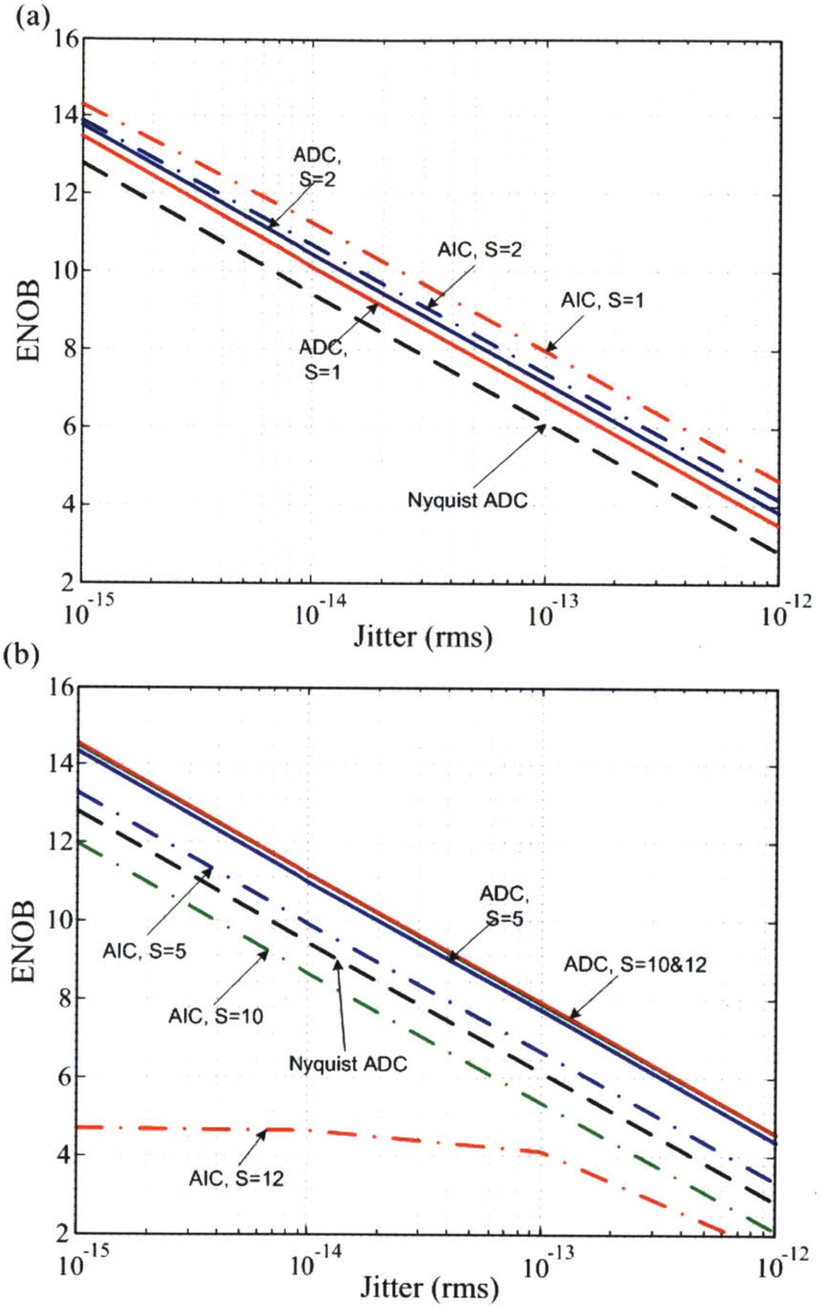

Figure 3-6: Jitter (rms) versus ENOB for (a) S = 1, 2, and (b) S = 5, 10, 12, (N=1000, M = 100 for all S).

(i.e. the Walden curve is a pessimistic estimate). This is due to the fact that the input signal, $f(t)$, does not always have all its spectra concentrated at the Nyquist frequency, and therefore, hence in the average-case, the performance of high-speed ADC is much better than the worst-case prediction of the Walden curve.

3.3.2 Effect of Aperture

So far, we assumed that both the mixer and the ADC have unlimited bandwidth, i.e. we ignore the aperture effects. However, in practice, they are indeed band-limited, and this non-ideality may significantly impact their performance at high frequencies. Figure 3-7 (a) shows the effect of aperture on the performance of both the AIC and the high-speed ADC when $S = 2$. The high-speed ADC system performance is plotted for T_w value of 5 picoseconds (ps) and 10 ps, recall that T_w stands for the integration period in the ADCs (i.e. width of the triangle in the sampler signal, see Figure 3-1). We chose T_w of 10 ps and 5 ps, as they are equivalent to ADCs with 64 GHz signal bandwidth (i.e. about three times of highest input signal frequency) and 128 GHz signal bandwidth. As Figure 3-7 shows, aperture can worsen the performance of the high-speed ADC system when the jitter is really small. However, as the jitter becomes bigger, jitter becomes the dominant source of error, diminishing the aperture effect. To compare, the AIC performance is plotted in the same figure for $T_r = 5$ ps and $T_r = 10$ ps, where here T_r is the rise/fall time in PN sequence waveform, see Figure 3-5. The rise time of $5 - 10$ ps is consistent with the performance of a state-of-the-art PN sequence generator [28,29]. As Figure 3-7 shows, aperture in the mixer stage can also significantly worsen the performance of the AIC system. For example, for the case that $T_r = 5$ ps and jitter (rms) $= 10^{-14}$, the aperture caused the AIC performance (ENOB) to drop from 11 bits to 6 bits. Finally, we perform the same evaluation for higher number of nonzero signal components $S = 10$, as shown in Figure 3-7(b). Similar jitter and aperture limitations are also observed at the higher S value. However, as S increases, the performance of the AIC system worsens due to reconstruction limitations.

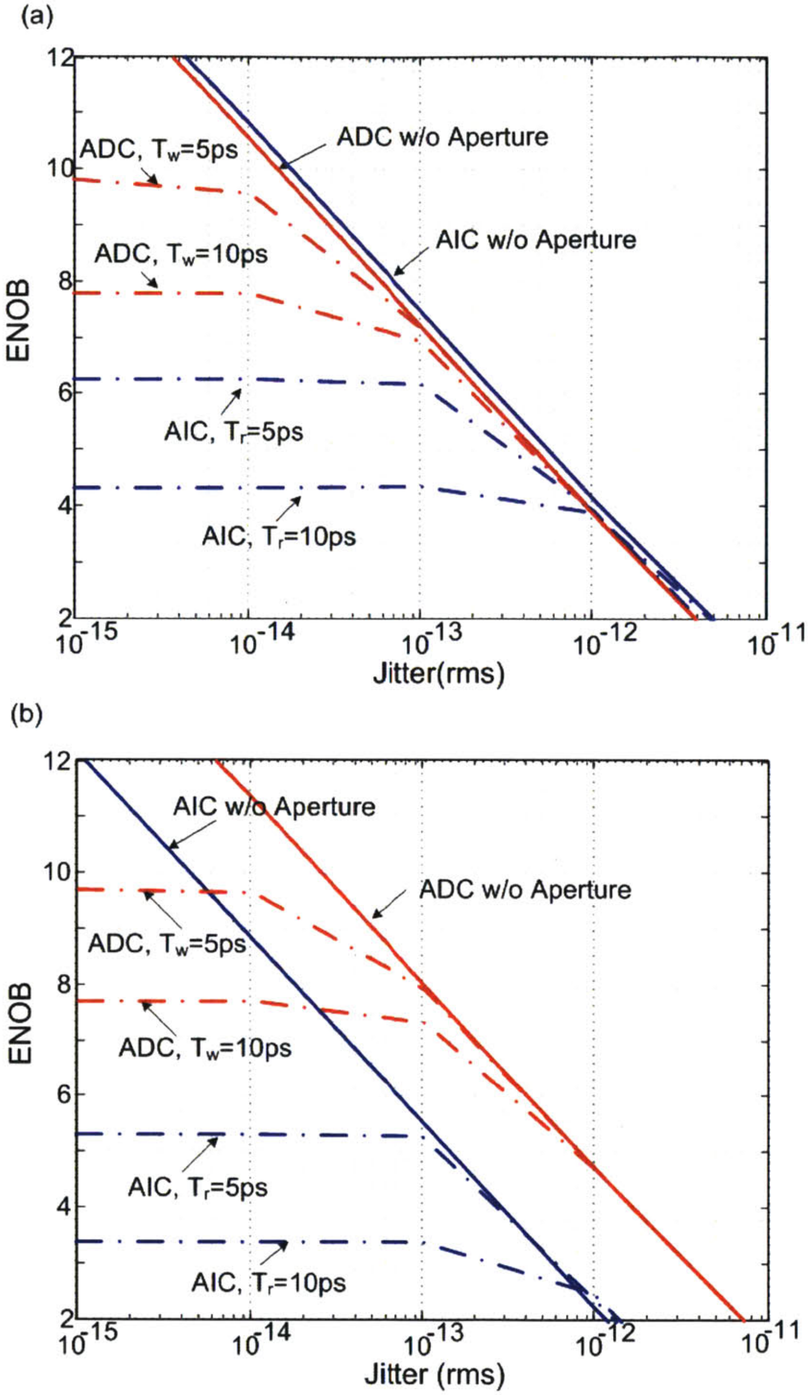

Figure 3-7: Performance of the AIC system versus the high-speed ADC system including aperture and jitter effects for (a) S = 2, and (b) S = 10 (N=1000, M = 100).

3.3.3 Sparse Sampling Matrix

Dense sampling matrices (that mix the input signal) are commonly used for CS based signal-acquisition systems. However, sparse matrices are also a viable option [30] , whereby using sparse matrices can potentially relax memory requirements. Another potential benefit of sparse matrices is that the injected jitter noise at the mixer stage becomes smaller and it may potentially improve AIC performance. This is due to the fact that jitter occurs only when a transition occurs in the sampling waveform, and waveforms made from sparse matrices have fewer transitions. Figure 3-8 shows sampling waveforms generated from dense and sparse matrices.

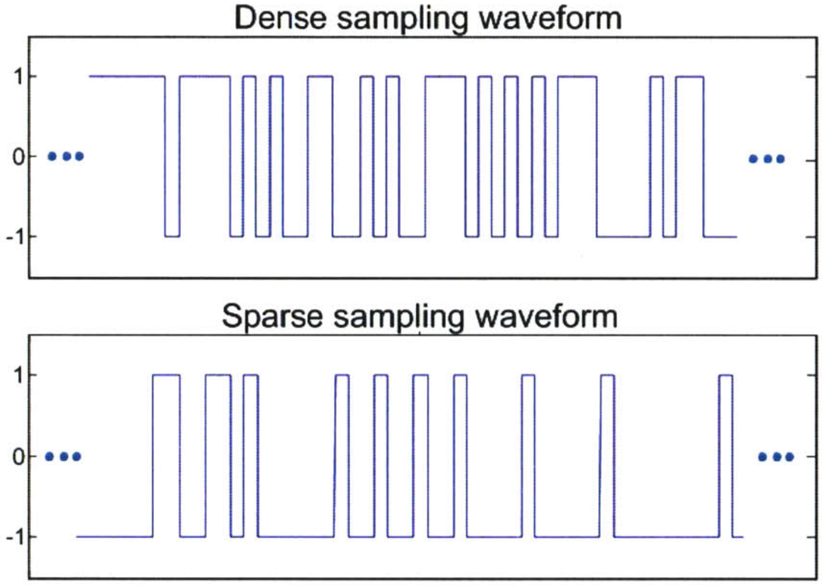

Figure 3-8: Dense sampling waveform versus sparse sampling (the latter has roughly 60% less transitions).

In this section, we examine whether or not sparse matrices can really allow the AIC system to be more jitter-tolerant. We used a sparse sampling matrix Φ we generated similar figures to 3-6 and 3-7, and our findings were that empirical results did not improve at all. This is due to the fact that even though the sparse waveforms made the noise smaller, they also made measurements y_i smaller, and as a result the measurement SNR is not improved at all and AIC performance stays the same.

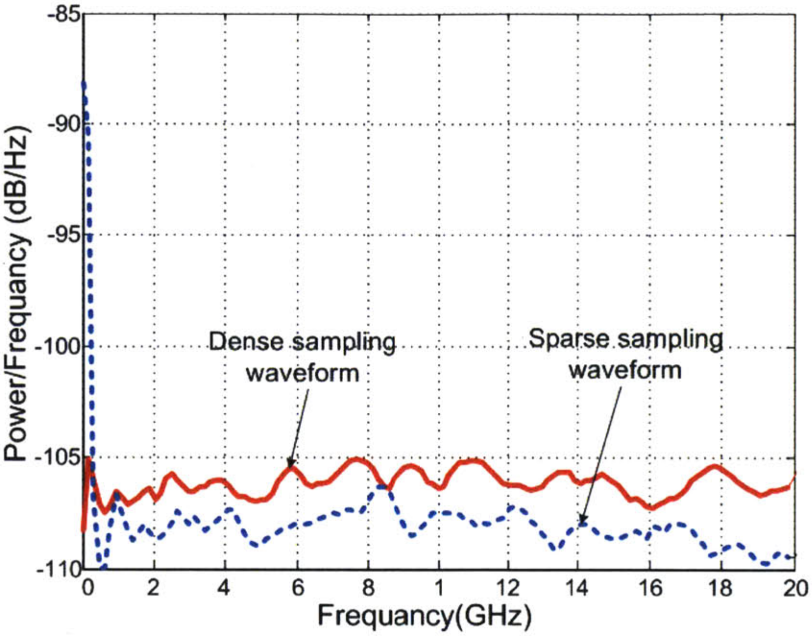

Figure 3-9: Power spectral densities of both dense and sparse sampling waveforms.

For more intuition, consider a high frequency, pure tone input, and some sampling waveform. In the frequency domain, the spectrum of the sampling waveform convolves with that of the single tone (at high frequency), and a shifted version of the sampling waveform spectrum will be created. The integration block attenuates high frequency and only passes the spectrum of the (shifted) sampling waveform that is located near DC. Now the frequency content of the (shifted) sampling waveform near DC, is simply the frequency content of the non-shifted sampling waveform at some high frequency. Hence, only if the original (non-shifted) waveform had large frequency components at that high frequency, then bigger measurements will be seen at the output of the integrator. However, observe Figure 3-9, which plots the power spectrum densities of both sparse and dense sampling waveforms (both waveforms normalized to have the same energy). Notice that at high frequencies, sparse sampling waveforms have lower power than dense sampling waveforms. Hence, the sparse sampling waveforms will generate smaller measurements. In conclusion, sparse sampling matrices will simultaneously degrade both signal and noise and as a result do not improve the AIC performance.

3.3.4 Performance Evaluation Summary

In conclusion, both AIC and high-speed ADC systems suffer from jitter and aperture non-idealities. For the high-speed ADC system, these non-idealities appear in the sampling stage, while for the AIC system, they appear in the mixing stage. Both jitter and aperture are frequency dependent, and since the mixer stage is still required to work at the Nyquist frequency this stage limits AIC performance in high bandwidth applications. To make matters worse, the AIC system performance degrades when the number of signal components, S, increases. This contrasts with high-speed ADC performance, where at a higher S the performance improves (recall that this is due to a different scaling up to V_{swing}). Finally, we also investigated sparse sampling matrices, where we found that while intuition may suggest the opposite, sparse sampling waveforms are still as susceptible to jitter and aperture, as compared to dense sampling matrices.

In the next section, we evaluate and compare the powers of both the AIC and high-speed ADC systems. Recall that AIC systems use slower-rate, sub-Nyquist ADCs, whereby the rate reduction in ADCs will result in some power savings. However, do note that the AIC architecture employs not one, but multiple ADCs, and also requires other circuits such as the integrator and mixer. Hence it is not immediately clear if the AIC system is more power-efficient than the high-speed ADC system. To answer this question, we first provide power models for both high-speed ADC and AIC systems and then we use these models to analyze the relative energy efficiency of both AIC and high-speed ADC systems, across important factors such as resolution, receiver gain and signal sparsity.

3.4 Energy Cost of High-Speed ADC and AIC

In this section, our objective is to present power models of both high-speed ADC and AIC systems, and compare their power-efficinecy for a range of different application parameters. These power models $P_{ADC,sys}$ and $P_{AIC,sys}$ are first given upfront (derivation follows later) as follows:

$$P_{ADC,sys} = 2BW_f \left[\underbrace{FOM \cdot 2^{ENOB}}_{ADC} + \underbrace{3C_1 \cdot G_A^2 \cdot 2^{2ENOB}}_{amplifier} \right], and \tag{3.10}$$

$$P_{AIC,sys} = 2BW_f \left[\begin{array}{l} \underbrace{\frac{M}{N} \cdot FOM \cdot 2^{ENOB}}_{ADCs} + \underbrace{M \cdot C_2}_{integrators} + \\ \underbrace{\frac{M \cdot N}{4} \cdot 3C_1 \cdot G_A^2 \cdot 2^{2ENOB}}_{amplifiers} \\ , \end{array} \right], \tag{3.11}$$

where BW_f is signal bandwidth, FOM is ADCs figure-of-merit (i.e. measuring the power per sample per effective number of quantization step), C_1 and C_2 are technology constants , and G_A is the amplifier gain [6]. The tunable parameters for the AIC system are N, M, $ENOB$, and the gain G_A, while for the high-speed ADC system they are only $ENOB$, and the gain G_A. Note that the gain G_A is set differently for the AIC system (i.e. in 3.11), as compared to the high-speed ADC system (i.e. in 3.10). In the high-speed ADC system, the ADC directly samples the input signal, while in the AIC system the ADCs sample the output of the integrator, which is an accumulated signal (see Figure 3-2). Since the accumulated signal has larger range than the original signal, the required amplifier gain G_A to accommodate the ADCs input range is potentially much lower in the AIC system than in the high-speed ADC system, for the same application. It should be noted that the required gain G_A depends on the application and the signal of interest.

Beside difference in G_A, the main difference between the power of the AIC system,

$P_{AIC,sys}$, and the high-speed ADC, $P_{ADC,sys}$, is an extra factor of $\frac{M \cdot N}{4}$in the AICs amplifier power and an extra factor of $\frac{M}{N}$ in the AICs ADCs power. The reason for these extra factors is described later in this section. It should also be noted that ADCs, utilized in the AIC system, have much lower FOM than a high-speed ADC since they work at much lower frequency. In our evaluation, $FOMs$ of 0.5, 1 and 5 pJ/conversion-step are used to represent a range of possible efficiencies for state-of-the-art and future high-speed ADC designs [5] while FOM of 100 fJ/conversion-step is used for AIC system, consistent with the general performance of state-of-the-art moderate-rate ADCs [6].

In the next section, we proceed with the derivation of 3.10 and 3.11. To do this, we build on the power models proposed in [6], with more focus on the noise constraints, as well as emphasizing detailed differences between the power models used in the AIC and the high-speed ADC systems. We then use 3.10 and 3.11 to evaluate and compare the energy costs of both systems.

3.4.1 High-Speed ADC System Power Model

The total power of the high-speed ADC system, 3.10,is simply the sum of the ADC power and the amplifier power.

For the ADC, the power can be expressed as:

$$P_{ADC} = FOM \cdot 2^{ENOB} \cdot 2BW_f, \tag{3.12}$$

where FOM is the ADC figure-of-merit, BW_f is signal bandwidth, and ENOB equals the ADCs resolution [6].

For the amplifier, the minimum required power is typically determined by the input referred noise ($v_{ni,rms}$). Using another figure-of-merit NEF (known as the noise efficiency factor), introduced in [31]

$$NEF = v_{ni,rms} \sqrt{\frac{2I_{amp}}{\pi \cdot V_T \cdot 4k \cdot T \cdot BW_f}}, \qquad (3.13)$$

where I_{amp} is the current drawn by the amplifier, the required power for the amplifier in the high-speed ADC system can then be described by:

$$P_{amp} = V_{DD} I_{amp} = V_{DD} \cdot \frac{NEF^2}{v^2_{ni,rms}} \cdot \frac{\pi \cdot V_T \cdot 4k \cdot T \cdot BW_f}{2}. \qquad (3.14)$$

In addition, here the total output noise of the amplifier needs to be less than the quantization noise of the ADC (see Figure 3-3(b)) which results in the following constraint on the amplifier output noise:

$$v^2_{ni,rms} \cdot G^2_A \leq \frac{V^2_{DD}}{12 \cdot 2^{2ENOB}}, \qquad (3.15)$$

where G_A is the amplifier gain, $ENOB$ is the resolution of ADC, and the ADCs input range is equal to V_{DD}.

Using 3.15 we can obtain a lower bound on the quantity $\frac{1}{v^2_{ni,rms}}$, which we substitute into 3.14 to obtain the minimum power required by the amplifier as:

$$P_{amp} = 2BW_f.12 \cdot 2^{2ENOB} \cdot \frac{G^2_A NEF^2}{V_{DD}} \cdot \pi \cdot V_T \cdot k \cdot T. \qquad (3.16)$$

Hence, using 3.12 and 3.16, the total power of the high-speed ADC system, $P_{ADC,sys}$, equals 3.10.

3.4.2 AIC System Power Model

Figure 3-10 shows a detailed block diagram of a single branch of the AIC system (out of M branches). The total power 3.11 of the AIC system is simply the sum of the ADC power, integrator power, and the amplifier power.

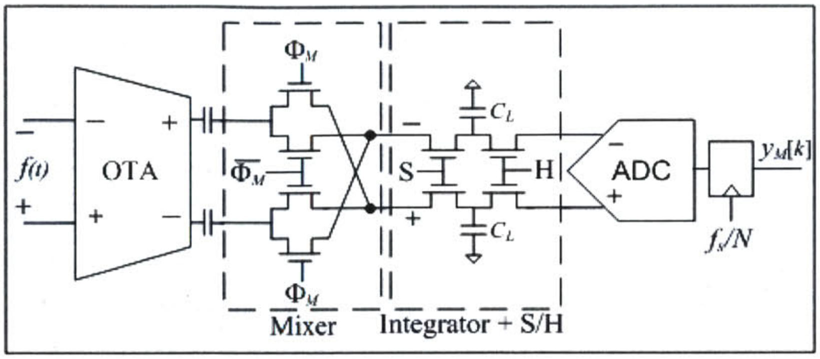

Figure 3-10: Circuit block diagram of an AIC branch taken from [6].

For the ADCs power, we account for M ADCs, each sampling at f_s/N:

$$P_{ADCs} = \left(\frac{M}{N}\right) \cdot FOM \cdot 2^{ENOB} \cdot 2BW_f. \tag{3.17}$$

The integrator power and the power due to switching of the integrator and Sample and Hold (S/H) circuits can be modeled by:

$$P_{\text{int}} = M \cdot V_{DD}^2 \left(\frac{C_L}{16} + C_G\right) \cdot \frac{f_s}{N}, \tag{3.18}$$

where C_L is the integrating capacitor and C_G is the total gate capacitance of the switches where it is negligible compared to C_L (see Figure 3-10). In addition, it is assumed that the common mode reset is at $0.5V_{DD}$ and the voltage swing is $\pm 0.25V_{DD}$. As described in [6], the lower bound on the size of the integrating capacitor (C_L) to functionally act as an integrator can be described by:

$$C_L > 10\pi \cdot N \cdot C_p, \tag{3.19}$$

where C_p is the capacitance at the dominant pole. Combining 3.18 and 3.19, the minimum power required by integrator can be expressed as:

$$P_{\text{int}} = 2BW_f \cdot \frac{M \cdot V_{DD}^2 . 10\pi \cdot C_p}{16}, \tag{3.20}$$

For the operational transconductance amplifier (OTA) power in the AIC system, the expression 3.16 needs to be modified to:

$$P_{amp} = 2BW_f \cdot 3M \cdot N \cdot 2^{2ENOB} \cdot \frac{G_A^2 NEF^2}{V_{DDA}} \cdot \pi \cdot V_T \cdot k \cdot T. \tag{3.21}$$

where 3.21 differs from 3.16 in the appearance of the parameters M and N, and missing a constant factor of 4. With array of M amplifiers in the AIC system

$$P_{amp} = M \cdot V_{DD} I_{amp} = M \cdot V_{DD} \cdot \frac{NEF^2}{v_{ni,rms}^2} \cdot \frac{\pi \cdot V_T \cdot 4k \cdot T \cdot BW_f}{2}. \tag{3.22}$$

As we will explain later on, the constraint of the output noise will now be

$$v_{ni,rms}^2 \cdot G_A^2 \cdot N \leq \frac{4V_{DD}^2}{12 \cdot 2^{2ENOB}}. \tag{3.23}$$

Finally, using 3.23 to get a lower bound on the quantity , we substitute that lower bound in 3.22 to obtain 3.21.

The AIC system requires the total integrated output noise to be less than the quantization noise of the ADC (see Figure 3-3(a)). In the AIC system, we are integrating over N samples modulated by a pseudorandom binary sequence (PRBS) and hence the accumulated noise in the output of integrator increases by a factor of $\sqrt{N}$. Since the total output noise must still be kept smaller than the quantization noise, the input referred noise ($v_{ni,rms}$) needs to be adjusted by a factor of $\sqrt{N}$to keep the total output noise smaller than the quantization noise. Finally, the reason for an extra factor of 4 in 3.23 is because the input of the ADC is differential in the AIC system (see Figure 3-10). Therefore, the input range of the ADC is $2V_{DD}$ differentially which accounts for the additional factor of 4.

We next analyze the energy-efficiency of the two systems using our power models (i.e. 3.10 and 3.11).

3.4.3 Relative Power Cost of AIC versus High-Speed ADC

The AIC system power, $P_{AIC,sys}$ 3.11, is a function of the $ENOB$, M and amplifier gain G_A. As mentioned earlier, the gain G_A needs to be set differently in both AIC and high-speed ADC systems; G_A needs to be set higher in the high-speed ADC system, whereby the relative ratio between the gains depends on application and the signal of interest. For example, in our cognitive radio setup, the relative ratio between G_A gains is about 20. Figure 3-11 plots the total power 3.10 and 3.11 versus ENOB, for both high-speed ADC and AIC systems, and also for different M in our cognitive radio setup. In Figure 3-11(a), we compare system power for relatively small gain scenario (large input signal) where G_A is set to 40 and 2 for the high-speed ADC and the AIC system, respectively. In Figure 3-11(b) we investigate a higher required gain scenario (small input signal) where G_A is set to 400 and 20, respectively. Note that although the power costs are plotted over a wide range of ENOB, high ENOB values are achievable only when jitter noise and aperture error are very small. As to be expected, when the amplifier gain is low, the AIC power flattens for ENOBs less than 5 to chip thermal limits. since the power is dominated by the integrator power (independent of ENOB). For higher resolutions (i.e. higher ENOB), the amplifier power becomes dominant in the AIC system, since it depends exponentially on ENOB. The main takeaway from Figure 3-11 is that at lower gain requirements and low to moderate resolutions (4-6 ENOB, which are also achievable for practical jitter and aperture values), AICs have the potential to be 2-10X more power-efficient than high-speed ADCs. Figure 3-11 also shows that increasing M increases the AIC power, as the number of components scales upwards with increasing number of measurements. However, increasing M also improves the CS reconstruction, which enables higher ENOB for larger S in the AIC system. Finally, note that the grayed areas in the plots show impractical regions due to chip thermal limits.

To get a sense of potential AIC advantages in other applications, we consider different gains (and also relative ratios between gains) for both AIC and high-speed

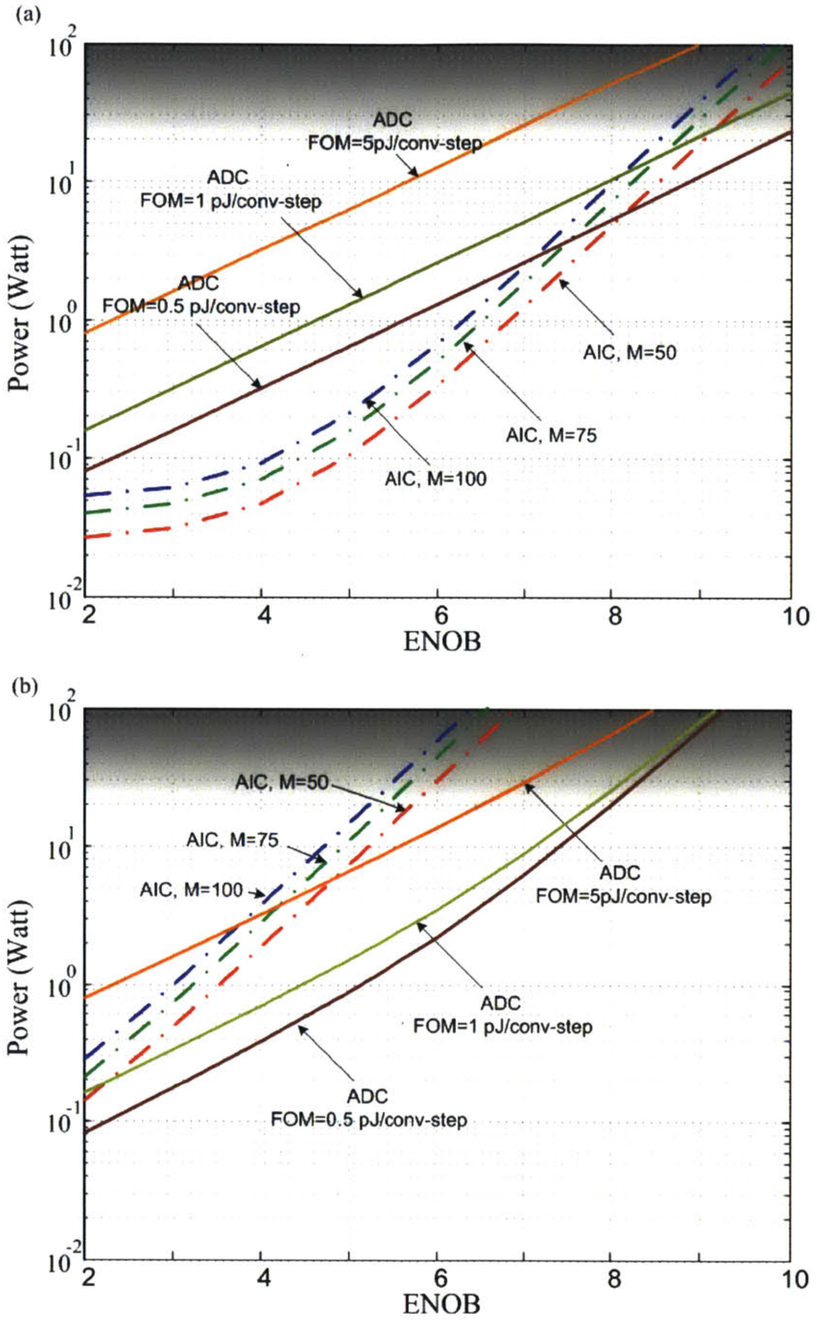

Figure 3-11: Power versus required ENOB for an application which require (a) low amplifier gain (GA=40 in the high-speed ADC system and $G_A = 2$ in the AIC system), and (b) high amplifier gain (G_A=400 in the high-speed ADC system and $G_A = 20$ in the AIC system).

ADC systems. Figure 3-12 shows the power of both systems versus ENOB for different values of gain G_A when $M = 100$. Note that both the systems have different dependence on amplifier gain G_A. For the AIC, the power increases as G_A increases, but on the other hand, the high-speed ADC power changes very little since the power of the single amplifier is not dominant. However, for a high-speed ADC with a very low FOM, amplifier power becomes dominant for high ENOB and as a result the high-speed ADC system power increases with G_A. In conclusion, the AIC system has lower power/energy cost and enables roughly 2-10X reduction in power for applications that require low amplifier gain and low to moderate resolution.

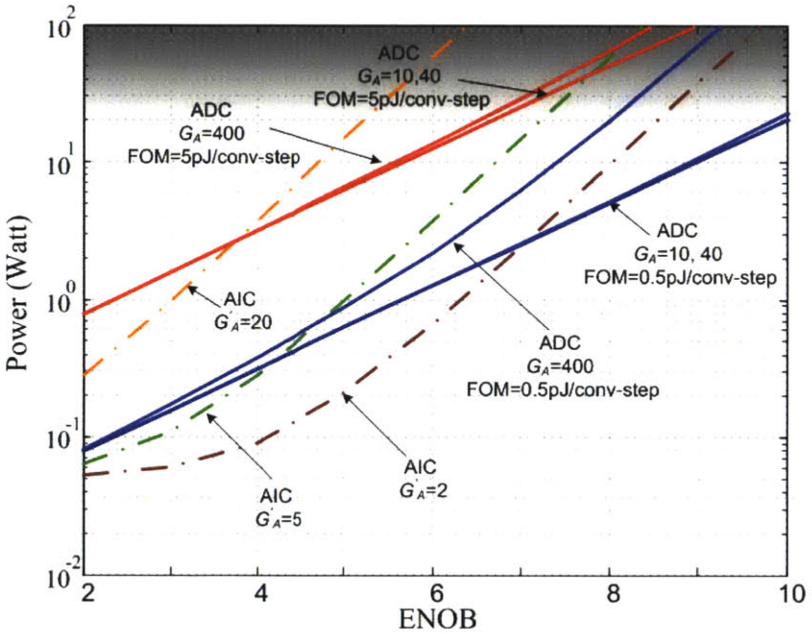

Figure 3-12: Power for the required ENOB and different receiver gain requirements, $N = 1000$ and $M = 100$.

3.5 Summary

In this chapter, we compared both energy cost and performance limitations of AIC and high-speed ADC systems, in the context of cognitive radio applications where the input signal is sparse in the frequency domain. Our findings report that jitter and aperture effects in the mixing stage of AIC systems limit their resolution and performance. Hence, this contests the proposal that AICs can potentially overcome the resolution and performance limitations of sampling jitter and aperture error in high-speed Nyquist ADCs. We show that currently proposed AIC topologies have no advantage over high-speed ADCs, and are actually even more sensitive to jitter and aperture errors. We also show that sparse matrices do not improve the resolution performance of AIC. Finally, using realistic power models for both AIC and high-speed ADC systems, we show that AICs have the potential to enable a 2-10X reduction in power for applications where low signal gain and low to moderate resolution are acceptable.

Chapter 4

Hardware Implementation of a CS Decoder

In the previous sections, we examined the performance of CS-based systems in different applications. We showed that such systems can enable higher power-efficiency in communications systems. However, due to the complexity of the decoder/reconstruction algorithm, signal reconstruction is slow, and, as a result, these systems might not be suitable for real-time applications. Thus, the hardware solution has become crucial for higher performance and cost effectiveness. Hardware implementation of the decoders/algorithms can enable higher power-efficiency and also provide higher speed by using parallel/pipelined architecture. In this thesis, we designed an end-to-end system evaluation framework to evaluate the performance of different reconstruction algorithms. Then, we evaluated them in terms of performance (reliability), implementation scalability, speed and power-efficiency. Finally, by considering these factors, we chose a suitable reconstruction algorithm to implement.

4.1 Reconstruction Algorithms

Several different schemes have been proposed for solving the l_1-minimization. However, we chose to compare the performance of only three well-known algorithms: Matching Pursuit (MP), Orthogonal Matching Pursuit (OMP), and Approximate Message Passing (AMP). We chose MP since it is known for its speed and relative simplicity, and we chose OMP and AMP since they are known for their performance. In general, there is a trade-off between performance and complexity of the algorithm. Here, MP has the lowest complexity while the OMP and AMP are more complex and have longer running time for each iteration. For example, OMP requires matrix inversion, and AMP requires square-root computation and a significant number of matrix multiplications in each iteration which make them more complex to implement in hardware compared to MP. On the other hand, OMP and AMP require a smaller number of iterations to reconstruct the signal, and they also have better performance.

Among these algorithms, AMP has the best performance as it performs the same as l_1 minimization. However, AMP has higher complexity and longer running time compared to the other two algorithms which make it unsuitable for some applications which require high speed/throughput reconstruction. For example, in the AIC system used in a cognitive radio system, MP might be a better option as this application requires fast reconstruction. In addition, it does not require a very precise reconstruction, and it has higher tolerance to the reconstruction error. Therefore, the performance of these algorithms should be compared in terms of both speed and quality of reconstruction in practical environments to decide which one would be a better choice for a specific application.

4.1.1 Matching Pursuit

Matching Pursuit (MP) is an iterative greedy algorithm that finds the sparse solution x subject to $y = \Phi\Psi x$, where Ψ is the basis matrix, Φ is the sampling matrix, and y is measurements [32]. A flow chart of the algorithm is shown in Figure 4-1, where r_t denotes the residual/error at t-th iteration. MP iteratively improves its estimate of the signal by choosing the column of the matrix A (where $A = \Phi\Psi$) that has the highest correlation with the residual r_t. Then, it subtracts the correlated column from the approximation error/residual and then iterates the procedure on the newly obtained approximation error. The algorithm stops if the norm of the residual falls below a threshold, or if the number of iterations (k) reaches to the limit L. Note that even if we perform M iterations of MP, it is not guaranteed that we will obtain an error of zero, though the asymptotical convergence of MP for $k \to \infty$ has been proven [32].

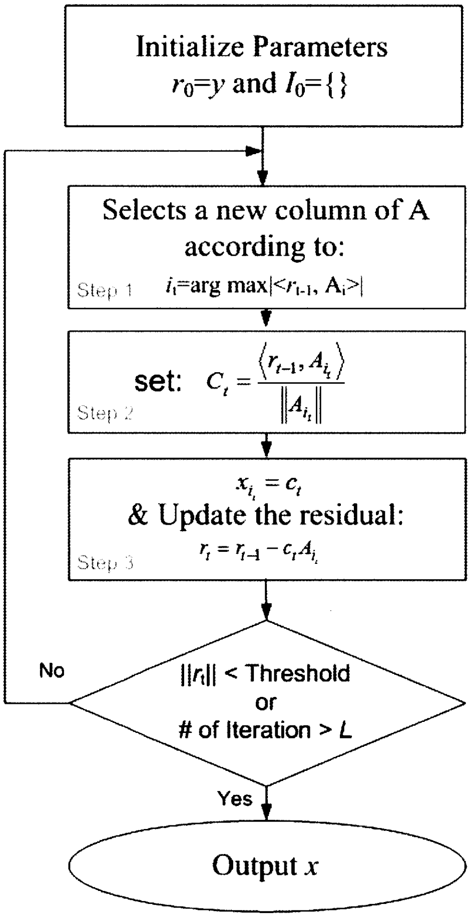
Initialize Parameters
r_0=y and I_0={}
Selects a new column of A according to:
Step 1
i_t=arg max|<r_{t-1}, A_i>|
set: $c_t = \frac{\langle r_{t-1}, A_{i_t} \rangle}{\|A_{i_t}\|}$
Step 2
$x_{i_t} = c_t$
& Update the residual:
$r_t = r_{t-1} - c_t A_{i_t}$
Step 3
No
||r_t|| < Threshold
or
# of Iteration > L
Yes
Output x

Figure 4-1: Flow chart for MP algorithm.

4.1.2 Orthogonal Matching Pursuit

Figure 4-2 shows the flow chart for OMP [33] . The selection criterion of the OMP algorithm is the same as the MP algorithm. The main difference between MP and OMP is in their projection step. At each iteration of MP, the measurements are projected on only selected column of matrix A, and, hence, the residual may not be orthogonal to the subspace span by A_i unless the columns of A are orthogonal to each other. In contrast to MP, at each iteration of OMP, the measurements y are projected on the range of all previously selected columns of matrix A, and, as a result, the newly derived residual is orthogonal not only to the immediately selected column, but also to all the columns selected at previous iterations. As a consequence, once a column is selected, it is never selected again in subsequent iterations.

4.1.3 Approximate Message Passing

Approximate message passing (AMP) has been recently proposed as an effective algorithm to reconstruct a sparse signal from a small number of incoherent linear measurements [34]. It has been shown that this algorithm performs exactly the same as l_1 minimization while it is running fast. AMP flowchart is shown in 4-3 where z_t and x_t are residual and the estimate of the signal at time t (t-th iteration), respectively. Note that $< u >= \sum_{i=1}^{N} u(i)/N$, and $\delta = M/N$. Here,$\eta_t()$ is a threshold function. Many threshold policies have been proposed for AMP. However, we can simply set the threshold to the magnitude of the M-th largest coefficient in absolute value [35].

4.2 Performance Evaluation

Since most CS theory is based on asymptotic bounds, a simulation framework must be established to evaluate the performance of reconstruction algorithms. As we already mentioned, CS can be used for any signal which is sparse in time domain or any other domain. Hence, in this section, we first evaluate the performance of

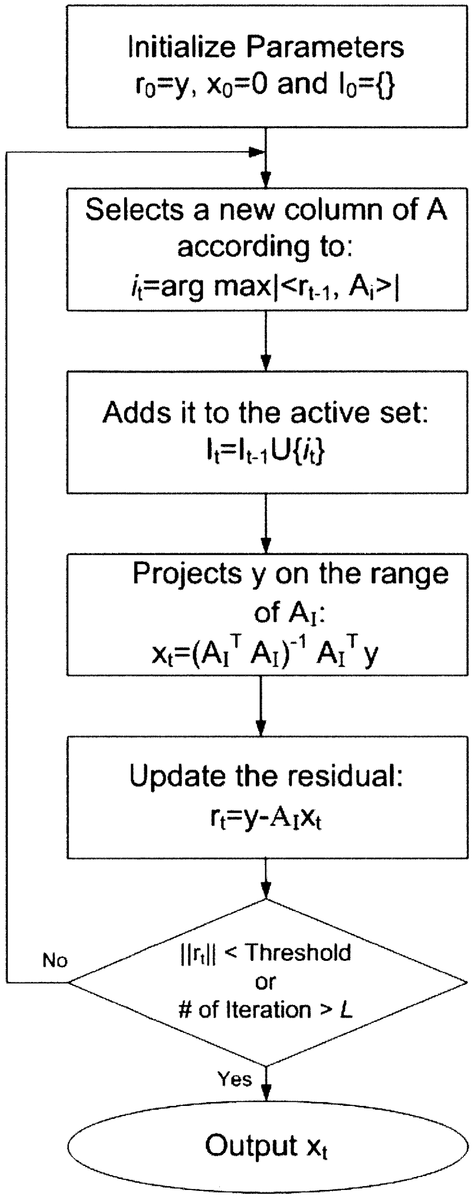

Figure 4-2: Flow chart for OMP algorithm.

MP, OMP and AMP algorithms when the signal of interest is time-sparse. Then, we do the same evaluation for frequency-sparse signal (i.e. signal is sparse in frequency domain) as an example signal which is sparse in other domain.

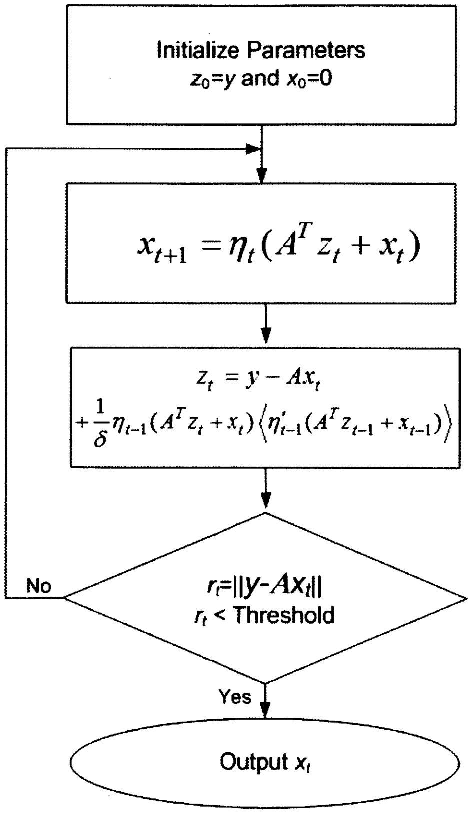
Initialize Parameters
$z_0=y$ and $x_0=0$
$x_{t+1}=\eta_t(A^T z_t+x_t)$
$z_t=y-Ax_t$
$+\frac{1}{\delta}\eta_{t-1}(A^T z_t+x_t)\langle\eta'_{t-1}(A^T z_{t-1}+x_{t-1})\rangle$
No
$r_t=\|y-Ax_t\|$
$r_t <$ Threshold
Yes
Output x_t

Figure 4-3: Flow chart for AMP algorithm.

4.2.1 Time-sparse Application

In this section, we evaluate the performance of MP, OMP and AMP algorithms when signal of interest is time-sparse (i.e. the signal is sparse in time domain). To illustrate the performance comparison results, random signals of length $N = 1000$ with S non-zero coefficients are generated by drawing on a uniform random distribution over [-1,1] to assign the sign and magnitude, and over [1,1000] to assign the position of each non-zero value in the S-sparse signal. The generated signal is corrupted by white Gaussian noise and is quantized to Q bits. Then, the quantized signal is multiplied by a sampling matrix to generate CS measurements. Finally, the CS measurements are used to reconstruct the signal.

We first evaluate the performance of the algorithms in terms of probability of reconstruction failure. Here, we set the failure threshold to PRD of -2dB which corresponds to $ENOB$ of 8 bits. In other words, a reconstruction is called a failure if the reconstructed signal has $PRD > -2dB$ (i.e., resolution of the reconstructed signal is lower than 8 bits). Note, in this example, Q and input noise variance is chosen to give signal PRDs of lower than failure threshold in the original input signal.

Probability of Failure

The probability of failure for MP, OMP and AMP is plotted in Figure 4-4 where the signal of interest is time-sparse and the number of measurements (M) is 50. As S increases, the performance of the algorithms worsens. As figure 4-4 shows, OMP performs better than AMP for input signal with sparsity level (S) higher than 5. Although, here, OMP performs better that AMP for this set-up and some range of sparsity, its performance might become worse than AMP for some other cases

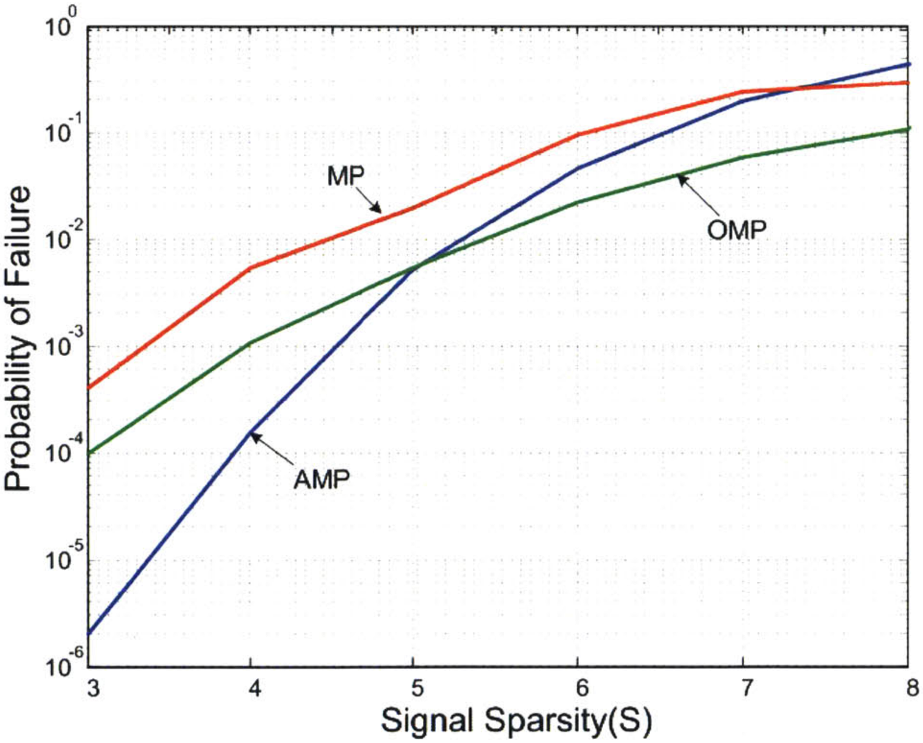

Figure 4-4: signal sparsity versus probability of reconstruction failure for MP, AMP and OMP when the signal of interest is sparse in time domain ($N = 1000$, $M = 50$, Failure threshold set to $PRD = -2dB$).

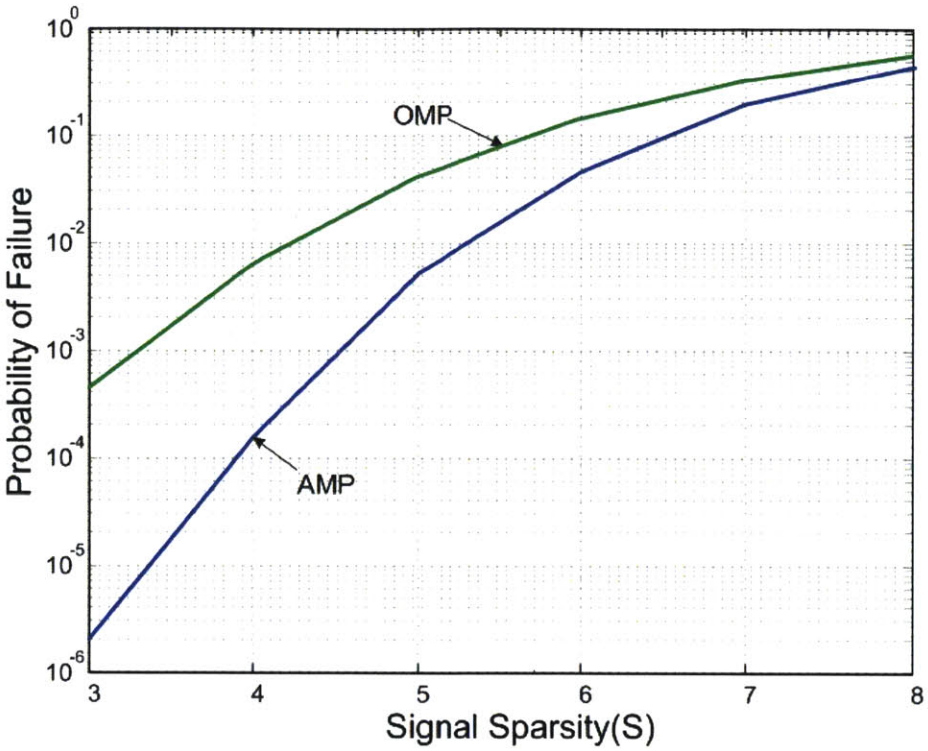

Figure 4-5: signal sparsity versus probability of reconstruction failure for AMP and OMP when the signal of interest is sparse in time domain and is bounded (i.e. $0.5 \leq |x| \leq 1.0$) .

(i.e. different block size, signal and etc). For example, Figure 4-5 shows the same evaluation when the signal of interest is sparse in time domain and is bounded (i.e. $0.5 \leq |x| \leq 1.0$). As it can be seen, in this case, OMP performance is worse than AMP for the whole range of sparsity level. As figure 4-4 shows, the MP algorithm perform worse than both OMP and AMP. However, since MP is less computational complex, it can still be a good choice for applications which require fast reconstruction but does not require very low probability of failure.

Required Number of Measurements

Increasing the number of measurements M can improve the quality of reconstruction. Hence, the other factor which needs to be considered to evaluate the performance of algorithms is the number of measurements, M, required to meet a desired probability of failure. Figure 4-6 shows the required M to achieve the probability of failure of 1% and 0.1%. As it is shown, there is no significant difference in the required M for OMP versus AMP, while MP requires slightly higher number of measurements to achieve the same probability of failure as the other two algorithms.

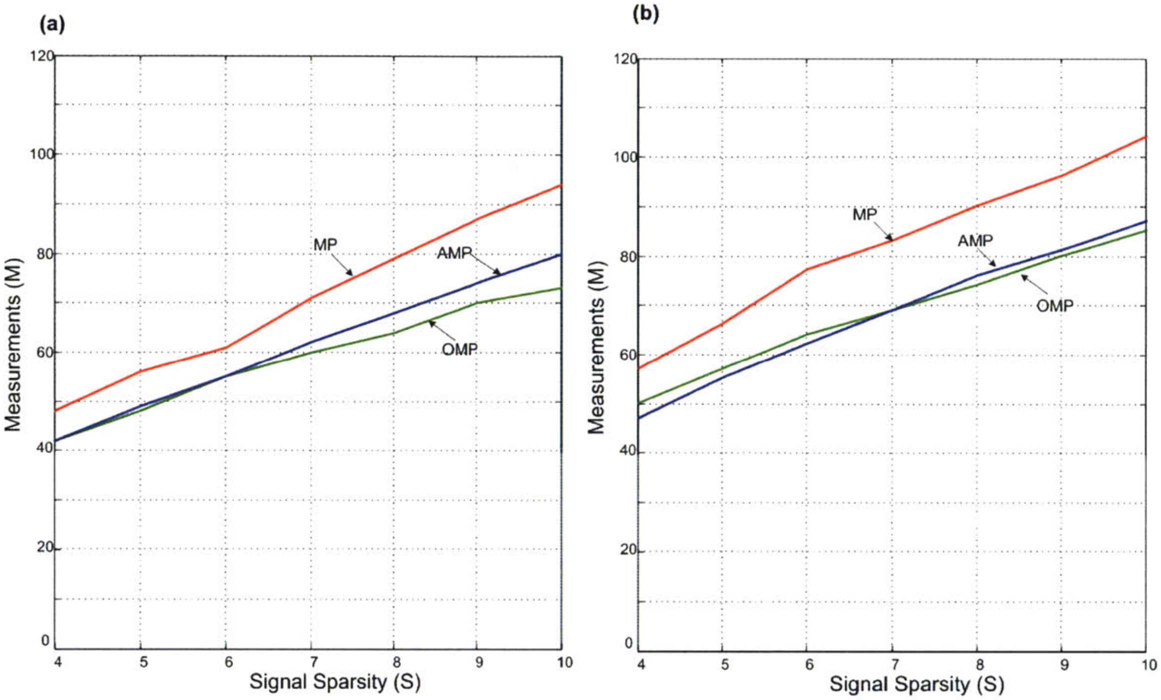

Figure 4-6: signal sparsity versus required number of measurements for MP, AMP and OMP when the signal of interest is sparse in time domain ($N = 1000$); (a)Failure rate=10^{-2} , (b) failure rate=10^{-3}.

4.2.2 Frequency-sparse Application

In this section, we perform the same evaluation as 4.2.1. However, instead, we assume the signal of interest is frequency-sparse (i.e. the signal is sparse in the frequency domain). One example application is the AIC used in cognitive radio. To illustrate the performance comparison results, we use the same signal model explained in Chapter 3. Figures 4-7 and 4-8 show the probability of failure and the required number of measurements curves (like figure 4-4 and 4-6) for frequency-sparse signals. We see

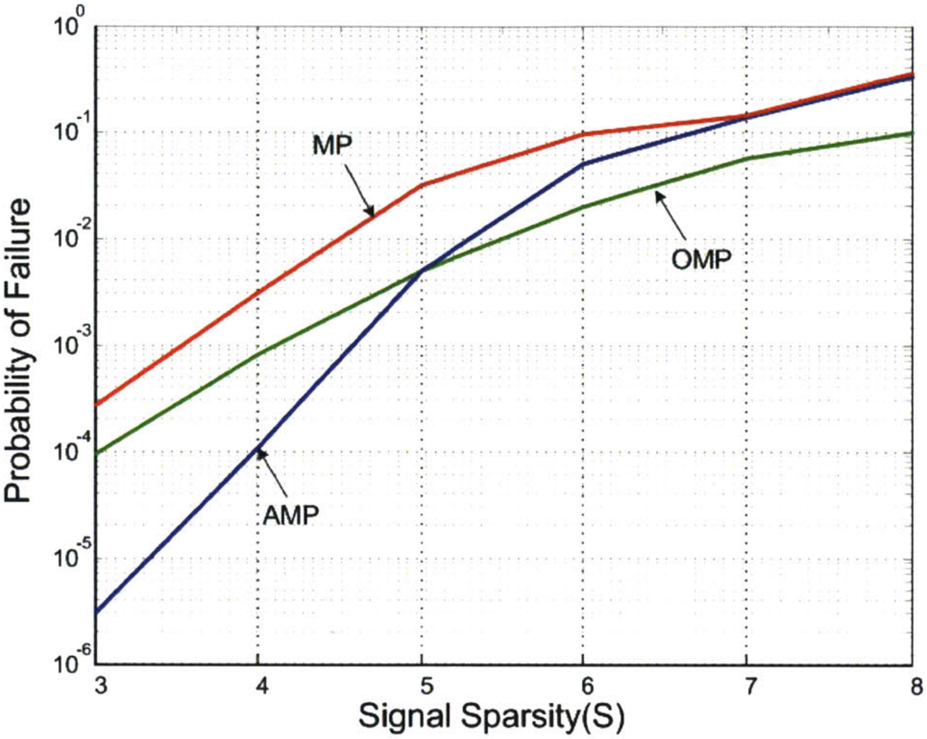

Figure 4-7: signal sparsity versus probability of reconstruction failure for MP, AMP and OMP when the signal of interest is sparse in frequency domain (N=1000, M=50, Failure threshold set to PRD=-2dB).

similar observations as the time-sparse signal in the performance of the algorithms.

4.2.3 Evaluation Summary

In conclusion, depending on the application, one algorithm may be more suitable than the others. For example, for an application which requires fast reconstruction but does not need a very low probability of failure, MP may be a more suitable choice. On the other hand, AMP can provide higher performance at the cost of reconstruction time and higher complexity.

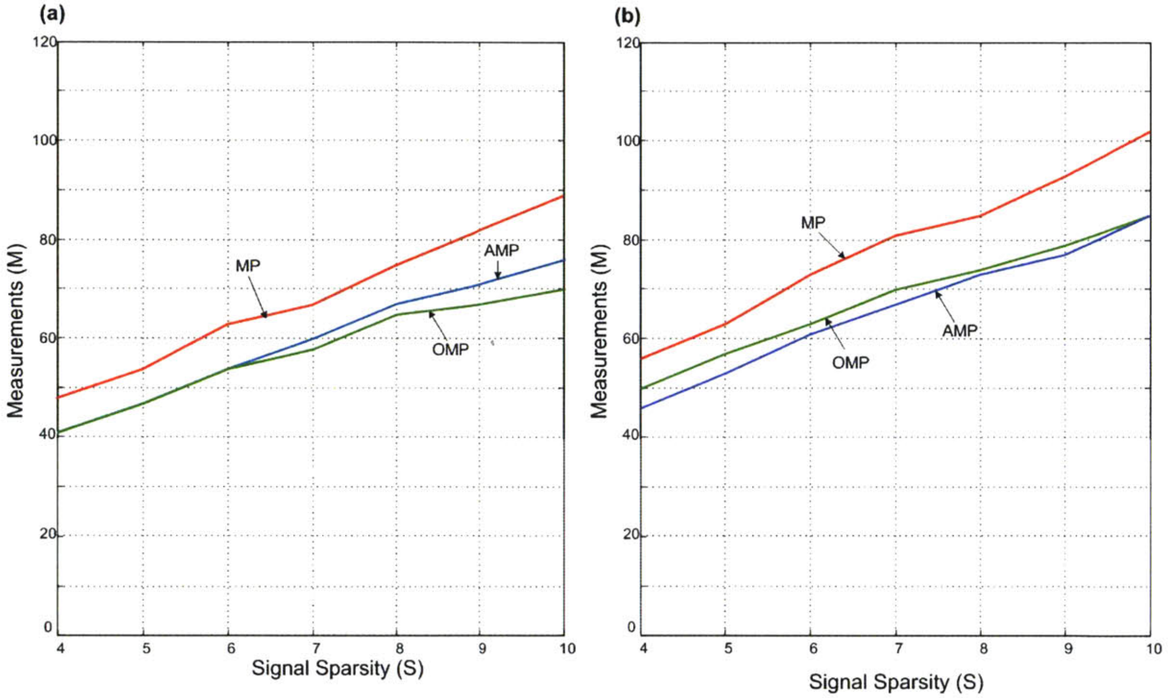

Figure 4-8: signal sparsity versus required number of measurements for MP, AMP and OMP when the signal of interest is sparse in frequency domain ($N = 1000$); (a)Failure rate=10^{-2} , (b) failure rate=10^{-3}.

4.3 MP Implementation

In the previous section, we evaluated the performance of the MP, OMP and AMP algorithms in terms of probability of failure and complexity. We illustrated that the MP algorithm can be a suitable choice for applications which require high throughput without requiring a very low probability of failure. Due to MP's low complexity, the hardware implementation of this algorithm requires less chip area and potentially has lower power consumption. In this section, we first present a very-large-scale-integration (VLSI) architecture for the Matching Pursuit (MP) algorithm for generic applications. Then, we present the implementation result in terms of throughput, chip area and power consumption.

4.3.1 VLSI Design Methodology

To be able to achieve the highest gain (i.e. higher throughput and lower power consumption) from a VLSI implementation of an algorithm compared to its software implementation, the number of utilized parallel/pipelined stages must be optimized. On one hand, maximizing the number of parallel/pipelined stages can potentially increase the throughput of the system, but on the other hand, it has higher power consumption and requires more chip area. In general, there is a trade-off between throughput, power consumption and area in a VLSI implementation.

One well-known approach to reduce the power consumption of a VLSI design is to reduce the supply voltage, but use parallel and pipelined architecture to compensate the performance degradation due to reduced power supply level. Figure 4-9 depicts the design space for a VLSI design and the relationship between throughput, power consumption and area. As it can be seen, for the same supply voltage, using more parallel stages in the design can increase the throughput at the cost of higher power consumption and larger area. This figure also shows that for the same throughput,

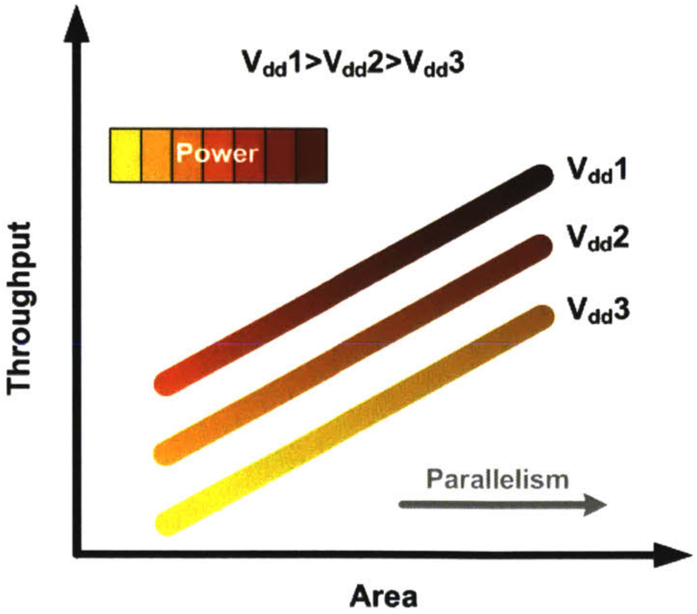

Figure 4-9: VLSI design space.

utilizing more parallel stages allow us to use lower supply voltage and potentially improves the power consumption at the cost of larger area. To understand better these tradeoffs, lets consider a simple hardware which multiplies two vectors of length 10. This hardware can be simply implemented using only one multiplier. Since there is only one multiplier and 10 multiplications are required, the block takes 10 clock cycles to calculate the result of vectors multiplication. However, the throughput of this block can be improved by using multiple multipliers working in parallel. For example, if we use 10 multipliers working in parallel, it takes only one cycle to calculate the multiplication of two vectors. However, the drawback is that we require more chip area, and also power consumption goes higher since multiple multipliers are working simultaneously.

4.3.2 Design Specification

Our implemented architecture can be used for many Compressed Sensing applications. However, here, our aim is to have a decoder which can be used for our discussed AIC system in Chapter 3. This means that we try to design an MP decoder to reconstruct the signal length of 1000 ($N = 1000$) from 100 measurements ($M = 100$). The reported results are based on 10 iterations and ENOB of 8 bits in the recovered signal.

4.3.3 VLSI Architecture

Before designing the MP algorithm in hardware, its performance should be evaluated in the fixed-point arithmetic. This evaluation helps in choosing the resolution (i.e. the number of bits) required in each operation of the MP algorithm to achieve the required ENOB in the recovered signal (in our case, ENOB=8). Our MATLAB evaluation shows that to achieve 8 bits of resolution in the recovered signal, most of the algorithm's operations can be done in 8 bits except the vector correlation which is required in step 1 of the algorithm (see figure 4-1). This operation must be done in at least 12 bits (i.e. each element of matrix A, vector y and vector r must have at least 12 bits of resolution). This is because we are adding the result of 100 ($M = 100$) multiplications to calculate the vector correlation. Since the quantization error of 100 numbers(i.e. 100 products) are added, an extra 3-4 bits (i.e. $log_2\sqrt{M}$) bits are required to achieve ENOB of 8 bits in the correlation value and, consequently, we can achieve 8 bits of resolution in the recovered signal.

As shown in the MP algorithm flowchart (see figure 4-1), this algorithm includes 3 main steps. Each step can be designed as a separate block when they work in series. It should be noted that these blocks cannot be pipelined since the input of the first step at each iteration depends on the output of the previous iteration. However, noting that the MP algorithm requires many multiplications in each step which can be executed in parallel, each step can be implemented using multiple parallel stages. In order to optimize the throughput of the system, we first need to evaluate the number of operations (multiplications) required in each step. The first step calculates the correlation of the residual vector r with each column of the measurement matrix A. In other word, it executes $N \cdot M$ multiplications. The second block (step) requires only one multiplication since we use a LUT for the division operation, and finally the last block requires M multiplications. It is evident that, the first block is the bottleneck of the system as it requires a lot more multiplications compared to the other blocks. For example, for the case where $N = 1000$ and $M = 100$, more than 99% of all required operations to reconstruct the signal are done by the first block. Hence, utilizing parallel stages in the first stage can significantly improve the throughput of the system.

As previously discussed, a VLSI design can be implemented using different numbers of parallel stages. However, not all of them are optimized in terms of power consumption and area. In the MP decoder, the design of the first block (which requires $N \cdot M$ multiplications) can be done using only one multiplier which takes $N \cdot M$ clock cycles to compute the correlation of all columns of matrix A with vector r. It can also be implemented by using up to $N \cdot M$ multipliers. If we use $N \cdot M$ multipliers working in parallel, it takes only one cycle to calculate the correlation of all columns of the matrix A with the vector r. However, using that many multipliers will require significant chip area. In this project, in order to outline the potential design space and find the optimal number of parallel stages, we explore three different designs. In each design, we use a different number of parallel stages, and we evaluate them in terms of required area, power and throughput. The first design employs only one multiplier, the second design uses 10 (i.e. $M/10$) multipliers, and the last design uses 100 (i.e. M) multipliers to calculate the correlation between all the columns of matrix A with vector r . It should be noted that the last design requires only one clock cycle to calculate the correlation of two vectors and, consequently, requires N clock cycles to calculate the correlation of all the columns of matrix A with vector r.

Figure 4-10 depicts the high level block diagram of our proposed MP architecture. The design consists of three main units. The first unit (MaxCorrelation Unit) calculates the correlation between each column of the measurement matrix A ($A = \phi \cdot \psi$ where ϕ is the sampling matrix and ψ is the basis matrix) and the current residual r (y for the first iteration). The outputs of this unit are the value of the maximum correlation and the index of the column that has the highest correlation with the residual. The second unit (Normalization Unit) normalizes the value of maximum correlation (i.e. output of MaxCorrelation Unit) by the norm of the selected column. In this block, a divider is required, which is implemented by using a multiplier and a look-up-table (LUT). Finally, the last unit (Update Unit) updates the residual by subtracting the selected column from the previous residual.

Figure 4-11 shows the simplified block diagram of the MaxCorrelation unit. This unit employs the vector correlator module to calculate the correlation of two vectors. Then, it calculates the absolute value of the correlation and compares it with the old value of maximum correlation. Finally, it updates the maximum correlation value if it is needed. As already mentioned, we evaluate the performance of three different designs where each design uses a different number of multipliers working in parallel. Figure 4-12 depicts the block diagram of the vector correlator used in our second and

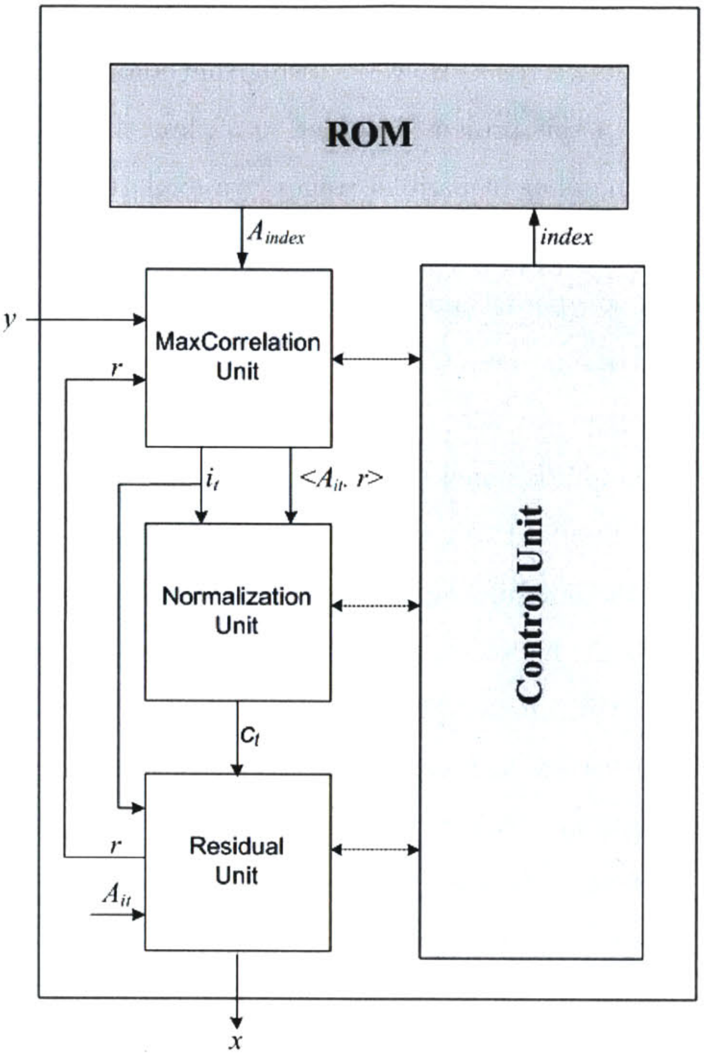

Figure 4-10: High-level block diagram of the MP decoder.

third design. As shown, the second design includes 10 multipliers and, consequently, it requires 10 cycles to calculate the correlation of two vector of size 100. However, the third design includes 100 multipliers and it takes only one cycle to calculate the correlation of two vectors. It should be noted that although the third vector correlator architecture is potentially 100 times faster than first architecture(i.e. vector correlator with one multiplier), it also requires an area which is approximately 100 times larger, and consumes more power.

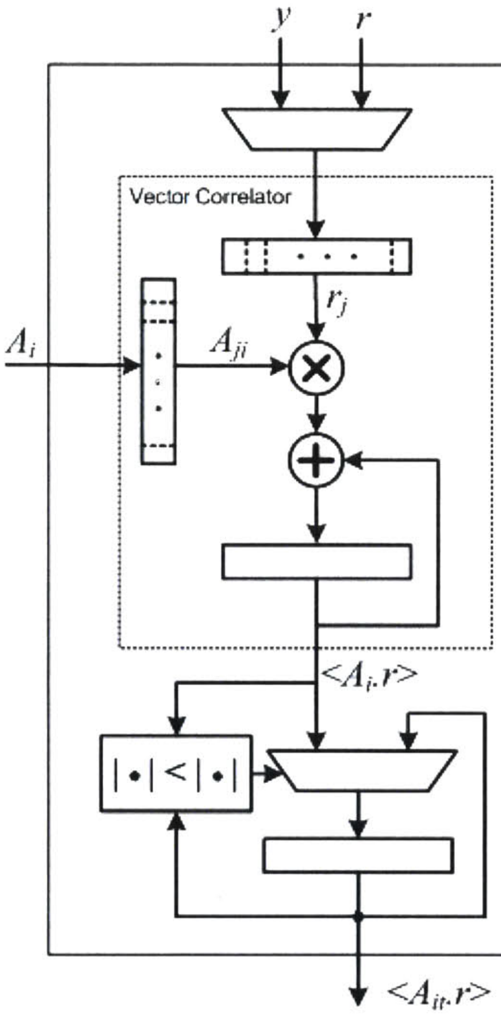

Figure 4-11: Simplified hardware block diagram of the MaxCorrelation unit.

4.3.4 Implementation Results

We synthesized and placed-and-routed our design using 45nm SOI technology. Figure 4-13 shows the layout of the MP ASIC implementation for the case when 10 multipliers are used in the design. Post-layout simulations were run for the three designs to obtain power consumption, area, and timing estimates. The results are provided in Tables 4.1 and 4.2. It should be noted that these results are for a decoder that has 8 bits of resolution in the recovered signal and performs 10 iterations of the MP algorithm.

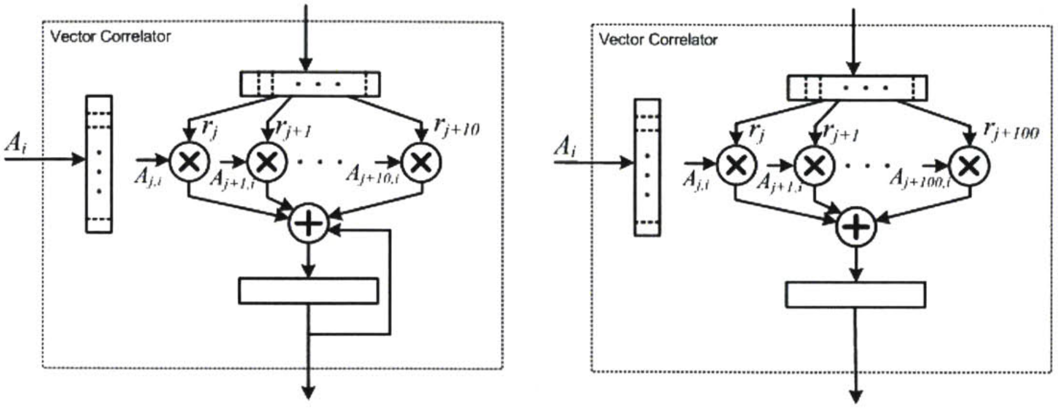

Figure 4-12: Hardware block diagram of the vector correlator using (a) 10 multipliers, and (b) 100 multipliers.

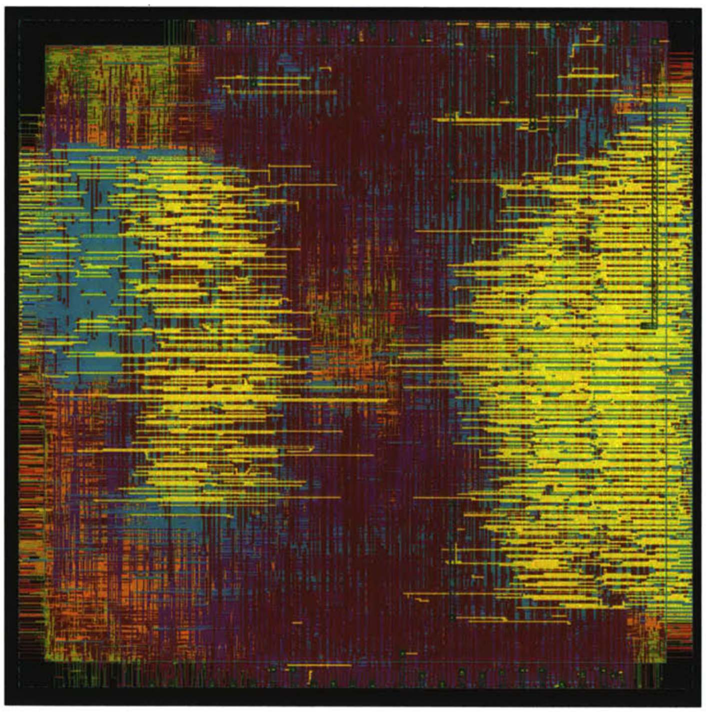

Figure 4-13: Layout of the implemented MP decoder (with 10 multipliers).

Table 4.1: MP decoder implementation results in 45nm SOI process, (resolution=8 bits, number of iterations = 10, Vdd=1V).

MP Decoder	with 1 multipler	with 10 multiplers	with 100 multiplers
Throughput(ksamples/s)	538	3952	11190
Power Consumption(mW)	66.42	96.23	202.5
-Leakage Power (mW)	1.73	2.06	3.72
-Dynamic Power (mW)	64.69	94.17	198.78
Chip Area (mm^2)	0.25	0.28	0.49
Energy Cost(nJ/sample)	123.4	23.1	18.1
FOM (pJ/sample-step)	482	90.2	70.6

Table 4.2: MP decoder required chip area breakdown

MP Decoder	with 1 multipler	with 10 multiplers	with 100 multiplers
MaxCorrelation Unit	14.2%	23.8%	53.7%
Normalization Unit (excluding LUT)	0.26%	0.2%	0.1%
Residual Unit	9.7%	8.6%	5.2%
Control Unit and registers for vector x	75.8%	67.4%	41%

The throughput of the MP decoder with one multiplier is 538 Ksamples/s while it has the power consumption of 66.42mW and requires chip area of 0.25 mm^2. By comparing these results with the results for the MP decoder with 10 multipliers, we can see that using 10 multiplier stages in parallel improves the throughput to 3.9 Msample/s. It should be noted that although we used 10 multipliers instead of one, the throughput has increased by only a factor of ~ 8. This is due to the fact that using parallel stages in the vector correlator module only reduces the number of required cycles by that block, and it does not affect the number of overhead cycles required to read/write the data from registers and the cycles required by other blocks. Hence, the total number of cycles required to reconstruct a block of signal has reduced only by factor of ~ 8 by using 10 multipliers in parallel. The other observation is that the required area has only slightly increased since most of the chip area is taken up by the registers used to keep the values. For example, we require $N \cdot 8$ registers (in our case 8000) to keep the elements of the vector x (the recovered signal).

From the results, it is evident that increasing the number of parallel stages to 100 multipliers improves the throughput to 11.190 Msample/s. This throughput is only ~ 21 times higher than the throughput of the decoder with one multiplier. In order to fairly compare these designs, the energy-cost per sample is calculated and reported in the Table 4.1. The energy cost has not improved significantly in the decoder with 100 multipliers (compared to the decoder with 10 multipliers). This is due to the fact that, in the design with 1 multiplier, the throughput is mainly limited by the vector correlator module. Hence, by using more parallel stages in this module, the throughput increases significantly. However, there is a point at which, increasing the number of parallel stages cannot help anymore since the throughput becomes limited by the other blocks. As a result, employing too many multipliers increases the power consumption while it only has insignificant effects on the throughput. This observation is shown in figures 4-14 and 4-15. For the case with a large number

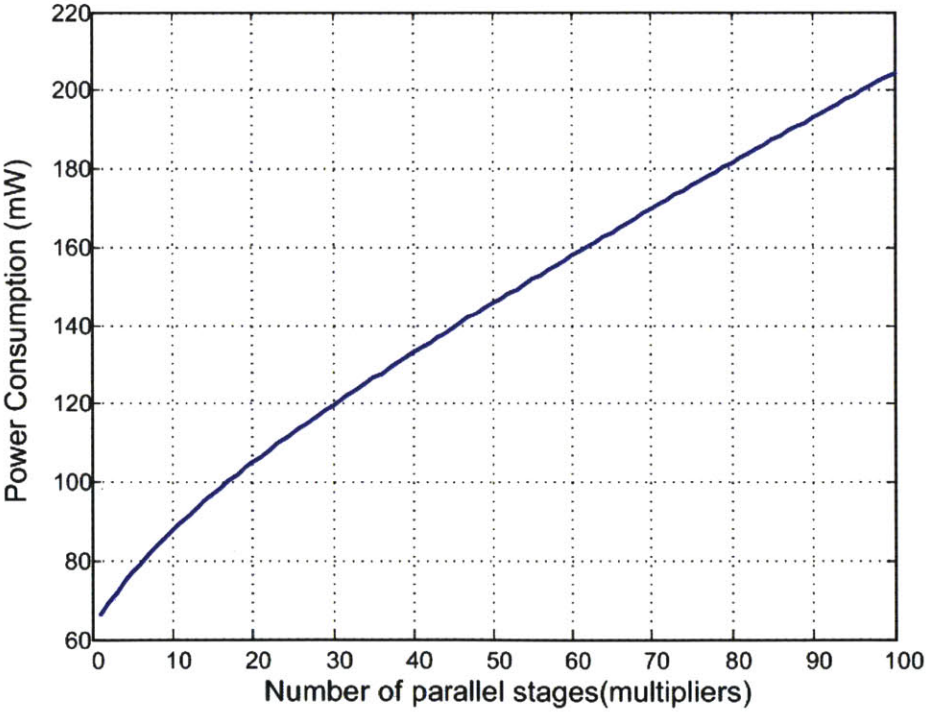

Figure 4-14: Number of multipliers used in the vector correlator module versus the power consumption of the MP decoder.

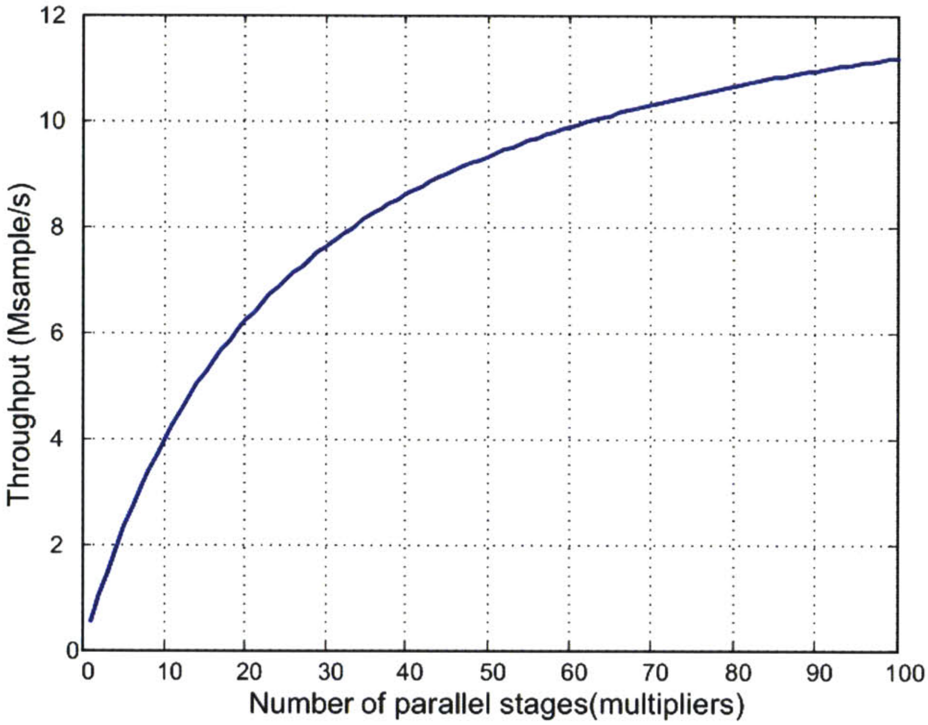

Figure 4-15: Number of multipliers used in the vector correlator module versus the throughput of the decoder.

of multipliers, adding an extra multiplier increases the power consumption while the throughput is slightly changed. Finally, figure 4-16 shows the number of multipliers versus energy-cost. The optimum number of multipliers is 40, which means the MP decoder has the lowest energy-cost when 40 parallel stages (multipliers) are used in the vector correlator module. The achieved throughput and energy cost of this MP decoder (with 40 multipliers) are 8.6 Msample/s and 15.45nJ/sample, respectively. Notice that since the energy-cost versus number of multipliers is pretty flat from 30-100, if area constraint is more critical than energy, designs with larger than 40 multipliers may be preferred.

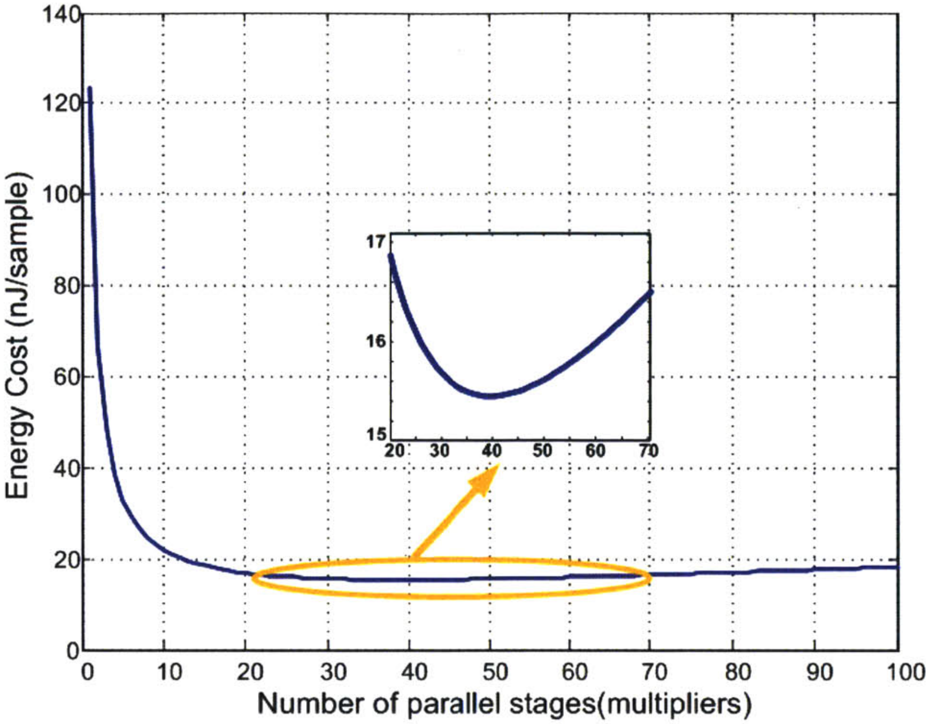

Figure 4-16: Number of multipliers used in the vector correlator module versus the energy-cost of the MP decoder.

4.3.5 Supply Voltage Scaling

As previously mentioned, the power efficiency of a VLSI design can potentially be improved by using parallel stages and reducing the supply voltage to reach the minimum energy operating point. In the previous section, we examined the performance of three different designs with different number of parallel stages. In particular, we explored the effects of parallelism on power consumption, area, and throughput. Here, we describe the model that can be used to estimate the decoder performance at scaled supply voltage.

As explained in [36] , the delay of a CMOS device can be modeled by:

$$T_d = \frac{C_L \times V_{dd}}{I} = \frac{C_L \times V_{dd}}{\frac{\mu C_{ox}}{2}(W/L)(V_{dd} - V_t)^{\gamma}}, \tag{4.1}$$

where C_L is the load capacitance, V_{dd} is the supply voltage, V_t is the threshold voltage of the transistor, and μC_{ox}, W, and L are device parameters. Since the maximum clock frequency is inversely proportional to the delay, the relationship between supply voltage and maximum clock frequency can be expressed as follows:

$$\frac{f'}{f} = \frac{V_{dd} \cdot (V'_{dd} - V_t)^{\gamma}}{V'_{dd} \cdot (V_{dd} - V_t)^{\gamma}}, \tag{4.2}$$

where the threshold voltage V_t and gama are respectively around 0.3V and 1.1 for 45nm SOI technology. Clearly, reducing the supply voltage decreases the maximum clock frequency and, consequently, decreases the throughput of the system. As it was already mentioned, reducing the supply voltage (V_{dd}) can also decrease the power consumption. The switching power dissipation in a digital circuit can be expressed as:

$$P_{switching} = \alpha \cdot C_L \cdot V_{dd}^2 \cdot f_{clk}, \tag{4.3}$$

where α is the node transition activity factor, C_L is the load capacitance, V_{dd} is the supply voltage and f_{clk} is the operating clock frequency [36]. Note that the switching power consumption decreases quickly with reducing the supply voltage as a combined effect of V_{dd}^2 and f_{clk}. Using our post-layout evaluation results, and the delay and power models, we evaluated the power consumption and throughput of the three designs with different supply voltages. Figures 4-17 and 4-18 show the throughput and power consumption versus supply voltage. Finally, to evaluate the efficiency of the designs in different supply voltages, we have also plotted the energy-cost of the designs versus supply voltage in Figure 4-19. It can be seen that decreasing the supply voltage can improve the energy-cost of the decoder. The energy-cost of the MP decoder with 40 multipliers can be improved to the optimum value of 2.54 nJ/sample while the decoder works at 0.35V and has the power consumption and throughput of 3.43mW and 1.35 Msamples/s, respectively. To use this design for the AIC proposed in Chapter 3 (i.e. 40 Gsamples/s), $\sim$ 30000 MP decoders must be employed to work in parallel which results in a total power consumption of 102 W. Given the desired 8

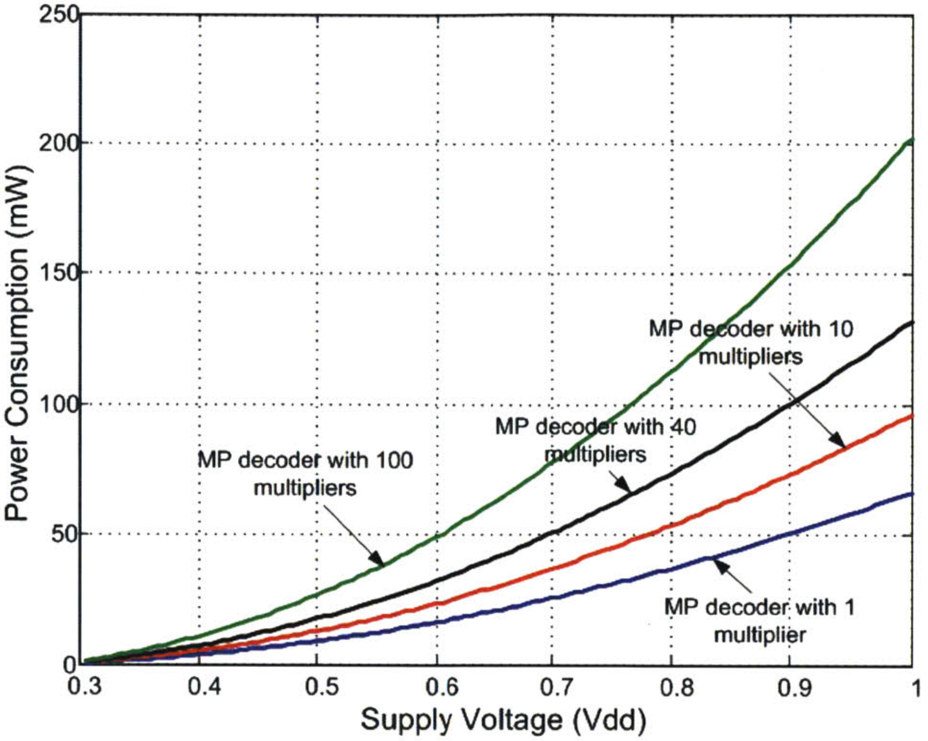

Figure 4-17: Supply voltage (V_{dd}) versus the power consumption of the MP decoder.

bit resolution, this would represent energy cost of $\sim$ 10pJ/conversion-step, which is close to the energy cost of high-rate sampling ADCs today [5]. If the task of AIC is not just to recover the time-domain waveform but actually to do the spectral sensing, then AIC alone may be more energy efficient than ADC plus FFT. Here, our goal was to optimize (reduce) the energy-cost at the cost of more area. However, we should point out that for the same throughput (i.e. 40Gsamples/s), the required number of the MP decoder can be reduced at the cost of slightly higher energy-cost. For example, the total number of required MP decoders can be reduced to $\sim$ 15800 (i.e. equivalent to $\sim 80mm \cdot 80mm$) while the design works at supply voltage of 0.4V and has the energy-cost of 2.83 nJ/sample . This hardware can be implemented by placing an array of $\sim 8mm \cdot 8mm$ ASIC dies on a board while each die includes 158 MP decoders.

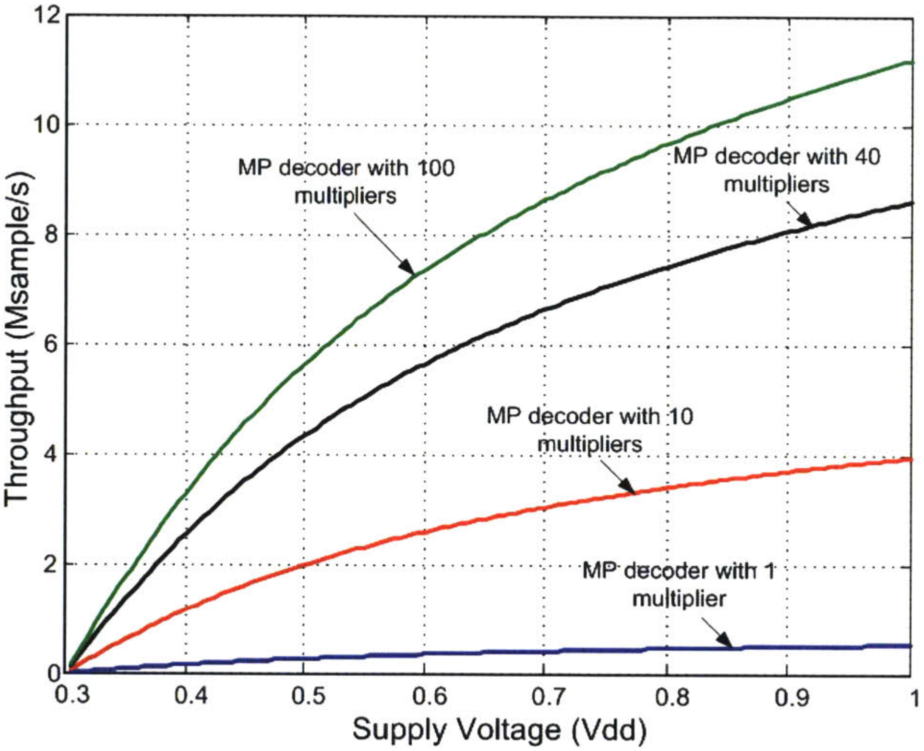

Figure 4-18: Supply voltage(V_{dd}) versus the throughput of the MP decoder.

4.4 Summary

In this chapter, we first compared the performance of MP, OMP and AMP algorithms in terms of probability of failure and hardware complexity. We illustrated that the MP algorithm is suitable for applications that require high speed, but can handle lower signal quality. Finally, we proposed a hardware architecture for the MP decoder, and we implemented it in 45nm SOI technology. We improved (optimized) the energy-cost of the decoder by using the optimum number of parallel stages and reducing the supply voltage. The optimized MP decoder has the energy-cost and throughput of 2.54 nJ/sample and 1.35 Msample/s, respectively, while it works at 0.36V. However, with the same chip area, the throughput can be improved by increasing the supply voltage at the cost of higher energy. Besides, using more parallel stages will further increase the throughput at the cost of larger chip area and higher

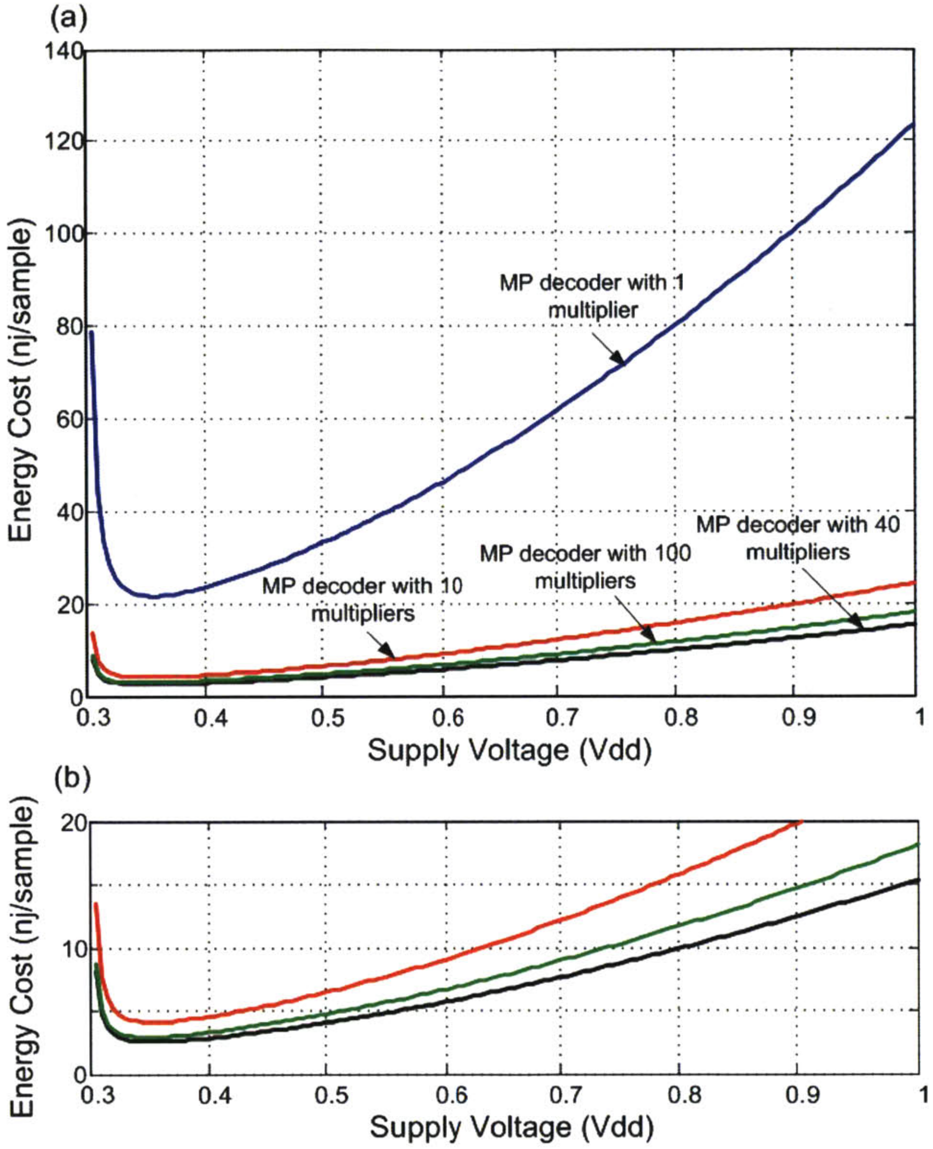

Figure 4-19: (a)Supply voltage(V_{dd}) versus the energy-cost of the MP decoder, (b)zoomed-in area.

energy. We showed that the optimal number of multipliers in the design depends on chip area and energy-cost constraints.

Chapter 5

Conclusion

Compressed sensing (CS) has been proposed as a method for recovering sparse signals from fewer measurements than ordinarily used in Shannon's sampling theorem [14]. Recently, many CS-based communications systems/devices have been proposed as a potential solution to overcome the high power consumption and performance limitation of traditional systems. CS-based wireless sensors and analog-to-information converters (AIC) are two well-known examples of CS-based systems. However these systems have yet to be evaluated in hardware. This thesis filled this gap, by designing end-to-end system evaluation frameworks and examining the impact of circuit impairments on performance limitations and energy-cost of AICs. Furthermore, we evaluated the energy-performance design space for CS in the context of a practical wireless sensor system. Finally, we designed and implemented a CS decoder (MP algorithm) in a 45nm SOI technology. The design was evaluated in terms of chip area, power consumption, and throughput.

We compared both energy cost and performance limitations of AIC and high-speed ADC systems in the context of cognitive radio applications where the input signal is sparse in the frequency domain. Our findings suggest that jitter and aperture effects in the mixing stage of AIC systems limit their resolution and performance. Hence, this contests the proposal that AICs can potentially overcome the resolution and performance limitations of sampling jitter and aperture error in high-speed Nyquist ADCs. We show that currently proposed AIC topologies have no advantage over high-speed ADCs, and are actually more sensitive to jitter and aperture errors. Finally, using realistic power models for both AIC and high-speed ADC systems, we showed that AICs have the potential to enable a 2-10X reduction in power for applications where low signal gain and low to moderate resolution are acceptable.

For CS-based wireless sensors, we have shown that, under practical constraints, CS can be an efficient and robust compression algorithm for wireless sensor applications where the signal of interest is sparse. Furthermore, we showed that CS can enable on the order of 10X reduction in transmission energy when compared to raw quantized data for applications requiring modest resolution performance (8 bits, PRD 1%). Furthermore, we have shown that CS is robust to channel error. We also proposed a diversity scheme for CS that provides greater than 10X improvement in recovered signal quality and requires no additional transmission costs. In addition, we demonstrated that the design framework and presented analysis is applicable to real world signals, such as ECGs, with a similar order of magnitude reduction in transmission energy costs.

Finally, we evaluated and compared the performance of MP, OMP and AMP algorithms in terms of probability of failure and hardware complexity. Then, We designed and implemented the MP algorithm in 45nm SOI process. Depending on the constraints (chip area versus energy-cost), different number of parallel stages may be preferred. We showed that the energy-cost of the AIC system including the reconstruction cost is close to the energy cost of high-rate sampling ADCs today. However, if the task of AIC is not just to recover the time-domain waveform but actually to do the spectral sensing, then AIC alone may be more energy efficient than ADC plus FFT.

Bibliography

[1] Fred Chen, Fabian Lim, Omid Abari, Anantha Chandrakasan, and Vladimir Stojanović. Energy-Aware Design of Compressed Sensing Systems for Wireless Sensors under Performance and Reliability Constraints. *Transactions on Circuits and Systems (TCAS)*, 2012.

[2] Michiel Van Elzakker, Ed Van Tuijl, Paul Geraedts, Daniel Schinkel, Eric Klumperink, and Bram Nauta. A 1.9μW 4.4fJ/Conversion step 10b 1MS/s Charge Redistribution ADC. In *IEEE Solid-State Circuits Conference (ISSCC)*, pages 244–246, 2011.

[3] Ying-zu Lin, Soon-jyh Chang, Yen-ting Liu, Chun-cheng Liu, and Guang-ying Huang. A 5b 800MS/s 2mW Asynchronous Binary-Search ADC in 65nm CMOS. In *IEEE Solid-State Circuits Conference (ISSCC)*, pages 80–82, 2009.

[4] Yun Chai and Jieh-tsorng Wu. A 5.37mW 10b 200MS/s Dual-Path Pipelined ADC. In *IEEE Solid-State Circuits Conference (ISSCC)*, pages 242–243, 2012.

[5] Y.M Greshishchev, J Aguirre, M Besson, Robert Gibbins, Chris Falt, Philip Flemke, Naim Ben-hamida, Daniel Pollex, Peter Schvan, and Shing-chi Wang. A 40GS/s 6b ADC in 65nm CMOS. In *IEEE Solid-State Circuits Conference (ISSCC)*, volume 35, pages 390–392, 2010.

[6] Fred Chen, Anantha P Chandrakasan, and Vladimir M Stojanović. Design and Analysis of a Hardware-Efficient Compressed Sensing Architecture for Data Compression in Wireless Sensors. *IEEE Journal of Solid-State Circuits (JSSC)*, 47.

[7] R.H. Walden. Analog-to-digital converter survey and analysis. *IEEE Journal on Selected Areas in Communications*, 17(4):539–550, April 1999.

[8] B. Murmann. A/D converter trends: Power dissipation, scaling and digitally assisted architectures. *2008 IEEE Custom Integrated Circuits Conference*, 1:105–112, September 2008.

[9] Fred Chen, A.P. Chandrakasan, and V. Stojanovic. A signal-agnostic compressed sensing acquisition system for wireless and implantable sensors. In *Custom Integrated Circuits Conference (CICC)*, pages 1–4. IEEE, 2010.

[10] Reid R Harrison, Paul T Watkins, Ryan J Kier, Robert O Lovejoy, Daniel J Black, Bradley Greger, and Florian Solzbacher. A Low-Power Integrated Circuit for a Wireless 100-Electrode Neural Recording System. *IEEE Journal of Solid State Circuits*, 42(1):123–133, 2007.

[11] Naveen Verma, Ali Shoeb, Jose Bohorquez, Joel Dawson, John Guttag, and Anantha P Chandrakasan. A Micro-Power EEG Acquisition SoC With Integrated Feature Extraction Processor for a Chronic Seizure Detection System. *IEEE Journal of Solid-State Circuits*, 45(4):804–816, 2010.

[12] Catalina Monica Fira and Liviu Goras. An ECG Signals Compression Method and Its Validation Using NNs. *Transactions on Bio-medical Engineering*, 55(4):1319–1326, 2008.

[13] Emmanuel J Candes and Terence Tao. Decoding by Linear Programming. *IEEE Transaction on Information*, 51(12):4203–4215, 2005.

[14] David L Donoho. Compressed Sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006.

[15] Emmanuel J Candes and Terence Tao. Near-Optimal Signal Recovery From Random Projections : Universal Encoding Strategies ? *Transactions on Information Theory*, 52(12):5406–5425, 2006.

[16] Vivek K Goyal. Coding : Compression. *IEEE Signal Processing Magazine*, 18(5):74–94, 2001.

[17] Joseph Mitola. Cognitive Radio for Flexible Mobile Multimedia Communications. In *IEEE Inernational Workshop on Mobile Multimedia Communication*, volume 22102, pages 3–10, 1999.

[18] Moshe Mishali and Yonina C Eldar. From Theory to Practice : Sub-Nyquist Sampling of Sparse Wideband Analog Signals. *IEEE Journal of Selected Topics in Signal Processing*, 4(2):375–391, 2010.

[19] Jason Laska, Sami Kirolos, Yehia Massoud, Richard Baraniuk, Anna Gilbert, Mark Iwen, Martin Strauss, and Ann Arbor. Random Sampling for Analog -to-Information Conversion of Wideband Signals. *IEEE Dallas/CAS Workshop on Design*, 2006.

[20] Jason N Laska, Sami Kirolos, Marco F Duarte, Tamer S Ragheb, Richard G Baraniuk, and Yehia Massoud. Theory and Implementation of an Analog-to-Information Converter using Random Demodulation. In *International Symposium on Circuits and Systems*, pages 1959–1962, 2007.

[21] Sami Kirolos, Jason Laska, Michael Wakin, Marco Duarte, and Dror Baron. Analog-to-Information Conversion via Random Demodulation. In *IEEE Dallas/CAS Workshop on Design, Applications, Integration and Software*, pages 3–6, 2006.

[22] Joel A Tropp, Jason N Laska, Marco F Duarte, Justin K Romberg, and Richard G Baraniuk. Beyond Nyquist : Efficient Sampling of Sparse Bandlimited Signals. *IEEE Transaction on Information Theory*, 56(1):520–544, 2010.

[23] Xi Chen, Zhuizhuan Yu, Sebastian Hoyos, Brian M Sadler, and Jose Silva-martinez. A Sub-Nyquist Rate Sampling Receiver Exploiting Compressive Sensing. *IEEE Transaction on Circuits and Systems I*, 58(3):507–520, 2010.

[24] Tamer Ragheb, Sami Kirolos, Jason Laska, Anna Gilbert, Martin Strauss, Richard Baraniuk, and Yehia Massoud. Implementation Models for Analog-to-Information Conversion via Random Sampling. In *Midwest Symposium on Circuits and Systems (MWSCAS)*, pages 325–328, 2007.

[25] Moshe Mishali. Efficient Sampling Of Sparse Wideband Analog Signals. In *IEEEI, 25th Convention*, pages 290–294, 2008.

[26] Justin Romberg. Compressive sensing by random convolution. *SIAM J. Imag. Sci*, 2(4):1098–1128,, 2009.

[27] M Jeeradit, J Kim, B Leibowitz, P Nikaeen, V Wang, B Garlepp, and C Werner. Characterizing Sampling Aperture of Clocked Comparators. In *IEEE Symposium on VLSI Circuits*, pages 68–69, 2008.

[28] Jaeha Kim, Brian S Leibowitz, and Metha Jeeradit. Impulse Sensitivity Function Analysis of Periodic Circuits. In *IEEE/ACM International Conference on Computer Aided Design (ICCAD)*, pages 386–391, 2008.

[29] Timothy O Dickson, Ekaterina Laskin, Imran Khalid, Rudy Beerkens, Jingqiong Xie, Boris Karajica, and Sorin P Voinigescu. An 80-Gb / s 2 31 1 Pseudorandom Binary Sequence Generator in SiGe BiCMOS Technology. *IEEE Journal of Solid-State CircuitsSolid-State Circuits*, 40(12):2735–2745, 2005.

[30] By Anna Gilbert and Piotr Indyk. Sparse recovery using sparse matrices. *Proceedings of the IEEE*, 98(6):937–947, 2010.

[31] Michel S J Steyaert, Willy M C Sansen, and Andchang Zhongyuan. A Micropower Low-Noise Monolithic Instrumentation Amplifier For Medical Purposes. *IEEE Journal of Solid-State Circuits (JSSC)*, 22(6):1163–1168, 1987.

[32] S. Mellat and Zhifeng Zhang. Matching Pursuits With Time-Frequency Dictionaries. In *IEEE Transactions on Signal Processing*, 1993.

[33] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad. Orthogonal Matching Pursuit : Recursive Function Approximation with Applications to Wavelet Decomposition. In *27th Asilomar Conference on Signals, Systems and Computers*, pages 1–5, 1993.

[34] David L Donoho, Arian Maleki, and Andrea Montanari. Message-passing algorithms for compressed sensing. *National Academnc Science USA*, 2009:1–6, 2009.

[35] Arian Maleki, David L Donoho, and Andrea Montanari. Analysis of Approximate Message Passing Algorithm. In *44th Annual Conf. Information Sciences and Systems*, 2010.

[36] Anantha Chandrakasan and R. W. Brodersen. Minimizing Power Consumption in Digital CMOS Circuits. *IEEE*, pages 498–523, 1995.