Low Power Data Acquisition for MicroImplant Biometric Monitoring of Tremors
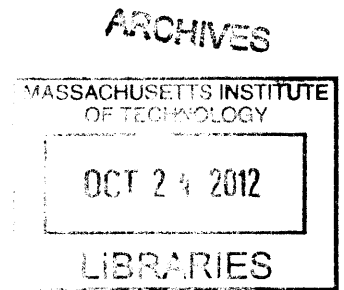
by

Tania Khanna

B.S., Cornell University (2005)

S.M., Massachusetts Institute of Technology (2008)

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

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2012

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

Certified by . . . .
Joel Dawson
Associate Professor
Thesis Supervisor

Accepted by . . . .
Leslie Kolodziejski
Chairman, Department Committee on Graduate Students

Low Power Data Acquisition for MicroImplant Biometric Monitoring of Tremors

by

Tania Khanna

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

Abstract

In recent years, trends in the medical industry have created a growing demand for implantable medical devices. In particular, the need to provide doctors a means to continuously monitor biometrics over long time scales with increased precision is paramount to efficient healthcare. To make medical implants more attractive, there is a need to reduce their size and power consumption. Small medical implants would allow for less invasive procedures, greater comfort for patients, and increased patient compliance. Reductions in power consumption translate to longer battery life. The two primary limitations to the size of small medical implants are the batteries that provide energy to circuit and sensor components and the antennas that enable wireless communication to terminals outside of the body. The theory is applied in the context of the long term monitoring of Parkinson's tremors. This work investigates how to reduce the amount of data needing to acquire a signal by applying compressive sampling thereby alleviating the demand on the energy source. A low energy SAR ADC is designed using adiabatic charging to further reduce energy usage. This application is ideal for adiabatic techniques because of the low frequency of operation and the ease with which we can reclaim energy from discharging the capacitors.

Keywords: SAR ADC, adiabatic, compressive sampling, biometric, implants

Acknowledgements

I would first like to thank my family for all of their support and attention. Without my parents' hard work and patience, I would not be who I am. I am amazed every day at how they do it, and it is that effortlessness that I hope to achieve. My brother and sister have set an excellent example for me in life, and I can only give my thanks for that. My family has been there whenever I needed anything, and their constant support and tolerance of me is priceless.

I owe gratitude towards my incredible network of friends, who have always been there for me when I needed encouragement or discouragement. They have been a great source of energy, and I thank them mostly for their cheerleading without pressure. I appreciate the drinks and the diversions, your friendship over the years is worth at least a couple of pages in my memoir

I must thank my advisor, Dr. Joel Dawson. He has always given me great advice and direction, especially in recognizing when I need it. In addition to the guidance he has given me, he has always been optimistic when it has come to my research motivating me to keep going. This work could not have been completed without his help and encouragement.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The healthcare system of today, while seeming miraculous at times, has many trouble areas. A major, tangible issue is that of patient non-compliance. Non-compliance, or non-adherence, refers simply to incapable or stubborn patients—those who do not comply, or adhere, to medical instructions such as following a dosage schedule, recording medically-relevant activity as requested, attending scheduled medical visits, etc. To mitigate the effects of non-compliance, patient interaction is something that needs to be minimized if not eliminated.

It is especially difficult to monitor biometric signals in any long-term way without patient compliance. Commercially available devices exist to monitor sleep [1], blood pressure [2], and blood glucose levels [3]. They all require extensive patient interaction. In the case of blood pressure and glucose monitors, the patient is responsible to take a recording, thereby making continuous monitoring unlikely. Additionally the monitors themselves are large and require proper use to get an accurate data point. In the case of some sleep monitors, like the Actiwatch, continuous monitoring can be achieved if the watch is worn continuously. However, with daily life it is easy to neglect to wear it for a

significant period of time. Furthermore, the data is transferred via wired communication and requires wired recharging. The most reliable way to obtain continuous long-term data is with an implant that requires little to no operation by the patient.

An implant is a frightening notion, and adding a surgery to place it inside the body can be a "deal breaker". While biometric monitoring is important and can be critical, the risk and pain associated with a surgery may seem too high a cost. If we can eliminate the surgery by making the implant injectable by a medical syringe and needle, we can bring patient monitoring to a large population. The company Positive ID [4] had in the past developed an injectable identification sensor called VeriChip, and is currently working on a product with a similar form factor to monitor glucose levels. The limitations imposed by designing an injectable long-term biometric monitoring implant are explored in this work and proposes a minimally invasive microImplant platform.

A question remains: why is biometric monitoring is so important? One significant case is that of monitoring Parkinson's tremors. Parkinson's disease is a neurodegenerative disorder that has no cure, but there are several treatments that can prolong an individual's life and improve their quality of life. Unfortunately, patients are often misdiagnosed for years and lose several years of treatment due to a lack of a clear diagnostic test. The current diagnosis method depends on a doctor recognizing a tremor or other physical manifestations and is a very subjective and qualitative measure that is easily and understandably missed. By attempting to quantify this diagnostic with biometric monitoring, we can improve patient care. Furthermore, with early diagnosis, there is hope for improved clinical research leading to better treatment methods.

This work introduces an energy efficient method for monitoring Parkinson's tremors by using compressive sampling and proposes a hardware technique to sample sparsely in real time to reduce the number of samples converted and written to memory to acquire the relevant signal. The use of compressive sampling has the potential to greatly reduce the energy consumption of the implant—in the memory and the ADC. Writing to memory is costly in energy. A non-volatile memory for use in RFID tags was designed in [5] that reported an energy consumption of 5.4 nJ/bit or 43.2 nJ/byte. Comparatively, if we had an 8-bit ADC with an energy consumption of only 1 pJ/conv is four orders of magnitude lower. A reduction in samples of 75% would reduce the write energy consumption significantly. This comparison argues strongly for reducing the number of samples that are stored, as opposed to storing a great many samples and then using compression to eliminate those that are not necessary. In this work, we explore compressive sampling as a means of saving energy here. In addition, we also minimize the energy consumption of the data conversion by designing a SAR ADC using adiabatic charging.

Though monitoring Parkinson's tremors is a critical use of biometric monitoring, the motivation behind this work is to design a *platform* for monitoring. The hope is that by replacing the sensor, we can monitor any biometric. In being able to monitor a biometric reliably, we can improve patient care. For example, it is often the case that a hospital patient be required to stay admitted in the hospital for patient monitoring, which may be as simple as checking a patient's temperature to check for a fever. This could cause a patient to remain in the hospital for days past the necessary stay. However, if a patient

could have a reliable monitor of their temperature, they could be allowed to possibly venture out of the hospital for hours if not days depending on the situation.

The far future vision of biometric monitoring is that every individual would have their significant biometrics continuously monitored. This data can then be used to not only detect any troubling signals as early as possibly, but it enable a framework for individualized patient care. A patient's data can be used to get a detailed history, and personalize care and treatment towards an individual.

1.1   Thesis organization

Chapter 2 describes the microImplant platform and the high level challenges associated with designing a long-term injectable monitor. The nature of the implant requires a strict form factor and use model imposing tight constraints on the system design. This chapter discusses the design choices made as a result of the tight constraints and highlights the importance of designing the system as a whole in order to meet the constraints.

Chapter 3 aims to explain the background theory behind compressive sampling (CS) and in specifics as it relates to information and theory and data acquisition. The sparsity and incoherence conditions are discussed and explained in practical terms. Then the theory is put into practice for a simple example of a sum of sinusoids, which is sparse in frequency mimicking the tremor signal spectrum that this works is focused on monitoring. The various parameters involved in applying CS theory are analyzed and their effect on the recovery is presented.

In Chapter 4, the biometric of gross motor activity to monitor Parkinson's tremor is described in an effort to motivate the use of compressive sampling. The work from

Chapter 3 is then applied on real tremor data to fine tune the recovery algorithm for signal recovery. A hardware implementation for random-like sampling is proposed, and in simulation is applied to real tremor data.

Chapter 5 discusses the design of a low energy SAR ADC using adiabatic charging to reduce energy consumption. A dynamic comparator with digital offset correction is presented to reduce static power loss to leakage levels. The adiabatic charging is implemented with a self-stabilizing ramp circuit that is analyzed for the 180 nm process and the power savings are analyzed for this application. The measurement results are presented with the figure of merit for the ADC. Conclusions and contributions of this work are given in Chapter 6.

Chapter 2

MicroImplant Platform

This work proposes the design of a minimally invasive implant *platform* for biometric monitoring with a specific form factor and use model that impose constraints onto the underlying system blocks and the circuits contained within them. The proposal for the implant is a reconfigurable radio-frequency identification (RFID) transponder that contains a rechargeable power source and periodically records a specific biometric signal to be transmitted wirelessly. In the past, devices have been successfully surgically implanted. The aim of the implant platform targeted here is to be minimally invasive, or not requiring surgery for implantation and eliminating the risks attached to surgical implantation. In place of surgery, the implant would be injected via a hypodermic needle into the appropriate location in the body directly. Making the implant injectable imposes severe constraints on the form factor of the implant and also on the power consumption of its electronics. In sub-section 2.2 a use model is described informing the design and making the case for circuit techniques being advantageous. While the implant system is separated into individual blocks, the constraints put upon them individually inform the design of the entire system. In order to achieve the strict form factor and use model, they

must be designed in conjunction with each other. This chapter details the constraints stemming from both the form factor and the use model, and how they impact the design of the signal sensing and processing block.

2.1.   Form Factor

Hypodermic needles are categorized by their outer diameters described as a gauge, where a smaller gauge number indicates a larger diameter. A 12-gauge needle has an outer diameter of 2.769 mm and a nominal inner diameter of 2.159 mm dictating the entire implant package would need be no more than 2 mm wide and 2 mm high. There is more flexibility on the length, since a hypodermic needle can be several centimeters long. To settle on a length, we invoke one more desired metric, which is that the implant be easily injectable, and, if injection is not preferred, *discreetly* wearable. In this aspect of the design we stray from strict electrical engineering to product design, where aesthetics and "feel" matter as much as hard metrics like power consumption. Working with our medical collaborator Dr. Seward Rutkove at Beth Israel Deaconness Medical Center, we settled on a length specification of 5 mm. With this strict form factor constraint there is only enough space for an energy storage element and a single integrated circuit (IC) containing the entire system. A representation of the entire package is show in Figure 2-1.

While a battery could fit this small form factor, the limited recharge cycles would limit the lifetime of the implant and require battery replacement via surgery. The platform as depicted in Figure 2-1 shows the energy storage element as an ultracapacitor, which are able to achieve volumetric energy densities and have very high power density
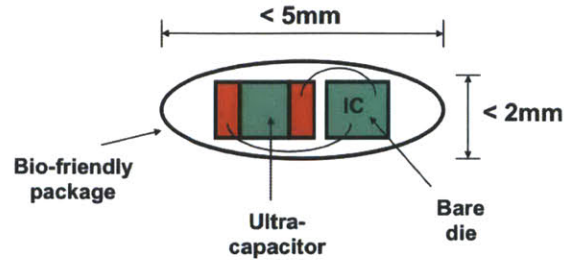
Figure 2-1. Proposed minimally invasive biometric monitoring platform.

with practically unlimited recharge cycles. The charge-discharge lifetime of ultracapacitors allow them to outlive the devices they provide energy for, making them environmentally friendly. Typical battery chemistries wear out in just a few years, leaving behind toxic waste that pose a serious disposal and safety hazard. Moreover, memory effects, loss of electrolyte, non-ideal charge cycles, leakage, and geometries pose challenges when using batteries. Lastly, ultracapacitors have the potential to be integrated on chip with the use of carbon nanotubes. [6] An innovative power management technique for this platform was designed in conjunction with this work in [7].

The IC contains the entire system for long-term biometric monitoring and, as seen in Figure 2-2, can be portioned into three major blocks: 1) energy management, 2) signal sensing and processing, and 3) data collection and transmission. The energy management block was examined in detail in [7], where an ultracapacitor array stacking technique and variable ratio switched capacitor voltage regulator provide a steady reference to act as a voltage supply for the IC. This work addresses the signal sensing and processing block by implementing adiabatic charging techniques in a SAR ADC design and applies compressive sampling to Parkinson's tremor monitoring. The remaining block consists

of memory storage of the data and an RF modulated switch to transmit data when the implant is in the presence of a reader using the same technique as a traditional passive RFID transponder.
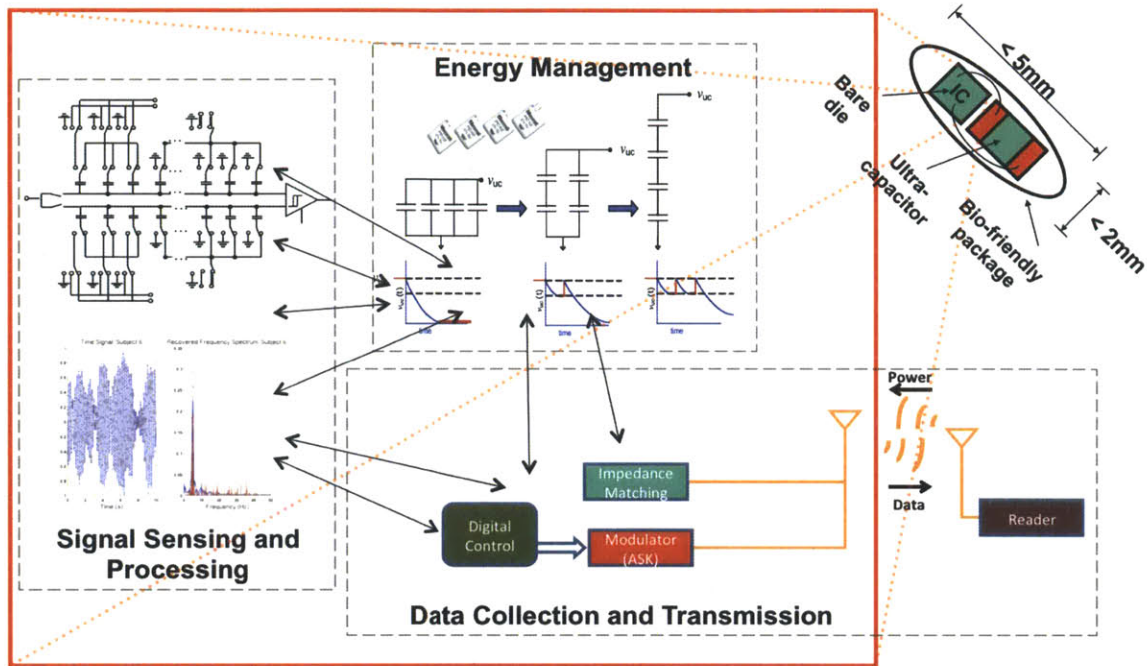
Figure 2-2.   System blocks for proposed platform IC.

As explored in [7], ultracapacitors have many advantages. A distinct disadvantage is their lower energy density relative to that of batteries. Given the fixed form factor, it is clear that the optimal design strategy involves outsourcing as much of the energy demands of the system outside the implant as possible, leaving the reader, for example, to do any and all post-processing of the data. A central idea of this dissertation involves taking this line of reasoning a step further and exploiting compressive sampling techniques to take a significantly lesser number of samples for analog-to-digital conversion. In so doing, we reduce the energy consumed by an ADC both through the advanced signal processing of compressed sampling and also by allowing it to run at as

slow a conversion rate as possible. A comparator based SAR ADC is an ideal choice for the platform because of its small area requirements, and we further reduce the energy consumption by using a dynamic comparator with digital offset correction and employing adiabatic charging techniques on the capacitor array. The ADC design and measurement results are described in detail in Chapter 5.

The form factor also impacts the data transmission block via the ultracapacitor by limiting the energy available. Data transmission via a wireless transceiver would severely tax any on-chip energy source. By using RFID techniques, we can again outsource the energy demands outside the system. By storing data on chip and transmitting only in the presence of an external reader, we leverage the power supplied by the reader to collect the relevant data. Using a modulated switch we can transmit binary data by modifying the impedance detected by a reader's reflected radio wave. However, the antenna must still be designed to fit within our package shown in Figure 2-1. Work done in [8] shows that the optimal frequency for transmitting through biological media like skin and tissue exists in the GHz-range. This makes possible reducing the antenna size to sizes acceptable for implantable devices. The antenna would be shared between data transmission and ultracapacitor charging, and as part of [7] fractal antennas were explored to show their effectiveness in reducing antenna area without sacrificing electrical antenna length. Though RFID techniques significantly reduce the energy required to transmit data, reduction in the amount of data to be transmitted would improve the use model of the implant, which is further discussed in Section 2.2.

2.2.   Use Model

A major challenge for any work done in the medical space centers on the psychology of the patient using the device. It is easy to dismiss non-compliant patients as stubborn or wrong-headed, but the truth is that any implant that requires rigid, daily discipline on the part of the patient effect is doomed to help a very small percentage of the population. The psychology and emotions a person deals with when facing a chronic illness is not to be underestimated. Additionally, in elderly patients with neurodegenerative diseases—like Parkinson's disease—are often not physically or mentally able to meet such discipline. Part of the answer for a device designer is to make the final product so easy to use that it has almost no effect on the daily life of the patient. An easy example of a badly done product is an implant that requires the patient to sit still for an hour each day while it charges. Such impositions should be avoided at all costs.

As discussed in Chapter 1, commercial wearable monitoring devices exist, but for many of them patient compliance is a major issue. While eliminating any patient interaction would be ideal, this is impractical for long-term monitoring, as the implant would have to accommodate an energy source capable of lasting more than a day without charging and as a result be too large to be implantable. The use model proposed here is for a patient to (1) recharge the ultracapacitor and (2) transmit data once per day via an RFID near-field reader by holding the implant and reader in close proximity for less than one minute. Again, ultracapacitors are well suited for this requirement because of their speedy charge time.

While memory is typically seen as "free" because of the advancements in technology to reduce memory size and power, the amount of memory needed to store a day's worth

of monitoring data can approach area sizes too large for the proposed form factor. For example, in the case of monitoring Parkinson's tremors—discussed in great detail in Chapter 4—we are required to sample at a minimum of 40 Hz. If we monitored the signal with 8-bit samples for 24 hours, the required memory size would be 27.65 Mbits. This memory capacity translates into an active area of 8.3 mm^2 and 2.7 mm^2 in a 180 nm and 90 nm process respectively for a 6T-SRAM memory with minimum sized transistors. The memory is able to fit in the proposed package size, but would leave little room for anything else. Reducing the amount of data to transmit would not only save on transmit time, but reduce memory capacity to a practical size for the implant.

Another component to the implant's use model is the application for which it is intended. This work proposes the long-term monitoring of Parkinson's tremors. The tremor signal is acquired by using an accelerometer to detect gross motor activity, however, the time representation is not relevant in clinical settings. The relevant information is contained in the frequency spectrum, and much analysis has been done to distinguish Parkinson's tremors from other, less serious tremors not associated with neurodegenerative diseases [9-10]. Since tremor signals are sparse in frequency, compressive sampling can be employed to sample at less than the Nyquist rate effectively reducing the data storage requirements. Compressive sampling theory is explained in detail in Chapter 3 and is applied to Parkinson's tremors specifically in Chapter 4. A 75% reduction in samples can be made while still being able to detect the tremors. In the future, making the monitoring event driven would further reduce the data size. For example, monitoring can be only activated when movement beyond some threshold is detected, or with additional control circuitry when a tremor is detected. A scheme for

24

detecting tremors in real time was detailed in [11], but this would add significantly to the energy usage.

Both the form factor and use model directly affect the system design of the implant and place strict constraints on the operation, but the constraints placed are such that the entire implant system must be considered and designed as whole. Each system component's design must be integrated to meet the strict specifications. Therefore understanding the application, in this case long-term Parkinson's tremor monitoring, plays an important role in the system design.

Chapter 3

Compressive sampling theory

Perhaps the most fundamental concept in information theory is the Nyquist-Shannon sampling theorem, which states in [12]:

> If a function $f(t)$ contains no frequencies higher than W cps [hertz], it is
>
> completely determined by giving its ordinates at a series of points spaced
>
> 1/2 W seconds apart.

That is, if a signal has frequency content fully contained within the range $\left[-BW, BW\right]$,

then the signal can be completely recovered if is sampled at minimum rate f_N, where

$$f_N = 2 \cdot BW \ .$$

A critical distinction must be made between bandwidth and frequency content. A signal

can contain high frequencies, but the Nyquist frequency depends only on the frequency

band with non-zero amplitudes. A good illustration of this distinction can be found in

radio communications, where at the time of this writing it is common to have transmitted

signals centered at a carrier frequency of 2 GHz while occupying only a 5-MHz-wide

contiguous band of spectrum. Intuitively we can think of the signal being downconverted

from the carrier frequency and then sampling at twice the highest frequency for complete

signal recovery. In some systems it may even be appropriate to perform "subsampling," in which the sampling rate is purely a function of the signal bandwidth, with the carrier aliased down to baseband in place of an actual downconversion. The point is that knowing the bandwidth of the signal and not just the maximum frequency contained in the signal allows for sampling at a lower frequency than if it was assumed the bandwidth of a signal was $2 \cdot f_{\max}$.

Applying Shannon's insight gets particularly challenging, though, in one unusual case: when the spectrum of a given signal is *sparse*. That is, suppose that its Fourier transform is zero almost everywhere, except for in a small number of extremely narrow bands. In this case the total "occupied" spectrum or bandwidth is quite small, and it could be argued that there ought to be some way to extend the idea of the Nyquist rate to this sparse case, allowing for the sampling rate to be lower than twice the highest frequency component. It turns out that the answer to this question lies in the technique called compressive sampling (CS). CS is guaranteed to work, subject to certain assumptions of sparsity and incoherence, which will be talked about in section 3.1. CS has recently been explored in the realm of video imaging by leveraging the sparsity of images in the spatial domain (I.e. few pixels in a given resolution change from frame to frame) to compress videos without losing any video quality. In [13] CS theory is applied as a compression technique for electroencephalography (EEG) and electrocardiogram (EKG) signals that are sparse in the time domain. These techniques can be applied to signals sparse in any domain, and many biometric signals meet the criteria as shown in [13]. This work applies CS, not as compression but as a sampling technique, to Parkinson's tremors sparse in the frequency domain as discussed in Chapter 4.

3.1.   Background theory

Compressive sampling (CS) enables sampling a signal at a frequency less than twice the highest frequency component of that signal, while still being able to accurately recover the signal; in some cases, it can result in exact recovery. However, two major assumptions of sparsity and incoherence need to be true of the signal and of the sampling mechanism, which will be defined and explored in this section.

First, we consider the concept of compression as it is most commonly understood in digital signal processing. Compression techniques are used to reduce a specific data stream to a new data stream of smaller length and can be categorized as lossless or lossy. With lossless compression, redundancies in the original bit stream are leveraged to reduce the number of bits without any loss of information. Huffman encoding is a form of lossless compression where large bit patterns are encoded with different, shorter bit patterns, which are usually variable in length and determined by the creation of a Huffman tree where the average code length is limited by the Shannon entropy. Decompression, or recovery of the original bit stream, requires a table detailing this encoding, but no information is lost. In the case of lossy compression, insignificant information in the original bit stream is lost, but after decompression the bit stream recovered is perceptually accurate for the given application. A common example of this is JPEG compression where unnecessary visual information is purposefully discarded that the human eye is less sensitive to, therefore the recovered image is not perceived as distorted from the original image. Lossy and lossless data compression is analogous to accurate and exact recovery in compressive sampling.

Compression of data exploits redundancies and data insignificance to reduce data size and recover the original data accurately. In a similar way, CS techniques exploit sparsity and incoherence of a signal to sense significantly fewer samples than conventional information theory dictates and still recover the original signal accurately. Therefore, we can explore the three main concepts on which CS theory relies: 1) sparsity, 2) incoherence, and 3) recovery. With the assumption of a sparse signal, we can design an incoherent sampling technique and optimization formulation for recovery.

Sparsity refers to the idea of a signal being compressible, or more technically if we can represent our signal in some basis with a vector and that vector has few non-zero elements. Incoherence means low coherence, which is a measure of the correlation between two separate bases. In the context of CS theory, the coherence of a signal is measured between the signal's representation basis—most often the signal is sparse in this basis—and a sensing basis. The sensing basis is designed to be incoherent with the representation basis so we can implement incoherent sampling mechanisms to recover the desired signal. To more clearly discuss these two assumptions, we can start by defining some basics of data acquisition.

Typical data acquisition consists of finite sampling of a continuous-time signal, $f(t)$, to acquire a discrete-time signal that can be represented by a finite length vector $x \in \mathbb{R}^N$. If we sample against some set of waveforms, $\psi_i(t)$, we can relate them by

$$f(t) = \sum_{i=1}^{N} x_i \psi_i(t), \quad t = 1, \dots, N.$$

In other words we represent our signal as a superposition of the waveforms ψ_i, which vary depending on the application. To acquire a signal in the time domain the waveforms are

simple delta functions, but in the case of MRI imaging the basis is made up of sinusoids, thus making x a vector of Fourier coefficients. For ease of notation we can write $f = \Psi x$, where Ψ, called the representation basis, is the $N \times N$ matrix with the waveforms ψ_i as columns.

We can now define a sparse signal as one that can be represented on a small set of waveforms in the Ψ-domain. In other words, the representation vector x has only a small number significantly less than N of non-zero entries. CS theory claims to fully reconstruct the original representation vector x from some undersampled vector y described as

$$y_k = \langle f, \varphi_k \rangle \quad k \in M \ , \tag{1}$$

where φ_k are sensing waveforms making up a second sensing basis, Φ, and $M \subset \{1, \dots, N\}$ is a subset of length $M < N$. If the sparse signal were sensed in the Ψ-domain, it would seem impossible to design a mechanism for undersampling and capture the significant samples. However, assuming Φ and Ψ are incoherent bases, we can design simple sampling techniques and recover x with very high probability.

Coherence is defined between two bases and is the largest correlation between any two elements in them

$$\mu(\Phi, \Psi) = \sqrt{N} \cdot \max_{1 \le k, j \le N} \left| \langle \varphi_k, \psi_j \rangle \right|.$$

The lower the coherence, the more accurate recovery of the representation vector is possible. Intuitively we can think of sensing a sparse signal in the basis in which it is sparse. If we sense enough data points, we can get an accurate representation, but if we significantly reduce the number of sensed data points it is easy to imagine missing a major signal feature. However, if we sample in the sensing basis with low coherence to the sparsity basis, we get more dimensional information in each data point than were we to

30

sample in the sparsity basis. An example of a sparsity basis and sensing basis that are incoherent is shown in Section 3.2. The benefits of this are seen in the recovery of the representation vector x via optimization. An incoherent sensing basis results in a more dimensionally constrained feasible solution set to the optimization formulation, which is now discussed.

We are only interested in the undersampled case ($M < N$) so we look at the measurements, y_k, described in Eq. 1 that give us partial information about the representation vector x

$$y = \Phi \Psi x \ ,$$

where Ψ is an $N \times N$ matrix describing the representation basis where x is sparse and Φ is the $M \times N$ matrix with the sensing waveforms $\varphi_\square$ as columns describing the sensing basis. To reconstruct x fully, we can formulate an optimization problem to minimize the number of non-zero entries of x

$$\min_{x \in \mathbb{R}^N} \|x\|_{\ell_0} \quad \text{subject to} \quad y = \Phi \Psi x \ ,$$

where $\|x\|_{\ell_0} = \sum_{i=1}^{N} |x_\square|^0$ with the definition of $0^0 = 0$. While this would generate the sparsest x, the set of feasible solutions is intractable and impractical to optimize. The common practice in CS is to minimize the ℓ_1 norm

$$\min_{x \in \mathbb{R}^N} \|x\|_{\ell_1} \quad \text{subject to} \quad y = \Phi \Psi x \ , \tag{2}$$

where $\|x\|_{\ell_1} = \sum_{i=1}^{N} |x_i|$. The ℓ_1 norm lends itself well to optimization problems because subject to only linear equalities we can use linear program techniques with many efficient solution algorithms [14]. Much research has been done on recovery algorithms and the solution algorithms in [15-20] leading to the development of a collection of MATLAB routines called L1-MAGIC [21] used in this work.

Looking at the formulation in Eq. 2, we can see that sampling in an incoherent basis allows for a more complete picture of the solution space in all dimensions, whereas sampling in the sparsity basis is likely to ignore some dimensions and result in redundant constraints therefore a set of optimal solutions instead of a single solution. An analogy of this is optically sensing a cube in three dimensions. If you were to look at the cube such that it appeared as a square in two dimensions, there is no visually sampled data to determine that it is a cube and not infinitely long in the third dimension. However, if your visual angle is skewed, you now gain information in the third dimension and can clearly see the shape is a cube. Once an incoherent domain is identified for a given signal, it was shown in [14] that sampling randomly in that domain and then performing a linear optimization will recover the signal. Furthermore, [14] showed for a signal which in the Ψ-domain is S-sparse (i.e. the signal has S significant features), the signal can be exactly recovered with overwhelming probability by sampling uniformly at random in the Φ-domain with M samples

$$M \geq C \cdot \mu^2(\Phi, \Psi) \cdot S \cdot \log N , \tag{3}$$

where C was empirically shown to be $\sim$2-2.5 [22]. We see that the number of samples required is proportional to $\log N$, where N is the information length to be recovered. Eq. 3 also indicates with increasing sparsity and incoherence (i.e. decreasing S and coherence) the minimum number of samples needed for recovery decreases. In the context of this work, Parkinson's tremors are sparse in the frequency domain and exhibit incoherence with the time domain, therefore we can sample randomly in time with fewer samples than the Nyquist rate would require or implementing sparse sampling in the time domain.

3.2. Sparse sampling

Sparse sampling can refer to sampling in any domain, but in this work we use it to refer to specifically sampling in the time domain. We apply the CS theory described in Section 3.1 to the specific case of detecting a signal that is S-sparse in the frequency domain. We can write the time-frequency relationship as

$$\hat{f}(\omega) = \sum_{i=1}^{K} \alpha_i \delta(\omega_i - \omega) \overset{\mathcal{F}}{\Leftrightarrow} f(t) = \sum_{i=1}^{K} \alpha_i e^{i\omega_i t} \quad ,$$

where $f(t)$ is written as a superposition of sinusoids.

We can then take M random samples in time and minimize the ℓ_1 norm in the frequency domain rewriting Eq. 2 as

$$\min \sum_{\omega} |\hat{g}(\omega)| \quad \text{subject to} \quad g(t_m) = f(t_m), \quad m = 1, \dots, M \ . \quad (4)$$

· After recasting this as a linear program, we can solve and recover the signal in the frequency domain. Once the signal spectrum is recovered, the inverse Fourier transform can then be used to obtain the time signal; however, in the case of Parkinson's tremor monitoring the relevant information is in fact the frequency spectrum—specifically the frequency and amplitude of the various tremors.

To investigate this we can examine a simple input signal, which is the sum of four sinusoids with varying amplitudes at random frequencies in the 0-20 Hz frequency band thereby creating a signal that is S-sparse in the frequency domain where $S = 4$. Figure 3-1 shows this input in the time domain if we were to sample for $T = 10$ seconds at $f_N = 40\,\text{Hz}$ and the power spectral density as calculated with a fast-Fourier transform (FFT) with the exact random frequency locations circled. With this example, the sparsity

33

basis is the frequency domain and the sensing basis is the time domain with the recovery technique resulting in a vector representation of the frequency spectrum with length $N = 1024$ constrained in the frequency range 0-20 Hz. Using Eq. 3 and conservatively setting $C = 2.5$ with a worst case correlation we evaluate the number of samples to take, $M = 70$, resulting in 17.5% of the Nyquist required samples shown in Figure 3-1 as the circled samples.
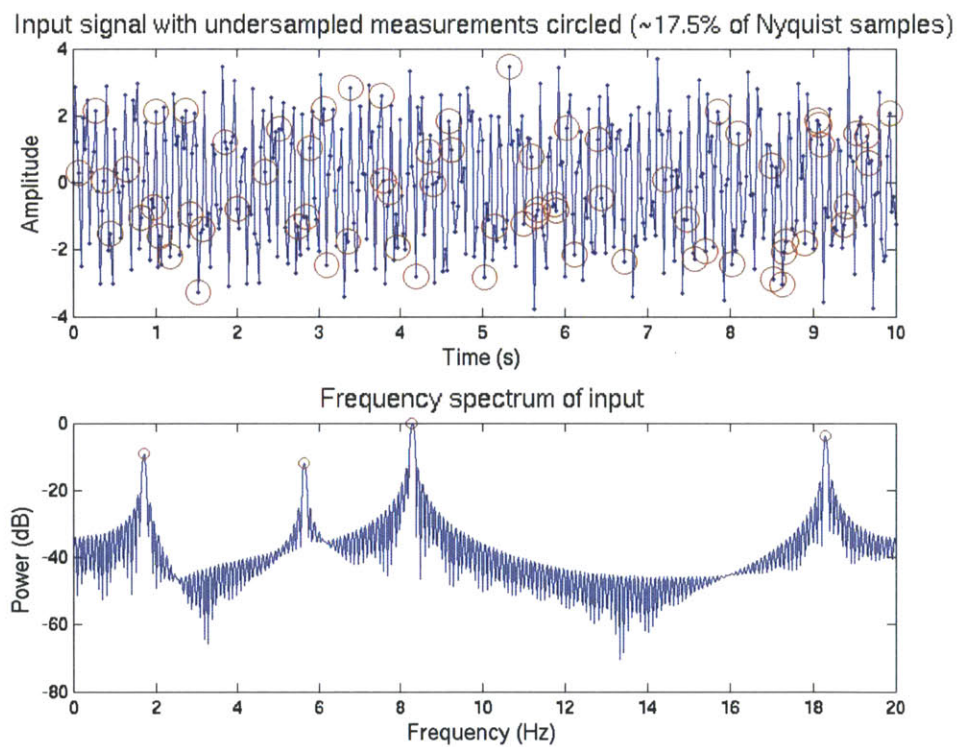

Figure 3-1.   Sample sum of sinusoids input signal and undersampled signal with power spectral density.

We know that the input signal is a sum of sinusoids, so the optimization program from Eq. 4 can be simply re-written as

$$\min \sum_{i=1}^{N=1024} |G(f_i)| \quad \text{subject to} \quad g(t_m) = \sum_{i=1}^{N=1024} \alpha_i \cdot \sin(2\pi f_i \cdot t_m), \quad m = 1, ..., M \ ,$$

34

where $G(f_i) = \alpha_i$, the weighted coefficient at the frequency f_i. Using the L1-MAGIC toolbox to minimize the ℓ_1 norm with equality constraints the frequency spectrum is recovered and the power spectral density $\left(20 \log \frac{|\alpha_i|}{\max |\alpha_i|}\right)$ is plotted in Figure 3. A cutoff of -75 dB was imposed so that if the calculated power value was less than the cutoff it was set to -75 dB. Looking at Figure 3-2 we can see that the relevant features of the frequency spectrum—the spike locations—are recovered within 10 mHz of the exact frequency locations, and in this ideal case the recovered spectrum is actual more accurate than the FFT of the full set of Nyquist required samples. The spike locations in this ideal input case are distinguished as any frequency with a power within the 30 dB of the maximum power.
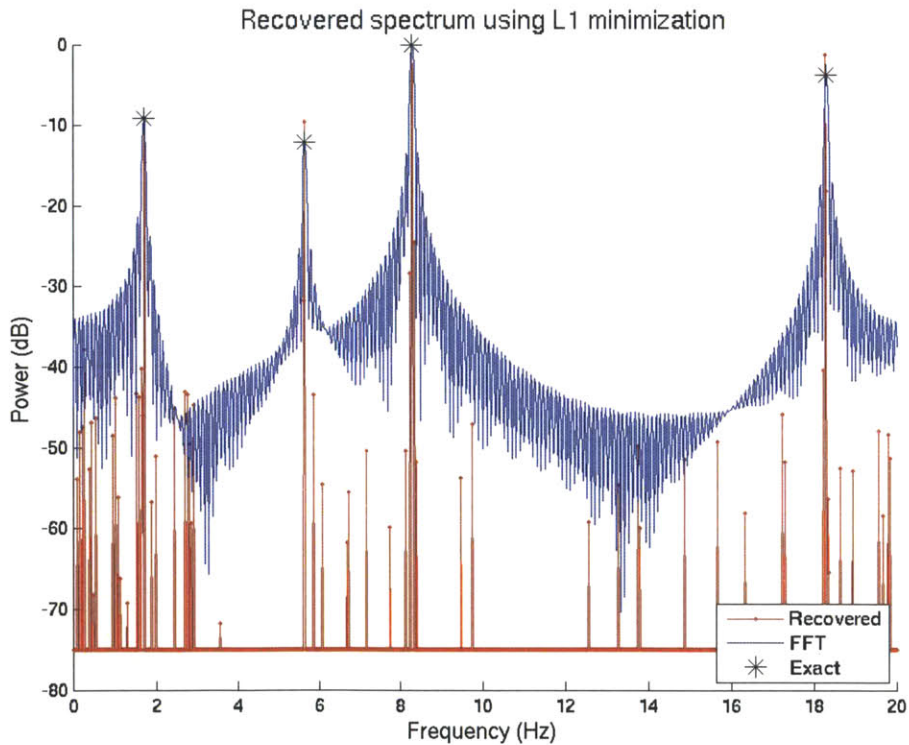

Figure 3-2.   Recovered frequency spectrum of sum of sinusoids input.

It is useful to investigate how the signal recovery is affected by changing the two relevant variables 1) percentage of Nyquist samples as altered by adjusting the optimization factor, C, and 2) input signal duration, T. We first need to define metrics to compare the quality of the recovered spectrums. The aim is to located the tremor frequency spikes, so we can define the maximum frequency error, $f_{\text{err, max}}$,

$$f_{\text{err, max}} = \max_{S'} \text{abs}(f_{j,\text{exact}} - f_{j,\text{rec}}), \quad j = 1, \dots, S',$$

where $f_{j,\text{exact}}$ is the exact frequency in the input signal, $f_{j,\text{rec}}$ is the recovered frequency, and S' is the number of frequency spikes indentified in the recovered spectrum. In this exploration where the input specturm is known, if the recovered spectrum contains erroneous spikes or misses spikes, the $f_{\text{err, max}}$ is set to -1, and the recovery is deemed a failure.

In addition to erroneous spikes, we can quantify the accuracy of the recovered spectrum, by defining a maximum power error, $p_{\text{err, max}}$,

$$p_{\text{err,max}} = \max_{N}(\log(p_{k,\text{rec}}) - \log(p_{k,\text{FFT}})), \quad k = 1, \dots, N,$$

where $p_{k,\text{FFT}}$ is the power calculated in the FFT of the input spectrum, $p_{k,\text{rec}}$ is the power in the recovered spectrum, and N is the recovery vector length. Because we expect all frequencies other than the input frequencies to be non-zero, the larger $p_{\text{err,max}}$ is the worse the accuracy of the recovered spectrum. An exceptionally high $p_{\text{err,max}}$ is an indication of erroneous spike detection.

Using the same setup as in the example in Figure 3-1, allowing the frequency locations to be set randomly, we can sweep C and recover the frequency spectrums. The recovered frequency spectrums are shown in Figure 3-3 for visual insight. Figure 3-4 plots both the $f_{\text{err, max}}$ and $p_{\text{err,max}}$ for each value of C, which is marked as the percentage

of Nyquist samples used for recovery. The $f_{err,\,max}$ stays below 10 mHz as soon as a threshold of number of samples is achieved and correct spike detection is achieved, however, the recovered spectrum $p_{err,max}$ decreases with increasing samples due to the increased constraints in the linear program concentrating the non-zero power levels around the frequency spikes. It is easy to see visually in Figure 3-3 that increasing C results in a more accurate spectrum, $p_{err,max}$ quantifies this accuracy.
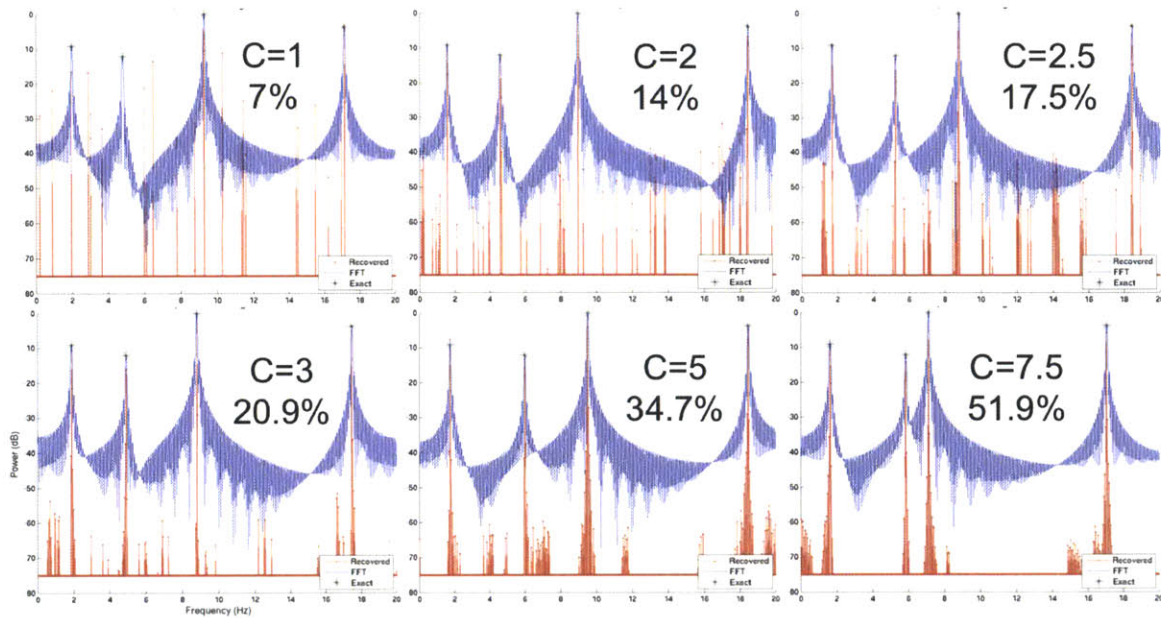

Figure 3-3.   Recovered frequency spectrums with increasing Nyquist samples.

Also following from the setup in the example in Figure 3-1, allowing the frequency locations to be set randomly, we now sweep T and recover the frequency spectrums. The percentage of Nyquist samples drops as T increases, so C is adjusted such that the percentage of Nyquist samples remains constant at 17.5%.
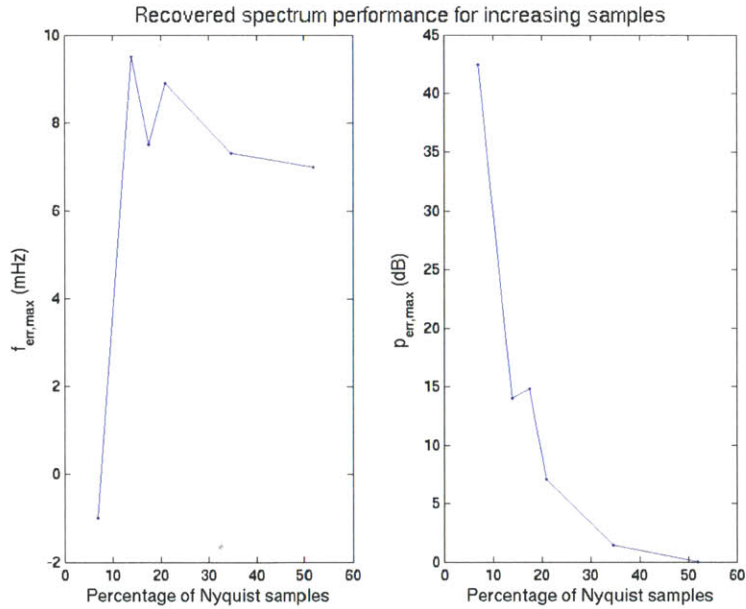
Figure 3-4.   Recovered spectrum performance for increasing Nyquist
samples.

The recovered freqneucy spectrums are shown in Figure 3-5 for visual insight. Figure 3-6 plots both the $f_{err,\,max}$ and $p_{err,max}$ for each value of T, as well as the percentage of Nyquist required samples in each case. The $f_{err,\,max}$ indicates the recovery failing for $T = 5$, and decreasing for increasing T. The $p_{err,max}$ decreases with increasing T; with the addition of more information, the recovery becomes more accurate. There is a tradeoff with increasing T: the linear program becomes more complex and takes longer to solve. The linear program with $T = 120$ solved on the order of minutes. This, however, does not increase the demands of the implant because it is all done on the reader side after data transmission.

Figure 3-5.   Recovered frequency spectrums with increasing *T*.
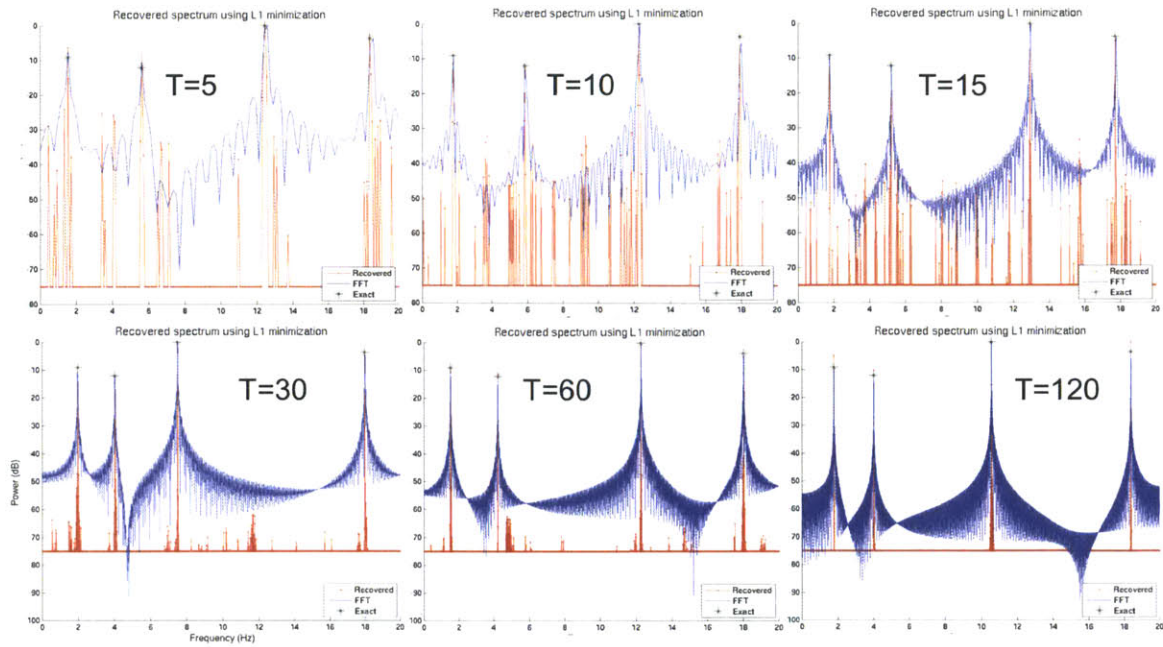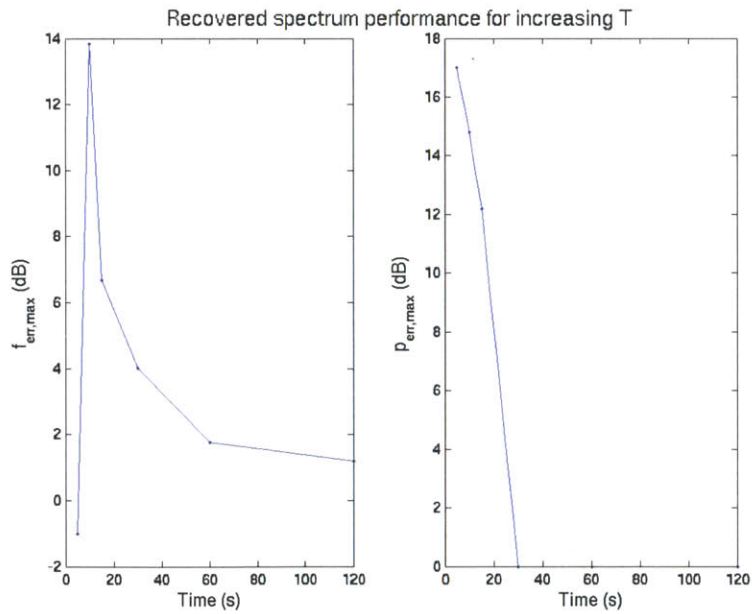

Figure 3-6.   Recovered spectrum performance for increasing *T* with
constant 17.5% Nyquist required samples.

We showed how the two relevant variables 1) percentage of Nyquist samples as altered by adjusting the optimization factor, C, and 2) input signal duration, T interact with each other to affect the recovery algorithm to both detect the frequency of the spikes and also eliminate erroneous spike detection. We can also add Gaussian white noise to create an input with an SNR of 15, and repeat the recovery experiment with $C = 2.5$ and $T = 10$. The recovered spectrum is plotted with a cutoff of -60 dB in Figure 3-7, and again the frequency locations are detected within 10 mHz. In the case of a noisy input, tremors were distinguished as any frequency with power within 20 dB of the maximum power. It is apparent that erroneous spike detection is complicated with introduction of a noise floor. Chapter 4 discusses the tremor spectrum and Parkinson's monitoring, and describes a design of a recovery algorithm specific to Parkinson's tremor monitoring.
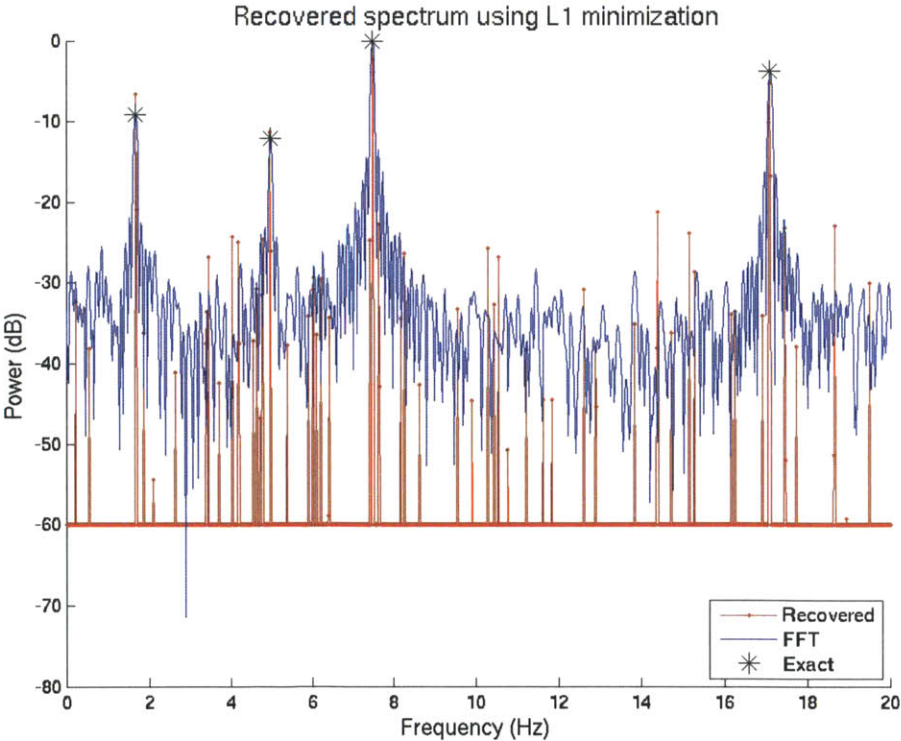

Figure 3-7. Recovered frequency spectrum of noisy sum of sinusoids input.

Chapter 4

Compressive sampling for tremor monitoring

In Chapter 3 we discussed the compressive sampling theory and the requirements behind exact recovery. Then the CS framework was practically applied as a sparse sampling technique and recovery algorithm on a simple input of a sum of sinusoids, which is sparse in the frequency domain and sampled incoherently in the time domain. Analysis was done on the parameters involved in the sampling and recovery to show their effect on the results. In this chapter we extend that to develop a framework for detecting and monitoring Parkinson's tremors.

4.1. Actigraphy and tremors

The monitoring of tremors is done via actigraphy, which is the long-term continuous measurement of activity with a small solid-state recorder. By monitoring the motion of a patient's wrist with an accelerometer a patient's tremors can be captured in time. However, the time representation of the data is not the relevant information. A time signal of a patient's gross motor activity, especially over long periods of time, would be a deluge of unnecessary data. In talking with Dr. Seward Rutkove of Beth Israel

Deaconness Medical Center, we learned that the frequency spectrum has the useful information: the tremor locations in frequency and their relative intensity. Long-term monitoring is useful for diagnosing and measuring the progression of a tremor. Monitoring efforts in the past [11] and [23] have tried to capture time scale information like tremor duration and number of tremor occurrences. However, in long-term monitoring these features are not important. The truth is that there is a lack of thorough understanding of the physical manifestations of Parkinson's disease and long-term continuous monitoring is needed to develop the necessary understanding to improve diagnoses and treatment.

As was discussed in Chapter 1, in diagnosing Parkinson's disease the tremor is the main symptom with no definitive test to confirm the presence of the disease. Many patients get misdiagnosed for several years because of the difficulty in assessing the tremor and other physical symptoms especially in elderly patients [24]. Tremors also vary with different influence factors like rest, mental concentration, walking and alcohol that make them difficult to diagnose and monitor with specificity within the context of doctor visits. With the severity of the progression of a neurodegenerative disease and the introduction of new treatments, the delay in treatment by several can be severe, and early diagnostic techniques are sought after [25]. Part of the cause of misdiagnosis is the existence of other tremors not as a result of Parkinson's disease.

While practical long-term monitoring has not been actualized, there has been research done on differentiating the different types of tremors that can exist in the body and in particular in reference to the frequency of the various tremors. A survey of various tremors is presented in [26]. Parkinson's tremors and essential tremors fall in the

frequency ranges 4-6 Hz and 5-10 Hz respectively. Understandably, Parkinson's tremors often get misdiagnosed as essential tremors. Essential tremor is defined as the presence of a tremor with an unknown cause and is more prevalent in the general population than Parkinson's tremors. In addition to these two common tremors, every person has physiologic tremors at low intensities at various frequencies from 0 to 20 Hz, therefore defining the effective bandwidth of the signal we wish to capture as 20 Hz. With no special techniques involved, we can guarantee complete acquisition of the data if we sample at a minimum frequency of 40 Hz.

4.2.   Compressive sampling of real tremor data

The band of frequencies is 20 Hz, it is sparse with only a few spikes at different frequencies, and while patients can have multiple tremors at different frequency, the most common presentation is a single tremor. Figure 4-1 shows tremor data from six subjects that were collected using low intensity velocity-transducing laser recording aimed at reflective tape attached to the subjects' finger recording the finger velocity. [27] All of the subjects show a Parkinson's tremor in the 4-6 Hz range. Subject 8 shows additional motor activity at two higher frequencies, subject 4 appears to have two tremors very close to each other in frequency. These patients represent the manifestation of tremors in Parkinson's patients and are used to devise the CS algorithm for tremor monitoring.

The goal is to set the two variables analyzed in Chapter 3: 1) percentage of Nyquist samples as altered by adjusting the optimization factor, C, and 2) input signal duration, T, such that the frequency of the tremors are detected, and in the case of multiple tremors, that the relative intensities are captured. The real tremor data suggests a sparsity index of

3 ($S = 3$). As a starting point, we can use the setup from Chapter 3, where $C = 2.5$ and $T = 10$, which translates into 13% of the Nyquist required samples. Then we can push the limits of these variables to minimize the percentage of samples while maintaining the ability to capture the tremor frequencies—the "important" features.
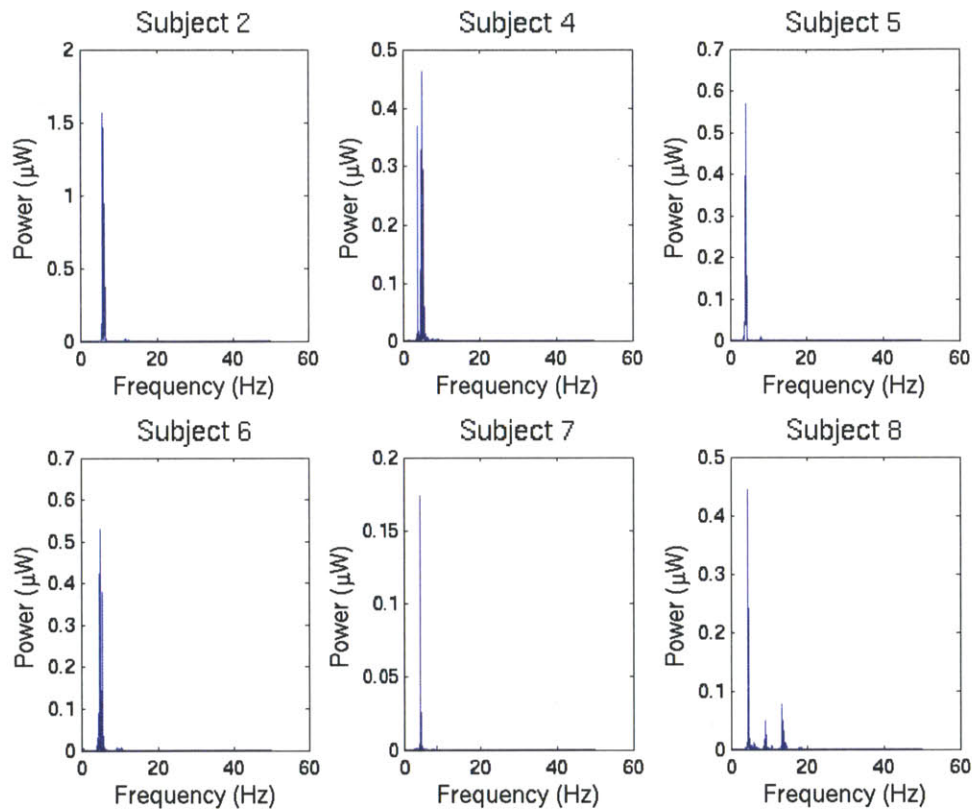
Figure 4-1.   Frequency spectrum for 6 subjects with Parkinson's tremors

Using a recovery algorithm with $C = 14$ and $T = 30$ translating to 26.7% of the Nyquist required samples, we recovered the frequency spectrums for all 6 subjects, which are shown in Figure 4-2. The window size, T, was limited to 30, because Subject 5 was limited in data length and only had 30 seconds of gross motor activity recorded. Full size representations are shown in Appendix A. The $f_{err,\,max}$ for each subject is plotted in Figure

44

4-3 and is shown to be less than 100 mHz. In the case of real tremor data, tremors were

distinguished as any frequency with power within 10 dB of the maximum power.
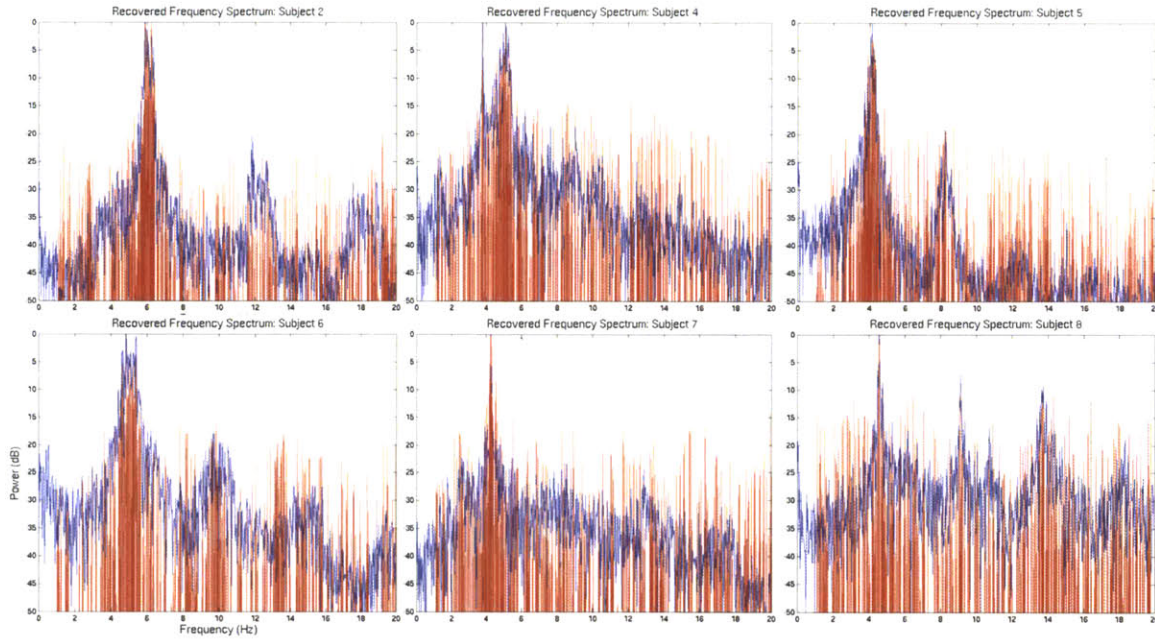

Figure 4-2.   Recovered frequency spectrums using 26.7% of the
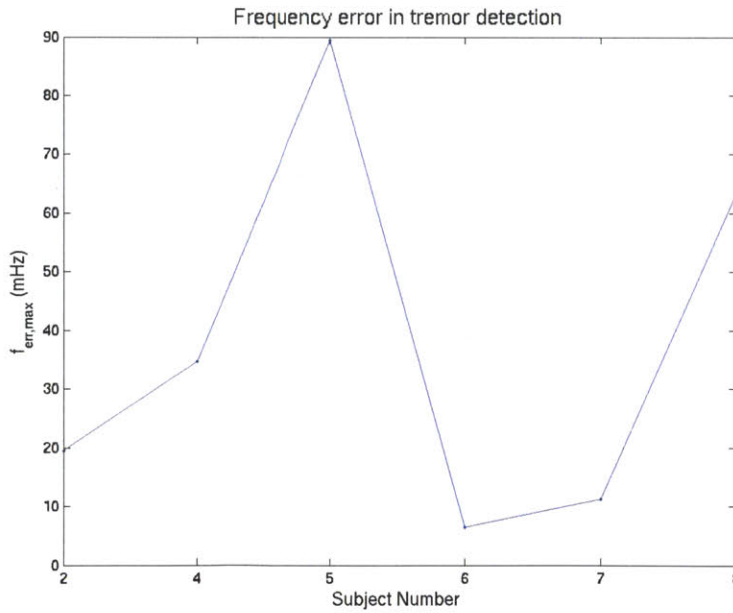
Nyquist required samples.

Figure 4-3.   Maximum frequency error in tremor detection for each

subject with compressive sampling recovery.

This is just one random sampling of each subject. We can run 100 test cases for each subject with their own random sampling and record the success and error in the recovery. A summary of these results are in Table 4-1. The detection success indicates how many cases resulted in correctly detected exactly the number of tremors known to be there, where again a tremor is identified as a signal with a power within 10 dB of the maximum power. We see that the recovery outlined works perfectly for Subjects 2, 5, 6 and 7 and works well for Subject 4, and seems to mostly fail for Subject 8. For each subject the mean $f_{err, max}$ is recorded as well. We see that the mean $f_{err, max}$ in below 100 mHz, except for the case of Subject 8 which is slight higher at 126.4 mHz.

Table 4-1.   Recovery success and error for 100 cases of random
sampling of real tremor data

Subject #	Detection Success	Mean $f_{err,max}$ (mHz)
2	100	27.4
4	84	44.5
5	100	81.8
6	100	73.0
7	100	6.5
8	30	126.4

The 100 case experiment suggests that the detection fails for Subject 8 the majority of the time, and this would be true if we were using an automated method for detecting tremors. This is not un-expected, and refers back to our analogy with lossy compression. We expect to lose some unimportant information, but retain all the important features. Just as in the case of JPEG compression, where perceputally irrelavent information is lost

but the image intergrity is retained, here too the perceptual integrity of the tremor locations in frequency is retained. The automated detection of tremors fails in the case of Subject 8, not because the spectral recovery failed but because the automated detection method used for analysis failed. Our aim is to provide a visual representation for a clinician to observe, and not automate the visual analysis.

We can look at two representative cases for Subject 8 shown in Figure 4-4 and Figure 4-5, where there is an automated detection success and a detection failure respectively. The two recovered spectrums are visually similar, and a trained individual would gain the relevant information. Additionally, in all cases the frequency with the peak power, which is the peak of interest is detected. The long-term nature of the monitoring makes an argument that this technique can successfully detect the freqency locations of a patient's tremor over long periods of time with a trained clinician processing the data.
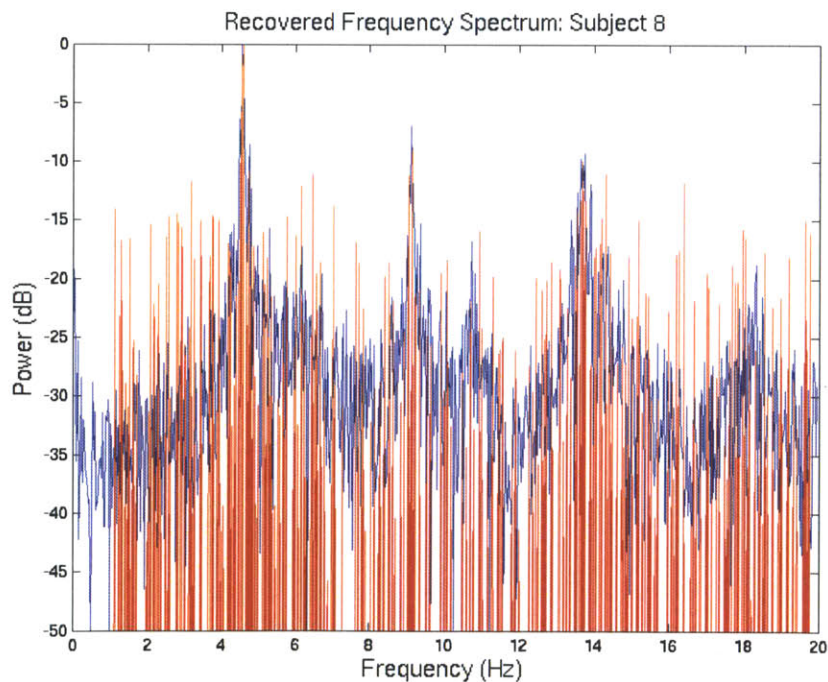

Figure 4-4.   Recovered frequency spectrums using 26.7% of the Nyquist required samples with automated detection success.
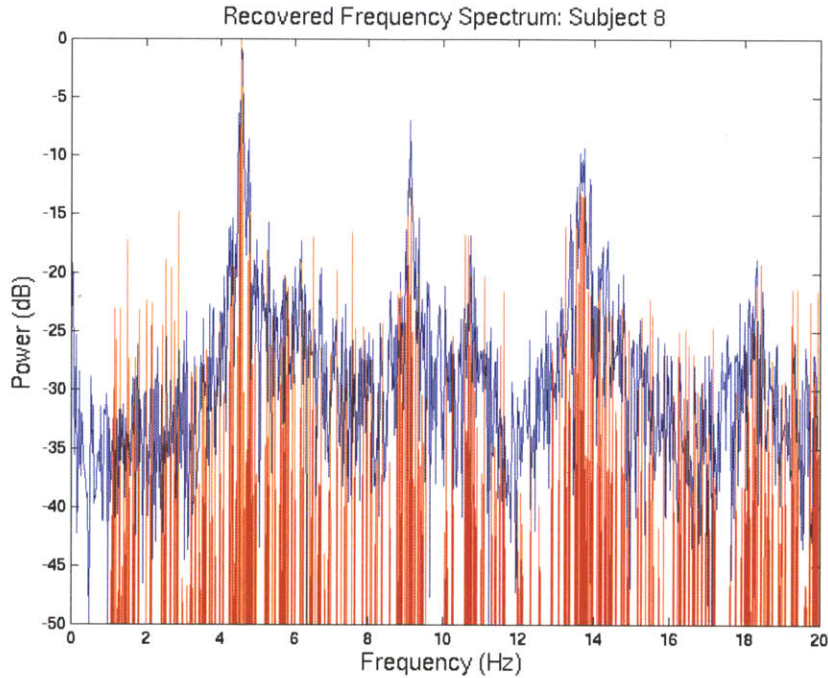
Figure 4-5.   Recovered frequency spectrums using 26.7% of the
Nyquist required samples with automated detection failure.

4.3.   Sparse sampling in hardware

Section 4.2 delineated the process of taking significantly less samples than the Nyquist rate would require and still recovering the input spectrum accurately.   However, the underlying belief is that the samples are random. This is an easy task if we record all of the data to the reader and finally a computer for post processing, which is what was done in section 4.2.   Realizing the random sampling concept in actual hardware is less straightforward, and less common. The aim is to reduce the energy consumed by writing to memory as well as converting with the ADC on the implant. We seek therefore to only convert and write the samples required as shown by our compressive sampling analysis,

and no more. The implementation of this desired strategy is described in the following paragraphs.

A hardware implementation of random sampling is to generate a pseudo-random number with a linear feedback shift register (LFSR). A maximal-length 8-bit Galois LFSR with three taps is shown in Figure 4-6. The LFSR output cycles through a predictable set of output combinations, and an 8-bit maximal length LFSR has a period of 255. That is, if the current state of the LFSR is known, then the output can be predicted. In this regard the output sequence is not random, but exhibits statistical randomness.
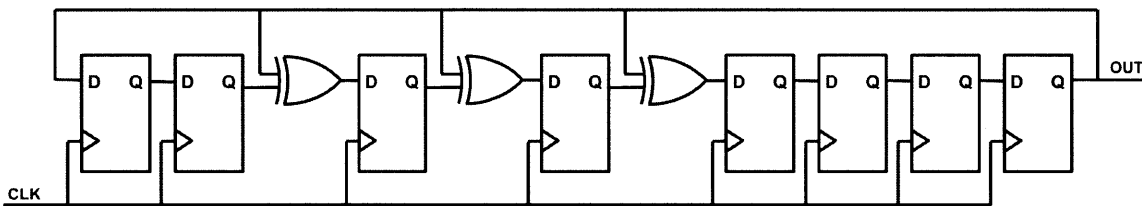

Figure 4-6.   8-bit Galois LFSR with 3 taps.

While the output stream generated with a maximal-length will exhibit statistical randomness, it will also be DC balanced to within one bit. If a "1" corresponds to a sample used and "0" to one that is not used, this results in capturing 50% of samples, which as we know from section 4.2 is more than necessary. One way to combat this is to change the location of the taps, which will break the balance property, but not enough to fall below 40%.

Our solution is to design three separate LFSRs with different tap locations as described in Table 4-2, and then logically AND the output sequences together. This generates a sequence that is a 1 only 26.3% of the time. This implementation fixes the percentage of samples for a given time window. That is to say, with an 8-bit LFSR, for

every 255 samples, we are guaranteed to acquire 26.3%. Using the output sequence as a write control to only convert and write to memory when logic high will allow us to sample sparsely on-chip without much overhead and reduce energy consumption by turning off conversion.

The initial seed of each LFSR was fixed as [1 1 0 0 1 0 0 1] in order to produce the ratio of ones to zeros given above. It follows that by changing the seed would alter the percentage of required samples, and a higher percentage would result in a better spectral recovery while a lower percentage would result in a worse spectral recovery.

Table 4-2.   8-bit LFSR tap locations to generate unbalanced sequence

LFSR #	Tap Locations
1	8, 5, 4, 2
2	7, 5, 4, 3
3	8, 6, 4, 2

In addition to sampling randomly, we must know when the random samples occur to complete the recovery algorithm. Using the LFSR to generate a random-like sequence with the desired ratio of "1"s simplifies the task, by simply transmitting the initial state the reader can know exactly which samples are chosen. The hardware method proposed here will enable us to sample sparsely in real time without a significant increase in overhead, and provide an almost 75% reduction in samples converted and written to memory.

4.4. Simulated tremor signal recovery

We developed code in MATLAB to emulate the hardware methodology outlined in section 4.3 and replaced the random sampling in post-processing with the pseudo-random sampling basis generated by the LFSRs. This new recovery technique was again applied to the real tremor data with $T = 30$. The fudge factor, C, is no longer used because the number samples are now determined by the LFSRs. The recovered spectrums are shown in Figure 4-7. Full size plots can be found in Appendix B. All tremors for each subject are detected and the $f_{err, max}$ for each subject is plotted in Figure 4-8 and is shown to be less than 100 mHz.

With the use of LFSRs, we can generate an incoherent sampling basis that is random-like and is able to recover the tremor locations with accuracy. It is important to remember that compressive sampling does not require random samples, but just suggest random samples as a sufficient incoherent sampling technique. Though the sequence generated here is not random and is, in fact, predetermined, it is effective in recovering the Parkinson's tremor data.
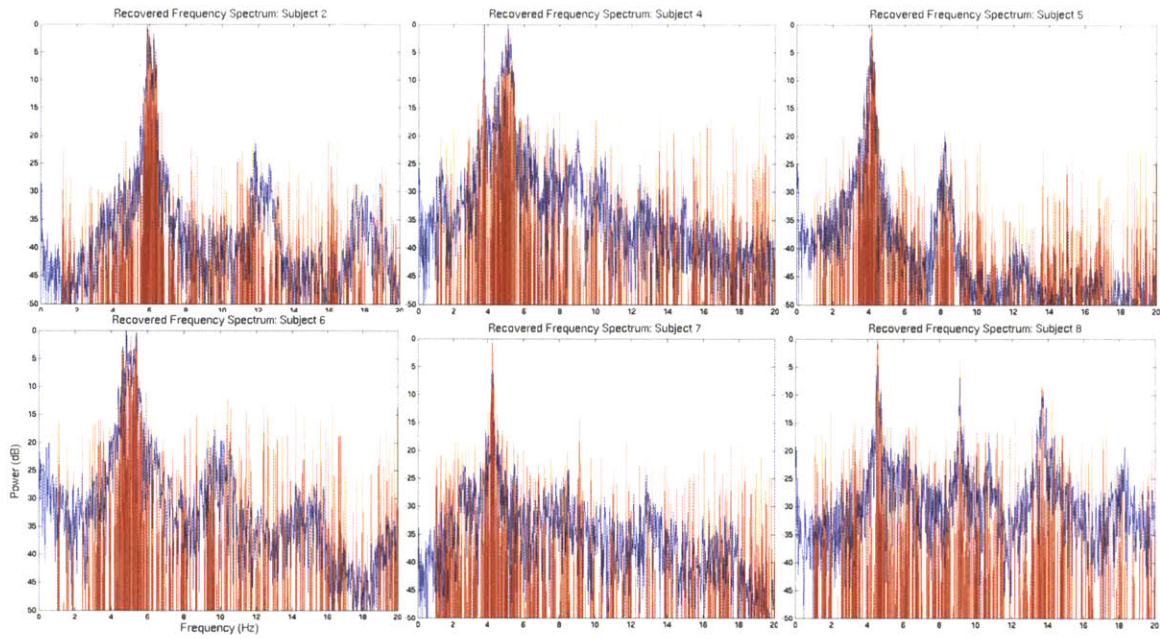
Figure 4-7.   Recovered frequency spectrums using 26.3% of the
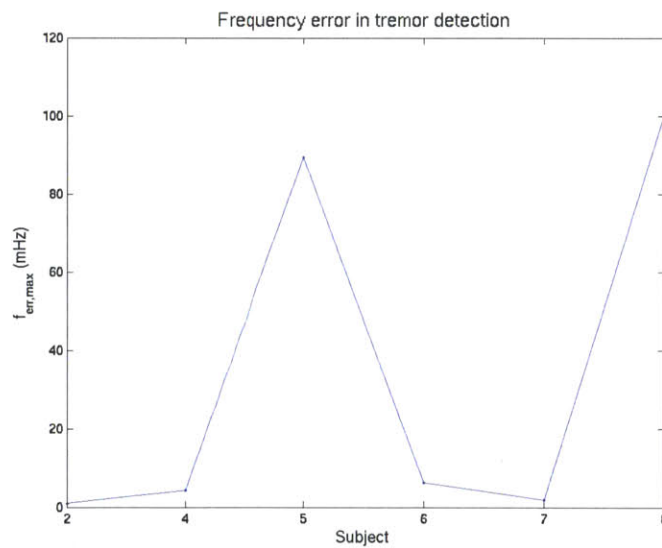Nyquist required samples using LFSR to sample randomly.

Figure 4-8.   Recovered frequency spectrums using 26.3% of the
Nyquist required samples using LFSR to sample randomly.

Chapter 5

Techniques for low power SAR ADC design

In the case of minimally sized implants with a limited energy source, the need for low-power, energy-efficient circuits becomes especially pressing. In the proposed use model, the system energy demands for data processing and transmission are pushed outside the implant and onto a reader via RFID communication. Compressive sampling is an example of pushing the demands of data processing off chip to save energy by writing to memory significantly less. The sparse sampling also motivates not expending energy to convert the unneeded samples, and motivates the design of a low energy ADC suited for this application and slow sampling rates contrary to modern ADCs with high sampling rates. The microImplant platform is one of the few systems that will benefit from the ongoing improvement in ADC techniques.

The microImplant platform proposed here assumes that the biometrics of interest are slow moving and low in frequency, as indeed tremor signals are. In this chapter, we consider the problem of choosing the best ADC technique for a system that requires ultra-low-power operation, that is free to sample slowly, that does not require very high resolution, and that will *utilize compressive sampling*. We can immediately eliminate

pipelined and flash ADCs, which can achieve needlessly high frequency conversion rates, and at the cost of power consumption and in the case of flash ADCs die area. [28] By implementing a successive approximation ADC, however, we can trade off speed for lower power consumption and smaller size. Furthermore, by employing a dynamic comparator, static power loss can be reduced in the comparator to leakage levels and the energy consumption in switching the binary-weighted capacitor array can be reduced by employing adiabatic circuit techniques. In this chapter, we propose that a SAR ADC that utilized adiabatic capacitor charging is ideally suited for the microImplant platform. We describe the design and implementation of such a converter.

5.1. ADC topology

A successive approximation ADC implements a binary search algorithm to convert an analog voltage into a stream of bits stored in a successive approximation register (SAR). We examine an N-bit, single-ended charge redistribution ADC consisting of a binary weighted capacitor array and an operation amplifier (opamp) comparator as shown in Figure 5-1. The SAR ADC has two modes of operation—sampling and bit cycling. During sampling the input voltage is sampled on the capacitor array by connecting the opamp in feedback and the capacitors to the input voltage, V_{in}. The feedback loop then opens and the capacitors connect to ground to begin the bit cycling phase. Before bit cycling begins, we note the voltage at node X is
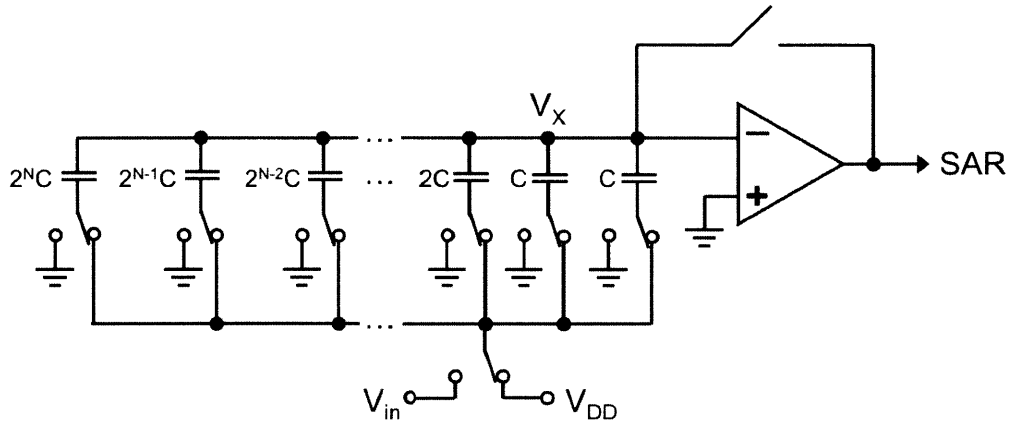
$$V_X = -V_{in} \ .$$

Figure 5-1.   Single-ended charge redistribution ADC topology

Bit cycling is the clocked switching of each capacitor from largest to smallest to redistribute charge on the array; the SAR controls the capacitor switching and also retains the bit decision results. For instance looking at the ADC in Figure 5-1 during the first bit cycle, the largest capacitor connects to V_{DD} injecting charge into the capacitor array evenly divided between the largest capacitor and the rest of the array. The voltage at node X, V_X, adjusts to

$$V_X = -V_{in} + \frac{V_{DD}}{2} ,$$

and is compared to ground in the comparator, which outputs the first bit decision sent to the SAR. We can see from Eq. 2 if V_{in} is greater than $V_{DD}/2$ the comparator outputs a logic '0', and if V_{in} is less then $V_{DD}/2$ the comparator outputs a logic '1'. If the bit decision results in a '1' the bit cycled capacitor is held at V_{DD}, otherwise the bit cycled capacitor is connected to ground and bit cycling resumes with the next largest capacitor until N bit decisions are made. Therefore it takes at least N clock cycles to complete one conversion making the sampling frequency at best equal to a factor of N smaller than the clock frequency or f_{clk}/N.

For better accuracy, we examine a differential topology shown in Figure 5-2. Conventional differential charge redistribution ADCs require two voltage references for sampling and bit cycling in order to maintain a single common mode throughout operation, which is necessary for accurate operation when an opamp comparator is used. In the sampling phase, the inputs of the comparator are pre-charged to V_{cm}, where $V_{cm} = V_{DD}/2$, and the bottom plates of both top and bottom capacitor arrays are connected to the differential input. The pre-charge switches open and the bottom plates of the capacitor arrays are set to V_{cm} holding the sampled input voltage before bit cycling begins. In bit-cycling operation, the bottom plates of the two corresponding capacitors being bit-cycled from both arrays are set to V_{DD}, and after comparison, depending on the bit-decision made, one is set to ground and the other is held at V_{DD}. The charge redistribution effects can be easily seen by looking at the voltage waveforms on the inputs of the comparator as shown in Figure 5-3.

In our microImplant application, where power management is essential, generating an extra voltage source will only add to our energy management budget. Because we already plan to replace the opamp comparator with a dynamic comparator, we can leverage this and redesign the topology using a single voltage supply thereby reducing the energy management demands. Figure 5-4 and Figure 5-5 show the differential topology with an 8-bit resolution and the new charge redistribution waveforms respectively. The charge usage is the same as in the double reference topology, but the charge redistribution is no longer done on both capacitor arrays identically. The dynamic comparator design is discussed in the next section.
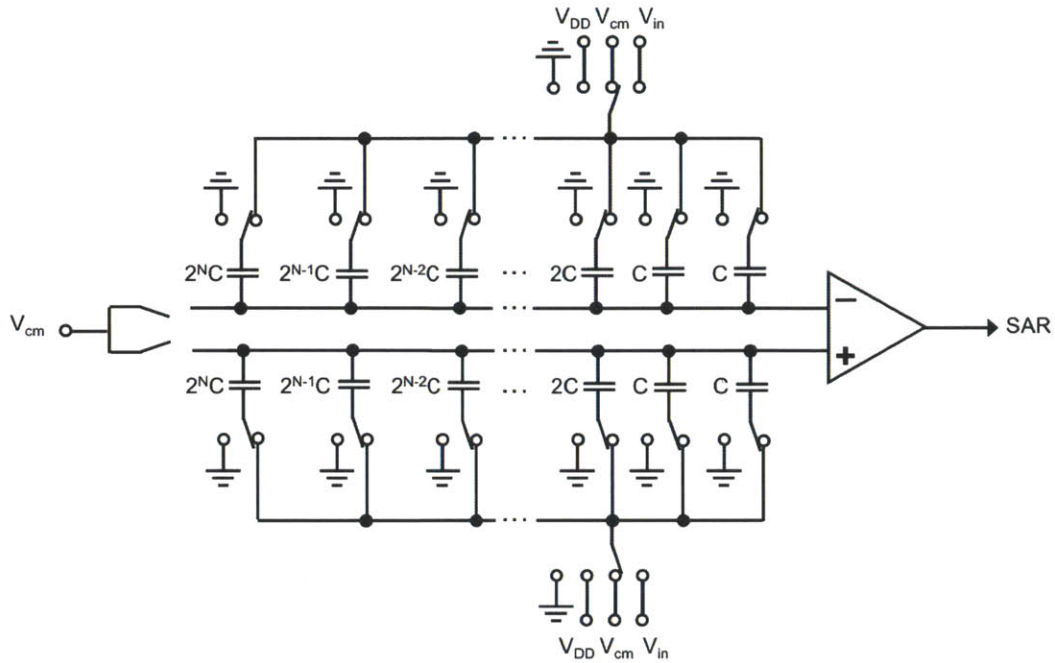
56

Figure 5-2.   Differential SAR ADC topology with N-bit resolution and
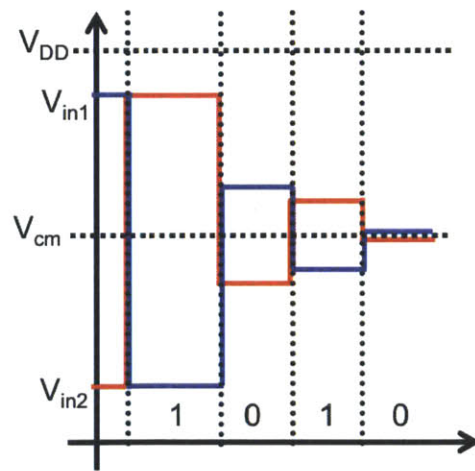opamp comparator

Figure 5-3.   Waveforms of comparator inputs demonstrating double
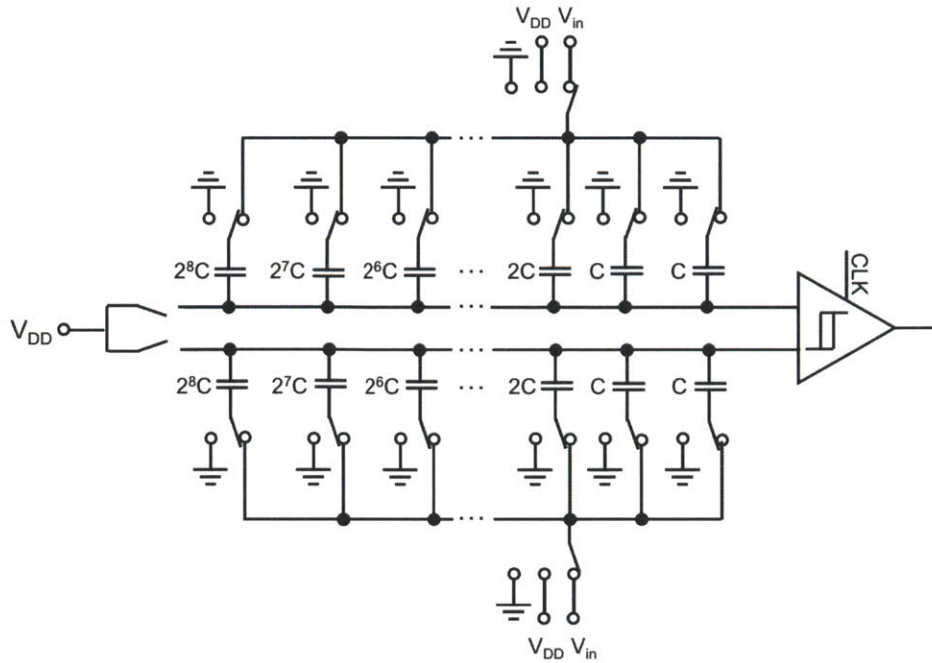reference supply charge redistribution

Figure 5-4.   Differential SAR ADC topology with 8-bit resolution and
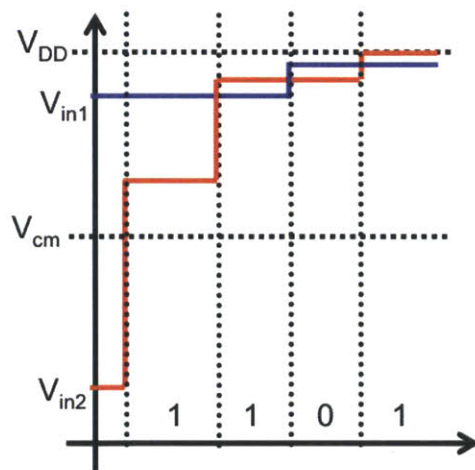dynamic comparator

Figure 5-5.   Waveforms of comparator inputs demonstrating single
reference supply charge redistribution

Intuitively, smaller capacitances in the capacitor array would amount to a smaller energy dissipation; however there are constraints imposed on the size by noise and fabrication processes. Thermal noise on capacitors, or kT/C noise, is represented as a noise voltage, v_n, given as follows:

$$v_n^2 = \frac{kT}{C} \, ,$$

where v_n is the rms noise voltage across the capacitor, C, T is the operating temperature in Kelvin, and k is Boltzmann's constant (1.38×10^{-23} JK^{-1}). Given that our ADC resolution is 8 bits, we can calculate an LSB (least significant bit) voltage with the assumption of $V_{DD} = 1$ V:

$$\mathrm{LSB} = \frac{1}{2^8} = 3.9 \, \mathrm{mV} \, .$$

If we impose

$$v_n < \frac{\mathrm{LSB}}{2} \, ,$$

then 1.09 fF $< C$ with T = 300 K. A capacitor that small is not realizable in a 180 nm process, so for fabrication reasons, the unit capacitor was set at 20 fF making the largest capacitor in the array 2.5 pF. It is worth noting how difficult the design becomes if even one or two more bits are required, as the size of the largest capacitor balloons up exponentially as the number of bits.

5.2. Comparator design

As discussed, a dynamic comparator is desirable in our application because of its lack of a static power draw and smaller area. The dynamic comparator has two stages—a

preamplifier and latch stage shown in Figure 5-6—and is clocked with the digital control signal CLK. When the CLK is low, the outputs of the preamplifier stage are pre-charged to VDD by transistors M4 and M5, and the outputs of the latch stage are pre-charged to GND by transistors M10 and M11. M14 and M15 also pre-charge the source terminals of transistor M8 and M9 to ensure there is no current flowing in the latch stage prior to the comparison phase when the CLK is high. The CLK then switches high, and the differential input voltage is amplified to trigger the latch stage for bit decision. The latch stage consists of two inverters in feedback with each other, which continue to hold the comparator output values while the CLK is high after the transistors have all stopped drawing any current.
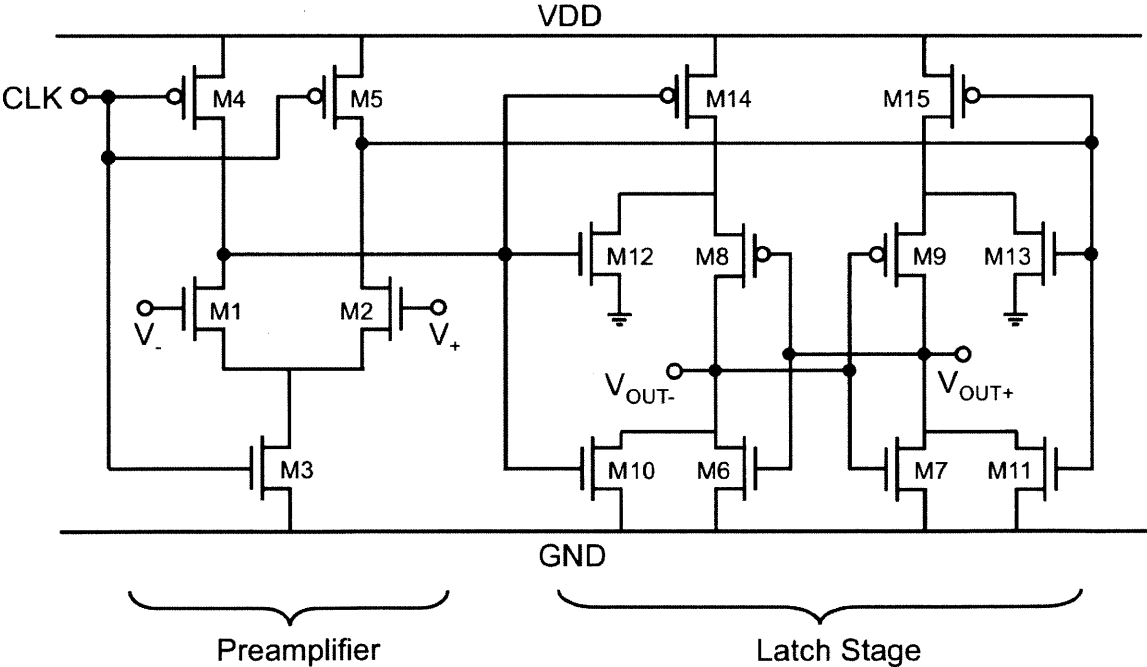

Figure 5-6.   Dynamic comparator topology.

This circuit implementation varies from the convention topology because the latch timing is determined by the output of the preamplifier stage and not an inverted CLK.

This is beneficial because only a single phase CLK is necessary, however, as a results the comparator will operate slower than the conventional design. This trade-off is willingly made because our CLK period is more than long enough to make this a worthwhile concession.

5.2.1. Digital offset correction

The dynamic comparator is used in the ADC to eliminate extraneous static current but is still susceptible to transistor mismatch. We employ the digital offset correction technique from [29] to compensate for this mismatch. Digital offset correction is ideal for this application because, again, we eliminate the need for static current and additional energy loss in the comparator.

A calibration phase is introduced prior to comparison, in which a charge pump adjusts the bias current inside the comparator to correct for offset caused by mismatch in the transistors. Figure 5-7 shows the calibration and comparison modes of the comparator. Two bias transistors are place in parallel with the input transistors (M1 and M2 in Figure 5-6) of the preamplifier stage. One is biased with voltage, V_b, and the other is connected to a calibration capacitor, C_{cal}, with the voltage, V_{cal}, across it. During calibration the inputs are tied together, and a charge pump adjusts V_{cal} by charging or discharging C_{cal} until the outputs alternate between "1" and "0". The charge pump in the offset calibration circuitry does include a static bias current, but because of the slow operating frequency of the comparator this static current is designed at leakage current levels. In future designs the charge pump can be switched off when not in the calibration mode.
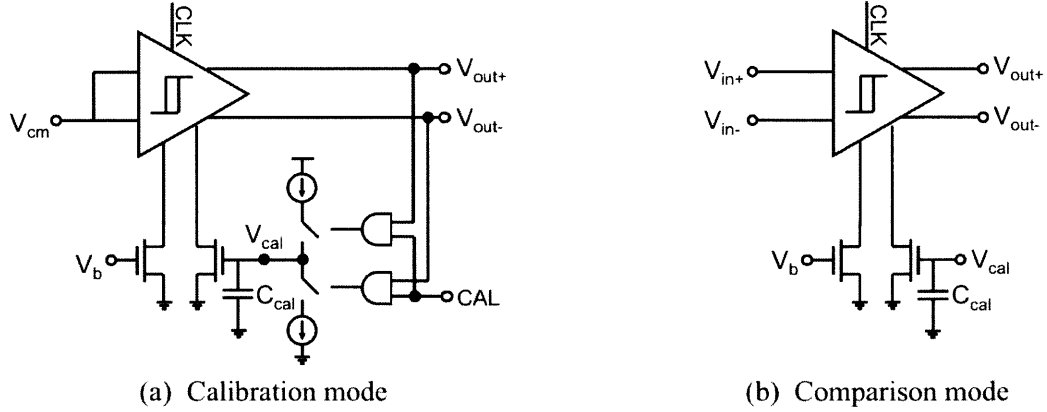
| (a) Calibration mode | (b) Comparison mode |

Figure 5-7.   Operating modes of dynamic comparator

According to [29] the calibration capacitor, C_{cal}, and sampling clock frequency, f_{clk}, are related as

$$V_{offset} = \frac{I_{CP}}{f_{clk} \times C_{cal}} ,$$

where V_{offset} is the input referred offset voltage and I_{CP} is the charge pump output current. Because of the low sampling clock frequency in this application, a higher clock frequency is applied during calibration mode to keep the calibration capacitor and charge pump output current small. Calibration is done for a period less than ½ of the clock period, T_{clk}, around every negative clock edge of the clock to ensure accuracy for the comparison on the positive clock edge. The calibration capacitance is chosen to minimize area and ensure V_{cal}, the voltage across C_{cal}, leaks by no more than one LSB voltage.

5.3.   Adiabatic charging

The binary weighted capacitors of the capacitor array enables the binary search algorithm implemented in the SAR ADC, but the charging and discharging of this array in each bit

decision is a major drain of energy in the ADC now that we have replaced the traditional opamp comparator with a low energy dynamic comparator. By employing adiabatic charging, we can reduce the energy loss, and with the aid of a self-stabilizing ramp circuit actually reuse charge that in conventional SAR ADCs would be wasted, which is advantageous with a non-traditional ultracapacitor supply.

The term adiabatic is defined as a process in which heat does not enter or leave a system [30]. From an electronics standpoint, this means a circuit or system that has no thermal loss or, more generally, no energy loss. By the material nature of circuit elements, we know this is impossible, so the term has been adopted in the circuit community to mean any time less than $\frac{1}{2}CV^2$ energy is lost by charging a capacitance, C, to the voltage, V. This expression is derived by looking at the simple circuit in Figure 5-8, where a capacitor is modeled in series with its equivalent resistance, R, and a lossless switch connected to a static voltage source, V. The capacitor is initially fully discharged with no voltage drop across it and fully charges up to V when the switch is closed.
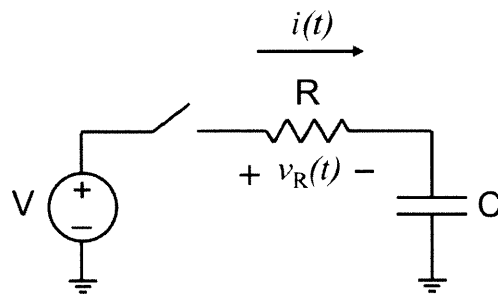

Figure 5-8.   Simple switched charging of a capacitor with a static
voltage source

The energy loss from the resistor is

$$|E| = \left| \int_0^\infty i(t) * v_R(t)\, dt \right| = \left| \int_0^\infty \frac{V}{R} e^{-t/RC} * V e^{-t/RC}\, dt \right| = \frac{1}{2}CV^2 \; . \qquad (5)$$

In Figure 5-9 we replace the static voltage supply with a time varying ramp voltage that

swing from 0 to V over a time period T

$$v_s(t) = \begin{cases} \dfrac{V}{T}t, & 0 \le t \le T \\ 0, & T < t \end{cases}$$

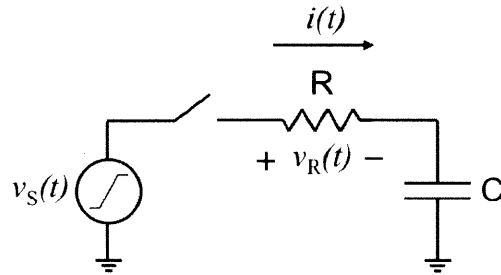and again close the switch at $t = 0$ with the capacitor initially discharged.

Figure 5-9.   Simple switched charging of a capacitor with a ramp
voltage source

We derive the differential equation pair

$$\dfrac{V}{T}t = v_C(t) + RC\dfrac{dv_C(t)}{dt} \quad \text{for} \quad 0 \le t \le T$$

$$V = v_C(t) + RC\dfrac{dv_C(t)}{dt} \quad \text{for} \quad T < t$$

where $v_C(t)$ is the voltage across the capacitor and with the initial conditions of

$v_C(0) = 0$ and $v_C(\infty) = V$. Solving for $v_C(t)$ and substituting the following identities

$$v_R(t) = v_S(t) - v_C(t)$$
$$i(t) = C\dfrac{dv_C(t)}{dt}$$

into Eq. 5 results in

$$|E| = \left| R\left(\dfrac{CV}{T}\right)^2 \left[\int_0^T \left(1 - e^{-t/RC}\right)^2 dt + \int_T^\infty \left(e^{T/RC} - 1\right)^2 e^{-2t/RC} dt \right] \right| .$$

After evaluating the integral and assuming $T \gg RC$ the energy lost in the resistor is

$$|E| = \frac{RC}{T} CV^2 \ .$$

With a $T > 2RC$, the energy loss reduces from the typical $\frac{1}{2} CV^2$. The assumption that

$T \gg RC$ is a reasonable one to make since modern CMOS processes have charging times

at nanoscale and the tremor signal we are monitoring requires a low sampling frequency,

we are ideally suited to extend T without any frequency penalty.

Circuits using inductors have been used to generate slow switching digital signals

mimicking ramp voltages for the purpose of driving digital circuitry [31], thereby

reducing energy loss due to parasitic capacitance charging and discharging. However, it

has been shown that the energy savings decrease with the square root of the charging

time, T [32].

Another technique to implement adiabatic charging is to employ stepwise charging.

Instead of charging a capacitor with a linear ramp over T seconds, we charge the

capacitor in N equal steps over T seconds. Using the results from a single unit step in Eq.

4 again, we write the resistor energy loss as

$$|E| = \left| N \left(-\frac{1}{2} C \left(\frac{V}{N} \right)^2 \right) \right| = \frac{1}{2N} CV^2 \ ,$$

thus reducing the energy loss by a factor of N. The idea of step charging was seen in [33]

to reduce the energy loss when charging and discharging large capacitors in a SAR ADC.

This technique is especially advantageous in our application with a limited energy

source, because in the discharge cycle the stored energy gets reclaimed by the tank

capacitors generating a voltage ramp used for charging and discharging, which is

discussed later in this section. Normally complex circuitry is necessary for energy

recovery with a battery; the most common type is seen in efficient charge recovery logic

(ECRL) [34]. However, in our case, the supply is a capacitor and we avoid this design requirement. Again, we take advantage of the low sampling frequency to decrease energy loss, because we are not designing for high frequency and can implement step charging with no sampling frequency penalty.

5.3.1 Self-stabilizing ramp circuit

The most reduction in energy loss by charging in N steps would be to have N uniformly distributed voltage sources; however, this is extremely impractical in energy management to maintain so many voltage sources. A self-stabilizing ramp circuit described in [35] uses large tank capacitors to store the discrete voltage levels. The topology used to create N steps between 0 and V_{DD} is shown in Figure 5-10.
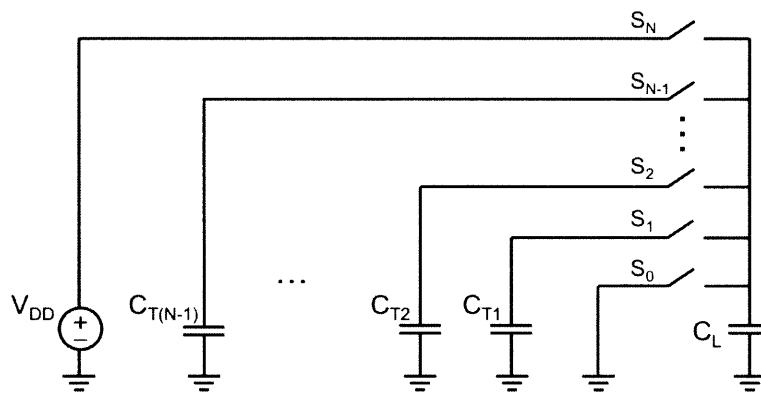

Figure 5-10.   Self-stabilizing circuit to generate an N-step ramp voltage.

The switches are closed in sequence one at a time from S_1 to S_N to effectively step charge C_L up to V_{DD}, and then closed in reverse order from S_{N-1} to S_0 to effectively step discharge C_L to ground. If the tank capacitors are sized much larger than the load capacitor, the voltages stored on them uniformly distribute between 0 and V_{DD}. To see why this happens, we can analyze the simple case of just one tank capacitor seen in Figure 5-11.

Initially both capacitors, C_T and C_L are discharged, and we start the switching sequence by closing S_1; the system remains unchanged because initially we have no
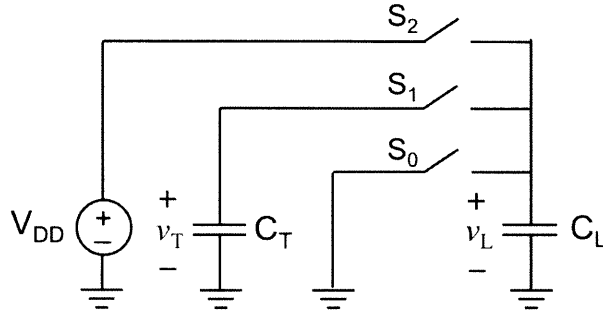

Figure 5-11. Self-stabilizing ramp circuit with one tank capacitor.

charge in the system. Second in the sequence, we close S_2 and charge C_L up to V_{DD} effectively injecting the system with a total charge of $Q_{tot} = Q_L + Q_T = C_L V_{DD}$. It is helpful to note that the total charge stored in the system is only dependent on C_L, because C_T is thus far still fully discharged. The load capacitor has now been step charged up, even though the first step was irrelevant because we started the sequence with no charge in the system. The switching sequence continues by closing S_1 implementing the first step in a two-step discharge of C_L; we have now created a system of two capacitors with different voltages and charge levels being switched together. The capacitors behave accordingly to conserve charge and therefore they share the total charge in the system, Q_{tot}, proportionally to their capacitances and settle to the same voltage level:

$$Q_L = \frac{C_L}{C_{tot}}(Q_{tot}) = \frac{C_L}{C_{tot}}(C_L V_{DD}) \tag{6}$$

$$Q_T = \frac{C_T}{C_{tot}}(Q_{tot}) = \frac{C_T}{C_{tot}}(C_L V_{DD})$$

$$v_L = v_T = \frac{Q_{tot}}{C_{tot}} = \frac{C_L V_{DD}}{C_{tot}}, \tag{7}$$

where $C_{\text{tot}} = C_{\text{T}}{+}C_{\text{L}}$. The second step in the two-step discharge completes by closing S_0; C_{L} is discharged to ground and C_{T} remains unchanged with the same Q_{L} and v_{L} calculated in Eq. 6 and Eq. 7 respectively.

The switching to this point completes one full charge-discharge cycle, and we can see that we have now introduced charge into the system, which remains on the tank capacitor. We can update the charge and voltage on C_{T} after another full charge-discharge cycle as

$$Q_{\text{T}} = C_{\text{L}} V_{\text{DD}} \left(\frac{C_{\text{T}}}{C_{\text{tot}}} + \frac{C_{\text{T}}^3}{C_{\text{tot}}^3} \right)$$

$$v_{\text{T}} = C_{\text{L}} V_{\text{DD}} \left(\frac{1}{C_{\text{tot}}} + \frac{C_{\text{T}}^2}{C_{\text{tot}}^3} \right)$$

and extending this after the Kth full charge-discharge cycle

$$v_{\text{T}} = C_{\text{L}} V_{\text{DD}} \sum_{i=0}^{K-1} \frac{C_{\text{T}}^{2i}}{C_{\text{tot}}^{2i+1}} \, . \qquad (8)$$

If we let K go to infinity, we can rewrite Eq. 8 and evaluate the geometric sum

$$v_{\text{T}} = \frac{C_{\text{L}} V_{\text{DD}}}{C_{\text{tot}}} \sum_{i=0}^{\infty} \left(\frac{C_{\text{T}}}{C_{\text{tot}}} \right)^{2i} = \frac{C_{\text{L}} V_{\text{DD}}}{C_{\text{tot}}} \left(\frac{1}{1 - \left(\frac{C_{\text{T}}}{C_{\text{tot}}} \right)^2} \right) \, .$$

By substituting in for C_{tot},

$$v_{\text{T}} = \frac{C_{\text{L}}^2 + C_{\text{L}} C_{\text{T}}}{C_{\text{L}}^2 + 2 C_{\text{L}} C_{\text{T}}} V_{\text{DD}}$$

and applying the assumption that $C_{\text{T}} \gg C_{\text{L}}$ results in

$$v_{\text{T}} \approx \frac{1}{2} V_{\text{DD}} \, ,$$

we can see that for the simple case in Figure 5-10 the voltage on the tank capacitor does converge to half of the supply voltage, V_{DD}, creating a 2-step voltage ramp.

Figure 5-12 contains a cartoon representation of the switching sequence when there

are two tank capacitors, which are again designed to be much larger than the load capacitor. Each capacitor is represented as a bucket of charge and assumes the tank capacitors have already settled to their nominal voltages and thus their charge does not fluctuate significantly. The equations given in the diagram outline the charge sharing effects between the load and tank capacitors and show convergence to the uniform voltages over continuous charge-discharge cycles. Figures 5-13 and 5-14 plot the tank voltages against the number of charge-discharge cycles completed for the $N = 3$ ramp circuit, where $C_{T1} = C_{T2} = 100\,\text{pF}$, $C_L = 10\,\text{pF}$, and $V_{DD} = 1\,\text{V}$ and the $N = 4$ ramp circuit, where $C_{T1} = C_{T2} = C_{T3} = 100\,\text{pF}$, $C_L = 10\,\text{pF}$, and $V_{DD} = 1\,\text{V}$ respectively.
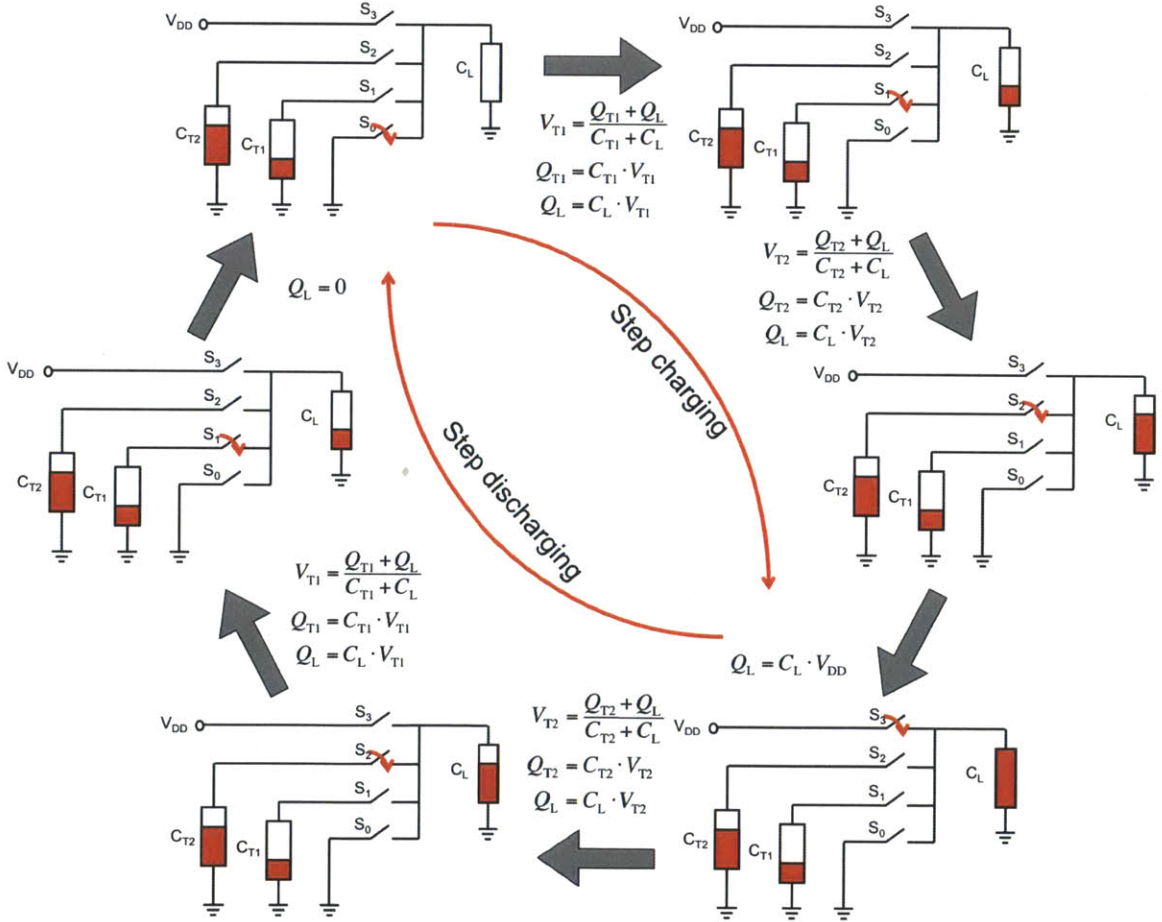

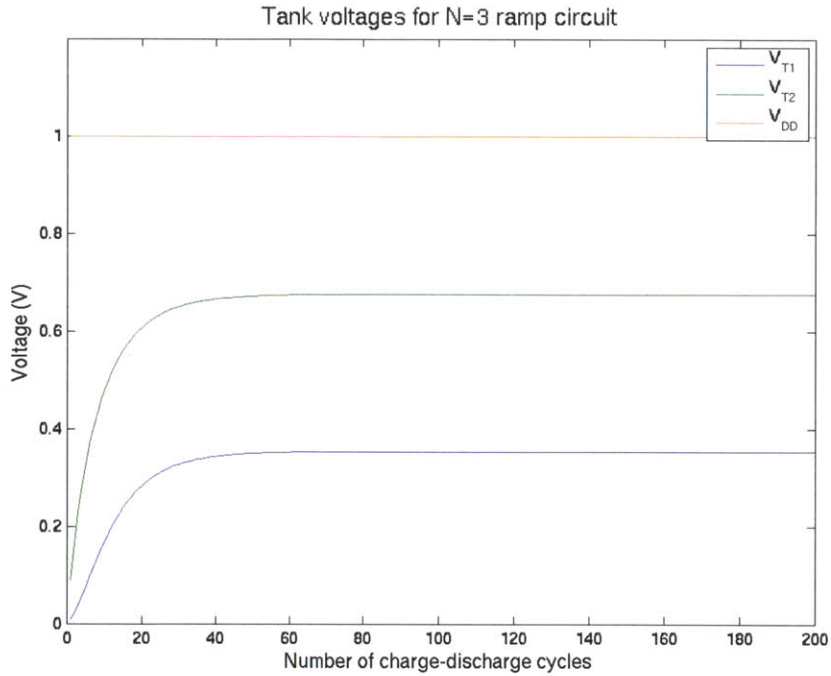Figure 5-12.   Cartoon drawing of initial charging of tank capacitors.

Figure 5-13. Uniform convergence of tank capacitor voltages for $N = 3$ ramp circuit with $C_{T1} = C_{T2} = 100$ pF, $C_L = 10$ pF, and $V_{DD} = 1$ V.
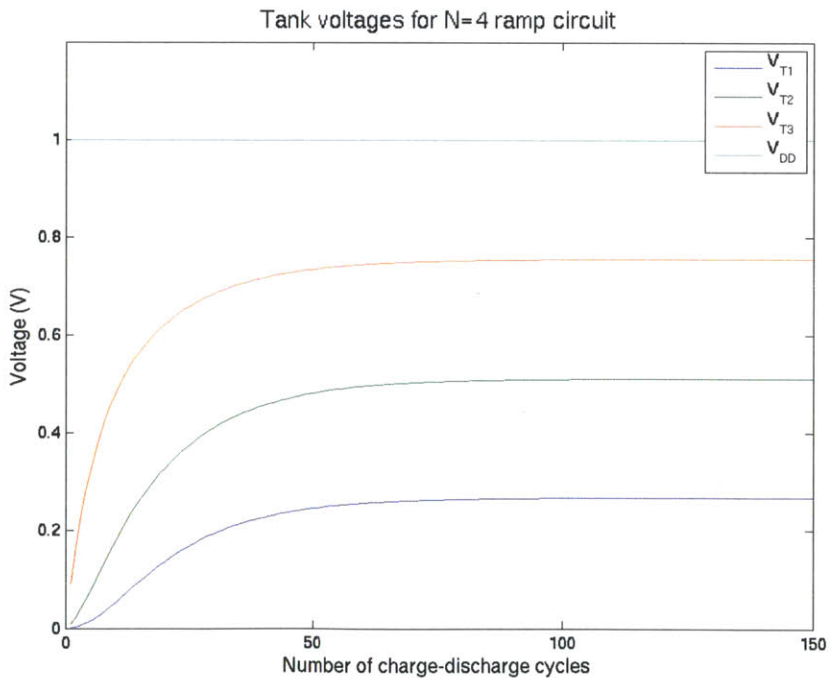

Figure 5-14. Uniform convergence of tank capacitor voltages for $N = 4$ ramp circuit with $C_{T1} = C_{T2} = C_{T3} = 100$ pF, $C_L = 10$ pF and $V_{DD} = 1$ V.

For the case of N tank capacitors, a proof was provided in [35] to show that the intermediate voltages converge and uniformly distribute between 0 and V_{DD} provided that C_T is much larger than C_L. The proof defines a deviance vector as the differences between the tank voltage and expected uniform voltage and shows that the ℓ_1 norm of deviance vector is decreasing. Furthermore, they show that the voltages on the tank capacitors settle to their values in M full charge-discharge cycles, where M is on the order of $N\left(\frac{c_T}{c_L}\right)$.

The aim was to implement adiabatic charging on the ADC using step charging to reduce the energy usage. Step charging was a viable solution to generate the necessary ramp voltage, but required maintaining intermediary voltages on chip combatting the energy savings. The self-stabilizing ramp circuit addressed this challenge, and an analysis of the settling and distribution of the uniform voltages was presented. The next section outlines the power cost-benefits for the NATIONAL 180 nm CMOST9T5V process that aided the design of the ADC.

5.3.2 Power analysis

It would follow that by continually increasing the number of steps, we introduce more energy savings. Ultimately, however, we reach a point of diminishing returns. Not only would the area increase with the addition of each tank capacitor, but also the additional switches would add to the power consumption thereby negating part of the intended power savings. We can compare the energy savings from step charging with the expended energy from switching in the ramp circuit and gain insight on how to optimally choose the number of steps to implement.

In [35] it was written that the total energy dissipation in a full charge-discharge cycle of the load capacitance is

$$E_{\text{tot}} = E_{\text{sw}} + E_{\text{diss}} \, ,$$

where E_{sw} is the energy used to drive the switches and E_{diss} is the energy dissipated from charging and discharging the load capacitance. Assuming that switch device i has a gate capacitance C_i, we can write

$$E_{\text{sw}} = \left(\sum_{i=1}^{N} C_i + \sum_{i=0}^{N-1} C_i \right) V_{\text{DD}}^2 \, , \tag{9}$$

where the first summation is the switching power require for a full step-charge up of C_{L} and the second summation is for a full step-discharge. Eq. 9 can be rewritten as

$$E_{\text{tot}} = \left(\sum_{i=1}^{N} C_i + \sum_{i=0}^{N-1} C_i \right) V_{\text{DD}}^2 + 2 \left(\frac{C_L V_{\text{DD}}^2}{2N} \right) .$$

We can plot the energy dissipation as a function of N and compare it with the case of traditional capacitor charging using approximate values for a given process to get a visual representation of the energy savings. Figure 5-15 plots the energy dissipation using parameters from a National CMOS9T5V 180nm process, where the switch device is a pass gate switch shown in Figure 5-16 ($C_i = 38.8 \text{ fF}$), $C_{\text{L}} = 10 \text{ pF}$, and $V_{\text{DD}} = 1 \, V$. While the energy dissipation decreases with increasing N, the rate of decrease drops off quickly for $N > 5$, and the overhead of design and area does not justify the minimal energy savings.

In Figure 5-15, the optimal point is at $N = 11$; however, from $N = 5$ to $N = 11$ the energy savings are small. For $N > 11$ the energy savings actually begin to decrease. In [35] further analysis was done to choose the optimal N and design of the switches.

However, as we can see in Figure 5-15, the curve is so shallow that this optimal N is larger than practical in terms of area and design complexity. For ease of design and proof of concept this work implements step charging with $N = 3$ and pass gate switches shown in Figure 5-16. From Figure 5-15 we can expect approximately a 30% improvement in energy, however increasing the number of steps by just one to $N = 4$ may be a worthwhile extension of this work.
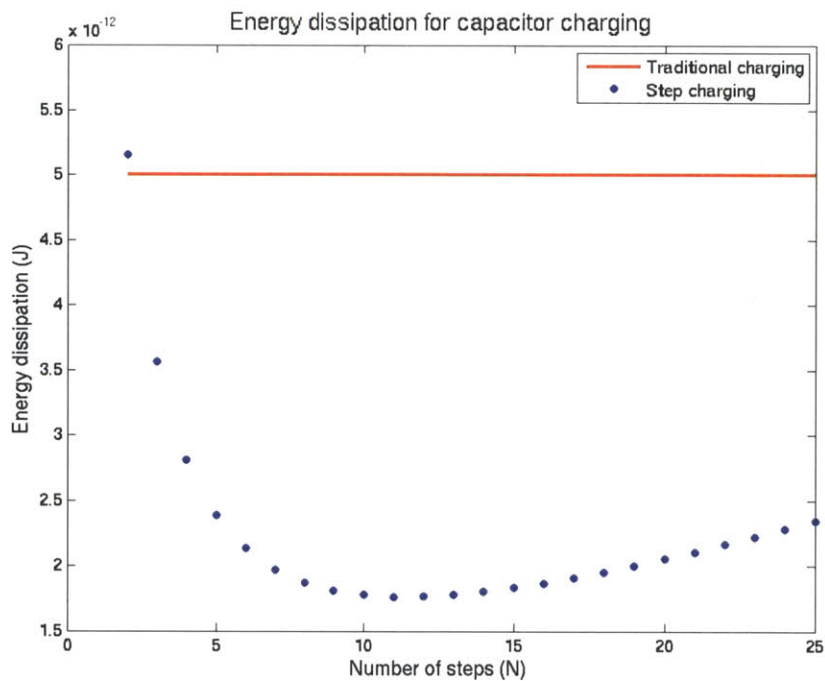

Figure 5-15. Energy dissipation for capacitor charging with $C_L = 10$ pF and $V_{DD} = 1$ V in a National CMOS9T5V 180nm process assuming pass gate switches ($C_i = 38.8$ fF).
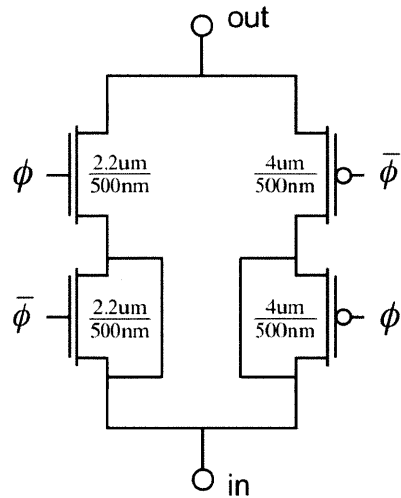
Figure 5-16.   Pass gate with extra transistors to offset charge injection
used in ramp circuit.

5.3.3 Control design

In our application we implemented a three step ($N = 3$) ramp circuit to step charge the
capacitor array during sampling and bit-cycling.   We generated a discretized ramp
voltage, V_{RAMP}, which is synchronized to the clock—seen in Figure 5-17—to enable both
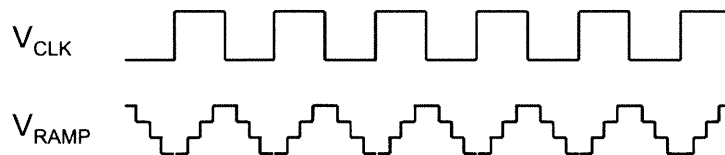step charging and step discharging during bit-cycling.

Figure 5-17.   Clock and ramp voltage waveforms.

Figure 5-18 shows the differential ADC topology with the switching
scheme for adiabatic charging. It is practical to only step-charge the three largest
capacitors—representing the three most significant bits (MSBs)—because the

advantages of step-charging diminish with the reduction of the load capacitor. In Figure 5-19, We plot the energy dissipation for varying load capacitance for $N = 3$ and $V_{\mathrm{DD}} = 1$ V using the pass gates described earlier and note that step-charging dissipates more energy than traditional charging for load capacitors less than 1.25 pF approximately. The load capacitance on the ramp circuit is greater than 1.25 pF until the fourth bit cycling period, so it is only practical to step charge on the three MSBs.
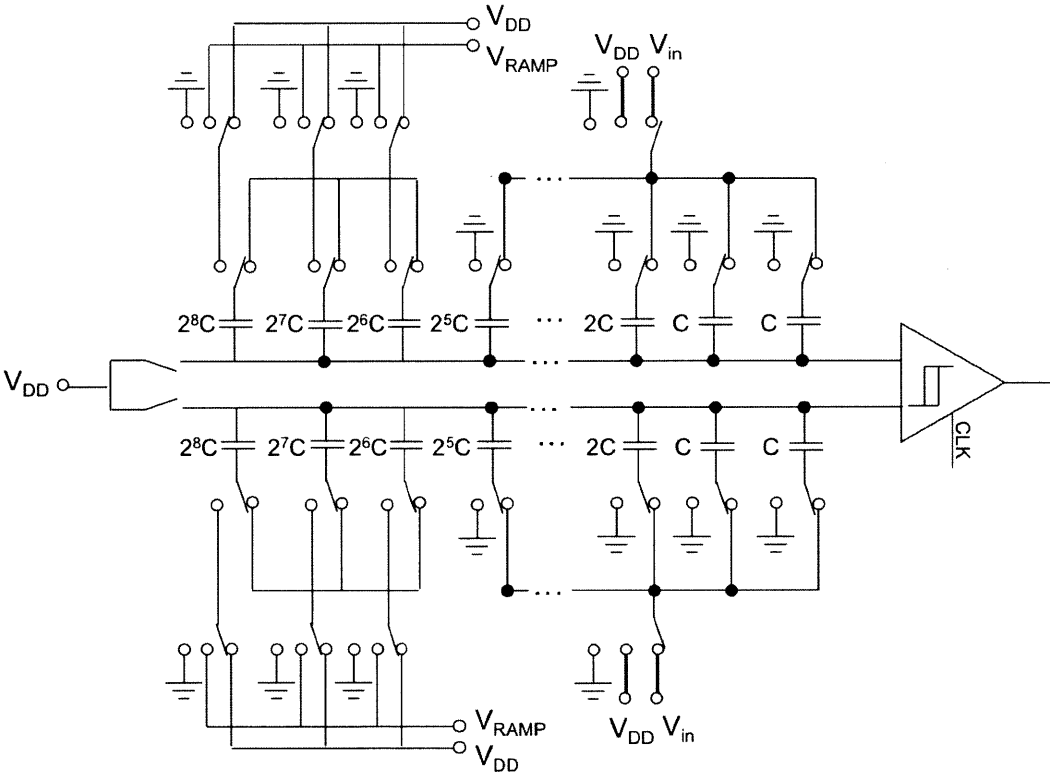
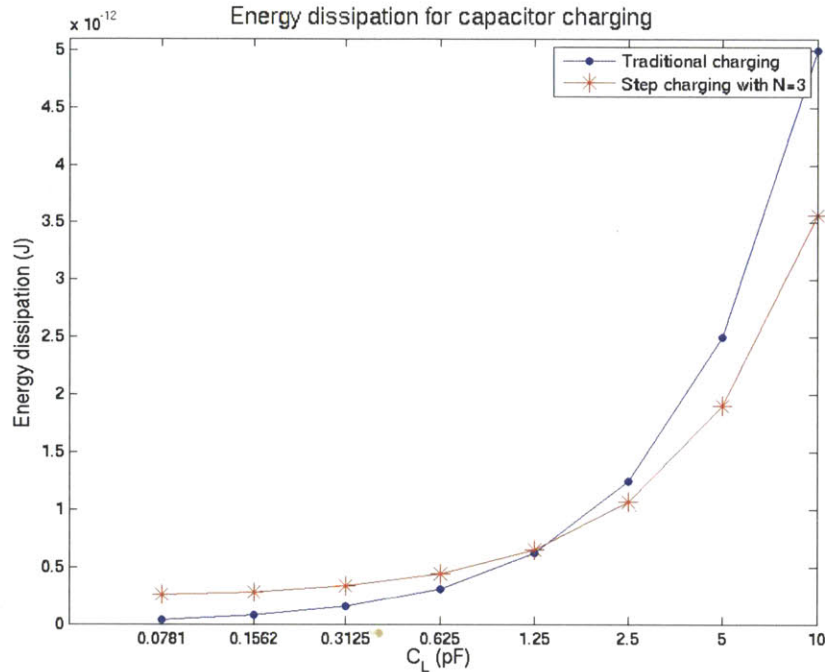Figure 5-18. ADC topology with adiabatic charging on the three MSBs.

Figure 5-19.   Energy dissipation for capacitor charging with $N = 3$ and $V_{DD} = 1$ V in a National CMOS9T5V 180nm process assuming pass gate switches ($C_i = 38.8$ fF).

During bit-cycling when charging the capacitor bottom plate from ground to V_{DD}, the bottom plate is connected to V_{RAMP}; after a bit-decision is made, the capacitor bottom plate is either held at V_{DD} or it remains connected to V_{RAMP} to charge back down to ground and then held, while the corresponding capacitor in the differential circuit is charged conversely after a bit-decision. To further reduce switching energy in the ramp circuit, a control signal was created to enable switching only while sampling and bit cycling of the three MSBs. Two non-overlapping clocks, φ and $\bar{\varphi}$, are already generated to signal the start of a new conversion and control sampling the input voltage. We can then generate a signal, ramp_enable, to enable switching in the ramp circuit only when step-charging is being implemented during bit-cycling. The clock and control signals φ

and ramp_enable are shown in Figure 5-20 along with V_{RAMP}.
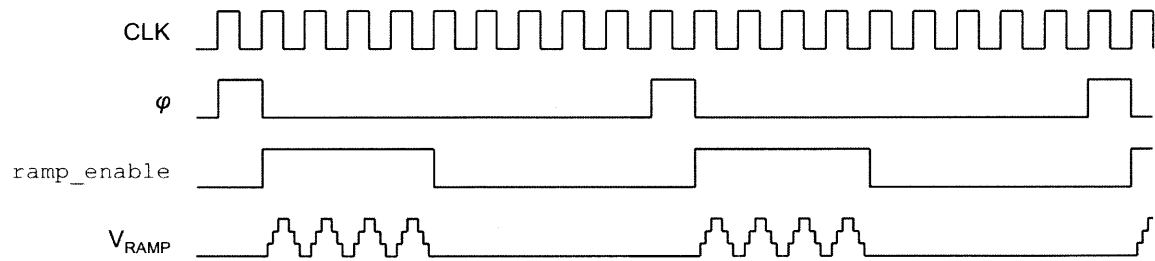

Figure 5-20. Control signals to reduce switching energy in the ramp circuit.

We note that there are two extra clock cycles in each conversion cycle thereby making $f_s = f_{clk}/10$. One is prior to sampling to ensure proper setup for conversion and one is after all 8 bit decision have been made to ensure that the data has sufficient time to write to memory. This is an insignificant decrease in sampling frequency since our application operates at such low frequencies, but nonetheless does affect the figure of merit (FOM) of the ADC.

It is also worthwhile to discuss the clocking of the ADC here. There are many control and clocking signals in a wide range of frequencies. There are three main clocks: 1) RAMP_CLK used in switching the tank capacitors in the self-stabilizing ramp circuit, 2) CLK used in the SAR ADC operation for conversion, and 3) CALIB_CLK used in the comparator during offset calibration. The RAMP_CLK is 8 times the frequency of the CLK signal, and CALIB_CLK is at a high frequency higher than 1 MHz in order operate the digital offset correction to calibrate the comparator to the accuracy needed. The ADC was designed for $f_s = 1\,\text{kHz}$, giving a CLK frequency of 10 kHz and RAMP_CLK frequency of 80 kHz.

77

5.4.   ADC measured performance

A test chip, with the ADC seen in Figure 5-21, is fabricated in a 180 nm National CMOS9T5V process. As mentioned previously the ADC was tested at $f_s = 12.5$ kHz to combat the error in the digital offset correction, however, only 7 bits are realized without missing codes.
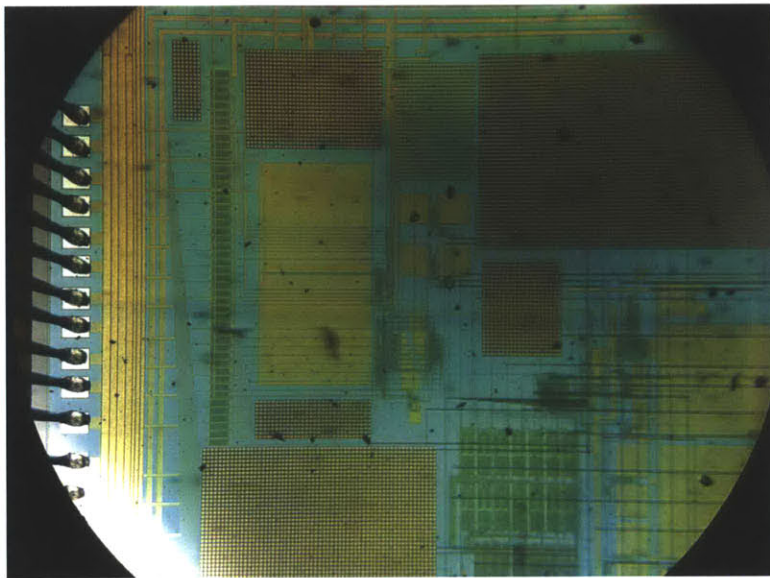

Figure 5-21.   Die photo of the SAR ADC in a National 180 nm
CMOS9T5V process.

Therefore the ADC performance presented here is with a 7-bit resolution. The output power spectrum is plotted in Figure 5-22 at 20 Hz, the maximum tremor frequency, and shows an SNDR = 41.18 dB. At an input of 5 kHz (Figure 5-23), the SNDR = 46.2 dB. The ADC has a maximum DNL of 0.9 LSB and INL of 2.4 LSB both of which are plotted in Figures 5-24 and 5-25. With a supply voltage of 1.25 V and $f_s = 12.5$ kHz the measured power was 204.7 µW.

The power reported includes the power hungry ring oscillator and flip-flop dividers to

generate the three clocks described in Section 5.4. The test chip did not have the capability to isolate the total ADC power on chip to get an accurate ADC measurement. The power consumption in the clock generation circuits and the ADC was found in simulation: $P_{ADC} = 1.55\ \mu W$ and $P_{CLK} = 193.75\ \mu W$. Using this ratio, we can adjust the measured power to give an estimated ADC power of $1.64\ \mu W$ and the estimated FOM reduces to 788 fJ/conv.

The ADC presented here was designed in the context of a microImplant and has a competitive FOM, but more importantly it implements adiabatic charging with a self-stabilizing ramp circuit with tank capacitors. With the unique ultracapacitor energy supply, we are concerned with the amount of charge removed more than the power used. The ramp circuit in conjunction with the step-charging allows us to more efficiently use the charge.
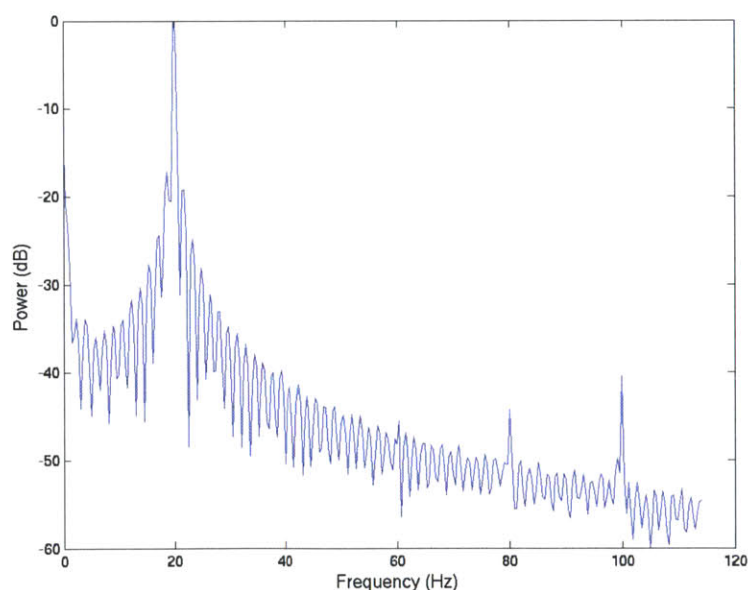

Figure 5-22. Output power spectral density with an input of 20 Hz
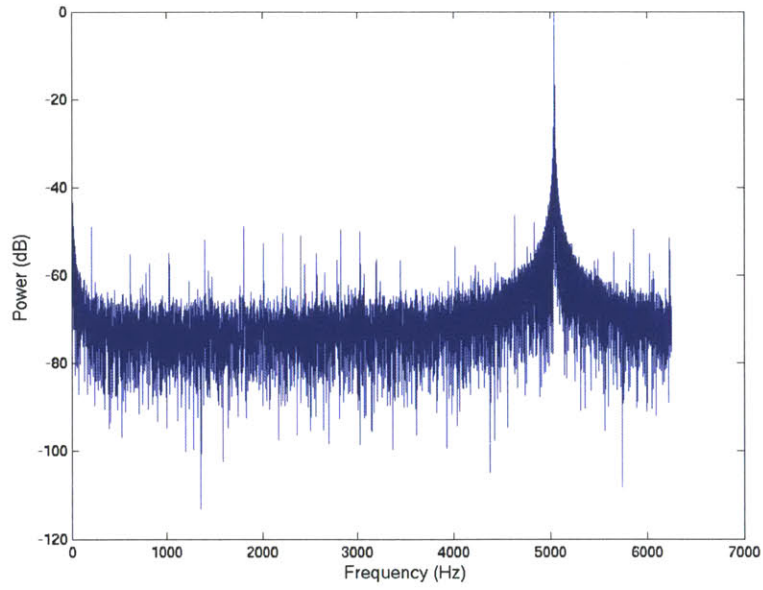resulting in SNDR of 41.18

Figure 5-23. Output power spectral density with an input of 5 kHz resulting in SNDR of 46.2 dB.
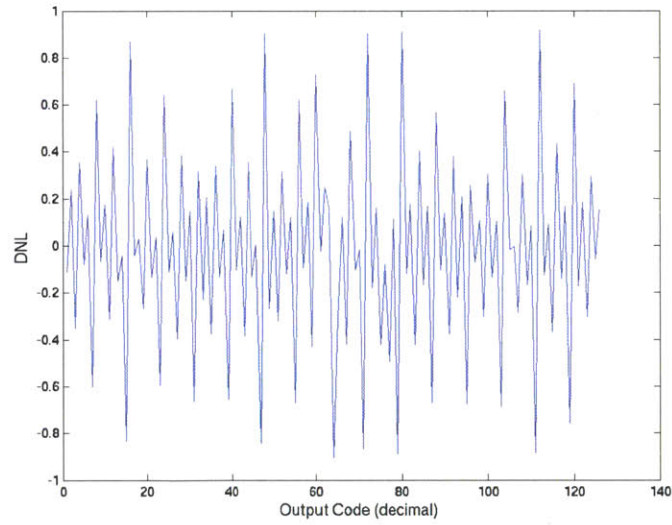

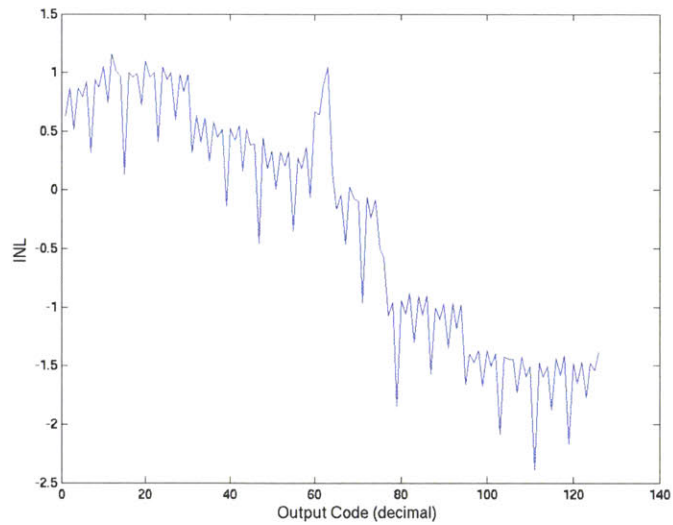Figure 5-24. Test chip measured DNL with maximum DNL of 0.9 LSB

Figure 5-25.   Test chip measured INL with maximum INL of 2.4 LSB

5.5.   Test chip design considerations

The ADC test chip had some design considerations that prohibited from having a better FOM. When the fabricated chip was delivered and subsequently packaged for testing, it was completely non-operational, and it was discovered that a layout error reversed the connections of the comparator outputs. A chip FIB was required to reverse these connections, but initial testing showed missing codes with a loss of two bits of resolution. It was discovered that the calibrate signal in the comparator was set incorrectly, such that instead of the comparator calibrating once every CLK cycle, it went six CLK cycles without calibrating during conversion. The leakage on the calibration capacitor, therefore, had the potential to convert incorrectly and reduce the bit accuracy. To combat this, the ADC was tested at a higher frequency, $f_s = 12.5$ kHz, and the supply voltage was raised from 1 V to 1.25 V; however missing codes persisted and only 1 bit of

81

resolution was gained back.

A dynamic comparator was suited for this SAR ADC to reduce static current, however the use of the digital offset correction technique was not. In implementing the offset correction, a charge pump was used that had a constant static power draw, when in the future it would be better to size the transistors in the comparator to mitigate the mismatch error instead of using minimal size devices as done here. Also, the leakage on the calibration capacitor was design in simulation such that the calibrated bias voltage, V_{cal}, would not fall more than 1 LSB, but this process should have over-compensated and ensured for a lesser error of 0.5 LSB. The leakage is another reason the digital offset correction is ill-suited for an ADC operating at such low frequencies.

Chapter 6

Conclusions and contributions

In this thesis we presented a discussion of the challenges associated with a minimally invasive microImplant for biometric monitoring and proposed a new technique for data acquisition for the long-term monitoring of Parkinson's tremors by devising a sparse sampling methodology to be implemented on-chip. The sparse sampling method relies on the implementation of compressive sampling to use fewer than the Nyquist required samples to reconstruct the tremor signal. Analysis on real tremor data was shown to reduce the percentage of random samples by 75% and still accurately detect the frequency at which the tremors occur.

The use of a random-like sequence generator allows sparse sampling to be done on chip and not rely on random in post-processing sampling. The use of three 8-bit LFSRs can generate an output sequence that is high only 26% of the time. In simulation we applied this predetermined, random-like sequence on the real tremor data and recovered the frequency spectrums. Implementing this on chip would allow us to not convert and write to memory roughly 26% of the required samples.

A SAR ADC with a FOM of 788 fJ/conv is presented that uses adiabatic charging to lower the energy loss by step charging the capacitor array. The sparse sampling technique would allow for an almost 26% reduction in sample conversion, and with the ADC presented here with an estimated FOM of 788 fJ/conv we can postulate that the charge per conversion effectively reduces to 205 fJ/conv since now only 26% the conversions are needed to acquire the same signal. This, of course, can be extended to any ADC and is not specific to the one presented here.

Further work to be done would be to demonstrate the entire system on-chip to show the sparse sampling in real time. Additionally the percentage of samples needed to recover the input spectrum could potentially be reduced more by using real tremor data with a longer duration to increase the window size and decrease the percentage while still recovering an accurate spectrum. There is also room for analysis and optimization in the random-like sequence to understand how the length and characteristic polynomial affect the generated sequence.
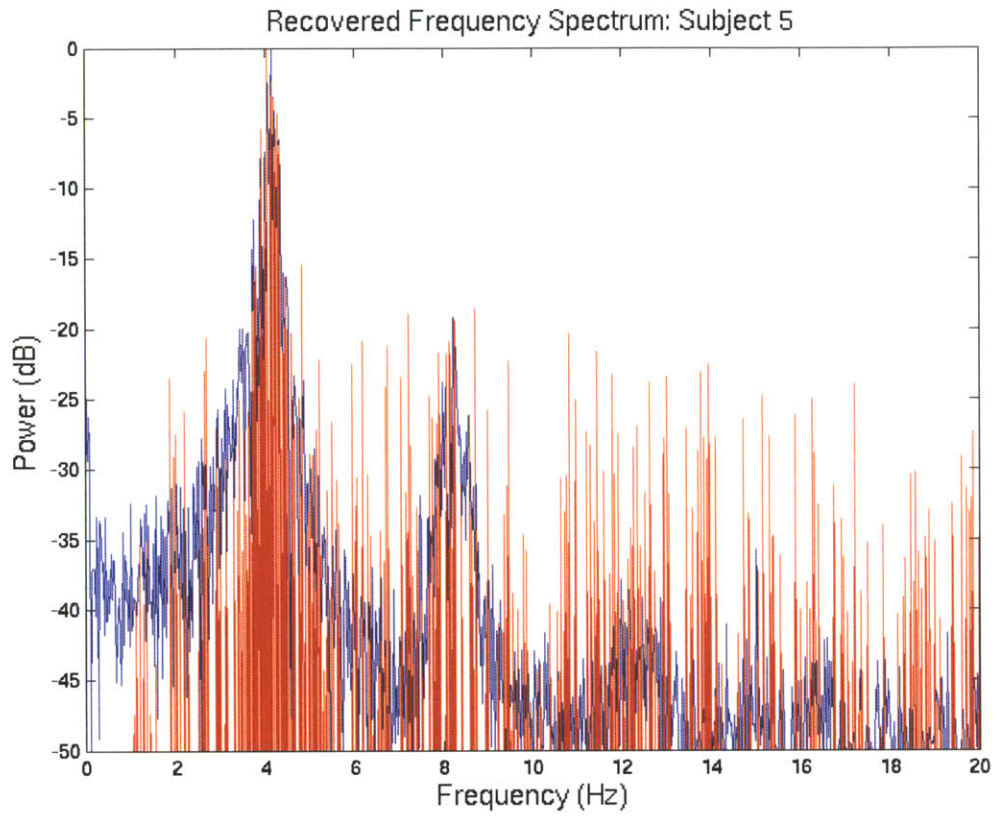
Appendix A

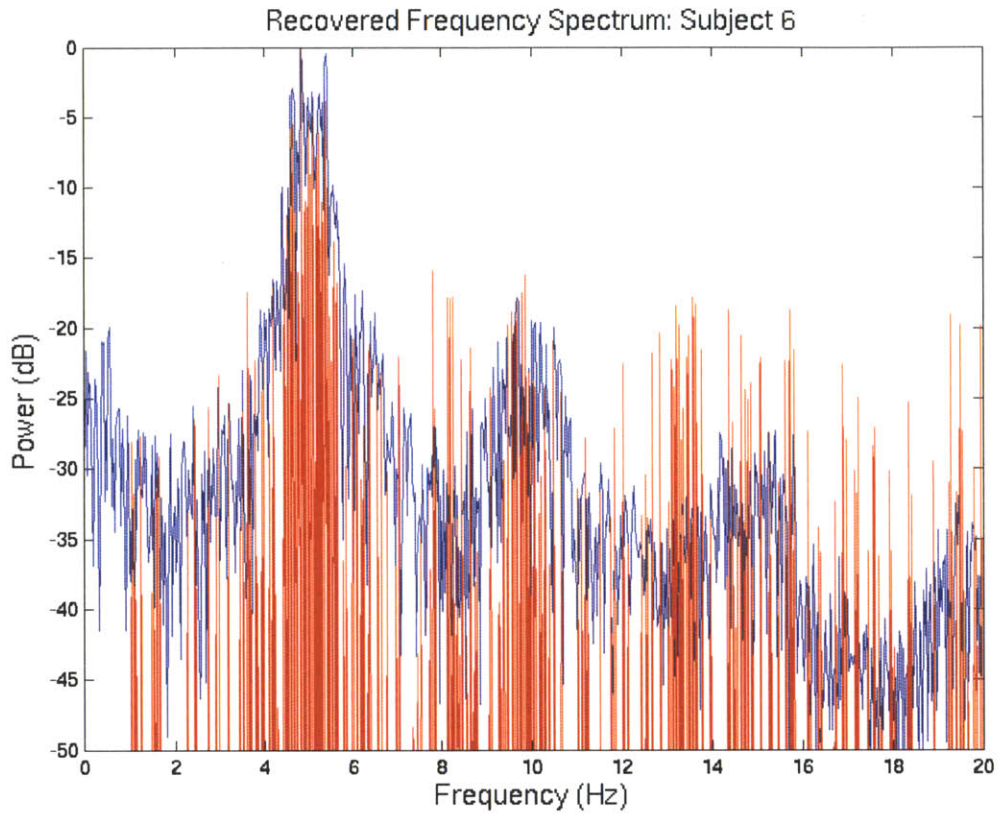# Real tremor data recovered spectrums using random samples
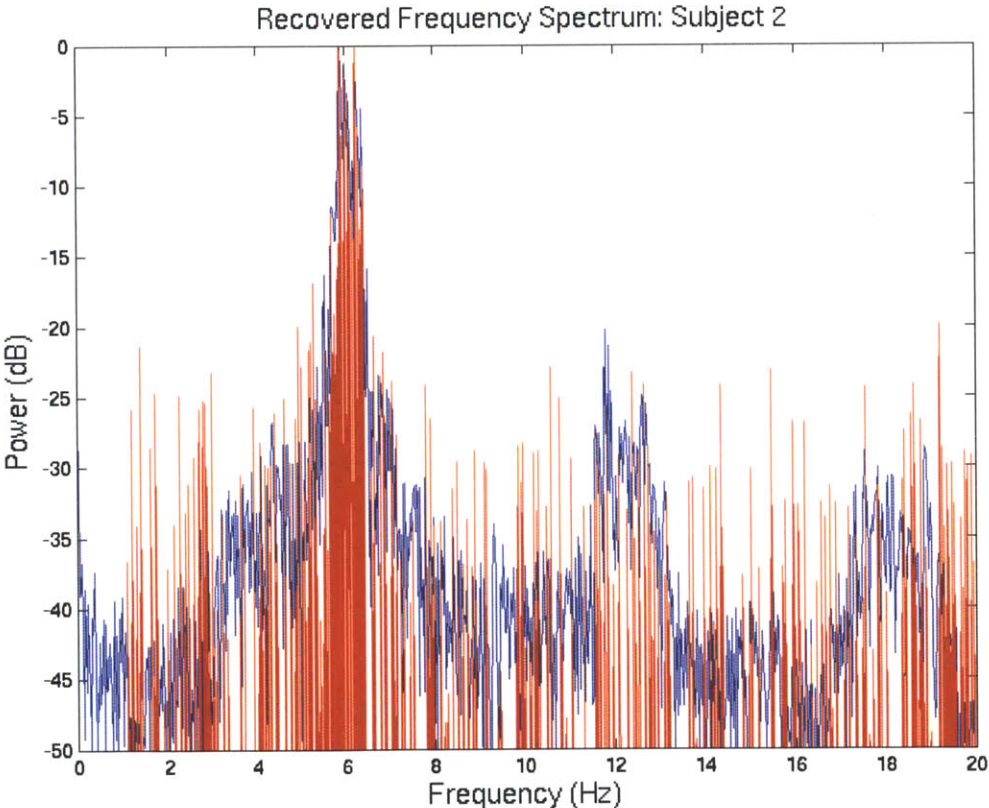
Recovered Frequency Spectrum: Subject 2

A-1.   Recovered frequency spectrums using 26.7% of the Nyquist
required samples for Subject 2.

Recovered Frequency Spectrum: Subject 4
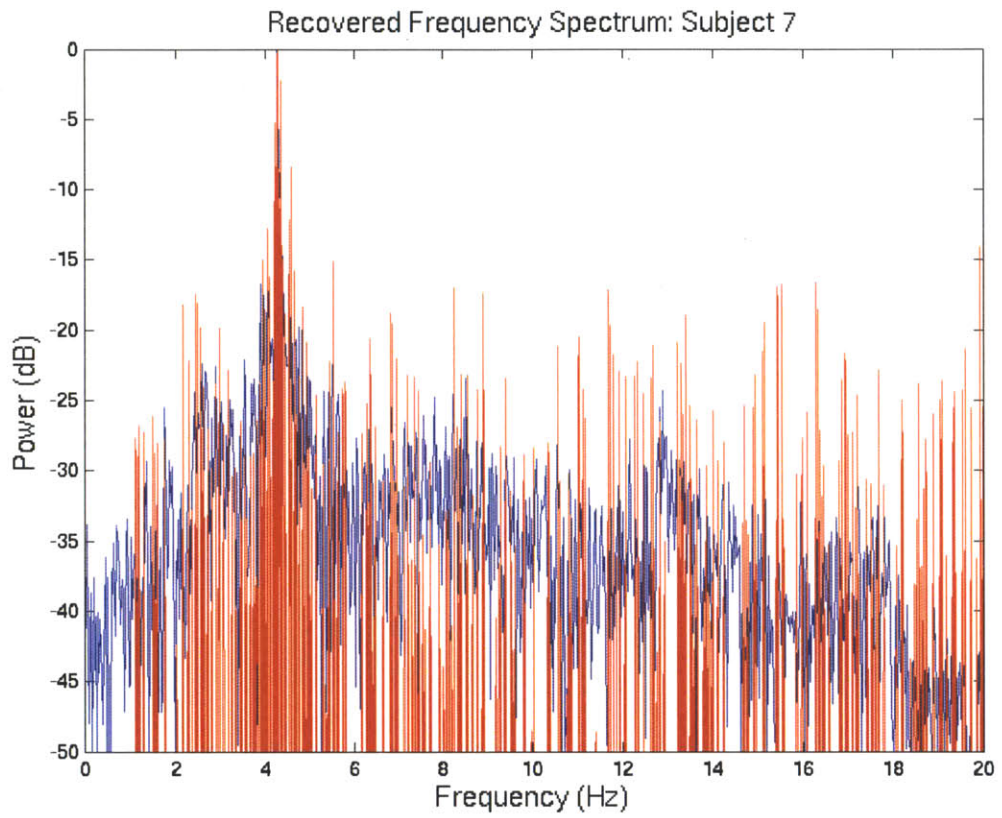
A-2.   Recovered frequency spectrums using 26.7% of the Nyquist required samples for Subject 4.

Recovered Frequency Spectrum: Subject 5
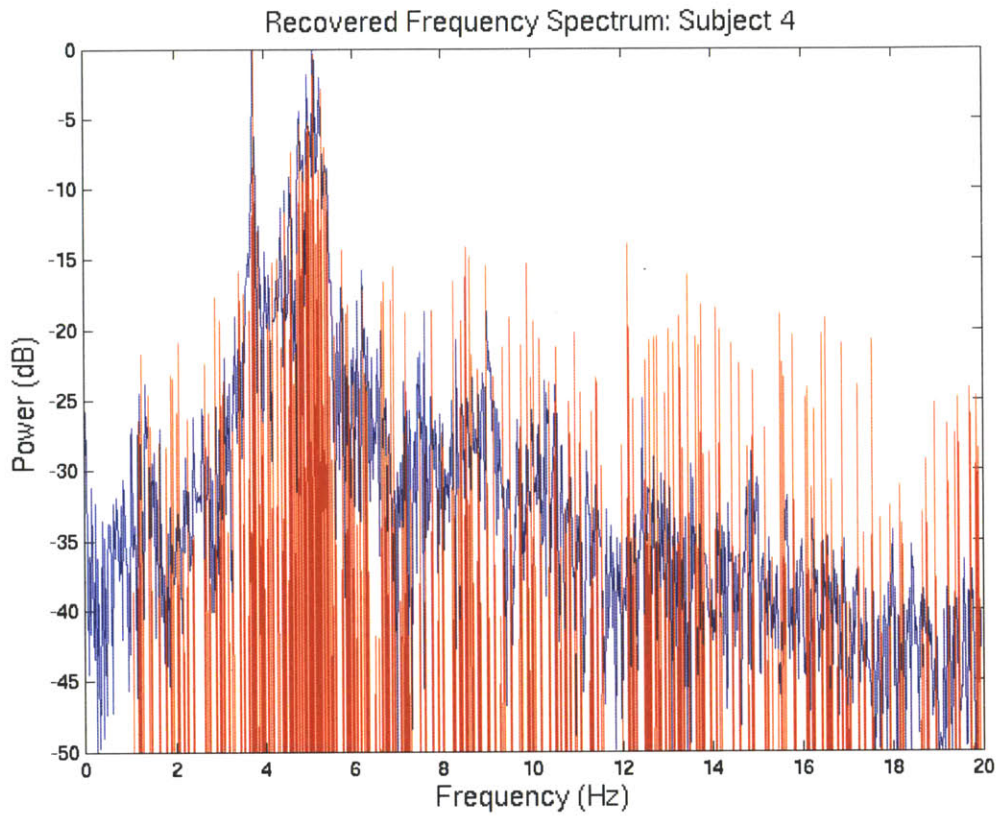
A-3.   Recovered frequency spectrums using 26.7% of the Nyquist
required samples for Subject 5.

Recovered Frequency Spectrum: Subject 6

A-4.   Recovered frequency spectrums using 26.7% of the Nyquist
required samples for Subject 6.

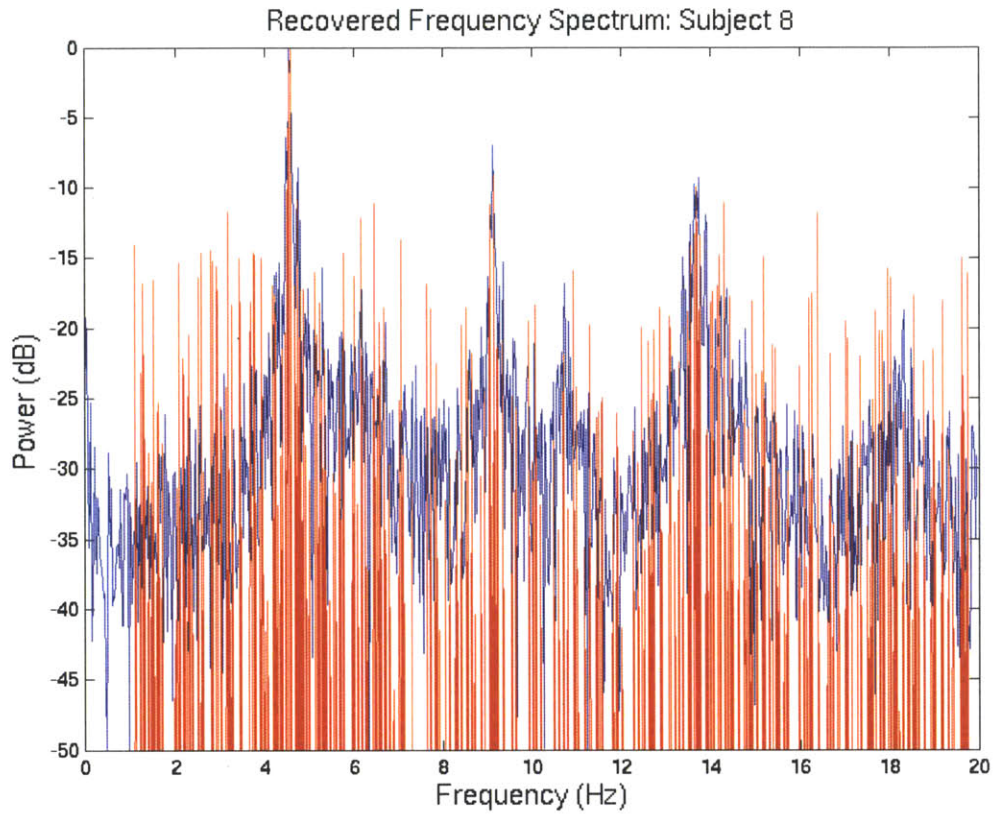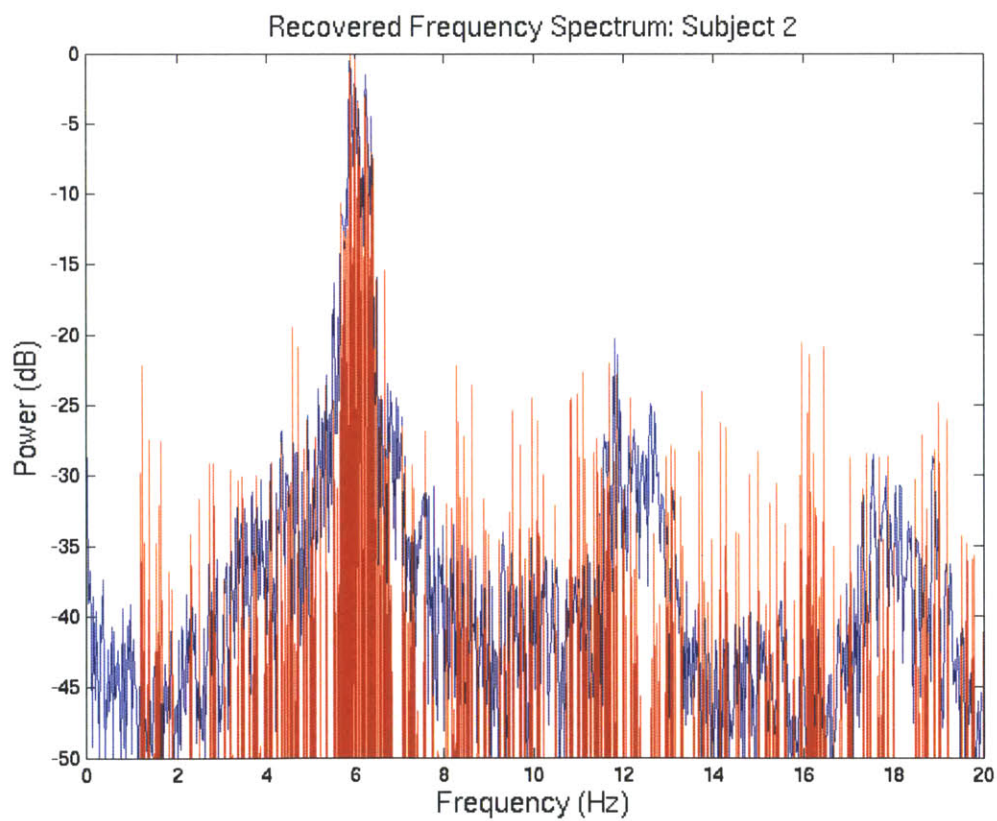Recovered Frequency Spectrum: Subject 7

A-5.   Recovered frequency spectrums using 26.7% of the Nyquist required samples for Subject 7.

A-6.   Recovered frequency spectrums using 26.7% of the Nyquist
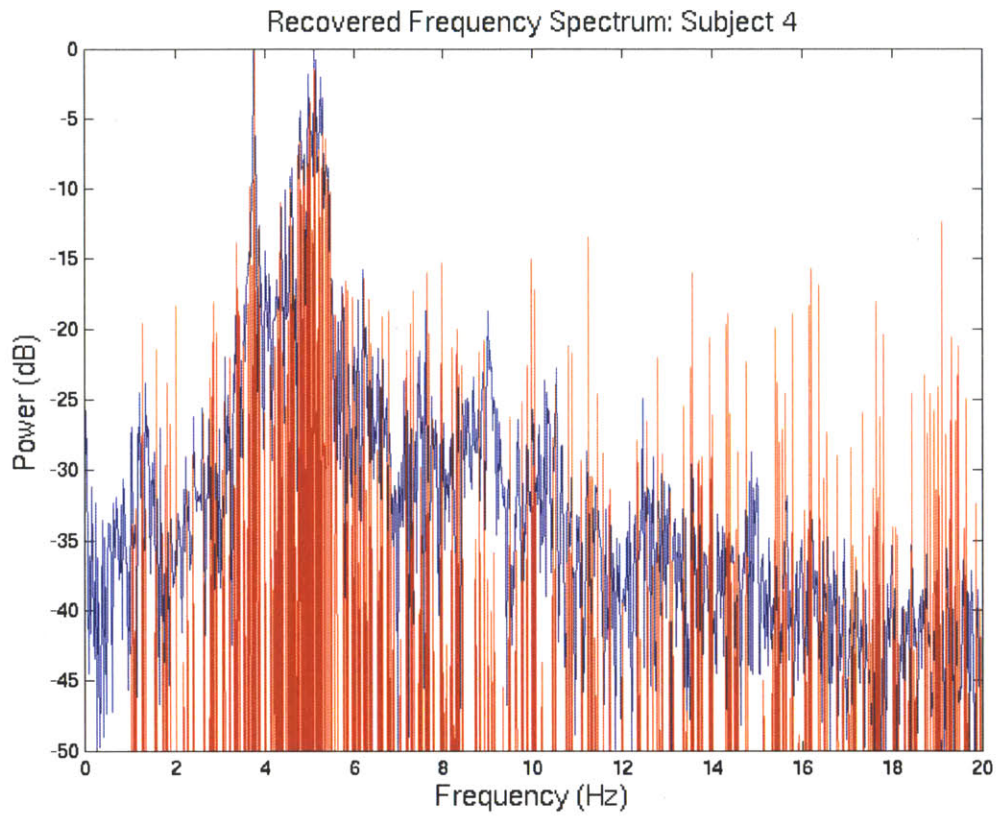required samples for Subject 8.
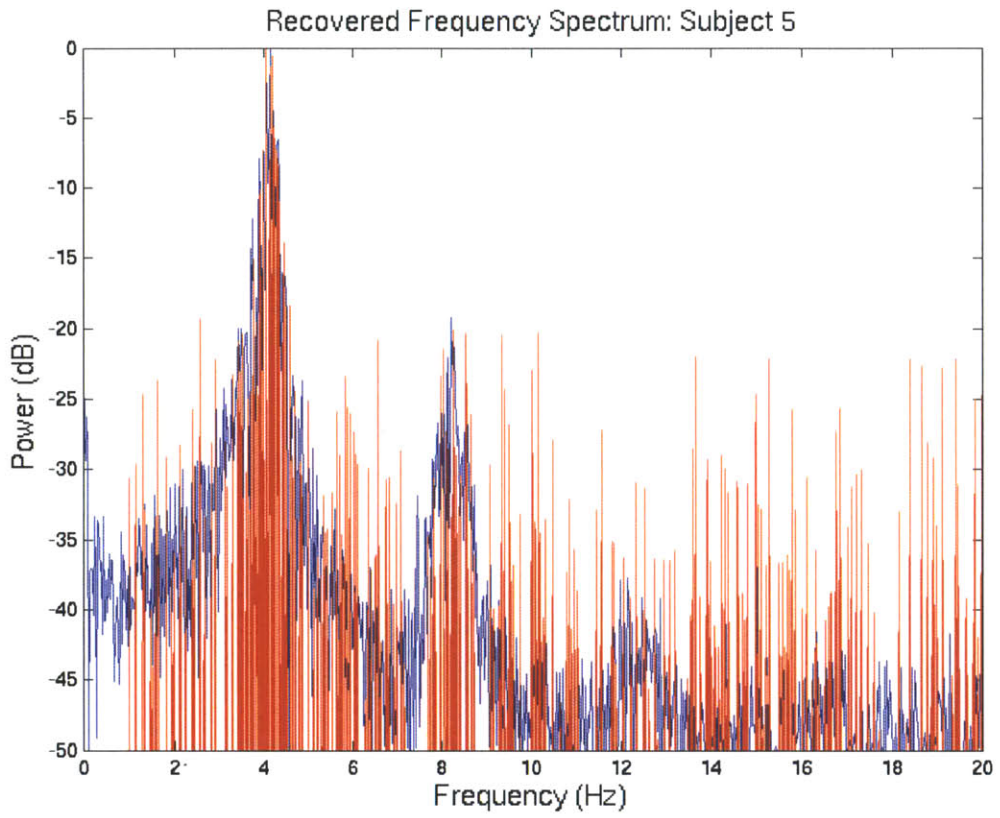
Appendix B

Real tremor data recovered spectrums using LFSR to generate samples

Recovered Frequency Spectrum: Subject 2

B-1.   Recovered frequency spectrums using 26.3% of the Nyquist
required samples for Subject 2 using an LFSR.

Recovered Frequency Spectrum: Subject 4

B-2. Recovered frequency spectrums using 26.3% of the Nyquist

required samples for Subject 4 using an LFSR.

Recovered Frequency Spectrum: Subject 5

B-3. Recovered frequency spectrums using 26.3% of the Nyquist required samples for Subject 5 using an LFSR.
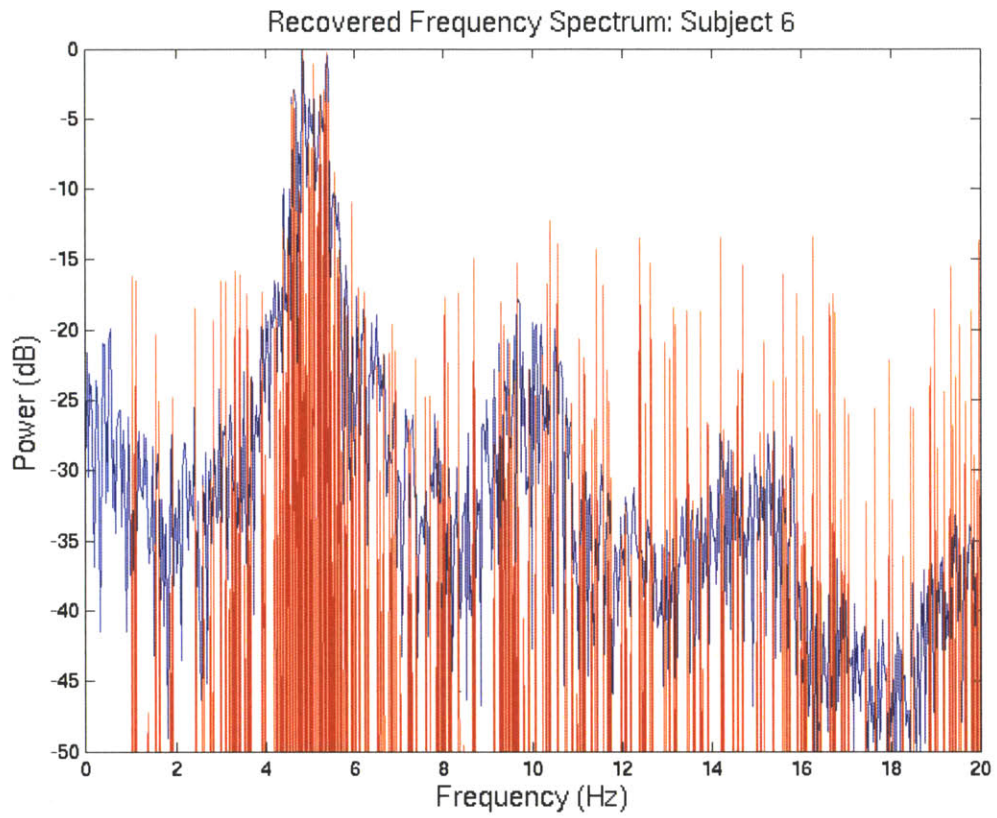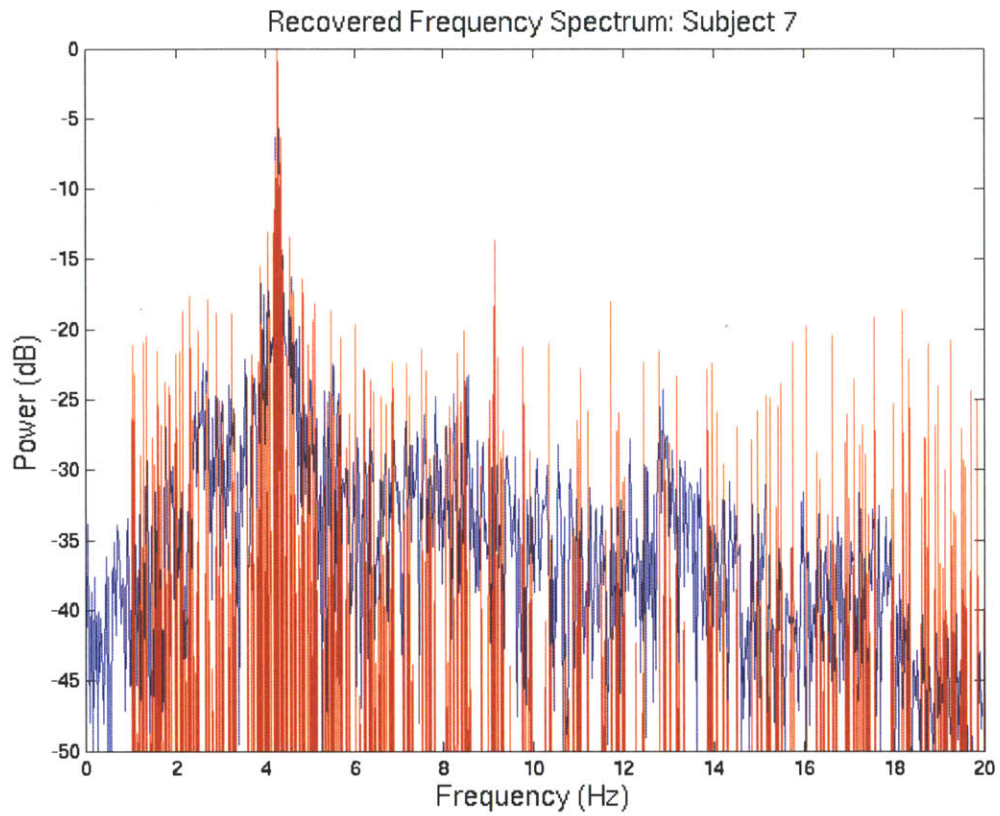
Recovered Frequency Spectrum: Subject 6
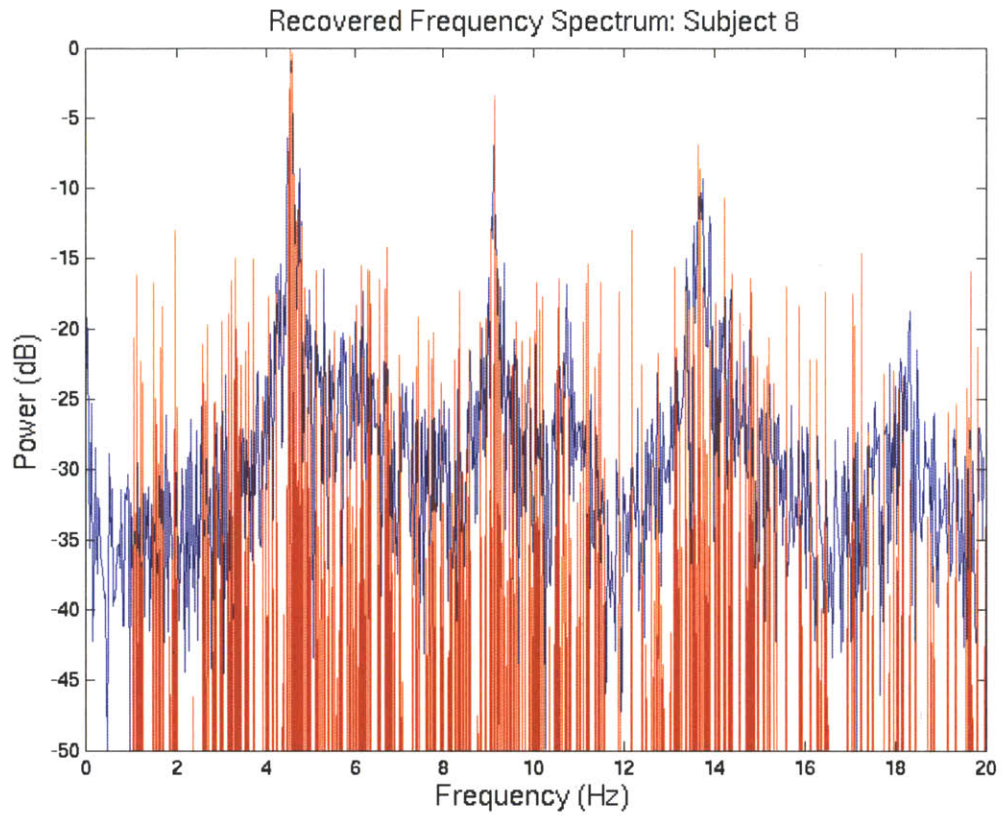
B-4. Recovered frequency spectrums using 26.3% of the Nyquist required samples for Subject 6 using an LFSR.

B-5.   Recovered frequency spectrums using 26.3% of the Nyquist
        required samples for Subject 7 using an LFSR.

Recovered Frequency Spectrum: Subject 8

B-6. Recovered frequency spectrums using 26.3% of the Nyquist required samples for Subject 8 using an LFSR.

Bibliography

[1]   Philips, Inc, "Actiwatch-Philips," Philips, Inc., 2004. Web. 20 Aug 2012.

[2]   Heart Rate Monitors USA, "LifeSource Blood Pressure Monitors," Heart Rate Monitors USA. 2012. 20 Aug 2012.

[3]   LifeScan, Inc, "Life First: One Touch," LifeScan, Inc. 2008. 20 Aug 2012.

[4]   PositiveID ™, "PositiveID Corporation," PositiveID ™. 2012. 20 Aug 2012.

[5]   P. Feng, Y. Li, N. Wu, "An ultra low power non-volatile memory in standard CMOS process for RFID tags," Custom Integrated Circuits Conference, Sept. 2009.

[6]   J. Schindall, "The charge of the ultra - capacitors," IEEE Spectrum, Nov. 2007.

[7]   W. Sanchez, "Energy processing circuits for low-power applications Energy Management Techniques for Ultra-Small Bio-Medical Implants," in Ph.D. Thesis. MIT, 2012.

[8]   A. Poon, S. O'Driscoll, and T. Meng, "Optimal frequency for wireless power transmis- sion into dispersive tissue," Antennas and Propagation, IEEE Transactions on, vol. 58, no. 5, pp. 1739 –1750, May 2010.

[9]   J. B. Gao. "Analysis of amplitude and frequency variations of essential and

Parkinsonian tremors," Medical and Biological Engineering and Computing, vol. 42, pp. 345-349, March 2004.

[10] P. E. O'Suilleabhain and J. Y. Matsumoto, "Time-frequency analysis of tremors," Brain, pp. 2127-2134, June 1998.

[11] Eus J. W. van Someren,* Ben F. M. Vonk, Wim A. Thijssen, Johannes D. Speelman, Peter R. Schuurman, Majid Mirmiran, and Dick F. Swaab, "A new actigraph for long-term registration of the duration and intensity of tremor and movement," IEEE Transaction on Biomedical Engineering, vol. 45, no. 3, March 1998.

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

[13] F. Chen, "Energy-efficient Wireless Sensors: Fewer Bits, Moore MEMS," in Ph.D. Thesis. MIT, 2011.

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

[15] E. Candes and J. Romberg. "Quantitative robust uncertainty principles and optimally sparse decompositions," Foundations of Comput. Math., vol. 6, no. 2, pp. 227-254, 2005.

[16] E. Candes, J. Romberg, and T. Tao. "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," IEEE Trans. Inform.Theory, vol. 52, no. 2, Feb 2006.

[17] E. Candes, J. Romberg, and T. Tao. "Stable signal recovery from incomplete and inaccurate measurements," Pure and Applied Mathematics, vol. 59, no. 8, pp. 1207-

1223, March 2006.

[18] E. Candes and T. Tao. "Near-optimal signal recovery from random projections and universalencoding strategies," IEEE Trans. Inform. Theory, vol. 52, no. 12, Dec. 2006.

[19] E. Candes and T. Tao. "The Dantzig selector: statistical estimation when p is much smaller than n," Manuscript, May 2005.

[20] E. J. Candes and T. Tao. "Decoding by linear programming," IEEE Trans. Inform. Theory, vol. 51, no. 12, December 2005.

[21] E. Candes and J. Romberg. "L1-Magic," 2005. [Online]. Available: http://users.ece.gatech.edu/~justin/l1magic/

[22] J. Tropp. "Signal recovery from random measurements via orthogonal matching pursuit," IEEE Trans. Inform. Theory, vol. 53, no. 12, December 2007.

[23] A. Salarian, H. Russman, C. Wider, P. Burkhard, F. J. G. Vigerhoets, and Kamiar Aminian. "Quantification of tremor and bradykinesia in Parkinson's disease using a novel ambulatory monitoring system," IEEE Transaction on Biomedical Engineering, vol. 54, no. 2, February 2007.

[24] J. Jankovic. "Parkinson' disease: clinical features and diagnosis," Neurology Neurosurgery Psychiatry, vol. 79, pp. 368-376. 2008.

[25] D. B. Calne, FRCPC, B. J. Snow, MD, and C. Lee, MD, "Criteria for Diagnosing Parlunson's Disease," Annals of Neurology, vol. 32, pp. S125-S127, 1992.

[26] A. Anouti, W. C. Koller, "Tremor disorders-diagnosis and management," Western Journal of Medicine, vol. 162, pp. 510-513. 1995.

[27] A. Beuter, M.S. Titcombe, R. Richer, C. Gross, and D. Guehl, "Effect of deep brain stimulation on amplitude and frequency characteristics of rest tremor in Parkinson's disease," Thalamus and Related Systems, pp. 203-211, 2001.

[28] D. Johns and K. Martin, "Analog integrated circuit design," Toronto. Wiley. 1997. Print.

[29] M. Miyahara, Y. Asada, D. Paik and A. Matsuzawa, "A low-noise self-calibrating dynamic comparator for high-speed adcs," in IEEE Asian Solid-State Circuits Conference, Fukuoka, Japan, November 2008, pp 269-272.

[30] "adiabatic, a." OED Online. June 2004. Oxford University Press. 30 April 2012. <http://dictionary.oed.com/>.

[31] W. C. Athas and L. J. Svensson, "Reversible logic issues in adiabatic CMOS," Physics and Computation. pp. 111-118. Nov 1994.

[32] W. C. Athas, J. G. Koller, and L. J. Svensson, "An energy efficient CMOS line driver using adiabatic switching," IEEE Great Lakes Symposium on VLSI. 1994.

[33] M. va Elzakker, E van Tuijjl, P. Geraedts, D. Schinkel, E. Klumperink, and B. Nauta, "A 1.9uw 4.4fj/conversion-step 10b 1ms/s charge-redistribution adc," in IEEE Inter. Solid-State Circuits Conf., 2008, pp. 244-245.

[34] Y. Moon and D.-K. Jeong, "An efficient charge recovery logic circuit," IEEE Journal of Solid-State Circuits, vol. 31, no. 4, pp. 514–522, April 1996.

[35] L. J. Svensson and J. G. Koller, "Adiabatic charging without inductors," University of Southern California: Information Sciences Institute, Tech. Rep., February 1994.