

ASYNCHRONOUS SIGNAL PROCESSING FOR COMPRESSIVE DATA TRANSMISSION

by

Shuang Su

B. S. in Engineering, Zhejiang University, China, 2005

Master of Philosophy, The Chinese University of Hong Kong, 2007

Submitted to the Graduate Faculty of
the Swanson School of Engineering in partial fulfillment
of the requirements for the degree of

Master of Science

University of Pittsburgh

2011

UNIVERSITY OF PITTSBURGH
SWANSON SCHOOL OF ENGINEERING

This thesis was presented

by

Shuang Su

It was defended on

July 13th 2011

and approved by

Luis F. Chaparro, PhD, Associate Professor, Electrical and Computer Engineering
Department

Mingui Sun, PhD, Professor, Neurological Surgery Department

Zhihong Mao, PhD, Assistant Professor, Electrical and Computer Engineering Department

Thesis Advisor: Luis F. Chaparro, PhD, Associate Professor, Electrical and Computer
Engineering Department

ASYNCHRONOUS SIGNAL PROCESSING FOR COMPRESSIVE DATA TRANSMISSION

Shuang Su, M.S.

University of Pittsburgh, 2011

In this thesis we propose a power-efficient procedure useful in the acquisition of biological data in digital form without using high frequency samplers. The data is compressed so that transmission is limited to parts of the signal that are significant. Our procedure uses an asynchronous sigma delta modulator (ASDM) together with a time-to-digital converter (TDC) to obtain binary data that is transmitted via orthogonal frequency division multiplexing (OFDM). The asynchronous sigma delta modulator is a nonlinear feedback system that allows the representation of bounded signals by zero-crossing times of a binary signal. Using duty-cycle modulation, the ASDM is shown to be equivalent to an optimal level-crossing sampler. The zero-crossing times are measured with a time-to-digital converter that applies pulse-shrinking delay lines and requires no high-frequency clock. Reconstruction of the original signal is possible from the zero-crossing times of the ASDM output binary signal. ASDM time-domain compression is compared with discrete wavelet transform based data compression. For wireless data transmission, the orthogonal frequency division multiplexing (OFDM) reduces the bit error-rate in multipath fading channels. The performance of the proposed algorithm is illustrated using an electrocardiogram signal which fits the bursty characteristic appropriate for our procedure.

TABLE OF CONTENTS

1.0 INTRODUCTION	1
1.1 Background and Objectives	1
1.2 Synchronous and Asynchronous Data Acquisition	3
1.2.1 Conventional Analog to Digital Converters	3
1.2.2 Sigma-Delta Analog-to-Digital Converter	5
1.2.3 Level-Crossing Analog-to-Digital Converter	9
1.2.4 Asynchronous Sigma-Delta A/D Converter	10
1.3 Methodology	12
2.0 ASYNCHRONOUS DATA ACQUISITION	16
2.1 Time-Encoding Machine	16
2.2 Time-Decoding Machine	20
2.3 Time to Digital Converter	22
2.3.1 Delay Elements and RS Flip-Flops	22
2.3.2 Pulse-shrinking Delay Lines	25
2.3.3 Pulse-shrinking Delay Line Based TDC	25
2.4 Pros and Cons of Asynchronous Sigma Delta A/D Converter	27
3.0 DATA TRANSMISSION AND COMPRESSION	29
3.1 Orthogonal Frequency Division Multiplexing	29
3.2 Data Compression	32
3.2.1 Direct Data Compression	32
3.2.2 Data Compression Based on Discrete Wavelet Transform	34

4.0 SIMULATIONS	38
4.1 Time-encoding and Time-decoding	38
4.2 Time to Digital converter	38
4.3 Data Compression	40
4.4 Transmission Error Analysis	44
5.0 CONCLUSION AND FUTURE WORK	52
BIBLIOGRAPHY	53

LIST OF TABLES

1	Summary of CR and PRD results using different thresholds (direct compression)	48
2	Summary of CR and PRD results using different thresholds (DWT based compression)	48
3	OFDM specifications	50

LIST OF FIGURES

1	An ECG signal	2
2	Generalized Analog-to-Digital Conversion Process	3
3	Generalized Analog-to-Digital Conversion Process	6
4	First Order Sigma-Delta Modulator	6
5	The waveforms of an input signal $x(t)$, the integrator output $y(t)$, and comparator output $z(t)$ of a first order sigma-delta modulator	7
6	First order sigma-delta modulator in z-domain	8
7	Spectrum of a First-Order Sigma-Delta Noise Shaper	9
8	Comparison of (a) the uniform and (b) the level-crossing sampling schemes for the same sampling resolution, i.e., $\epsilon_{max} = \Delta$	11
9	Asynchronous sigma delta A/D converter	12
10	Model of proposed data acquisition, direct compression and transmission system	14
11	Model of proposed data acquisition, DWTcompression and transmission system	15
12	Block Diagram of Time-encoding Maching	17
13	The waveforms on the integrator output $y(t)$, and Schmitt trigger output $z(t)$ for the DC level input signal: $x(t) = d \leq c < b$, $d = \text{constant}$, $t \in R$	17
14	TEM Output: Duty-cycled Signal	18
15	Inverter	23
16	Delay element	23
17	RS Flip-Flop	24
18	pulse-shrinking delay line	25
19	Pulse separation	26

20	Coarse measurement by clocks	26
21	Examples of OFDM spectrum	31
22	Block diagram of a simple OFDM system	32
23	The coding stream of the significant and insignificant zero-crossing times	33
24	A three-level iterative filter bank: Forward DWT	36
25	A three-level iterative filter bank: Inverse DWT	36
26	The coding stream of the significant and insignificant coefficients	37
27	time-encoding processes for different signals	39
28	Time-decoding of a sinusoid using PSWFs	40
29	Time-decoding of an ECG signal using PSWFs	41
30	Comparison of different sampling methods. Quantization error of delay-lines based sampling is reset every 1 second.	42
31	ECG Signal	42
32	Compressed Time-encoded Signal	43
33	Extracted Time-encoded Signal	43
34	Compression Error	44
35	Signal reconstruction using PSWFs (ECG input is noisy)	45
36	Signal reconstruction using PSWFs (ECG input is clean)	46
37	Compression of coefficients	47
38	Original ECG signal vs. extracted signal using DWT	47
39	DWT based compression vs. direct compression	49
40	Bit error rate analysis	51

1.0 INTRODUCTION

Ubiquitous healthcare, which has been recognized as an emerging trend in the healthcare arena, allows personal health status to be monitored and managed automatically during daily life without institutional intervention [1]. In this regard, continuous measurement of cardiac electrical activity is one of the core technologies. However, with very limited battery power within a portable heart monitor, significant technical difficulties exist in wireless transmission of electrocardiographic (ECG) data acquired from a set of electrodes to an external mobile device or a computer. Therefore, the development of new algorithms for data acquisition and analog-digital conversion to support energy-efficient wireless transmission of ECG is an important task in ubiquitous healthcare. Two issues of special interest in this development are power consumption reduction and system simplification which are both essential in the constructions of smaller, lighter and more reliable wireless ECG monitors.

1.1 BACKGROUND AND OBJECTIVES

Biosignal acquisition and processing involve collection and analysis of data in an effort to understand and identify features of the signals. The manipulation and dissection of such data provide vital information on the condition of the person being monitored [2]. In this thesis, we use the ECG as an example to illustrate how data acquisition, processing and transmission can be performed with new signal processing strategies that are significantly different from the traditional ones.

The electrocardiograph (ECG, or EKG [from German word Elektrokardiogramm]) is a transthoracic electrical activity of the heart externally recorded by a set of skin surface

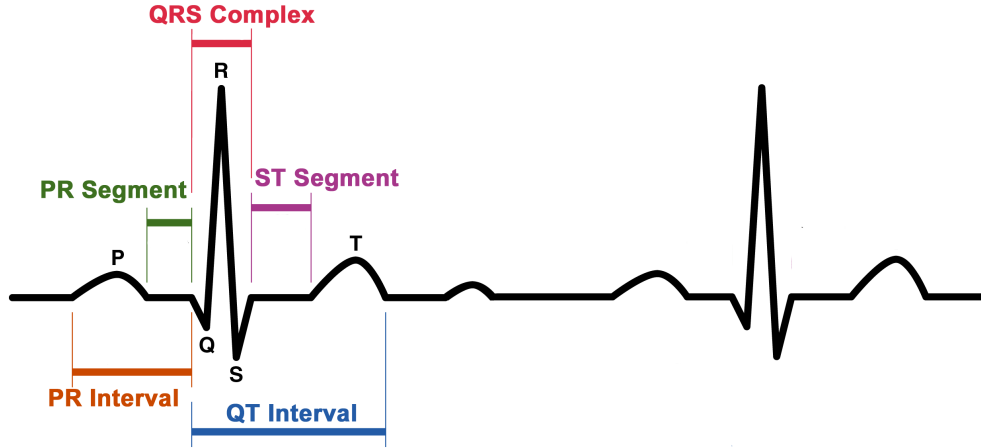


Figure 1: An ECG signal

electrodes around the chest. Like many other electrophysiological signals, the ECG is a bursty signal which exhibits high frequency contents. This signal feature makes ECG hard to sample and reconstruct (see Figure 1). Although acquisition and transmission of ECG can be performed using conventional analog and digital circuitries, the high frequency contents of ECG require a high sampling rate even though the samples between the bursty events are redundant. One objective of this research is to acquire ECG data in a special digital form without using high frequency samplers.

Another problem of the current data acquisition systems is the requirement of a high frequency clock. For example, in conventional sigma delta modulators, the speed of analog to digital converters (ADCs) is increased by using a continuous-time integrator in synchrony with a clock of very high frequency. This clock may lead to large power consumption (to be discussed next) and cause electromagnetic interference (EMI) [3]. Alternatively, asynchronous sigma delta modulators (ASDMs) to be studied by this thesis use a relatively low frequency clock, or no clock at all [4, 7]. A Time-encoding scheme converts the amplitude information to the time information in a modulated signal with varying duty cycles.

Another objective of this work is to reduce power consumption. A high-frequency clock consumes more power than a low-frequency one because of the required swings between logic

states within the integrated circuit. In traditional systems, a global synchronization strategy is often utilized in hardware and software architectures. The development of synchronous circuit seems to have reached a bottleneck since the increasing clock frequency entails several technological problems, such as significantly increased power consumption and clock skew [8]. In this work, we use the ASDM to reduce clock frequency which will lead to a significant decrease in power consumption. It has been shown that, in radio-frequency applications, the energy spent on transmitting one single bit is equivalent to the energy to execute one thousand 32-bit arithmetic operations [11]. In ECG monitors, a great portion of energy used is spent by the transmitter. Therefore, it is essential to study data acquisition and signal processing methods to decrease the work load in data transmission.

1.2 SYNCHRONOUS AND ASYNCHRONOUS DATA ACQUISITION

1.2.1 Conventional Analog to Digital Converters

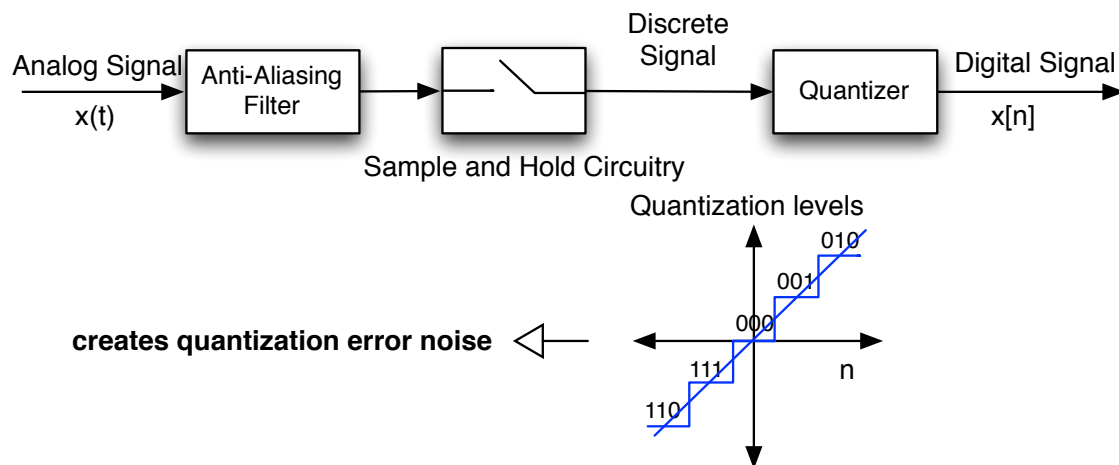


Figure 2: Generalized Analog-to-Digital Conversion Process

Figure 2 illustrates the conventional Analog-to-Digital (A/D) conversion process that transforms an analog input signal $x(t)$ into a sequence of digital values corresponding to

the samples at a regular sampling rate of $f_s = 1/T_s$, where T_s denotes the sampling interval. Although the idea of conventional A/D converters is simple and straightforward, it has some practical issues when it comes to high performance systems. First, the lowpass filter, also called the anti-aliasing filter, must have flat response over the frequency band of interest (passband) and attenuate the frequencies above the Nyquist frequency (stopband) enough to put them under the noise floor. Also, the anti-aliasing filter may cause non-linear phase distortion. Since the analog anti-aliasing filter is the limiting factor in controlling the bandwidth and phase distortion of the input signal, a high performance (and expensive) anti-aliasing filter is required to obtain high resolution and minimum distortion. In addition to an anti-aliasing filter, a sample-and-hold circuit is required. Although the analog signal is continuously changing, the output of the sample-and-hold circuitry must be constant between samples so the signal can be quantized properly. Each of these reference levels is assigned a digital code. Based on the results of the comparison, a digital encoder generates the code of the level the input signal is closest to. The resolution of such a converter is determined by the range of the quantizer and the number of the reference levels that are predefined. For high-resolution Nyquist samplers, establishing the reference voltages is a serious challenge [12]. For example, a 16-bit A/D converter, which is the standard for high accuracy A/D converters, requires $2^{16} - 1 = 65535$ different reference levels. The spacing of these levels could be beyond the limit of current VLSI technologies.

A practical A/D converter first transforms $x(t)$ into a discrete digital signal, $x(nT_s)$, where each sample is expressed with infinite precision in amplitude. Then each sample is approximated by a digital code, i.e., $x(nT_s)$ is transformed into a sequence of finite precision or quantized samples $x[n]$. This quantization process introduces errors. This quantization error is on the order of one least-significant-bit (LSB) in amplitude, and it is quite small compared to full-amplitude signals. However, as the input signal gets smaller, the quantization error becomes a larger portion of the total signal. The quantization error is defined as the difference between the digital signal and the discrete signal:

$$e[n] = x[n] - x(nT_s) \tag{1.1}$$

For an input signal which is large compared to an LSB step, the error $e[n]$ is assumed to be a random quantity uniformly distributed in the interval $(-\Delta/2, \Delta/2)$. Then for a uniform sampling A/D converter, the noise power (variance), σ_e^2 , of a B -bit A/D converter can be found as [13]:

$$\sigma_e^2 = \mathbf{E}[e^2] = \frac{1}{\Delta} \int_{-\Delta/2}^{\Delta/2} e^2 de = \frac{\Delta^2}{12} \quad (1.2)$$

where \mathbf{E} denotes expectation value, and $\Delta = \frac{R}{2^B-1}$ is the quantization step size (assuming uniform quantization) and R the range of the quantizer. The random noise assumption results in the total quantization noise being distributed equally in the frequency domain, from $-f_s/2$ to $+f_s/2$. Then the level of the noise power spectral density (PSD) can be expressed as:

$$N(f) = \sqrt{\frac{\sigma_e^2}{f_s}} \quad (1.3)$$

where f_s is the sampling frequency. It is seen that by using an oversampling A/D converter, we can reduce the quantization noise PSD linearly, but the noise is still uniformly distributed across the entire frequency range. The signal-to-quantization noise ratio (SQNR) is defined as

$$\text{SQNR} = 10 \log_{10} \frac{\sigma_s^2}{\sigma_e^2} \quad (1.4)$$

where σ_s^2 is the signal power (variance). Since quantization noise is the only noise source of an ideal A/D converter, the SQNR is equal to the signal-to-noise ratio (SNR). Therefore, the SNR of an ideal A/D converter depends on the number of bits B and range R . Increasing the resolution of a quantizer by increasing B or decreasing R results in the increase of the SNR.

1.2.2 Sigma-Delta Analog-to-Digital Converter

Various known techniques may reduce the quantization noise. Sigma-Delta Analog-to-Digital Converter is the most popular form of analog-to-digital conversion used in audio applications

[20]. As shown in Figure 3, a sigma-delta converter consists of three parts: an analog modulator, a digital filter and a decimation circuit (the latter two are served as demodulator). The modulator (shown in Figure 4) consists of an integrator (sigma) and a comparator (delta), a clock, a D-latch and a feedback loop with a 1-bit digital-to-analog (D/A) converter. Note that the clock rate, which here is also the sampling rate, is usually 64 times higher than the frequency of the input signal in order to produce a sufficient number of bitstream pulses. The more bitstream pulses are produced the better is the approximation of the input signal by the average bitstream.

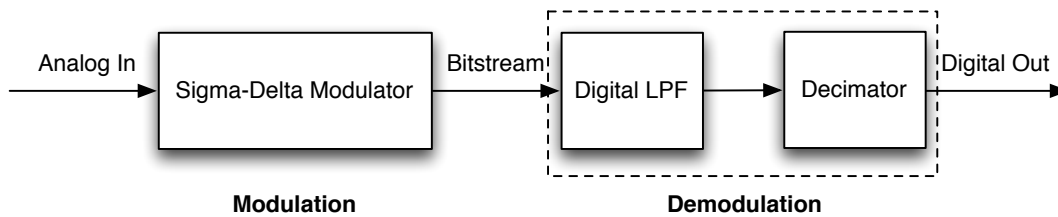


Figure 3: Generalized Analog-to-Digital Conversion Process

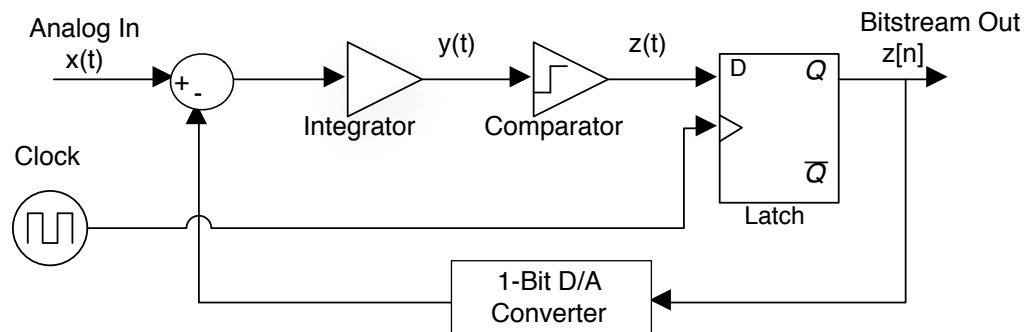


Figure 4: First Order Sigma-Delta Modulator

Figure 5 shows the input and output of the sigma-delta modulator. The local average of the modulator output $z[n]$ tracks the analog input $x(t)$. When the input is near zero, the value of the modulator output varies rapidly between a plus and a minus full scale with approximately zero mean.

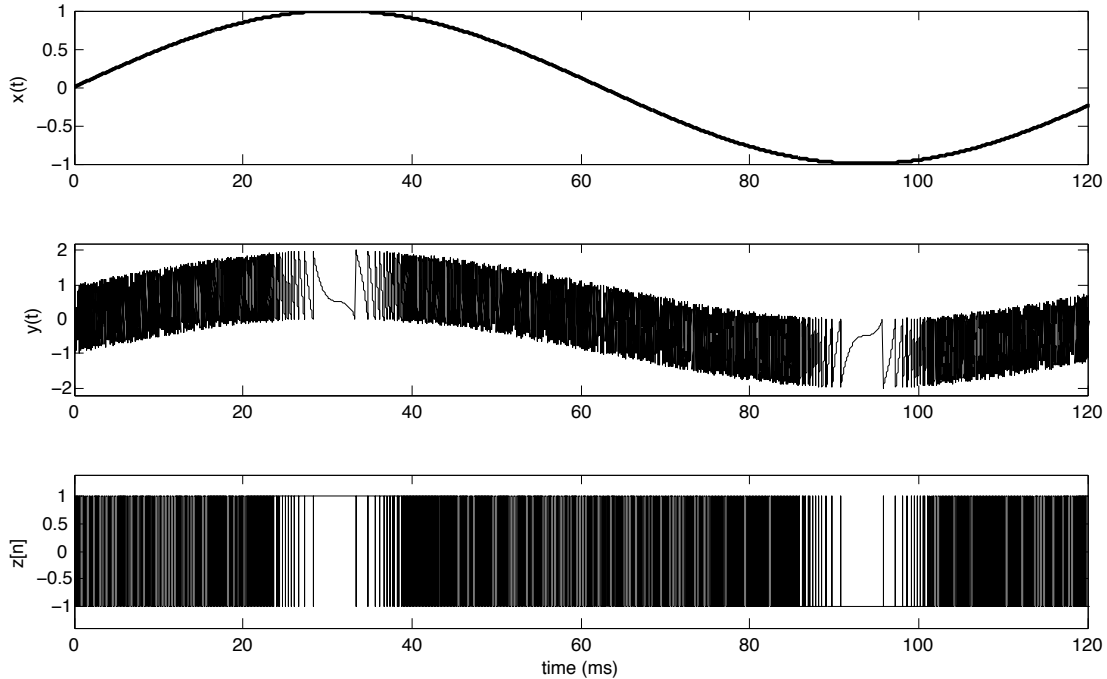


Figure 5: The waveforms of an input signal $x(t)$, the integrator output $y(t)$, and comparator output $z(t)$ of a first order sigma-delta modulator

To analyze the power spectrum density of the quantization error of a sigma delta modulator, we again assume that the error $e[n]$ brought in by the comparator is random in time. Therefore, $e[n]$ has the power spectrum density shown in Eq. 1.5:

$$E(f) = \sqrt{\frac{\sigma_e^2}{f_s}} \quad (1.5)$$

The z-domain block diagram of sigma delta A/D converter is shown in Figure 6, where the summing node to the right of the integrator represents a comparator. It is here that sampling occurs and noise $e[n]$ ($E(z)$ in z-domain) is added into the model. The transfer function of the integrator is

$$I(z) = \frac{1}{1 - z^{-1}} \quad (1.6)$$

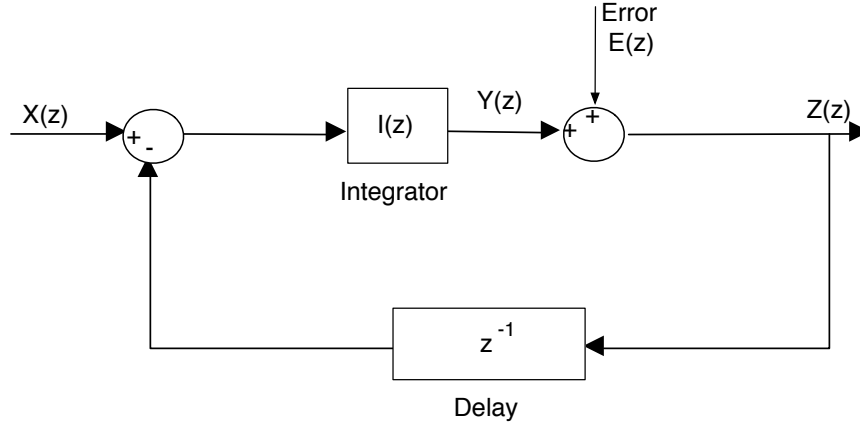


Figure 6: First order sigma-delta modulator in z-domain

When $E(z) = 0$, we have $Z(z) = X(z)$. The quantization error is defined as

$$e_{\sigma\delta}[n] = \mathcal{Z}^{-1}\{E_{\sigma\delta}(z)\} = \mathcal{Z}^{-1}\{Z(z) - X(z)\} \quad (1.7)$$

Then the quantization error in z-domain is

$$E_{\sigma\delta}(z) = Z(z) - X(z) = (1 - z^{-1})E(z) \quad (1.8)$$

Therefore, the power spectrum density of sigma delta modulator is

$$N_{\sigma\delta}(f) = |1 - e^{j2\pi f/f_s}|E(f) = |1 - e^{j2\pi f/f_s}| \sqrt{\frac{\sigma_e^2}{f_s}} \quad (1.9)$$

It is seen that term $|1 - e^{j2\pi f/f_s}|$ highpass filters the noise and pushes the noise to a higher frequency band (see the black curve in Figure 7). Since the modulator is followed by a low pass filter, most of the quantization noise will be filtered. In other words, the signal is left unchanged as long as its frequency content does not exceed the filters cutoff frequency, but the feedback loop pushes the noise into a higher frequency band. With higher order modulators the noise is even more suppressed. This is called **noise shaping**. Oversampling pushes out the noise, but it does so uniformly, that is, the spectrum is still flat. Noise shaping

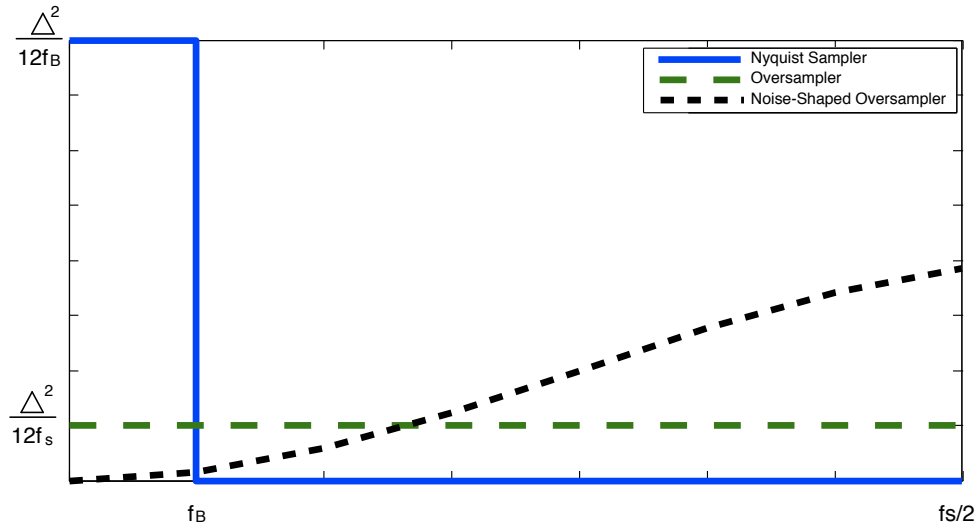


Figure 7: Spectrum of a First-Order Sigma-Delta Noise Shaper

changes that. Conservation still holds, the total noise is the same, but the amount of noise present in the audio band is decreased while simultaneously increasing the noise out-of-band, then the digital filter eliminates it. Comparison of the PSD of Nyquist sampler, oversampler and noise-sharped sampler is shown in Figure 7.

Another attribute shines true above all others for sigma-delta A/D converters and makes them a superior A/D converter: simplicity. The comparator can be viewed as a 1-bit quantizer. The simplicity of 1-bit technology makes the conversion process very fast, and very fast conversion allows the use of extreme oversampling.

1.2.3 Level-Crossing Analog-to-Digital Converter

A common characteristic of conventional A/D converter and sigma-delta A/D converter is that they use uniform (or periodic) sampling scheme, in which sampling is performed by measuring the value of the continuous signal every T seconds. In contrary, in the level-crossing scheme, the sampling operation is triggered if the input signal crosses pre-specified levels in amplitude [5]. The samples are not regularly spaced out in time: the more information

the signal has, the closer the samples. In many cases when clocks are not desirable, and low power consumption is required, level-crossing sampling is an alternative to conventional uniform sampling. However, both samples and their corresponding sample times are needed for reconstruction.

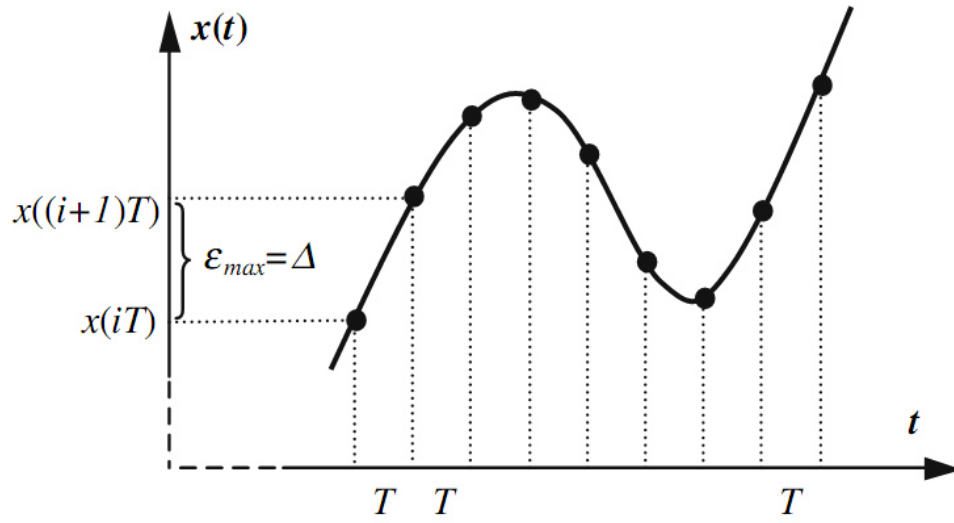
The only source of noise for an ideal level-crossing A/D converter is the quantization noise caused by limited resolution of the timer used to record the times when quantization levels are crossed. It is shown that the SNR of a level-crossing A/D converter is equivalent to that of a periodic sampling A/D converter. If a quantizer in a uniform sampling A/D converter is improved by one bit, a level-crossing sampling A/D converter has to double its frequency of the timer to obtain the same increase in the SNR [6].

The advantage of level-crossing sampling is the reduction of A/D conversion activity and energy consumption as the mean number of samples k per unit of time in the level-crossing scheme is lower than the sampling rate in the periodic one assuming that the sampling resolution is the same [5]. Figure 8 illustrates uniform sampling and level-crossing sampling. It is seen that for the same resolution Δ , level-crossing sampling takes fewer number of samples than uniform sampling during the same period of time.

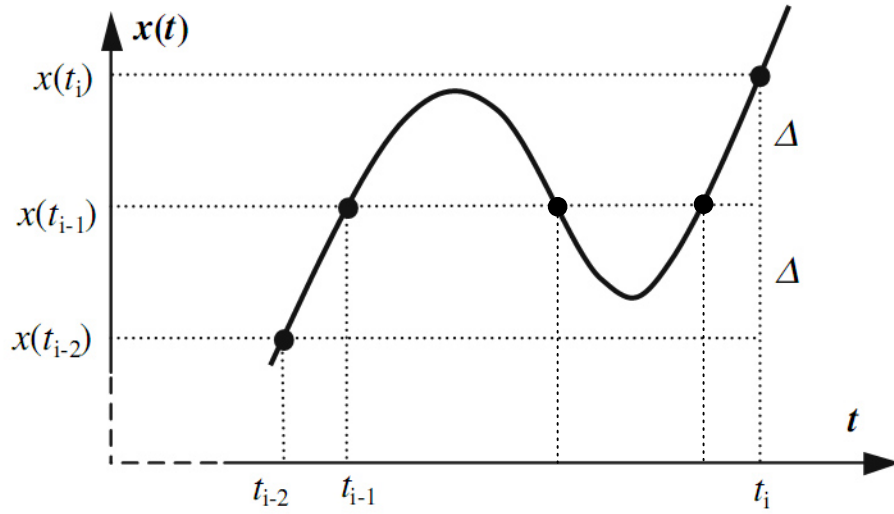
The problem of level-crossing sampling is that it is difficult to obtain optimal levels without statistical knowledge of the signal. In [5], the authors propose a sequential algorithm to obtain optimal levels.

1.2.4 Asynchronous Sigma-Delta A/D Converter

Another fast A/D converter is based on Asynchronous sigma-delta Modulation (ASDM), as shown in Figure 9. It is very similar to the sigma-delta modulation, but the ASDM separates the quantizer from the modulation loop, and the comparator is replaced by a Schmitt trigger. It generates an asynchronous rectangular wave with a duty cycle, which is approximately linearly dependent on the analog input signal. In addition, the ADSM can generate an instantaneous frequency, which is non-linearly dependent on the analog input signal [8]. The ADSM can be implemented without any clock and can be operated at low currents and supply voltages. It is equivalent to a optimal level-crossing sampler,



(a)



(b)

Figure 8: Comparison of (a) the uniform and (b) the level-crossing sampling schemes for the same sampling resolution, i.e., $\epsilon_{max} = \Delta$

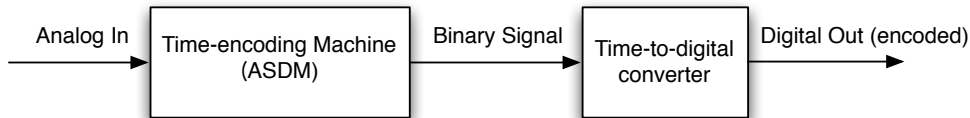


Figure 9: Asynchronous sigma delta A/D converter

thus generates less samples than the conventional A/D converter or the Sigma-Delta A/D converter. Further, since the ASDM is asynchronous, its output contains no quantization error. Therefore, the output of the ASDM is a direct representation of the analog input signal. Quantization is done in a second step, using a time-to-digital converter (TDC). Therefore, asynchronous sigma-delta A/D converters do not have the noise sharpening effect as sigma-delta A/D converters do.

The implementation of TDC can be divided into two categories, clock-led sampling and pulse-shrinking delay lines based sampling. Either ways has the potential to provide more accurate quantization than conventional A/D converter. The author of [25] has shown that under Nyquist-type rate conditions, quantization of a bandlimited signal in the time and amplitude domains are shown to be largely equivalent methods of information representation. However, the frequency references are more stable than the voltage reference sources so the time quantization is more accurate than the quantization in the amplitude domain [8]. Asynchronous sigma delta A/D converter is further discussed in chapter 2.

1.3 METHODOLOGY

In this thesis, we consider asynchronous, rather than conventional clock-led, devices to lower the power consumption and to minimize the electromagnetic field effects caused by clocks running at high frequencies. ECG signals from different sensors are processed by asynchronous sigma delta modulators (ASDMs). An ASDM is a non-linear feedback system that

represents the amplitude of its input by a sequence of zero-crossing times. Using the zero-crossing times only, we are able to recover the original continuous-time signal using sinc or prolate spheroidal interpolation.

Data digitization is based on the measurement of pulse widths of the time-encoded signal. Time measurement can be done by counting clock pulses during an input interval. However, given the lack of clocks required in bio-monitoring systems, a viable alternative is to use pulse shrinking delay lines [21, 22]. Delay lines use a logic gate delay as the time measurement unit. This method has good single-shot resolution but limited linear range. By combining the counter method and the delay line method, we not only achieve high single-shot resolution and large linear range, but also avoid electromagnetic interference caused by a high frequency clock, and reduce power consumption.

Orthogonal frequency division multiplexing (OFDM) is used for wireless transmission of the binary data resulting from the ASDM based A/D converter. The OFDM system is a multi-carrier system [14] very appropriate for transmission of the binary data from the ASDMs. At the receiver, the asynchronously modulated data can be recovered from the compressed data. Then we are able to reconstruct the original ECG signal. This system could be used for any other signal which has a bursty characteristic.

The ECG data is compressed before transmitted, in order to reduce the energy consumption in wireless transmission and alleviate network traffic. Two data compression methods are proposed. One is ASDM compression (direct compression) in time domain, which is equivalent to event-driven data transmission. This method takes advantage of the busy nature of many biosignals including ECG signals. It tends to reduce redundancy in the signal by examining the periodicity of the time-encoded signal by looking at a successive number of neighboring samples. Since the time-encoded signal carries the information about measurement of local averages of amplitude [9], it is natural to use the direct compression scheme. When the time-encoded signal becomes periodic, the corresponding ECG signal has steady amplitude and thus is considered as redundancy. Figure 10 shows an idealized model of the system I am proposing for ECG signal gathering and transmission using direct compression scheme.

The other compression method is Discrete Wavelet Transform (DWT) based compression. DWT is applied to the digitized signal $x[n]$ and then compression via redundancy reduction is applied in the transform domain rather than in the time domain. It is found that most of the coefficients for ECG signals are very close to zero, thus not containing much information, and do not need to be stored or transmitted [18]. Using this method requires us to reconstruct the ECG signal before compression. Therefore, the time-decoding is done at the transmitter but not the receiver. Figure 11 shows the idealized model of the system for ECG signal acquisition and transmission using DWT based compression scheme. Although DWT based compression can be used without ASDM, we still use ASDM based A/D converter to get the digital signal, because of the merits of this A/D converter discussed earlier. The direct compression and wavelet decomposition based compression methods are compared in Chapter 3.

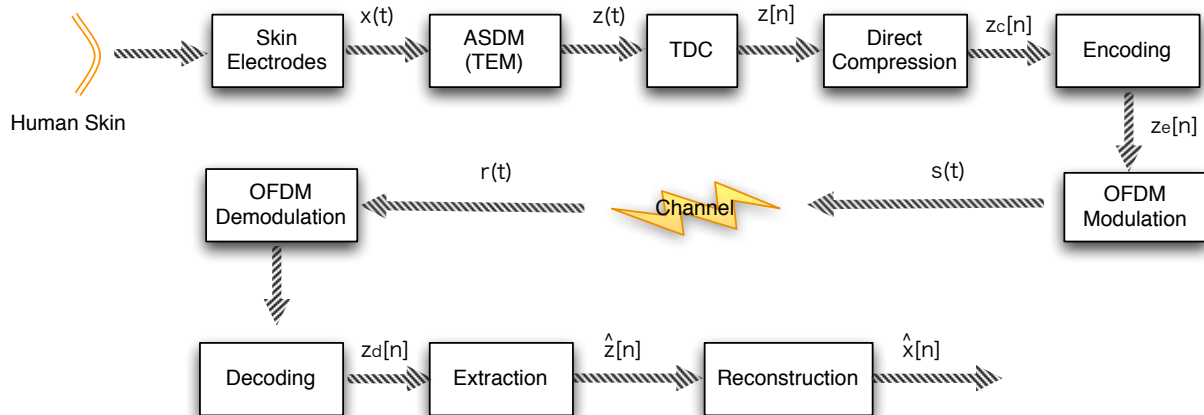


Figure 10: Model of proposed data acquisition, direct compression and transmission system

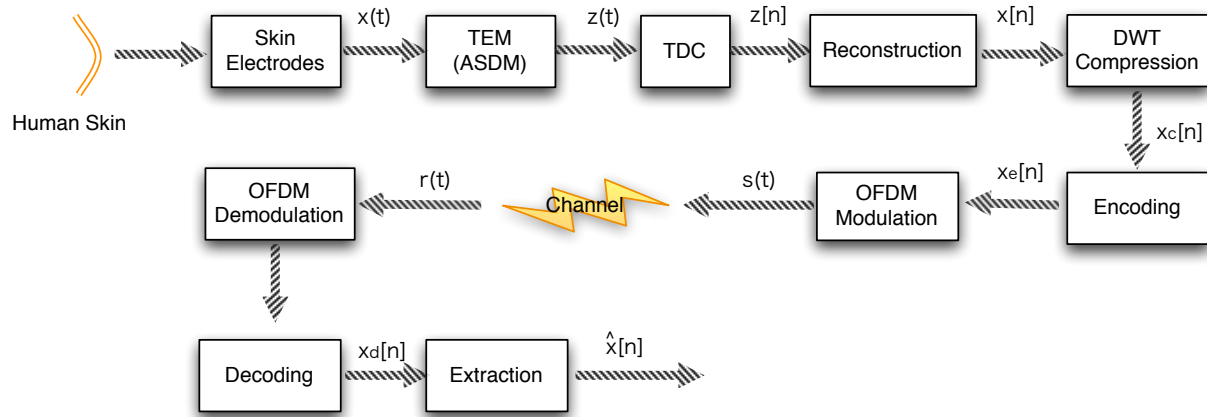


Figure 11: Model of proposed data acquisition, DWTcompression and transmission system

2.0 ASYNCHRONOUS DATA ACQUISITION

In this chapter, I will show how the ECG data collection can be accomplished without a high frequency clock using asynchronous sigma-delta A/D converters, and how the modulated data can be used to reconstruct the ECG signal.

An ASDM based A/D converter is similar to a synchronous sigma-delta A/D converter but it differs in that no sampling is done in modulation phase and as such no quantization noise is input into the modulator. Since it converts the amplitude information to time information, people also call the modulation phase a Time-Encoding Machine (TEM). The output of ASDM is then quantized in the second phase, which is call a Time-to-Digital Converter (TDC).

2.1 TIME-ENCODING MACHINE

The architecture of a TEM is shown in Figure 12. It is a nonlinear feedback system that transforms the amplitude information $x(t)$ into time information given by the zero-crossings of a binary continuous-time signal $z(t)$. The principle of the time-encoding operation is that the difference between the input $x(t)$ and the output $z(t)$ is integrated until one of Schmitt trigger thresholds $\{-\delta, \delta\}$ is reached. Then, the output of the trigger is reversed, and the integration starts again. In order to ensure that the signal $y(t)$ on the integrator output is a monotone increasing or decreasing function of the time, the maximum absolute input signal value c should be smaller than the Schmitt trigger output b :

$$|x(t)| \leq c < b \tag{2.1}$$

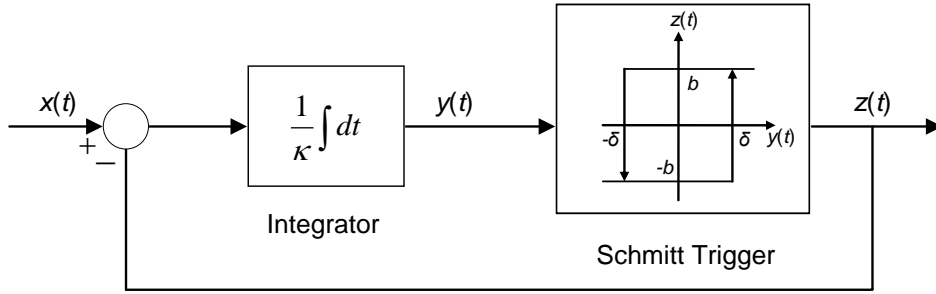


Figure 12: Block Diagram of Time-encoding Maching

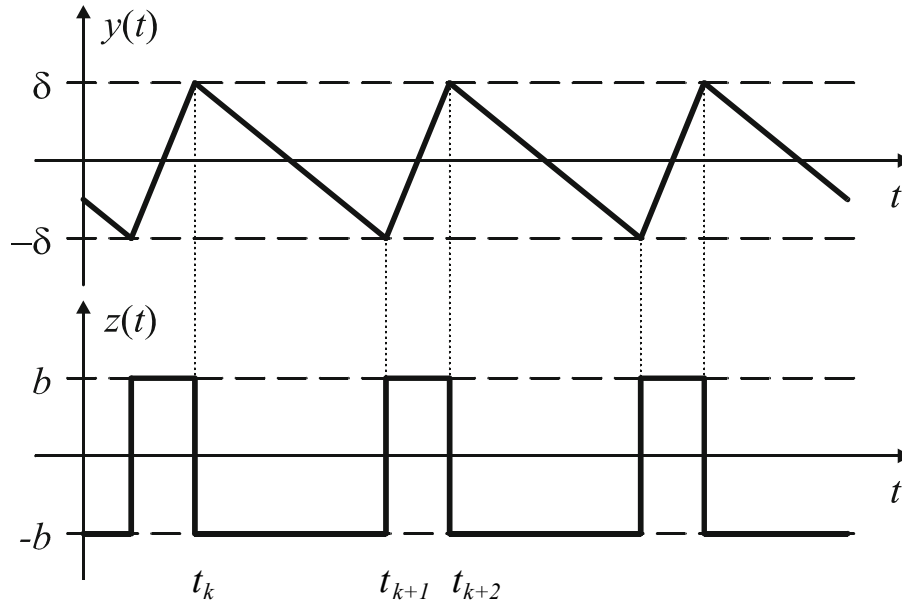


Figure 13: The waveforms on the integrator output $y(t)$, and Schmitt trigger output $z(t)$ for the DC level input signal: $x(t) = d \leq c < b$, $d = \text{constant}$, $t \in R$

Then the output of the integrator $y(t)$ at time $t_{k+1} > t_k$ in the ASDM is given by

$$y(t_{k+1}) = y(t_k) + \frac{1}{\kappa} \left[\int_{t_k}^{t_{k+1}} (x(\tau) - z(\tau)) d\tau \right] \quad (2.2)$$

Figure 13 shows the waveforms of $y(t)$ and $z(t)$ of a positive constant input. Assuming that the Schmitt trigger is in state $(-b, -\delta)$ at time t_k and it switches into state (b, δ) at time t_{k+1} , we have that

$$y(t_{k+1}) - y(t_k) = (-1)^k 2\delta \quad (2.3)$$

and

$$z(t_k) = b(-1)^{k+1}. \quad (2.4)$$

Replacing $z(t_k)$ in equation (2.2) gives

$$y(t_{k+1}) - y(t_k) = \frac{1}{\kappa} \left[\int_{t_k}^{t_{k+1}} x(\tau) d\tau - b(-1)^{k+1}(t_{k+1} - t_k) \right] \quad (2.5)$$

and finally replacing the left hand-side term by $(-1)^k 2\delta$ we find the following integral equation relating the input of the TEM with its parameters and the zero-crossing times:

$$\int_{t_k}^{t_{k+1}} x(\tau) d\tau = (-1)^k [-b(t_{k+1} - t_k) + 2\kappa\delta] \quad (2.6)$$

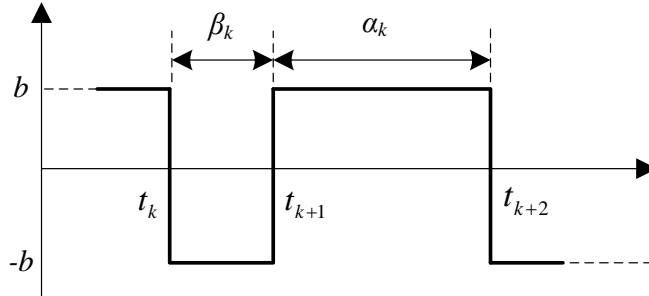


Figure 14: TEM Output: Duty-cycled Signal

It can be shown that output $z(t)$ of the TEM is a duty-cycled signal with only two possible values, b and $-b$ (see Figure 14). For $t_k \leq t \leq t_{k+2}$ and k an even integer, calling

$\beta_k = t_{k+1} - t_k$, and $\alpha_k = t_{k+2} - t_{k+1}$, the duty cycle is defined as the ratio $\alpha_k/(\alpha_k + \beta_k)$. If the signal $x(t)$ has a local average \bar{x} in the interval $t_k \leq t \leq t_{k+2}$, then the integral equation (2.6) gives

$$\begin{aligned}
(t_{k+2} - t_k)\bar{x} &= (-1)^k [-b(t_{k+1} - t_k) + 2\kappa\delta] + (-1)^{k+1} [-b(t_{k+2} - t_{k+1}) + 2\kappa\delta] \\
&= (-1)^k b [(-t_{k+1} + t_k) + (t_{k+2} - t_{k+1})] \\
&= b(\alpha_k - \beta_k)
\end{aligned} \tag{2.7}$$

which gives

$$\bar{x} = b \frac{\alpha_k - \beta_k}{(t_{k+2} - t_k)} = b \frac{\alpha_k - \beta_k}{\alpha_k + \beta_k} \tag{2.8}$$

which indicates that when $b = 1$, the difference of the time durations of two consecutive rectangular pulses divided by the total time duration of the two pulses equals the local average of the signal in the interval $t_k \leq t \leq t_{k+2}$. Thus if a level crossing sampler uses the local averages as the quantization levels the result is equivalent to that from the ASDM. Level crossing using local averages as quantization levels is optimal in the sense that it minimizes a local mean-square error, since no other statistics is available. It is important to understand that given the zero-crossing times and using the above duty-cycle interpretation only, the local averages of the signal can be reconstructed. On the other hand, the integral equation (2.6) indicates that it is possible to recover the actual values of the signal if we use the zero-crossing times and approximate the integral using a predetermined expansion for it. In [25, 7] it is shown that by assuming that a sinc or a prolate spheroidal interpolation can be used to recover a continuous-time signal from its samples, and by approximating the integral in (2.6) using the trapezoidal rule it is possible to find the interpolation coefficients to recover the original signal.

2.2 TIME-DECODING MACHINE

The sampling theorem states that, a signal $x(t)$ can be recovered exactly from its samples, $x[k] = x(kT)$, by the Whittaker-Shannon interpolation formula [13]:

$$x(t) = \sum_{k=-\infty}^{\infty} x(kT) \cdot \text{sinc}\left(\frac{t - kT}{T}\right) \quad (2.9)$$

provided that (i) the signal $x(t)$ is sampled uniformly at kT , for some sampling period T , and (ii) $x(t)$ is band-limited, that is, $T < \frac{\Omega_{max}}{\pi}$, where Ω_{max} is the maximum frequency present in the signal. The sinc function is defined as

$$\text{sinc}(t) = \frac{\sin(\pi t)}{\pi t} \quad (2.10)$$

If only nonuniform samples are available, the reconstruction problem can be presented as a projection using $\{\text{sinc}(t - t_k)\}$ as a frame for $x(t)$ so that we need to find the coefficients c_k 's

$$x(t) = \sum_{k=-\infty}^{\infty} c_k \cdot \text{sinc}(t - t_k) \quad (2.11)$$

or in matrix form:

$$\hat{\mathbf{x}} = \mathbf{S}\mathbf{c} \quad (2.12)$$

where \mathbf{S} is a matrix containing shifted sinc functions, and \mathbf{x} is a vector with entries the samples of the signal, \mathbf{c} the interpolation coefficients of the projection in terms of the sinc functions. However, this reconstruction process cannot be realized in practice since it implies that each sample contributes to the reconstructed signal at all time points, thus requiring summing an infinite number of terms. Instead, some type of approximation of the sinc functions, finite in length, has to be used. Lazar et al. [25]. proposed to approximate sinc functions by by complex exponentials:

$$\text{sinc}(t) \approx \sum_{m=-L}^L \alpha e^{jm\Omega_0 t} = \alpha \frac{\sin((L + 0.5)\Omega_0 t)}{\sin(0.5\Omega_0 t)} \quad (2.13)$$

where L is an arbitrarily large number not connected with $x(t)$, and $\Omega_0 = \frac{\Omega_{max}}{L}$, where Ω_{max} is the maximum frequency of $x(t)$.

Senay et al. [22] have shown that the Prolate Spheroidal Wave Functions (PSWF), or Slepian functions, are more appropriate than sinc functions for reconstruction. $x(t)$ is approximated by a PSWF projection:

$$x(t) \approx \hat{x}(t) = \sum_{n=0}^{M-1} \gamma_{M,n} \cdot \varphi_n(t) \quad (2.14)$$

where the value of M is chosen by making the frequency of the $\varphi_{M-1}(t)$ coincides with the frequency of $x(t)$. The PSWF projection can be written in matrix form:

$$\hat{\mathbf{x}} = \mathbf{\Phi} \mathbf{\Gamma} \quad (2.15)$$

The integral in equation (2.6) can easily be evaluated for each $t(k)$. Let $b = 1$ and

$$v_k = \int_{t_k}^{t_{k+1}} x(\tau) d\tau = (-1)^k [-(t_{k+1} - t_k) + 2\kappa\delta] \quad (2.16)$$

so that integral in 2.6 can be approximated by means of the trapezoidal rule as

$$\int_{t_k}^{t_{k+1}} x(\tau) d\tau \approx 0.5x(t_k)\Delta_t + \sum_{i=1}^{N_k-1} x(t_k + i\Delta_t)\Delta_t + 0.5x(t_{k+1})\Delta_t = \mathbf{q}_k \hat{\mathbf{x}} = \mathbf{q}_k \mathbf{\Phi} \mathbf{\Gamma} \quad (2.17)$$

where Δ_t is a small time step that satisfies $\Delta_t < \pi/\Omega_{max}$, and $N_k = (t_{k+1} - t_k)/\Delta_t$, and \mathbf{q}_k is the row vector with entries

$$q_{k,j} = \begin{cases} 0.5\Delta_t & j = t_k/\Delta_t, \text{ and } j = t_{k+1}/\Delta_t \\ \Delta_t & t_k/\Delta_t < j < t_{k+1}/\Delta_t \\ 0 & \text{otherwise} \end{cases} \quad (2.18)$$

Let \mathbf{v} be the row vector with entries v_k evaluated using 2.6, we have that

$$\mathbf{v} = \mathbf{Q} \mathbf{\Phi} \mathbf{\Gamma} \quad (2.19)$$

where \mathbf{Q} is the matrix with each column composed of vector \mathbf{q}_k . The coefficient vector $\mathbf{\Gamma}$ can be calculated using Equation (2.20), where $+$ represents the pseudoinverse operation. After finding out $\mathbf{\Gamma}$, we can use Eq. 2.15 to calculate $\hat{\mathbf{x}}$.

$$\mathbf{\Gamma} = [\mathbf{Q}\Phi]^+ \mathbf{v} \quad (2.20)$$

2.3 TIME TO DIGITAL CONVERTER

The zero-crossing times of the time-encoded signal $z(t)$ at the transmitter need to be digitized for compressive transmission. A conventional time-to-digital converter (TDC) measures the time intervals by counting clock pulses and giving a result in digital form.

The counter method has a wide linear range and good stability if the reference clock, a crystal oscillator for example, is accurate. However, in some applications, we don't want a high frequency clock because it consumes high power and causes electromagnetic interference corrupting the signal to be sampled. In this section, I discuss the implementation of TDCs using pulse-shrinking delay lines [22] combined with a low frequency clock.

2.3.1 Delay Elements and RS Flip-Flops

Delay elements and RS flip-flops are the building blocks of a pulse-shrinking delay line. One delay element consists of two CMOS inverters in series with different pull up and pull down strengths (see Figure 16). An inverter inverts the state of its input signal. For example, if the input of the inverter instantaneously changes from a high level to a low level ($V_{in} = 0$), ideally, the output would likewise instantaneously change to a high level ($V_{out} = 1$). However, because of the output capacitance, the output voltage will not instantaneously change state, but instead will change "slowly" with time. This is called a propagation delay. The delay time is a function of the aspect ratio (width to length ratio) of the nMOS or pMOS transistors [22]. A large aspect ratio increases the pull up strength of the pMOS transistor or the pull down strength of the nMOS transistor, and thus decrease the delay time of the inverter.

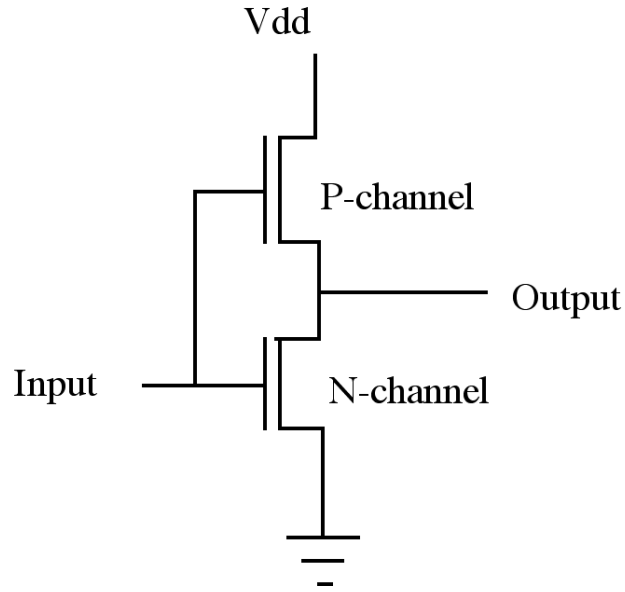


Figure 15: Inverter

If the pull up strength of the first inverting stage and the pull down strength of the second inverting stage is stronger, i.e., they have a smaller aspect ratio, the rising edge should be slower than the falling edge ($t_r > t_f$). This way the width of the pulse that propagates along the delay element decreases by the amount of the time difference $t_r - t_f$ (see Figure 16). By adjusting the bias voltage V_{bias} , the pulse-shrinking time per element can vary.

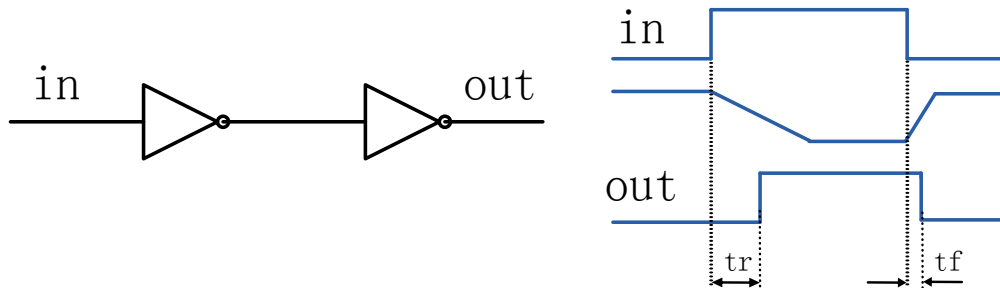


Figure 16: Delay element

A RS flip-flop is the simplest possible memory element. It is constructed by feeding the

outputs of two NOR gates back to the other NOR gates input (see Figure 17). The inputs R and S are referred to as the Reset and Set inputs, respectively. A RS flip-flop has three states:

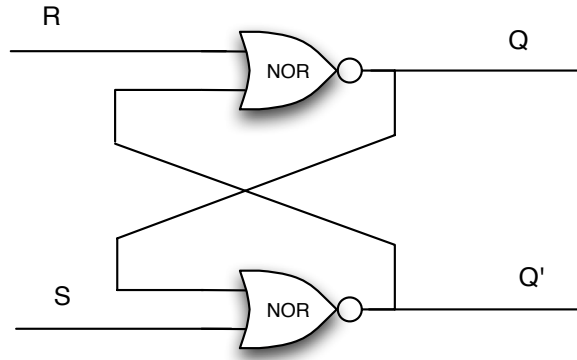


Figure 17: RS Flip-Flop

(1) $S=1$ and $R=0$: The output of the bottom NOR gate is equal to zero, $Q'=0$. Hence both inputs to the top NOR gate are equal to one, thus, $Q=1$. Hence, the input combination $S=1$ and $R=0$ leads to the flip-flop being set.

(2) $S=0$ and $R=1$: Similar to the arguments above, the outputs become $Q=0$ and $Q'=1$. We say that the flip-flop is reset.

(3) $S=0$ and $R=0$: Assume the flip-flop is set ($Q=0$ and $Q'=1$), then the output of the top NOR gate remains at $Q=1$ and the bottom NOR gate stays at $Q'=0$. Similarly, when the flip-flop is in a reset state ($Q=1$ and $Q'=0$), it will remain there with this input combination. Therefore, with inputs $S=0$ and $R=0$, the flip-flop remains in its state.

Note, the output Q' is simply the inverse of Q . It must be pointed out that there is one more combination of R and S needs to be avoided (it is called **forbidden state**). When, $S=1$ and $R=1$, as both NOR gates then output zeros, it breaks the logical equation $Q = \text{not } Q'$.

2.3.2 Pulse-shrinking Delay Lines

The principle of operation of a time interval measurement circuit based on pulse-shrinking delay lines is shown in Figure 18. One delay line consists of a chain of delay elements and RS flip-flops. The RS flip-flops are initially reset. The input S of RS flip-flops are connected to the output of delay elements. The pulse that propagates along the delay line decreases by $tr - tf$ after each delay element. We call $T = tr - tf$ the resolution of a delay line based TDC. The pulse width eventually shrinks to zero after several delay elements. The propagating pulse sets some RS flip-flops until it vanishes, after which the following flip-flops are left reset. The address of the last flip-flop that is set by the pulse is coded to the output. For example, if the resolution is 10 ns, and the last set flip-flop is number 5, then it is easy to calculate that the pulse width is 50 ns.

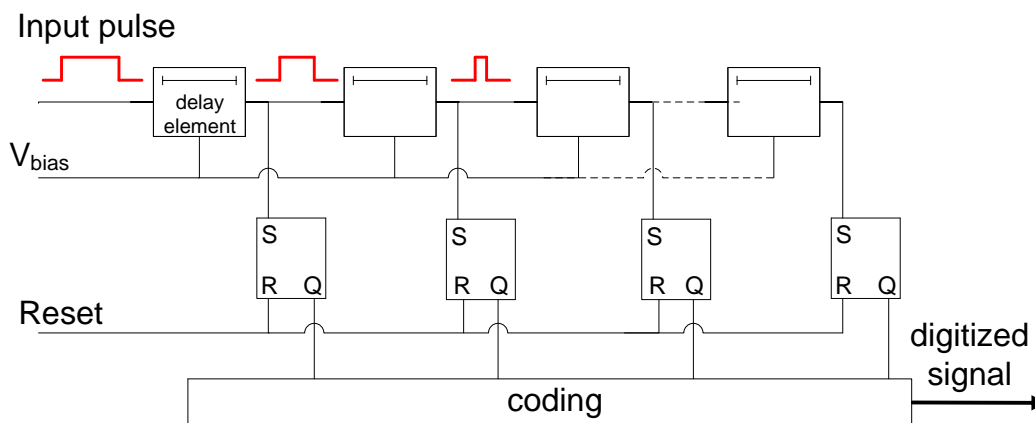


Figure 18: pulse-shrinking delay line

2.3.3 Pulse-shrinking Delay Line Based TDC

A pulse-shrinking delay line can only measure the width of one pulse. However, the time-encoded signal is a sequence of positive and negative pulses. In case of multiple pulses, we need to separate them and feed them into multiple delay lines and combine all measurement

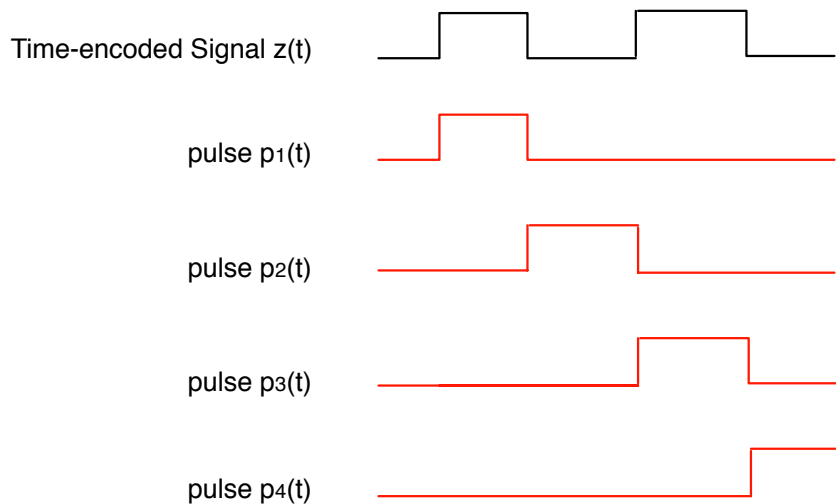


Figure 19: Pulse separation

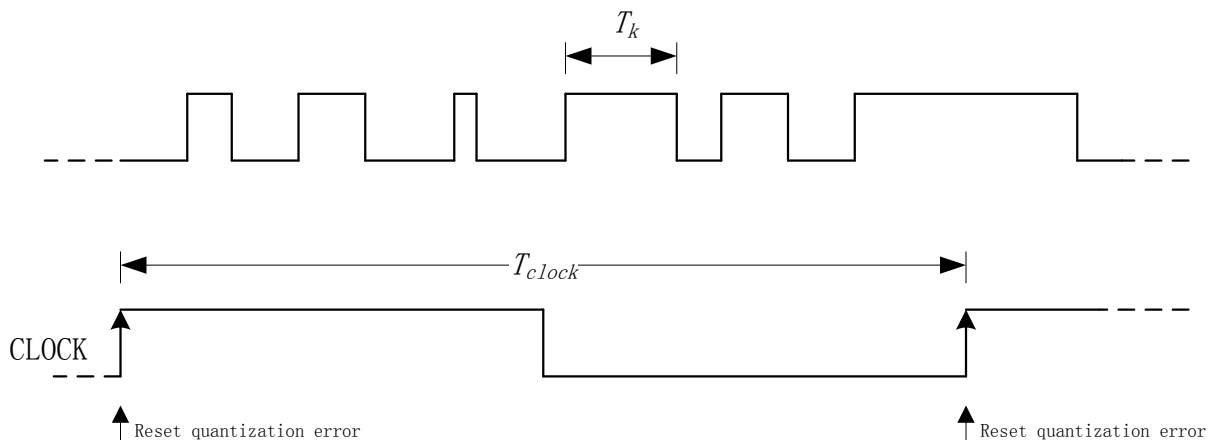


Figure 20: Coarse measurement by clocks

results. Pulse separation is illustrated in Figure 19. It can be done using edge triggered flip-flops. The digitized signal $z[n]$ is simply $\sum_{i=1}^N p_i[n]$, where N is the total number of pulses.

The resolution of a delay line determines its measurement error. The quantization error of one pulse is uniformly distributed between $(-T/2, T/2)$, and combining the measurement of n

delay lines together might cause the quantization error of the n^{th} pulse uniformly distributed between $(-nT/2, nT/2)$. However, by using a low frequency clock, the accumulated errors produced by pulse shrinking delay lines can be corrected at the beginning of each clock cycle.

Time intervals can be measured with pulse-shrinking delay lines with nanosecond or sub-nanosecond resolution [23]. We propose to combine the counter method and the delay line method, to achieve high resolution and wide linear range, and to avoid electromagnetic interference caused by a high frequency clock. Figure 20 shows the operation of our time-measurement system. We use a low frequency clock to do coarse measurement with resolution of T_{clock} . Pulse shrinking delay lines are used to measure each pulse T_k during each clock cycle and give refined measurements.

The choice of clock rate T_s affects the SNR and power consumption of the converter. Using a very low frequency clock can effectively reduce the power consumption but at the compromise of SNR as the quantization noise accumulates for a long time before it is reset. However, when a high frequency clock is not desirable, the proposed method is a better choice than the counter method.

2.4 PROS AND CONS OF ASYNCHRONOUS SIGMA DELTA A/D CONVERTER

The asynchronous sigma delta A/D converter is designed using an open loop architecture. It can prevent the clock jitter effect generated mainly from the feedback D/A converter in the conventional sigma delta A/D converter architecture [10]. Also, the circuits inside this A/D converter are really simple. Only three main blocks are required in this A/D converter design: hysteresis comparator and integrator for the TEM and inverters and flip-flops for TDC. This can ease the circuit design process for the asynchronous sigma delta A/D converter. Reducing number of major circuit blocks can reduce the overall power consumption of the A/D converter as well [8]. The use of a clock with much lower frequency than that of the signal to be sampled can prevent corruption caused by electromagnetic interference. The performance on asynchronous sigma delta A/D converter is directly proportional to the

resolution of the TDC. A basic TDC consists of inverters, which is the major element of the delay line, and RS flip-flops, which acts as a data sampler. The performance of digital circuits directly relates to the fabrication process. As CMOS fabrication process keeps improving these years, TDC resolution also increases [23].

The major noise source of an asynchronous sigma delta A/D converter is quantization noise due to limited time resolution, which is generated inside TDC only. The analog block, TEM, is isolated from these high frequency noise element. This results in a low noise floor scenario and high SNR at the TEM output. However, this might be a set back for the overall asynchronous sigma delta A/D converter. Since an asynchronous sigma delta A/D converter does not involve any noise shaping technique as the synchronous sigma delta modulator does, the quantization noise generated from the TDC shows up directly at the digital output. As a result, using existing CMOS fabrication process, the dynamic range of the synchronous sigma delta A/D converter is better than the one from asynchronous sigma delta A/D converter [10].

However, as CMOS fabrication process keeps enhancing, the performance of digital circuits, as well as the TDC resolution, improves. With better TDC resolution, the overall performance of the asynchronous sigma delta A/D converter should also improve. This provides a potential benefit on asynchronous sigma delta A/D converter over its conventional synchronous counterpart in the future.

3.0 DATA TRANSMISSION AND COMPRESSION

3.1 ORTHOGONAL FREQUENCY DIVISION MULTIPLEXING

Continuous measuring of heart electrical activity requires that data be transmitted wirelessly. The major problems with wireless transmission are: fading caused by multipath propagation, intersymbol interference (ISI) and shadowing [14]. Frequency selective fading causes some frequencies to be enhanced while others to be attenuated. Single-carrier techniques are vulnerable to fading in multipath propagation, especially when the bit rate is high. The frequency selective fading effect can be counteracted by applying some diversity scheme such as orthogonal frequency division multiplexing (OFDM), which is a multicarrier modulation scheme that is robust against ISI and fading caused by multipath propagation [14].

In OFDM, the data is divided into a number of sub-bands with carriers orthogonal to each other. Instead of transmitting in serial way, data is transferred in a parallel way. Only a small part of the data is carried on each subcarrier, and by lowering the bit rate per subcarrier, the influence of ISI is significantly reduced. Figure 21 shows an example of OFDM spectrum. At the central frequency of each subchannel, there is no crosstalk from other subchannels.

By dividing an entire channel bandwidth into many narrow sub-bands, the frequency response over each individual sub-band is relatively flat. Since each sub-band covers only a small fraction of the original bandwidth, equalization is potentially simpler than in a serial data system. In addition, by using cyclic prefix, the sensitivity of the system to delay spread can be reduced. The baseband data on each subcarrier is independently modulated (mapped)

commonly using some type of quadrature amplitude modulation or phase-shift keying. The baseband OFDM signal is expressed in Equation (3.3), where X_k are the data symbols, N is the number of sub-carriers, and T is the OFDM symbol time.

$$\nu(t) = \sum_{k=0}^{N-1} X_k e^{j2\pi kt/T}, \quad 0 \leq t < T, \quad (3.1)$$

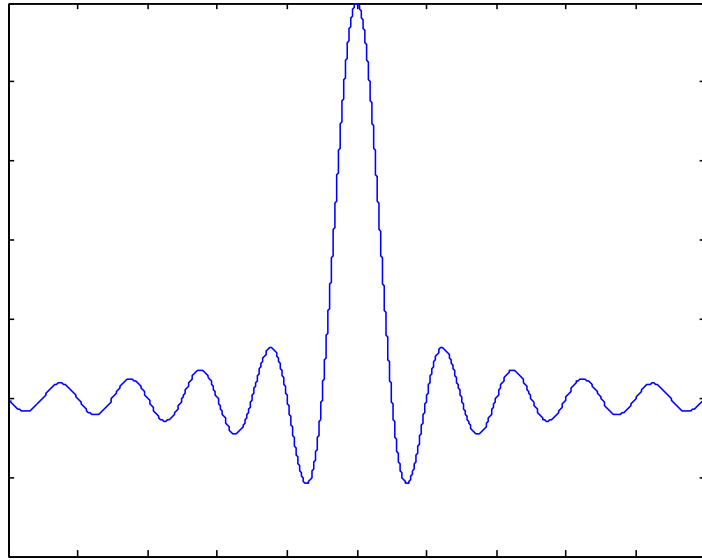
The subcarrier orthogonal property is satisfied:

$$\frac{1}{T} \int_0^T (e^{j2\pi k_1 t/T})^* (e^{j2\pi k_2 t/T}) dt = \frac{1}{T} \int_0^T e^{j2\pi(k_2 - k_1)t/T} dt = \delta_{k_1 k_2} \quad (3.2)$$

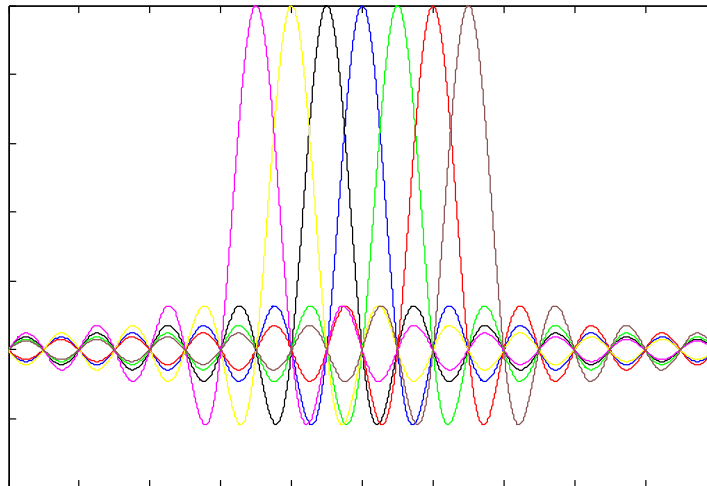
where the $*$ indicates the complex conjugate, and δ is the Kronecker delta. For wireless applications, the transmitted signal modulates an analog carrier of frequency f_c . The transmitted signal can be represented as:

$$s(t) = \Re \{ \nu(t) e^{j2\pi f_c t} \} \quad (3.3)$$

In practice, OFDM systems are implemented using a combination of inverse fast Fourier Transform (IFFT) and fast Fourier Transform (FFT) blocks (see Figure 22). After a serial to parallel transformation (S/P) the source symbols at the transmitter are treated as if they are in the frequency-domain. After mapping the symbols using QAM or PSK, an IFFT is computed on each set of symbols, resulting in a set of complex time-domain samples. After some additional processing (cyclic prefix insertion, up-converting, D/A conversion and etc.), the time-domain signal that results from the IFFT is transmitted across the channel. At the receiver, an FFT block is used to process the received signal and bring it into frequency-domain signal, which is an estimate of the original signal at the transmitter.



(a) A single subchannel



(b) 7 carriers

Figure 21: Examples of OFDM spectrum

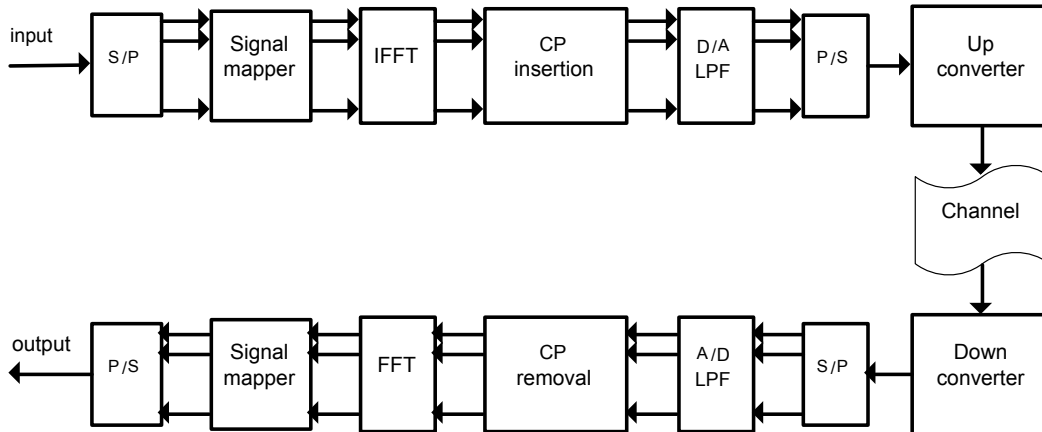


Figure 22: Block diagram of a simple OFDM system

3.2 DATA COMPRESSION

3.2.1 Direct Data Compression

Although digital storage media is not expensive and computational power has exponentially increased in last years, the possibility of ECG compression still attracts the attention, due to the huge amount of data that has to be transmitted, growing at a rate of 7.5-540 kB per minute per patient, depending upon the time and amplitude sampling rate and number of sensors. Different compression methods attempt to reduce the dimensionality of the non-stationary and quasi-periodical ECG signal, while retaining all clinically significant features including P-wave, QRS complex and the T-wave.

The main idea of compressive data transmission is to minimize the transmitted data to data corresponding to events of interest. As we can see in Figure 1, an electrocardiograph is a bursty signal where a large part of the signal does not have much change in amplitude, for example, the P-R segment, S-T segment, and the segment between two heart beats. Using Eq. 2.8, it is very easy to tell whether a signal $x(t)$ has a significant change in amplitude or not for a time interval from the periodicity of its corresponding time-encoded signal $z[n]$.

To save power in transmitting the ECG data we consider the difference between neighboring local averages,

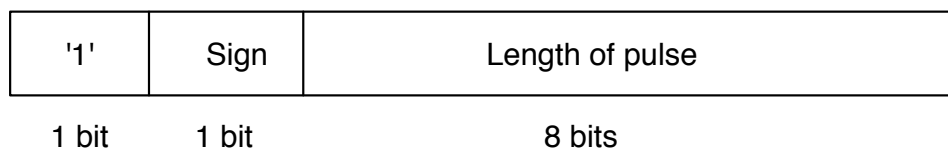
$$D(k) = \frac{\alpha(k) - \beta(k)}{\alpha(k) + \beta(k)} - \frac{\alpha(k+1) - \beta(k+1)}{\alpha(k+1) + \beta(k+1)} \quad (3.4)$$

and when it is small enough (under a certain threshold), we consider the corresponding signal segment insignificant. The threshold value L is defined as

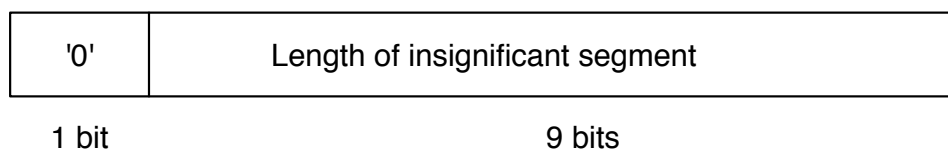
$$L = u \times \max\{D(k)\} \quad (3.5)$$

where u is called a thresholding factor.

Using a fixed frame size can save energy in wireless transmission, because it allows transceivers to turn their radio on exactly in time [26]. Therefore, instead of suspending the transmission when insignificant data segments are detected and resuming it when significant data segments are detected, we encode the whole data set. The data transmitted consists of: 1) zero-crossing times in significant segments and their corresponding length and sign; 2) insignificant segment starting times and their corresponding length. Figure 23 shows the coding stream.



(a) Representation of significant zero-crossing times



(b) Representation of insignificant zero-crossing times

Figure 23: The coding stream of the significant and insignificant zero-crossing times

At the receiver, we are able to decode the data stream to obtain several significant data segments with insignificant ones between them which correspond to when the signal $z[n]$ is

close to being periodic. The original $z[n]$ is recovered by filling in the insignificant parts with the last pattern of α and β of the significant segments. There can be extraction errors if the insignificant segments in the original $z[n]$ are not absolutely periodic.

The compression performances were computed in terms of the Percent Root-Mean Square Difference (PRD) and the Compression Ratio (CR) defined as follows:

$$\text{PRD} = \sqrt{\frac{\sum_{n=0}^N (z[n] - \hat{z}[n])^2}{\sum_{n=0}^N z[n]^2}} \quad (3.6)$$

$$\text{CR} = 1 - \frac{\text{length of output stream}}{\text{number of zero-crossing times in } z[n] \times 10} \quad (3.7)$$

where $z[n]$ denotes the time-encoded signal before compression and $\hat{z}[n]$ the extracted version. A low PRD value indicates a high reconstruction accuracy. The PRD and CR vary when different thresholds are chosen. A power-efficient transmission scheme requires the largest possible compression. However, there is a trade-off between CR and PRD — too large compression ratio result in large extraction error. An optimization routine has been developed to find the threshold level that yields to the highest CR and the lowest PRD. This has been achieved through minimizing the cost function

$$P = \text{PRD} + W/\text{CR}. \quad (3.8)$$

where W is the weighting factor. The selection of W is based on which is more important: high CR or low PRD.

3.2.2 Data Compression Based on Discrete Wavelet Transform

Data compression can also be achieved by transforming the data, projecting it on a basis of functions, and then encoding this transform. Because of the bursty nature of ECG signals, the transform used must be adequate for representing transient elements of a non-stationary signal. In order to save CPU time, the algorithm must be fast. Discrete Wavelet Transform (DWT) meets these criteria. Transient elements of a signal can be represented by a smaller

amount of information using DWT than would be the case if some other transform, such as the more widespread discrete cosine transform, had been used [15]. These coefficients sometimes can then be compressed more easily than the original signal because the information is concentrated in just a few coefficients. Actually, the DWT has emerged as a powerful mathematical tool in audio and image data compression [16] [17]. The author of [18] used DWT and QRS-Complex Estimation to compress ECG signals and a CR of 93% with a PRD 0.47 was reported.

The key issues in DWT and inverse DWT are signal decomposition and reconstruction, respectively. The basic idea behind decomposition and reconstruction is low-pass and high-pass filtering with the use of downsampling and upsampling respectively. One can choose the level of decomposition j based on a desired cutoff frequency. Given a signal $x[n]$ of length N , the DWT consists of $\log_2 N$ stages at most. The first step produces two sets of coefficients: approximation coefficients cA_1 , and detail coefficients cD_1 . These vectors are obtained by convolving $x[n]$ with the low-pass filter g for approximation, and with the high-pass filter h for detail, followed by decimation (downsampling):

$$cA_1[n] = \sum_{k=-\infty}^{\infty} x[k]g[2n - k] \quad (3.9)$$

$$cD_1[n] = \sum_{k=-\infty}^{\infty} x[k]h[2n - k] \quad (3.10)$$

The next step splits the approximation coefficients cA_1 in two parts using the same scheme, and producing cA_2 and cD_2 , and so on. The resulted wavelet coefficients are $c[n] = [cA_N, cD_N, cD_{N-1}, \dots, cD_1]$. Figure 24 shows how a 3-level DWT is done by cascading filter banks, where $\downarrow 2$ represents the downsampling operator by a factor of 2.

Figure 25 shows an implementation of a three-level inverse DWT, where \tilde{g} and \tilde{h} are low-pass and high-pass synthesis filters, respectively, and the block $\uparrow 2$ represents the upsampling operator by a factor of 2. The four coefficients are recursively combined to reconstruct the original signal $\hat{x}[n]$.

Data compression is easily done by thresholding the DWT coefficients to get significant ones and insignificant ones. The threshold value is calculated as

$$L = v \times \text{mean}\{E_c[n]\} \quad (3.11)$$

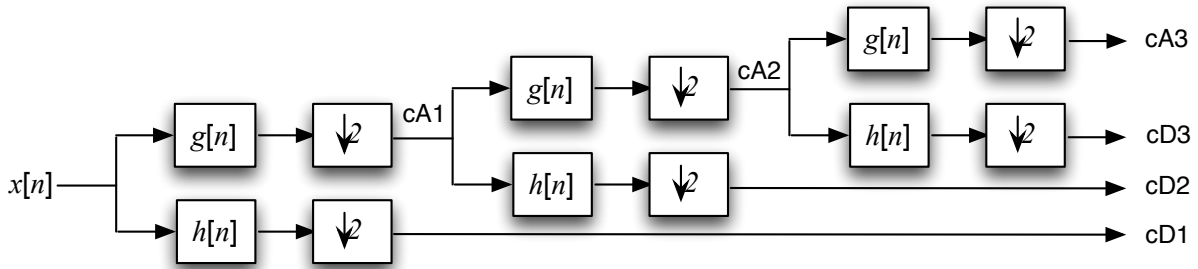


Figure 24: A three-level iterative filter bank: Forward DWT

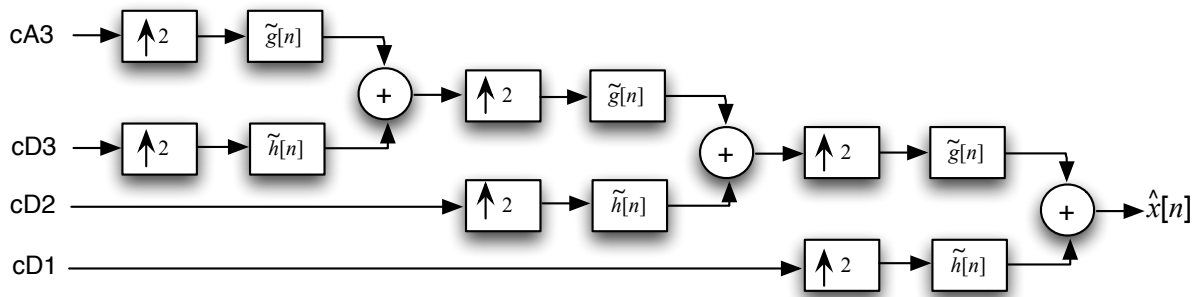


Figure 25: A three-level iterative filter bank: Inverse DWT

where v is the thresholding factor and $E_c[n]$ is calculated as the square of each coefficient, which represent the energy of each sub-band.

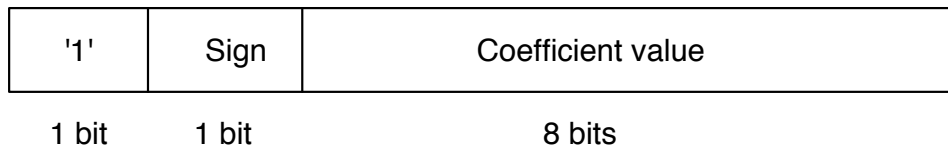
The transmitted data are the significant coefficients, and the lengths of segments of insignificant coefficients in binary forms. These requires coding. In this simulation, I used 11-bits to encode the original ECG signal. The significant and insignificant coefficients are coded separately. Figure 26 illustrates the coding stream. The runs of significant coefficients are coded as follows: The left most bit of '1' indicates the run of significant coefficients. The next bit of value '0' or '1' is the sign of the coefficient. The next 8 bits encode the value of the significant coefficient. The runs of insignificant coefficients are coded as follows: The left

most bit of '1' indicates the run of significant coefficients. The next 9 bits encode the length of the insignificant coefficient.

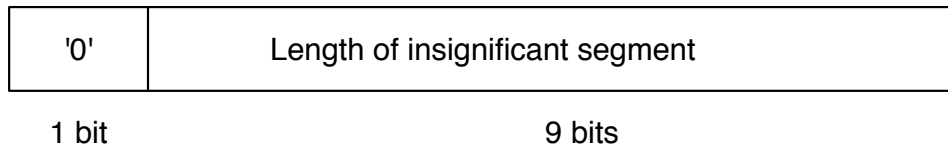
The PRD and CR for DWT compression are calculated using Equation (3.12) and (3.13). The optimal threshold can also be found using Equation (3.8).

$$\text{PRD} = \sqrt{\frac{\sum_{n=0}^N (x[n] - \hat{x}[n])^2}{\sum_{n=0}^N x[n]^2}} \quad (3.12)$$

$$\text{CR} = 1 - \frac{\text{length of output stream}}{\text{length of } c[n] \times 11} \quad (3.13)$$



(a) Representation of significant coefficients



(b) Representation of insignificant coefficients

Figure 26: The coding stream of the significant and insignificant coefficients

4.0 SIMULATIONS

In this chapter, I will show the MATLAB simulation result of time-encoding and time-decoding, time-to-digital conversion based on delay lines, and data compression, and analyze OFDM transmission bit error rate.

4.1 TIME-ENCODING AND TIME-DECODING

Figure 4.1 shows the time-encoding processes for different signals. In (a), the input $x(t) = 0$, the output $z(t)$ of the ASDM is a train of rectangular pulses of 50% duty cycle ($\alpha_k = \beta_k$). In (b), input $x(t)$ is a positive constant, the output $z(t)$ of the TEM has a duty cycle bigger than 50% ($\alpha_k > \beta_k$). It is seen that we obtain rectangular pulses with a duty cycle depending on the value of the amplitude of the input signal. A characteristic of these cycles is that the local average of the input signal $x(t)$ equals the average of the output signal $z(t)$ in each of the intervals $[t_k, t_{k+2}]$. Figure 28 and 29 show the time-decoding (using PSWFs) processes for a sinusoid and an ECG signal respectively. We can see that big reconstruction error appears at the places where frequency changes significantly.

4.2 TIME TO DIGITAL CONVERTER

In this simulation, I compare the performance of delay line based TDC with clock-based TDC. We compare the sampling result of delay line based TDC to illustrate how the quantization error is accumulated over time and how it can be reset by a low frequency clock.

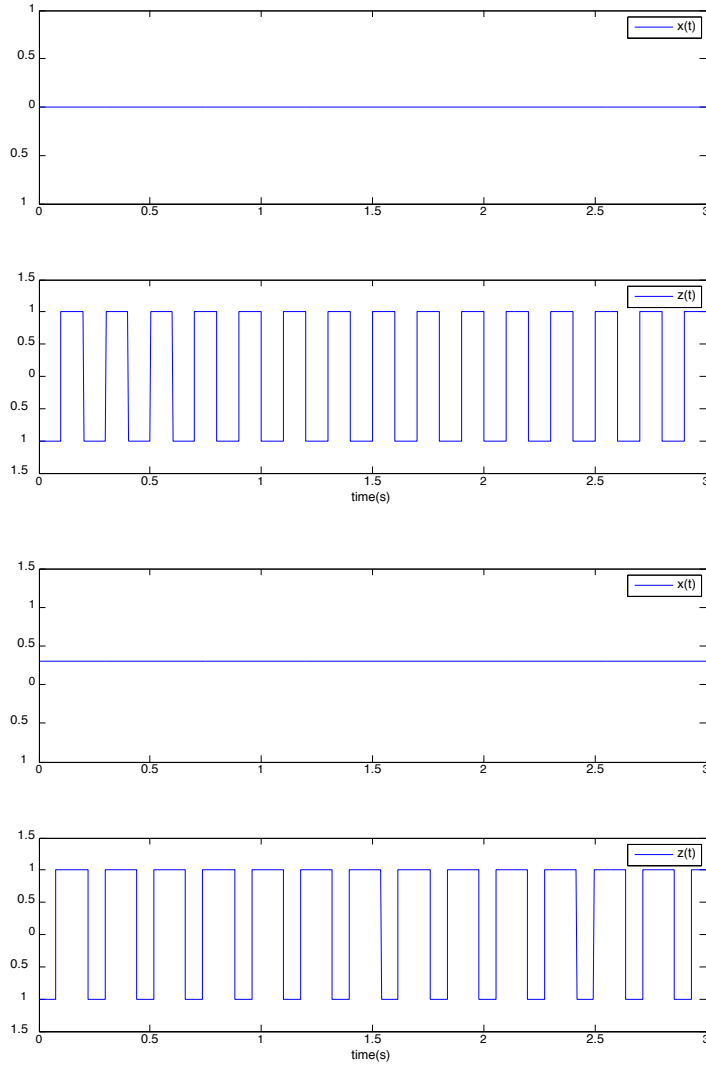
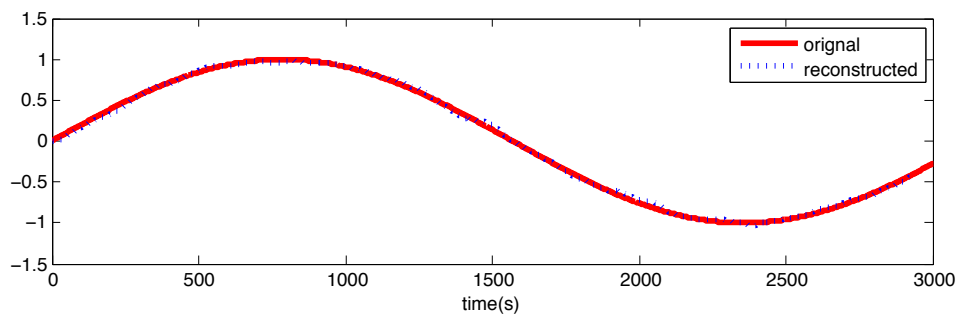


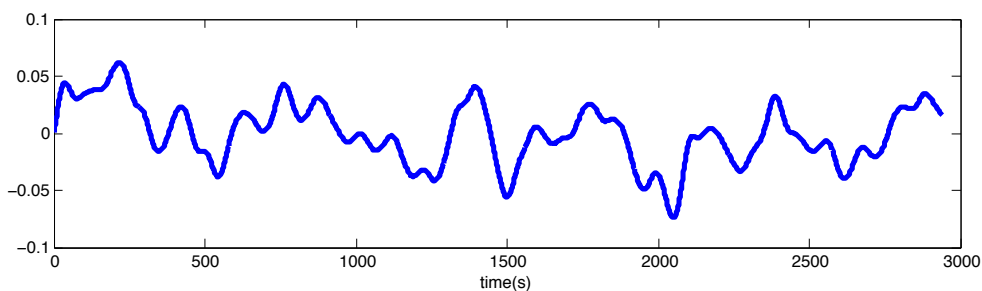
Figure 27: time-encoding processes for different signals

The delay line parameters are chosen as follows: $t_r = 0.011s$, $t_f = 0.001s$, so it is equivalent to a sampling rate $T_s = 0.01s$; the clock rate is $T = 1s$.

From Figure 30, we can see that sampling difference begins to appear in the measurement of 4th pulse, and it is passed to a few pulses behind. Quantization difference accumulates to $2T_s$ at the 9th pulse but is corrected to 0 by the clock rising edge at T . It behaves in the same accumulation-correction pattern during every clock cycle. However, as the CMOS



(a) Original signal vs. reconstructed signal



(b) Normalized reconstruction error

Figure 28: Time-decoding of a sinusoid using PSWFs

fabrication process keeps enhancing, the resolution of delay line based TDC can reach sub-nanoseconds. Therefore, the error accumulated in each clock cycle can be controlled to be very small and negligible.

4.3 DATA COMPRESSION

Figure 31 illustrates the direct compression of an ECG signal. It is seen from Figure 32 that only 33.5% of time-domain signal is considered to contain useful information and thus the corresponding zero-crossing times are transmitted. The segments with zero values indicate the insignificant segments. Data extraction (reconstruction from compressed data) is done at

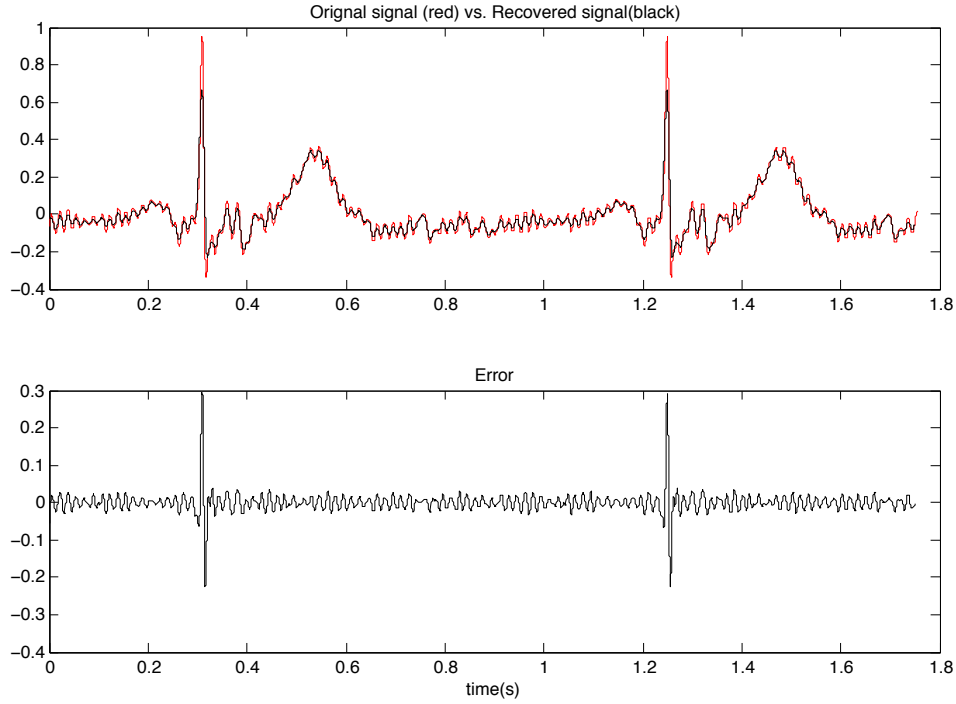


Figure 29: Time-decoding of an ECG signal using PSWFs

the receiver. Figure 33 shows the extracted signal $\hat{z}[n]$ in the case that the wireless channel is perfect (no noise, no multi-paths fading). Figure 34 shows the compression error $z[n] - \hat{z}[n]$.

Figure 35 and 36 show the reconstruction of ECG signals from compressed time-encoded signals. Comparing to the reconstruction from uncompressed time-encoded signal in Figure 29, we can clearly see that negative peaks appear ahead of the P-waves in the reconstructed signals. These peaks are produced from the compression of $z[n]$. It is seen that when the ECG signal is clean, such negative peaks are less obvious than when the ECG is noisy.

In DWT based data compression, the data to be compressed are the Wavelet coefficients $c[n]$. Figure 37 shows that around 90% of the coefficients are very small and are considered as insignificant. The extracted ECG signal from compressed wavelet coefficients is shown in Figure 38. It is seen that even at such a high compression ratio, the ECG signal still keeps all important features.

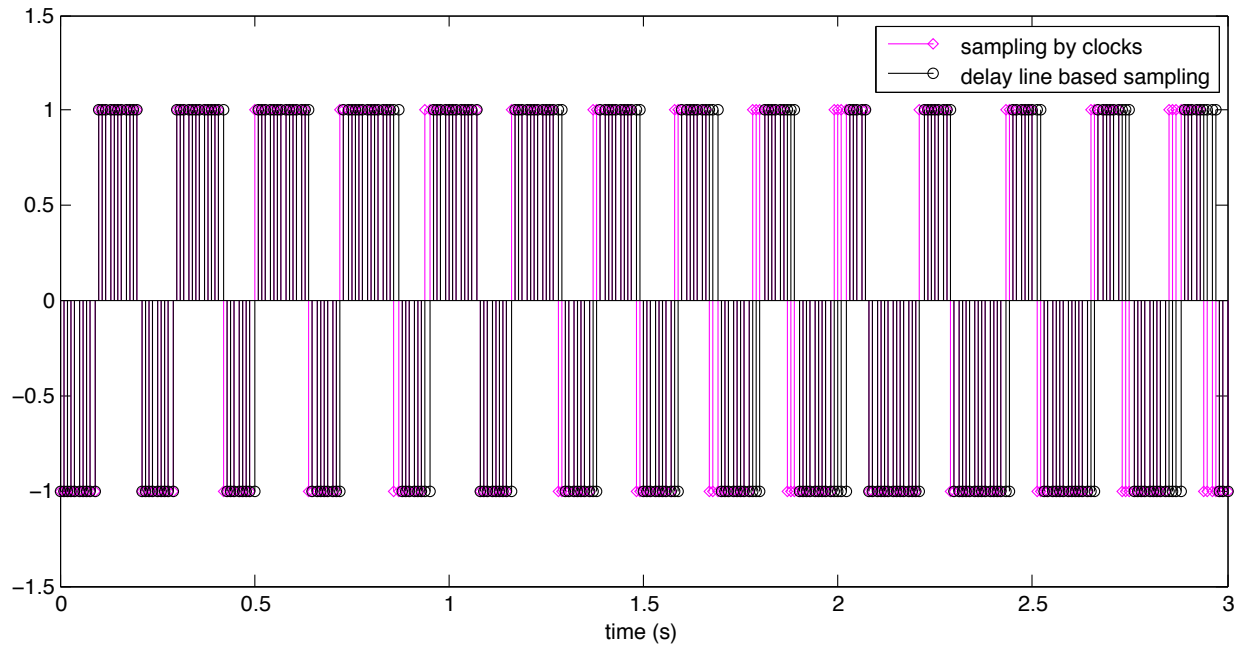


Figure 30: Comparison of different sampling methods. Quantization error of delay-lines based sampling is reset every 1 second.

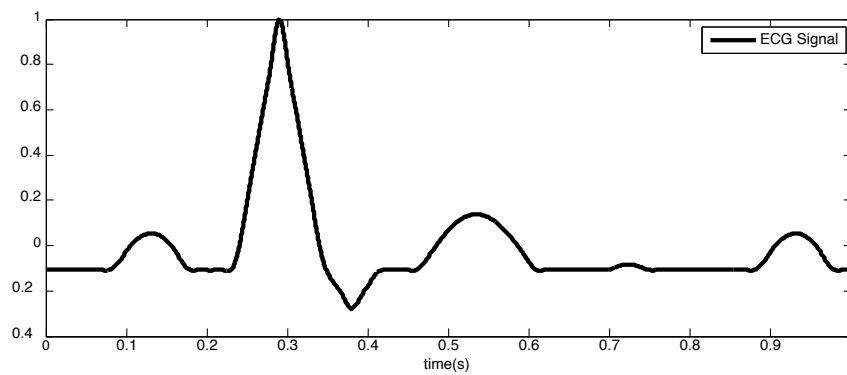


Figure 31: ECG Signal

In this thesis, we use exhaustive optimization to find the optimal threshold values. Table 1 and 2 show the CR and PRD values, and cost function values P when $W = 0.2$ under differ-

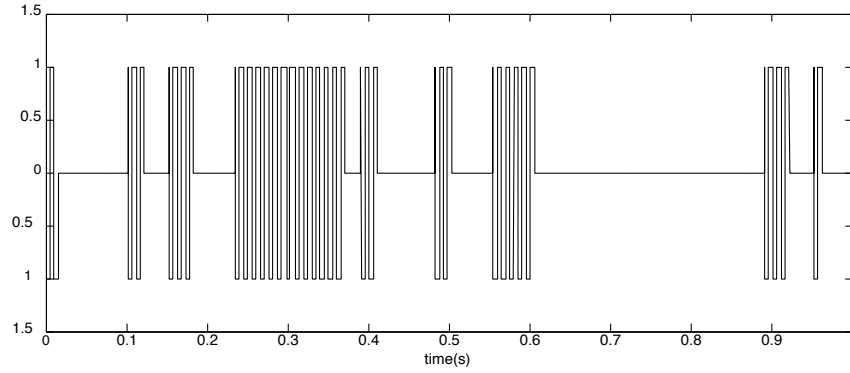


Figure 32: Compressed Time-encoded Signal

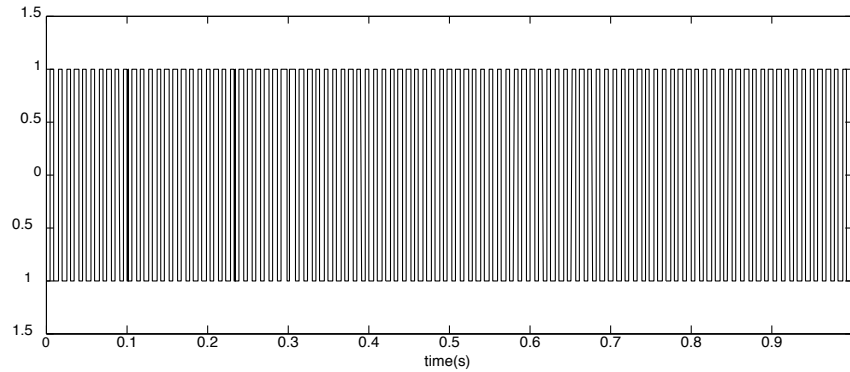


Figure 33: Extracted Time-encoded Signal

ent thresholds using direct compression and DWT based compression respectively. For direct compression, the cost function is minimized when $u = 0.08$. For DWT based compression, when $v = 0.06$, the cost function is minimized.

Figure 39 is the CR-PRD plot of both compression methods. It is seen that when $CR = 0.35$, the two methods achieve the same PRD (around 0.13). When CR is small, direct compression method has better PRD than that of DWT based method. However, when CR is large, DWT based compression has much better performance than direct compression.

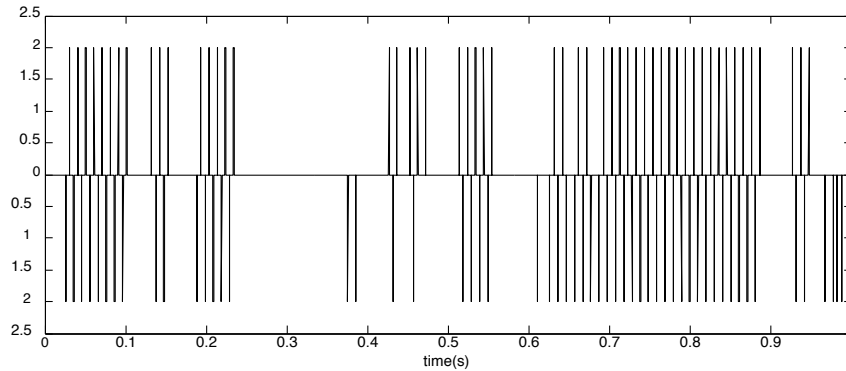


Figure 34: Compression Error

4.4 TRANSMISSION ERROR ANALYSIS

The OFDM transmission bit error rate (BER) is estimated under an additive Gaussian noise channel (AWGN channel) and Rayleigh multipath fading channel. The BER of OFDM is compared with that of single carrier transmission. In this simulation, the Rayleigh multipath channel was modeled as n -path channel with the real and imaginary part of each tap being an independent Gaussian random variable with zero mean and unit variance. The impulse response of the Rayleigh channel is,

$$h(t) = \frac{1}{\sqrt{n}} [h_1(t - t_1) + h_2(t - t_2) + \dots + h_n(t - t_n)] \quad (4.1)$$

where $h_n(t - t_n)$ is the channel coefficient of the n^{th} path. The term $1/\sqrt{n}$ is for normalizing the average channel power over multiple channel realizations to 1. The received signal in Rayleigh fading channel is

$$y(t) = h(t) * x(t) + n(t) \quad (4.2)$$

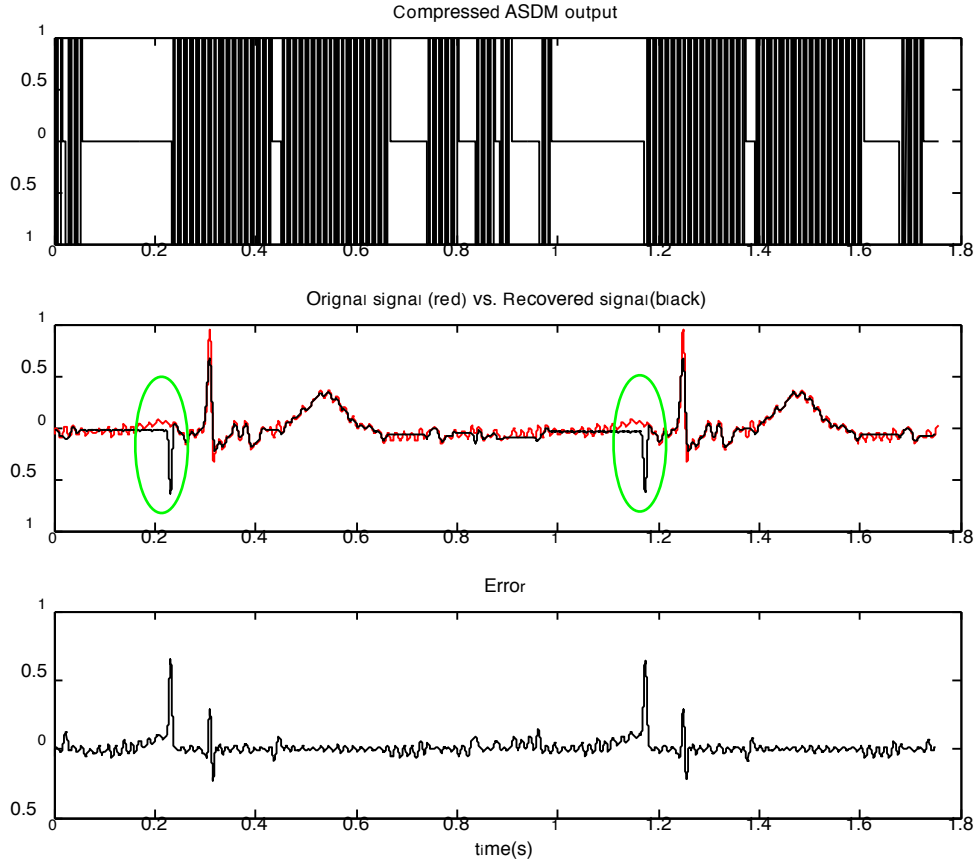


Figure 35: Signal reconstruction using PSWFs (ECG input is noisy)

where $*$ denotes convolution, $n(t)$ is the AWGN. In frequency domain, the received signal can be written as

$$Y = HX + N \quad (4.3)$$

Suppose the channel is known at the receiver, equalization can be performed at the receiver by dividing the received symbol Y by the apriori known H , i.e.

$$X = \frac{Y}{H} + \tilde{N} \quad (4.4)$$

where $\tilde{N} = N/H$ is the additive noise scaled by the channel coefficient.

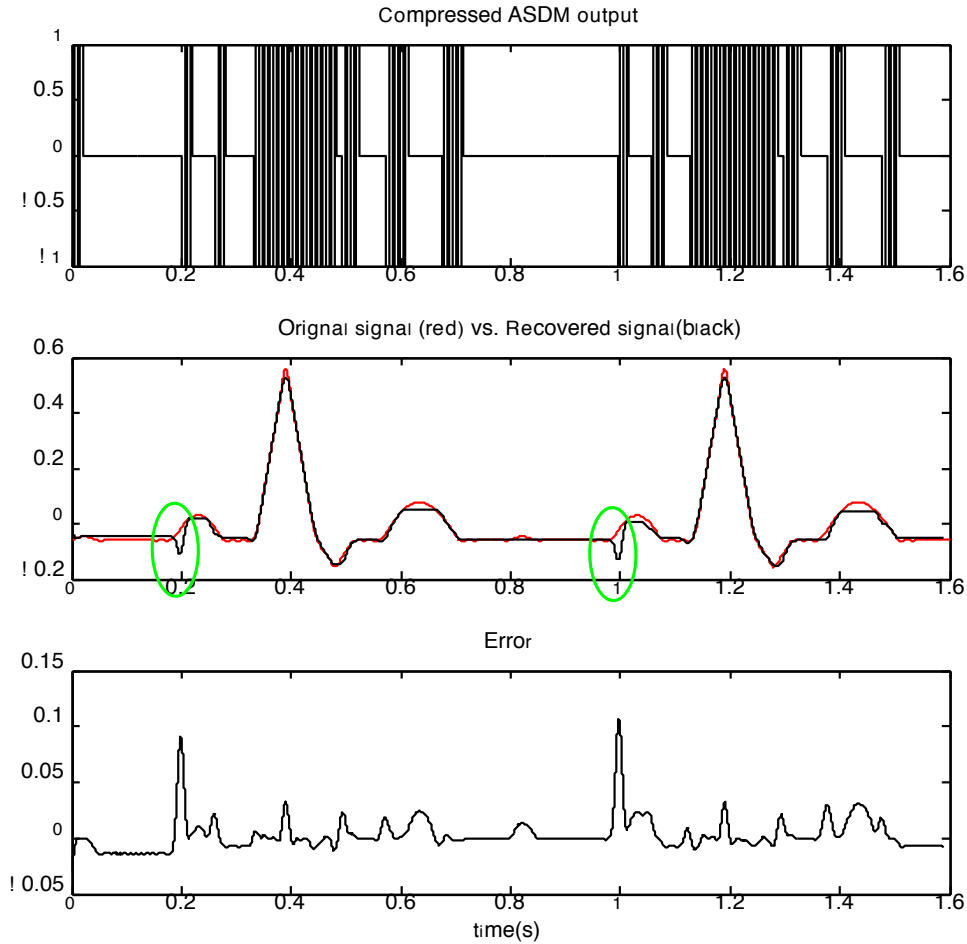


Figure 36: Signal reconstruction using PSWFs (ECG input is clean)

The OFDM system in this simulation is loosely based on IEEE 802.11a specifications (see Table 4.4).

The bit error is calculated as the number of error data at the receiver compared to the data at the transmitter. Figure 40 shows the BER performance under two different channel models. It is seen that under AWGN channel, OFDM transmission has the same BER as single carrier transmission. Under Rayleigh multipath fading channel, the BERs using both transmission schemes increase a certain amount. However, comparing the BER of OFDM transmission to that of single carrier transmission, it is still a significant reduction. Note

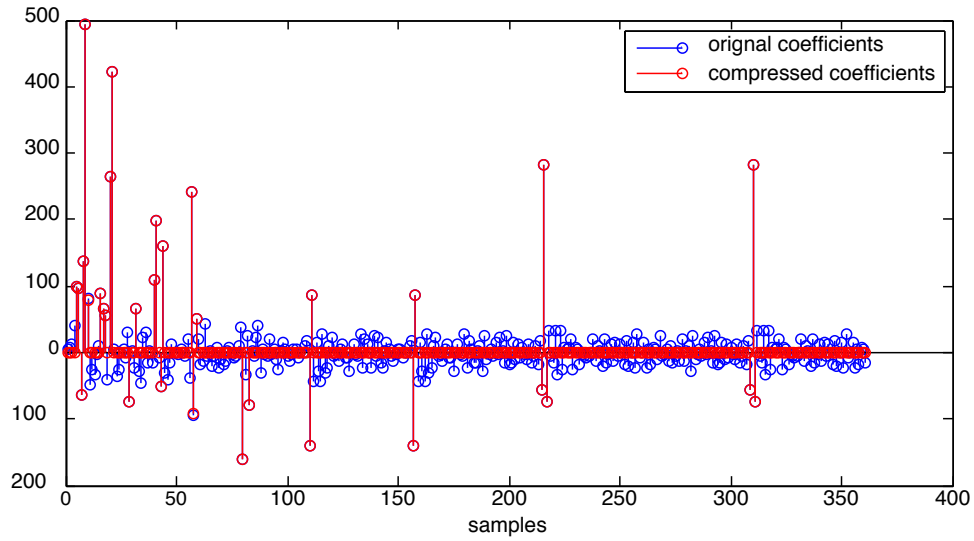


Figure 37: Compression of coefficients

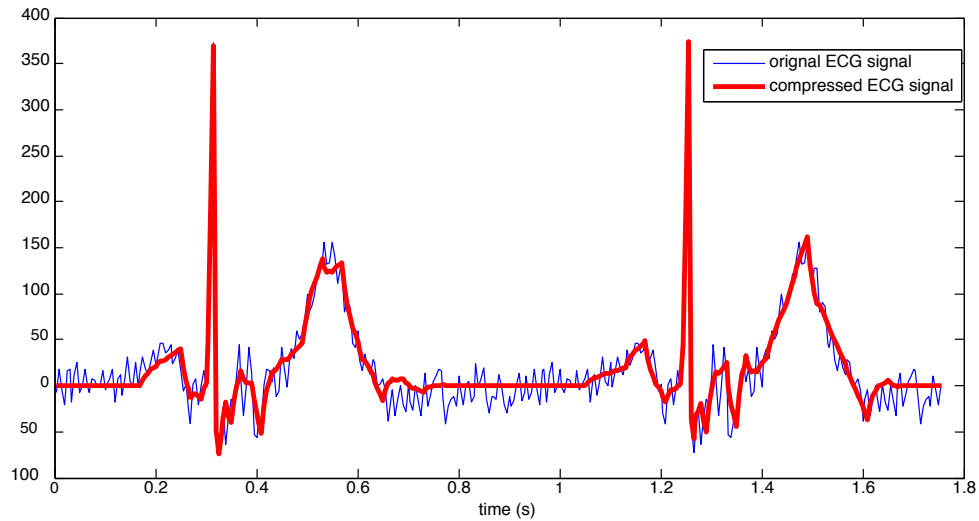


Figure 38: Original ECG signal vs. extracted signal using DWT

that without any coding or equalization process, the BER of single carrier transmission under Rayleigh channel with two paths is 0.35dB when SNR is 0dB, and the BER reduces to 0.23

Table 1: Summary of CR and PRD results using different thresholds (direct compression)

u	PRD	CR	P
0.02	0.0936	0.2457	0.9076
0.04	0.1194	0.3143	0.7557
0.06	0.1427	0.36	0.6986
0.08	0.1655	0.4065	0.6575
0.1	0.2178	0.4523	0.6600
0.2	0.2708	0.4981	0.6723

Table 2: Summary of CR and PRD results using different thresholds (DWT based compression)

v	PRD	CR	P
0.04	0.1551	0.4258	0.6248
0.05	0.1924	0.5618	0.5484
0.06	0.2312	0.6877	0.5220
0.08	0.269	0.7834	0.5243
0.10	0.3017	0.8388	0.5401
0.12	0.3234	0.864	0.5549
0.14	0.3414	0.8791	0.5689
0.16	0.3629	0.8942	0.5866

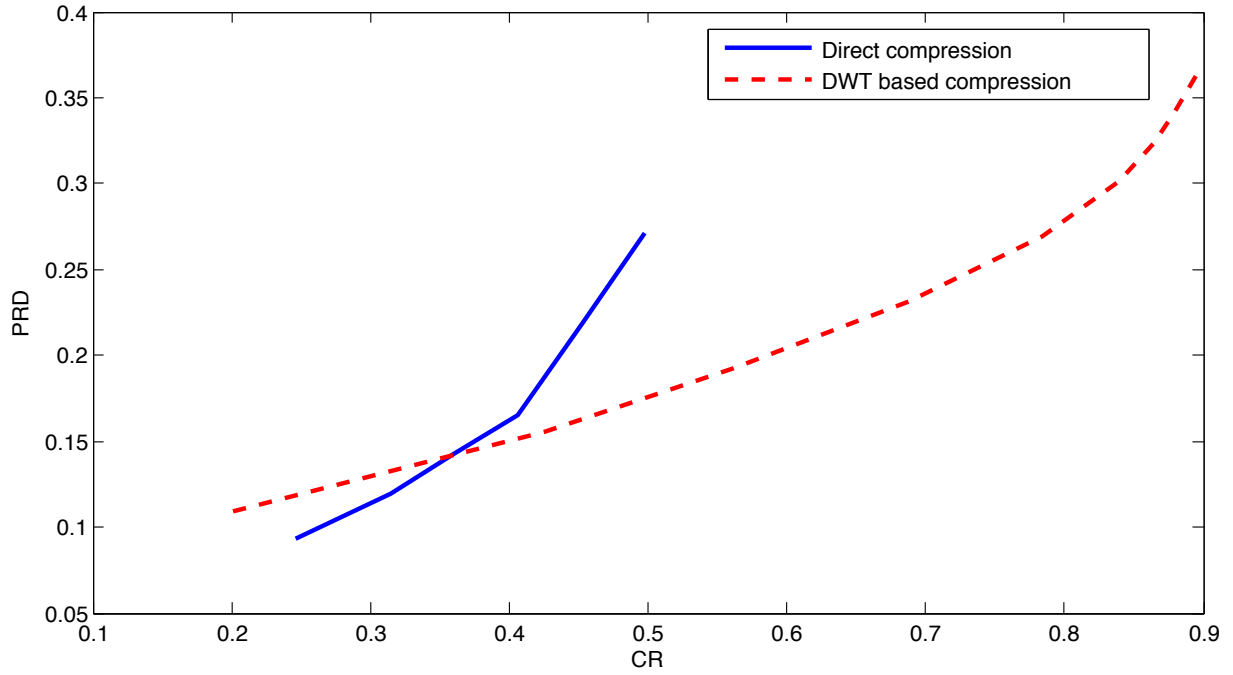
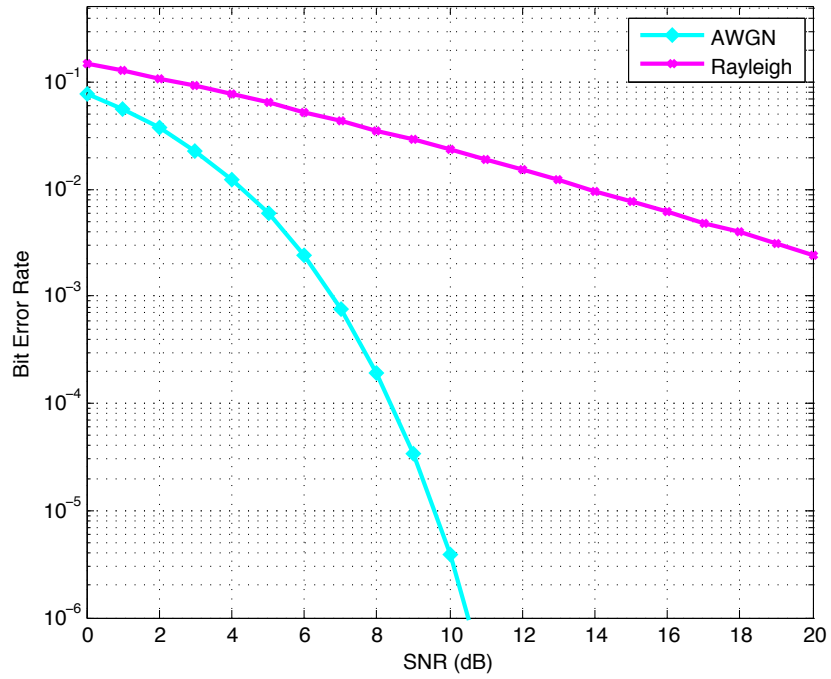


Figure 39: DWT based compression vs. direct compression

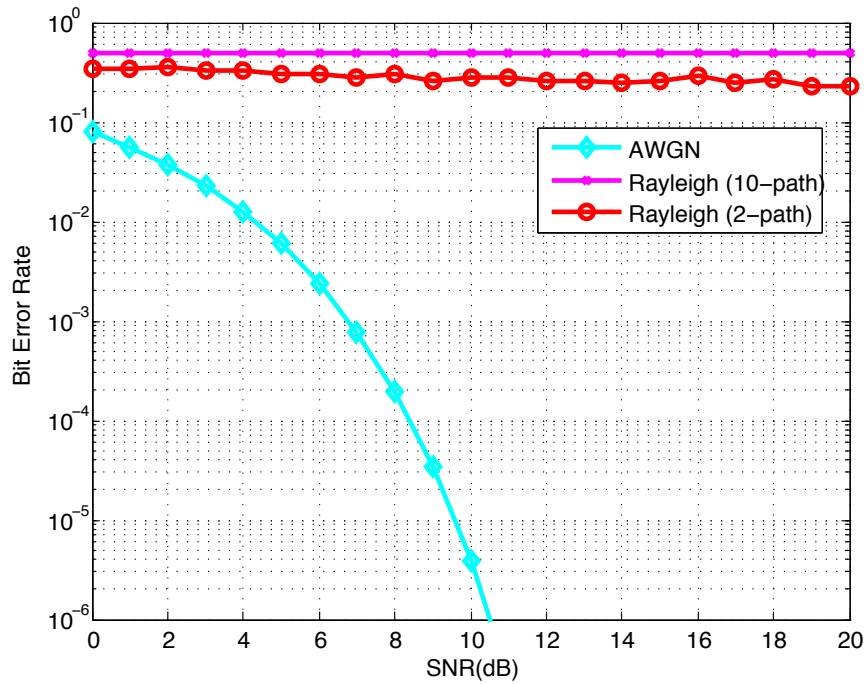
when SNR is 20dB. When the number of path increases to 10, the BER of single carrier transmission stays at 0.5 even if we increase the SNR, because multipath fading makes the received data totally scrambled and unrelated to the transmitted data.

Table 3: OFDM specifications

Parameter	Value
FFT length	64
Number of used subcarriers	52
FFT Sampling frequency	20MHz
Total symbol duration	4us
Subcarrier spacing	312.5kHz
Cyclic prefix duration	0.8us
Data symbol duration	3.2us



(a) BER for transmission using OFDM in AWGN and Rayleigh multipath channel (10-path)



(b) BER for signal carrier transmission in AWGN and Rayleigh multipath channel

Figure 40: Bit error rate analysis

5.0 CONCLUSION AND FUTURE WORK

In this thesis, an analog-to-digital conversion system using an ASDM and a TDC based on pulse shrinking delay lines that can be used in bio-monitoring systems is shown. The advantage of the proposed system is the lack of high frequency clocks and its low power consumption in data acquisition which makes it appropriate for bio-monitoring. Given that in wireless sensing, most of the power is consumed by the transmitter, an event-driven transmission scheme and DWT based compression scheme are proposed. The transmission bit error rate is reduced by using OFDM. The reconstruction of the original signal can be obtained by using PSWF reconstruction from non-uniform sampled signals and the results are satisfactory.

Although ASDM is simple and low-power, it requires intensive reconstruction procedures. It is desirable in power-critical applications where the lifetime of the self-powered sensors and transmitters is extremely important. Reconstruction should be carried out at some remote site where the required power is available. From this point of view, direct compression is more desirable than DWT based compression because the latter requires that time-decoding be done at the transmitter. An important goal is to find more efficient compression procedures that can be applied to time-encoded signal and can provide high compression while maintain low distortion to the signal.

BIBLIOGRAPHY

- [1] S. Bhardwaj, D. Lee, S. C. Mukhopadhyay, and W. Y. Chung, "Ubiquitous Healthcare Data Analysis and Monitoring Using Multiple Wireless Sensors for Elderly Person," *Sensors & Transducers*, Special Issue: Modern Sensing Technologies, Vol. 90, pp. 87-99, April 2008.
- [2] Wolpaw, J.R.; McFarland, D.J.; Vaughan, T.M., "Brain-computer interface research at the Wadsworth Center," *Rehabilitation Engineering, IEEE Transactions on*, Vol. 8, Issue 2, pp. 222-226, Jun 2000.
- [3] H. Wang, S. V. Rodriguez, C. Dirika, B. Jacoba, "Electromagnetic Interference and Digital Circuits: An Initial Study of Clock Networks," *Electromagnetics*, Vol. 26, Issue 1, pp. 73-86, Jan. 2006.
- [4] E. Allier, G. Sicard, L. Fesquet, and M. Renaudin, "A new class of asynchronous A/D converters based on time quantization," in *Proceedings of the Ninth International Symposium on Asynchronous Circuits and Systems*, Vancouver, Canada, May 12-15, 2003, pp. 196-205.
- [5] M, Miskowicz, "Efficiency of level-crossing sampling for bandlimited Gaussian random processes," *Proceedings of IEEE International Workshop on Factory Communication Systems WFCS*, 2006, pp. 137142.
- [6] Fredrik Hermansson, "Asynchronous analog-to-digital conversion with an application to frequency estimation," in *M.S. Thesis submitted to department of electrical engineering, KTH Royal Institute of Technology*, 2005.
- [7] Senay, S., Chaparro, L. F., Sun, M., and Sciabassi, R. J., "Time-frequency multiplexing for time encoded signals from brain computer interfaces," in *17th European Signal Processing Conference (EUSIPCO)*, Glasgow, Scotland, August 24-28. 2009, pp. 1186-1189.
- [8] D. Koscielnik and M. Miskowicz, "Asynchronous Sigma-Delta analog-to digital converter based on the charge pump integrator," *Analog Integrated Circuits and Signal Processing*, Volume 55, Number 3, 223-238, 2006.
- [9] S. Senay, L. F. Chaparro, M. Sun and R. J. Sciabassi, "Adaptive Level-Crossing Sampling and Reconstruction," *18th European Signal Processing Conference (EUSIPCO)*, 2010.

- [10] S. Y. Ng, "A continuous-time asynchronous Sigma Delta analog to digital converter for broadband wireless receiver with adaptive digital calibration technique," *Ph.D Dissertation*, Ohio State University, Electrical and Computer Engineering, 2009.
- [11] K. S. Lee, and K. Koh, "An Energy-Efficient Compression Algorithm for Wireless Communication," *International SoC Design Conference*, pp. 578-581, 2004.
- [12] K. Lakshmikumar, R. Hadaway, and M. Cope land, "Characterization and modeling of mismatch in MOS transistors for precision analog design," *IEEE J. Solid-State Circuits*, Vol. SC-21, pp. 1057-1066, Dec. 1986.
- [13] A. Oppenheim and R. Schafer, *Discrete-Time Signal Processing*. Prentice Hall. Third edition, 2010.
- [14] R. Prasad, *OFDM for wireless communications systems*. Artech House Publishers, 2004.
- [15] O. O. Khalifa, S. H. Harding, and A. A. Hashim, "Compression using Wavelet Transform," in *Signal Processing: An International Journal*, Vol. 2, Issue 5, 2010, pp. 17-27.
- [16] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image Coding using wavelet transform, in *IEEE Trans. Image Processing*, Vol. 1, No. 2, pp. 205-220, 1992.
- [17] D. Sinha, and A. H. Tewfik, "Low bit rate transparent audio compression using adapted wavelets," in *Signal Processing, IEEE Transactions on* , Vol. 41, Issue 12, 1993, pp. 3463-3479.
- [18] M. Abo-Zahhad, S. M. Ahmed, and A. Zakaria, "ECG Signal Compression Technique Based on Discrete Wavelet Transform and QRS-Complex Estimation," in *Signal Processing An International Journal (SPIJ)*, Vol., Issue 2, pp. 138-161, 2010
- [19] Senay, S., Akan, A., and Chaparro, L. F., "Sampling and reconstruction of non-bandlimited signals using Slepian functions," in *16th European Signal Processing Conference (EUSIPCO)*, Lausanne, Switzerland, August 25-29. 2008.
- [20] G. I. Bourdopoulos, A. Pnevmatikakis, V. Anastassopoulos, T. L. Deliyannis, *Delta-Sigma Modulators: Modeling, Design and Applications*. World Scientific Publishing Company , 2003.
- [21] Shahrzad Naraghi, "Time-based analog digital converters, " as *Ph.D dissertation submitted to The University of Michigan*, 2009.
- [22] E Risnen-Ruotsalainen, "Integrated Time-to-Digital Converters Based on Interpolation," *Analog Integrated Circuits and Signal Processing - Special issue: selected articles from the 1996 NORCHIP seminar*, vol. 15, pp. 49-57, May 1997.
- [23] M. Simpson, C. Britton, A. Wintenberg, and G. Young, "An integrated CMOS time interval measurement system with sub-nanosecond resolution for the WA-98 calorimeter," *IEEE J. Solid-State Circuits*, pp. 198-205, Feb. 1997.

- [24] D. Tse, *Fundamentals of Wireless Communication*, Cambridge University Press, 2005.
- [25] A. Lazar and L. Toth, "Perfect recovery and sensitivity analysis of time encoded bandlimited signal," *IEEE Trans. on Circuits and Sys.*, pp. 2060-2073, Oct. 2004.
- [26] Paul J.M. Havinga, Gerard J.M. Smit, "Energy-efficient wireless networking for multimedia applications," in *Wireless Communications and Mobile Computing*, Vol. 1, pp. 165-184, 2001.