Low Complexity Ultra Wide-band Receivers

Master Thesis

By

Matigakis Manolis

Submitted to the Department of Electronic & Computer Engineering in partial fulfilment of the requirements for the Master Degree.

Technical University of Crete

Advisor: Professor Liavas Athanasios

Co-advisor: Professor Sidiropoulos Nikolaos

Co-advisor: Assistant Professor Karystinos George

April 2011

This page was left blank intentionally.

Ευχαριστίες

θα ήθελα να ευχαριστήσω των επιβλέποντα καθηγητή μου, Καθηγητή Λιάβα Αθανάσιο, για την πολύτιμη υποστήριξη του και για την υπομονή που επέδειξε απέναντι μου. Επίσης θέλω να ευχαριστήσω τους καθηγητές κυρίους Λιάβα Αθανάσιο και Σιδηρόπουλο Νικόλαο, και τους επίκουρους καθηγητές Καρυστινό Γεώργιο και Μπλέτσα Άγγελο για τις πολύτιμες γνώσεις που μου προσέφεραν όλα αυτά τα χρόνια.

Contents

List of Figures

Abstract

During the last decade, there has been a growing interest, in both academia and industry, for ultra wideband (UWB) communications. A UWB system is defined as any radio system that has a 10 dB bandwidth larger than 20% of its center frequency, or has a 10 dB bandwidth equal to or greater than 500 MHz. In this thesis, we survey some low complexity UWB receivers. In particular, two design philosophies are studied: i) digital coherent receivers with 1 bit of analog to digital conversion resolution and ii) Noncoherent receivers. We also do a simulation study to compare their performance in terms of BER vs SNR curves.

Chapter 1

Introduction

During the last decade, there has been a growing interest in both academia and industry for ultra wideband (UWB) communications. A UWB system is defined as any radio system that has a 10 dB bandwidth larger than 20% of its center frequency, or has a 10 dB bandwidth equal to or greater than 500 MHz. Recently, the Federal Communications Commission has approved the unlicensed operation of UWB radios in frequencies from 3.1 GHz to 10.6 GHz, a 7.5 GHz wide band. In order to coexist with existing narrowband systems, such as 802.11a/b/g radios, severe limitation have been legislated for the transmit power of UWB devices. Figure 1.1 shows the spectral mask for UWB systems operating in an indoor environment.

UWB technology has the potential to become the new enabler for low-cost low-power wireless networks. Some of its features are:

1. high data rates,

2. availability of low-cost transceivers;

3. large instantaneous bandwidth enables fine time resolution for network time distribution and precision location capability;

4. robustness in dense multipath environments by exploiting many resolvable paths;

5. low power spectral density allows coexistence with existing users and has a Low Probability of Intercept (LPI);

6. data rate may be traded for power spectral density and multipath performance.

Potential application areas for UWB systems include wireless personal area networks (WPANs), high-speed mobile local area networks, wireless sensor networks, secure communications etc. [1, §1.2.7]. It is expected that many conventional principles and approaches used for short-range wireless communications will be reevaluated and a new industrial sector in short-range (e.g., 10 m) wireless communications with high data rate (e.g., 400 Mbps) will be formed. UWB technology has already been adopted by some industrial standards such as IEEE 802.15.3a (high data rate) and IEEE 802.15.4a (very low data rate).

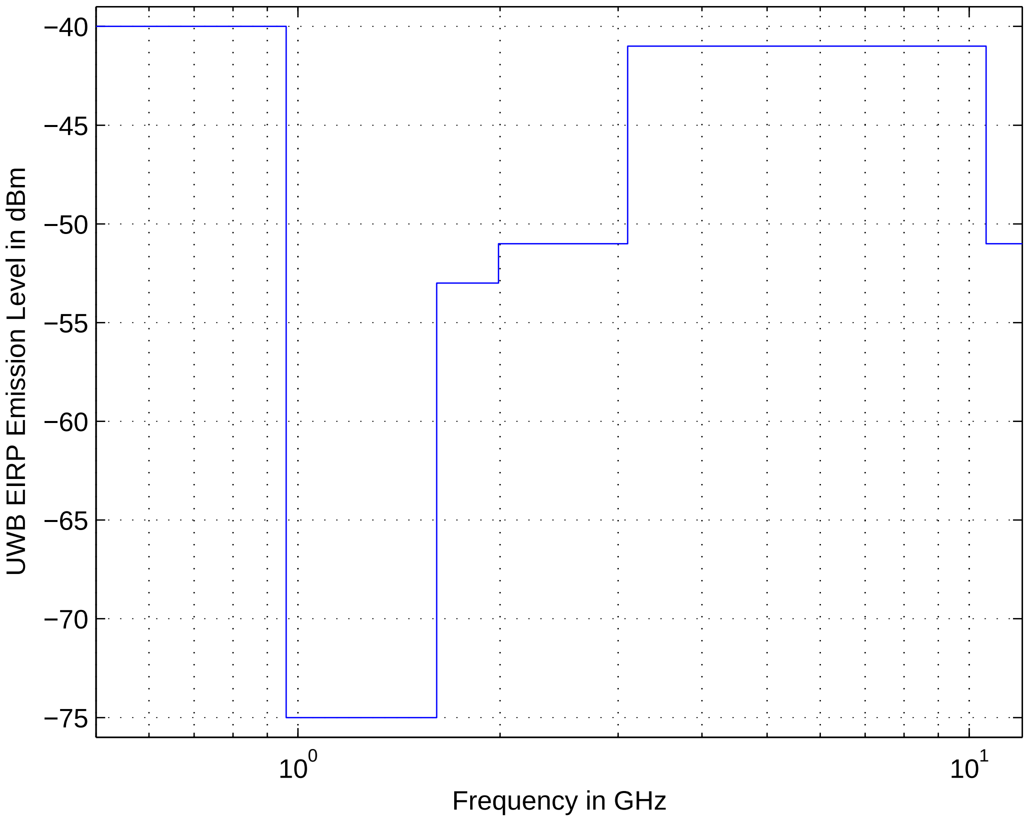

Figure 1.1: UWB Emission Limit for Indoor Hand-held Systems. The main reason for the large spectral gap between 960 MHz and 1.6 GHz has been enforced is to protect GPS receivers from interference.

1.1 UWB Modulation Types

UWB systems are divided into two categories depending on the way they map information symbols to waveforms. The first category uses orthogonal frequency division modulation (OFDM) to convey information and is known as Multicarrier Ultra Wideband (MC-UWB). In the second, category which is known as Impulse Radio Ultra Wideband (IR-UWB), each information symbol is mapped to a train of sparsely spaced narrow pulses. In the next two sections we will briefly describe these two types of UWB systems. However, the main focus of this thesis is on IR-UWB systems.

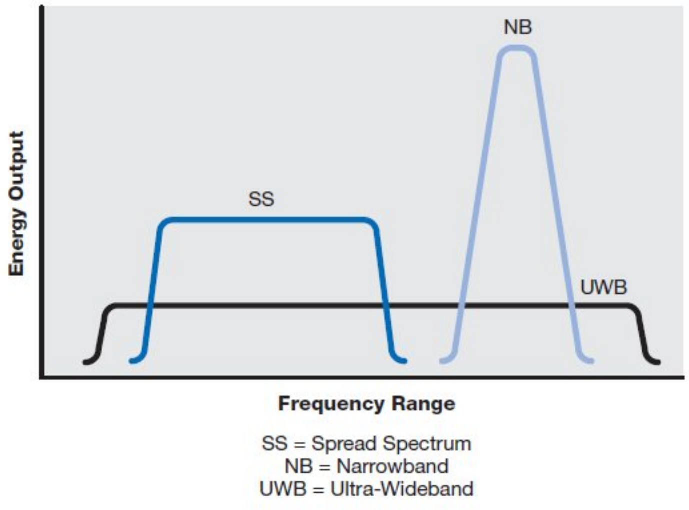

Figure 1.2: Qualitative comparison between narrowband, spread spectrum and UWB systems. IR-UWB can be considered as a form of spread spectrum scheme that uses narrow pulses which have most of their energy concentrated in a small time interval instead of using pseudo-random binary sequences to modulate a PAM signal where the energy is spread over a large time interval. The bandwidth used in UWB systems, however, is in general much greater than in typical DSSS systems in order to reduce the interference caused to other systems.

1.1.1 MC-UWB

In MC-UWB, information is conveyed in multiple closely spaced carriers that do not interfere with each other. This modulation is known as orthogonal frequency division mod-

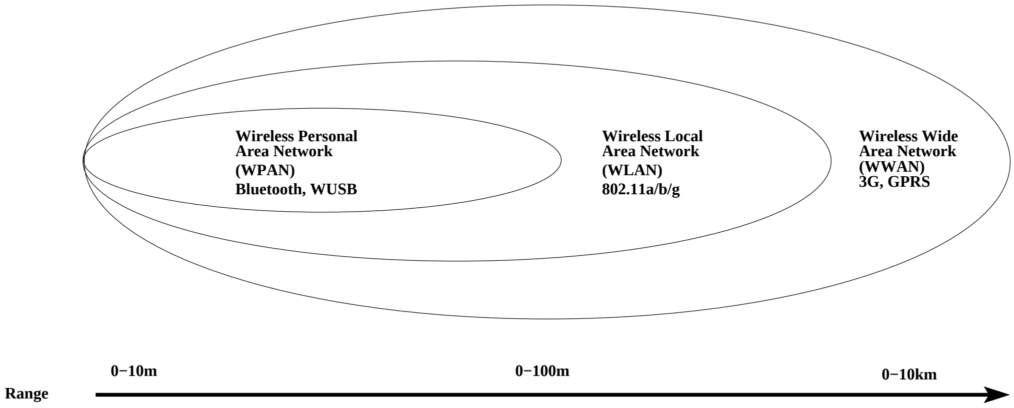

Figure 1.3: UWB systems belong to the wireless personal area network technologies since their range is limited to less than 10 meters.

ulation (OFDM) in the literature. OFDM is suitable for high data rate systems and it can be fully implemented in digital hardware using fast analog to digital (ADC) and digital to analog (DAC) converters. A baseband OFDM signal can be represented by

$$s(t) = \sum_{k=0}^{N-1} d_k e^{j2\pi f_k t}, \; 0 \le t \le NT_s \tag{1.1}$$

where N is the number of subcarriers per OFDM block, d_k is the complex symbol modulating the k-th subcarrier, $f_k = f_0 + k\Delta f$ is the subcarrier frequency and Δf is the frequency spacing between two adjacent subcarriers. It is necessary that $NT_s\Delta f = 1$ so that the subcarriers do not interfere with each other.

When $f_0 = 0$, the samples of $s(t)$ are given by,

$$s(nT_s) = \sum_{k=0}^{N-1} d_k e^{j2\pi \frac{nk}{N}}. \tag{1.2}$$

Therefore, we can use the inverse DFT of the complex symbols to retrieve the samples of the transmitted signal. This is one of main reasons that OFDM has become so popular since the inverse DFT can be computed using the FFT algorithm which has low implementation complexity.

Another important feature of OFDM is that, after preprocessing the received signal in the receiver, the detection of the symbols d_k is decomposed into independent detection problems. Specifically, the demodulated signal can be expressed as

$$r_k = h_k d_k + n_k \tag{1.3}$$

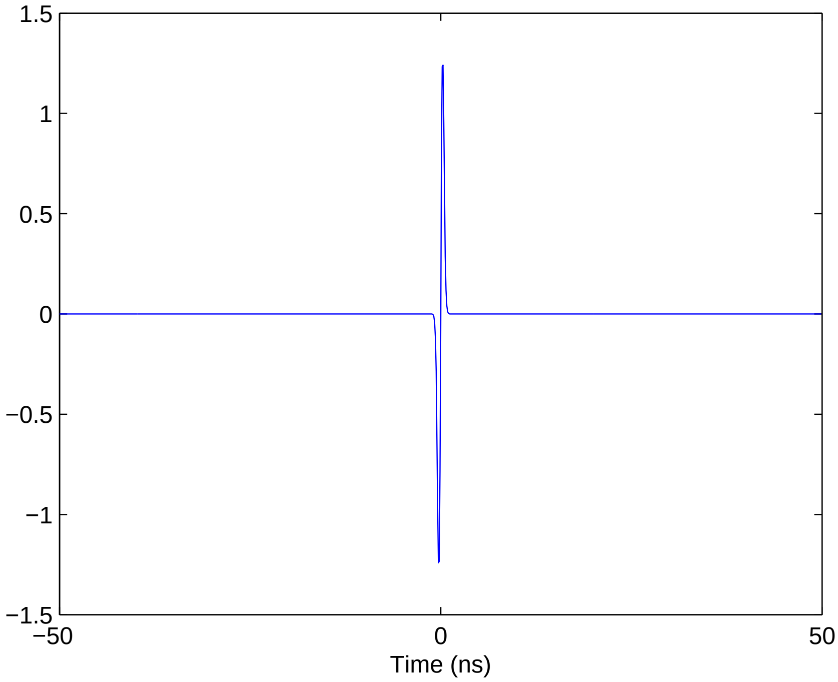

Figure 1.4: A first-order Gaussian pulse. This pulse is defined as $p(t) = te^{-ct^2}$ where $c > 0$ is a constant that determines the pulse duration and thus its bandwidth.

where $h_k \in \mathcal{C}$ is the channel frequency response at the k-th subcarrier frequency and $n_k \in \mathcal{C}, k = 0, \ldots, N-1$, are i.i.d random variables that represent the channel noise effect. OFDM can thus deal with large delay spreads without the need of a complicated channel equalizer which is important for UWB systems since as we will see UWB channels can be very long.

1.1.2 IR-UWB

Impulse radio doesn't use a modulated sinusoidal carrier to convey information. Instead, the transmit signal consists of a series of sparsely spaced baseband pulses. Typically, these pulses are extremely short, commonly in the subnanosecond range. Thus, the bandwidth is of the order of gigahertz. Each information symbol is conveyed by a series of pulses. The time interval where a single pulse is transmitted is called a frame. Information can be carried both in the amplitude and timing of the pulses depending on the modulation

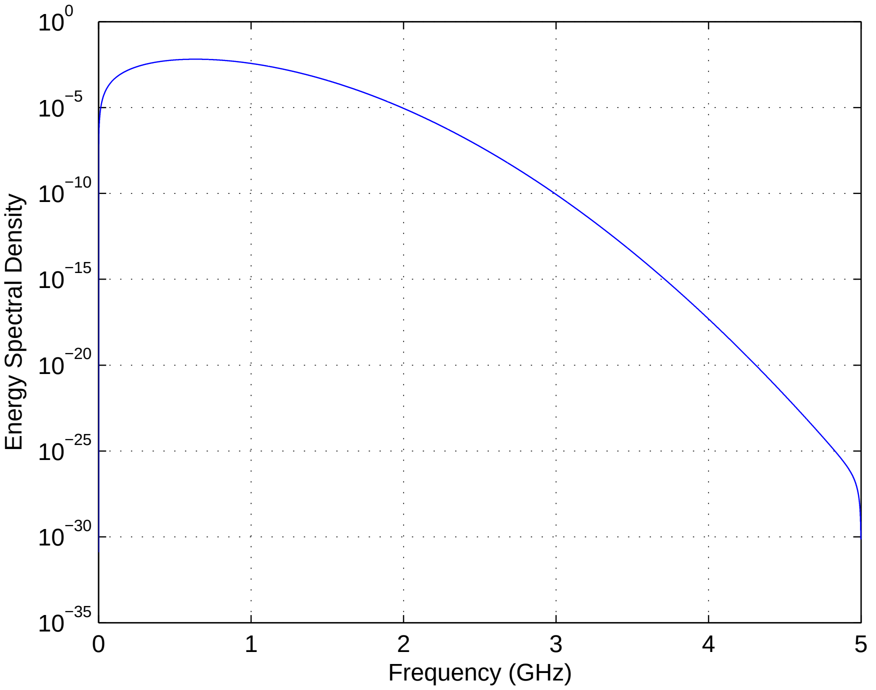

Figure 1.5: Energy spectral density of the first-order Gaussian pulse.

used. Additionally, each pulse is usually delayed using a pseudo random time hopping sequence. This is done to make the power spectral density smoother and to allow multiple users to access the channel simultaneously. The transmitted signal for a single symbol can be expressed as

$$b(t) = \sum_{j=0}^{N_f} p(t - jT_f - c_jT_c) \tag{1.4}$$

where $p(t)$ is called a monopulse, N_f is the number of pulses per symbol, T_f is the frame time and c_j is the pseudorandom spreading code which is unique for each user. Figure (1.4) shows a typical monopulse used in IR-UWB systems. The transmitted signal depends on the modulation type. Some of the modulation types used in IR-UWB are listed bellow.

Binary Pulse Position Modulation (PPM): In binary PPM, information is encoded in the position of a pulse relatively to some time instant. The signal model for this

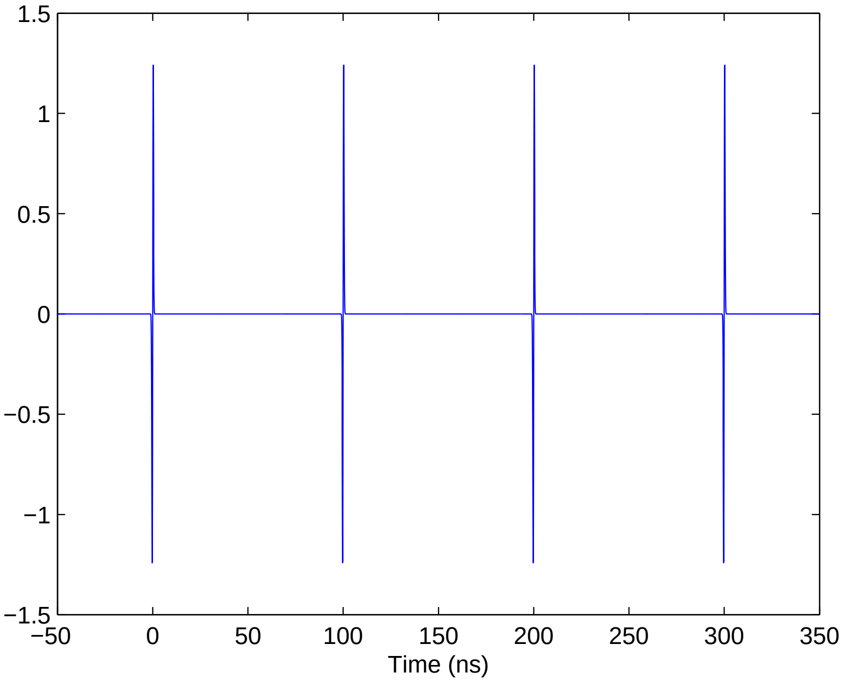

Figure 1.6: In this figure there is no pseudo-random time hoping code used. Using pulse trains like this for each symbol instead of one single pulse increases the energy per symbol but also allows sharing the channel by multiple users.

modulation is

$$s(t) = \sum_i b(t - iN_f T_f - \alpha_i \Delta) \tag{1.5}$$

where $\alpha_i \in \{0, 1\}$ are the information symbols and Δ is the delay used to differentiate the two symbols (See figure (1.7)).

Binary Pulse Amplitude Modulation (BPAM): In BPAM, information is encoded in the polarity of the transmitted pulses. The signal model for this modulation is

$$s(t) = \sum_i \alpha_i b(t - iN_f T_f) \tag{1.6}$$

where $\alpha_i \in \{\pm 1\}$.

On-Off Keying (OOK): In OOK modulation, information is encoded in the presence or

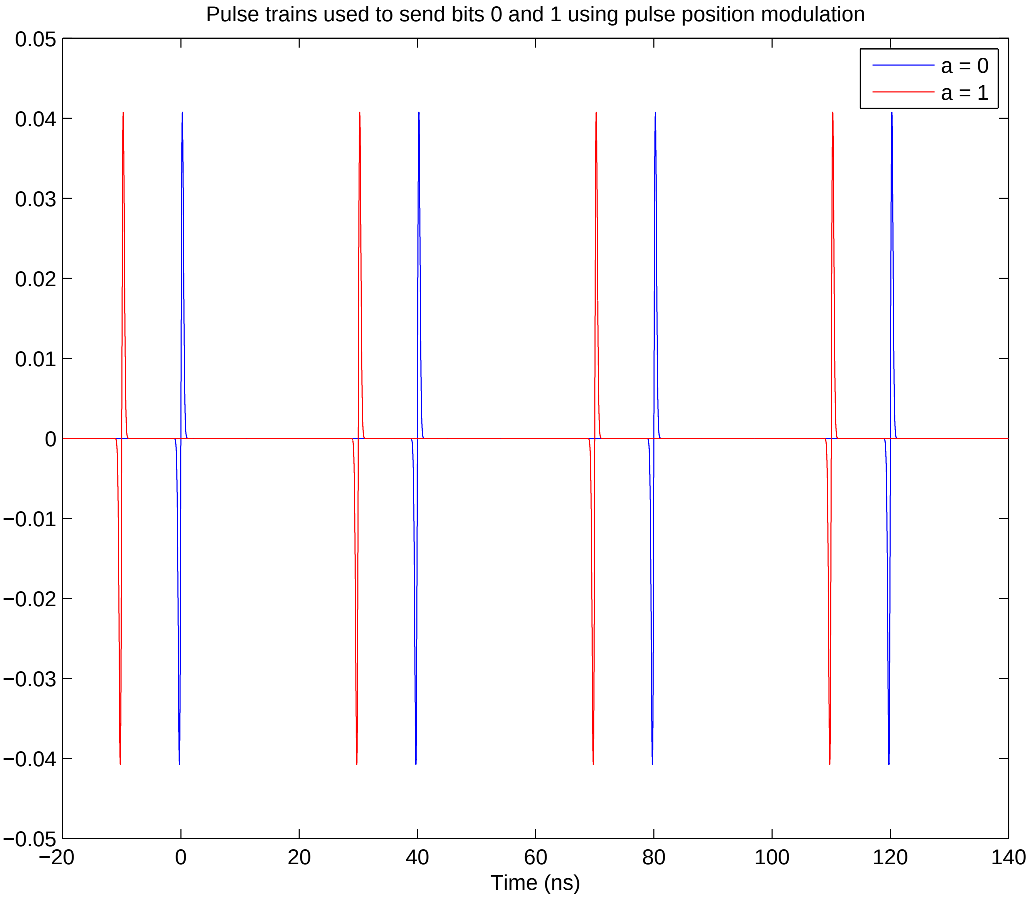

Figure 1.7: Binary Pulse Position Modulated pulse trains.

absence of a pulse. The signal model for this modulation is

$$s(t) = \sum_i \alpha_i b(t - iN_f T_f) \tag{1.7}$$

where $\alpha_i \in \{0, 1\}$.

1.1.3 Pros and Cons of Impulse Versus Multicarrier

MC-UWB systems can avoid causing or receiving narrowband interference by avoiding using certain carrier frequencies where narrowband systems operate. On the other hand,

MC-UWB	IR-UWB
Very fine control on the interference imposed	No control over interference
Complicated transmitter	Simple transmitter
Complicated Receiver	Simple receiver

Table 1.1: Comparison of MC-UWB and IR-UWB.

IR-UWB systems cannot shape their power spectral density to have arbitrary power allocation. For both modulations, diversity techniques can be used to reduce the impact of interference. The issue of interference handling to or from UWB systems remains one of the most important issues facing UWB systems.

Impulse radio transmitters are very simple devices unlike OFDM transmitters. They don't require any mixers since they transmit a baseband signal. So, from the transmitter side it is posible to achieve high throughput with low power and cost. This is a very appealing feature, especially for wireless sensor networks, where sensors can sometimes be transmit-only devices. On the other hand, impulse radio receivers can be quite complicated. However, researchers have come up with a collection of reduced complexity receivers. Table (1.1) summarizes the differences between multicarrier and impulse radio UWB systems.

1.2 UWB Signal Propagation and Channel Modeling

For IR-UWB systems the channel model that is commonly used represents the effect of the channel by a series of Dirac's pulses with random amplitudes and arrival times. The intuition is that each pulse represents one physical propagation path. So, the channel impulse response is expressed as

$$h(t) = \sum_{l=1}^{L} \gamma_l \delta(t - \tau_l), \tag{1.8}$$

where $\gamma_l \in \mathbb{R}$ and $\tau_l \in \mathbb{R}$ are the gain and delay of the l-th path, respectively, L is the total number of resolvable propagation paths, and $\delta(\cdot)$ is the Dirac delta function. Note that besides having a magnitude each coefficient γ_l also has a sign. The channel output can thus be written as

$$r(t) = s(t) * h(t) = \sum_{l=1}^{L} \gamma_l s(t - \tau_l) + w(t). \tag{1.9}$$

where $w(t)$ is additive white Gaussian noise. The AWGN assumption is common in the analysis of communication systems and it is a very good approximation for narrowband systems. Although the AWGN assumption is still used in most of the papers that study UWB systems, in practice, noise is colored due to narrowband interference and imperfections of the analog parts of the receiver. Figure (1.8) shows samples of UWB channel impulse responses.

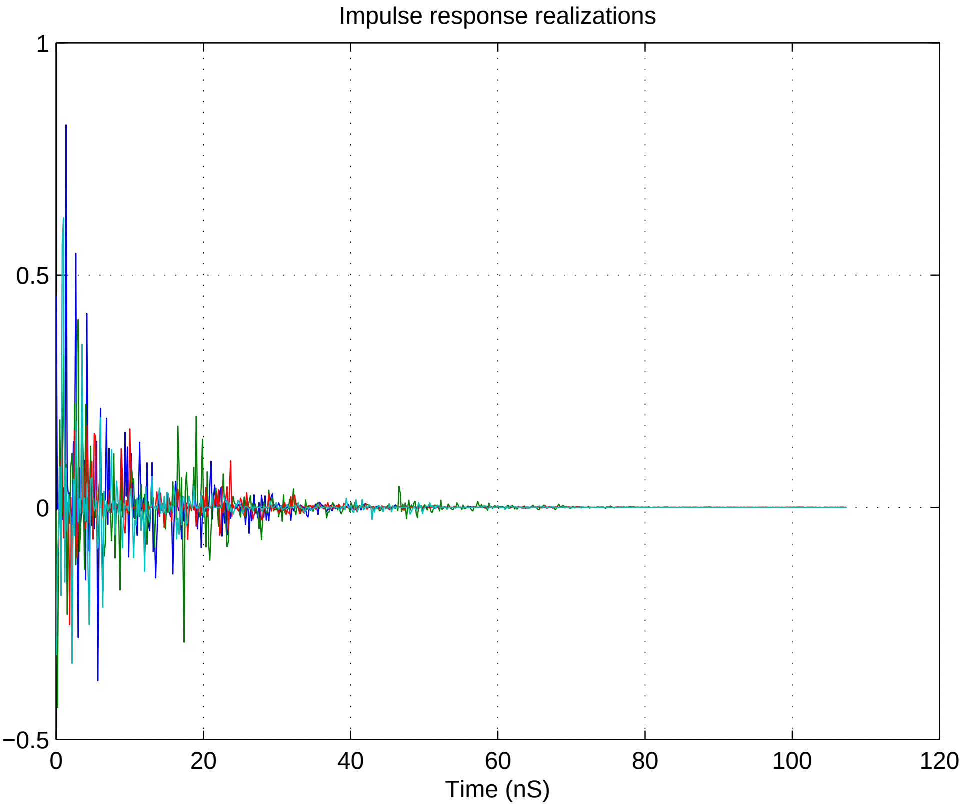

Figure 1.8: Random realizations of UWB channel impulse response in indoor environment.

1.3 UWB Receivers

UWB receivers can be divided in two categories. The first category, coherent receivers, requires channel knowledge at the receiver. This means that channel estimation has to be performed prior to demodulation. For instance, training based schemes for channel

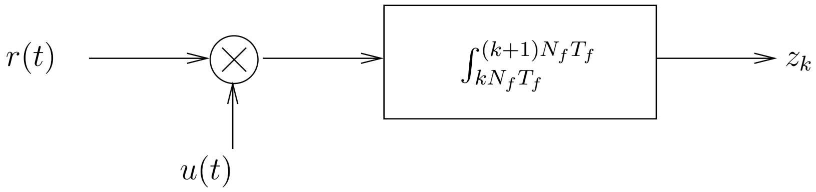

Figure 1.9: Matched filter receiver.

estimation, like the ones studied in [3], can be used prior to demodulation. The second category are the noncoherent receivers. These receivers don't require channel knowledge thus reducing the implementation complexity. On the other hand, as we are going to see, their performance is considerably worse than that of coherent receivers.

1.3.1 Coherent Receivers

Two kinds of coherent receivers have been studied in the UWB literature. The first is the matched filter receiver. Although the matched filter is optimal in terms of BER performance (assuming AWGN noise) its implementation complexity is extremely high. For instance, perfect knowledge of the channel impulse response is required for demodulation. Additionally, a device that generates arbitrary waveforms is needed. A receiver that can achieve a tradeoff between complexity and performance is the Rake receiver. Although optimal coherent receivers are complicated to implement, as we shall see in the next chapter, suboptimal receivers with good performance do exist. We briefly describe coherent receivers in the next two subsections.

Matched Filter Receiver

We will describe the MF receiver in the case of binary pulse position modulation. The transmitted signal is expressed as

$$s(t) = \sum_i b(t - iN_fT_f - a_i\Delta) \tag{1.10}$$

where $b(t)$ is an UWB waveform that is either delayed by Δ or not depending on the value of symbol $a_i \in \{0, 1\}$. This waveform is composed of a series of N_f narrow pulses $p(t)$, called monocycles, that are spaced T_f seconds apart as described by equation

$$b(t) = \sum_{j=0}^{N_f} p(t - jT_f - c_j T_c). \tag{1.11}$$

This makes the total duration of $b(t)$ equal to NT_f. The individual pulses are further randomly displaced from their normal position by $c_j T_c$ where $c_j \in \{0, N_{TH} - 1\}$ is a pseudo-random time hopping code used to smooth the final power spectral density of the transmitted signal and to allow multi-user channel access. The time shift Δ imposed by the modulated symbols a_i on the waveform $b(t)$ is an important factor in system performance but we will not dive into that problem here.

One assumption made in this model is that the last monocycle in a symbol does not overlap with the first monocycle of the next symbol. This is done to avoid ISI so that symbol by symbol detection is optimal.

The received signal is expressed as

$$r(t) = \sum_{l=1}^{L} \gamma_l s(t - \tau_l) + w(t) \tag{1.12}$$

where L is the number of channel paths γ_l is the l-th path strength, τ_l is the l-th path delay and $w(t)$ is the thermal noise which is assumed to be white Gaussian. This model assumes that the channel is static or at least varies very slowly so that it is almost constant for the duration of several data symbols.

The decision statistic for the k-th data symbol is calculated by correlating the received waveform with a template waveform $v(t)$

$$z_k = \int_{kN_f T_f}^{(k+1)N_f T_f} r(t) v(t - kN_f T_f) dt. \tag{1.13}$$

The optimal template waveform $v_{opt}(t)$ is [1]

$$v_{opt}(t) = \sum_{l=1}^{L_c} \gamma_l (b(t - \tau_l) - b(t - \tau_l - \Delta)) \tag{1.14}$$

and the corresponding decision rule is

$$\hat{\alpha}_k = \mathrm{sign(z_k)}. \tag{1.15}$$

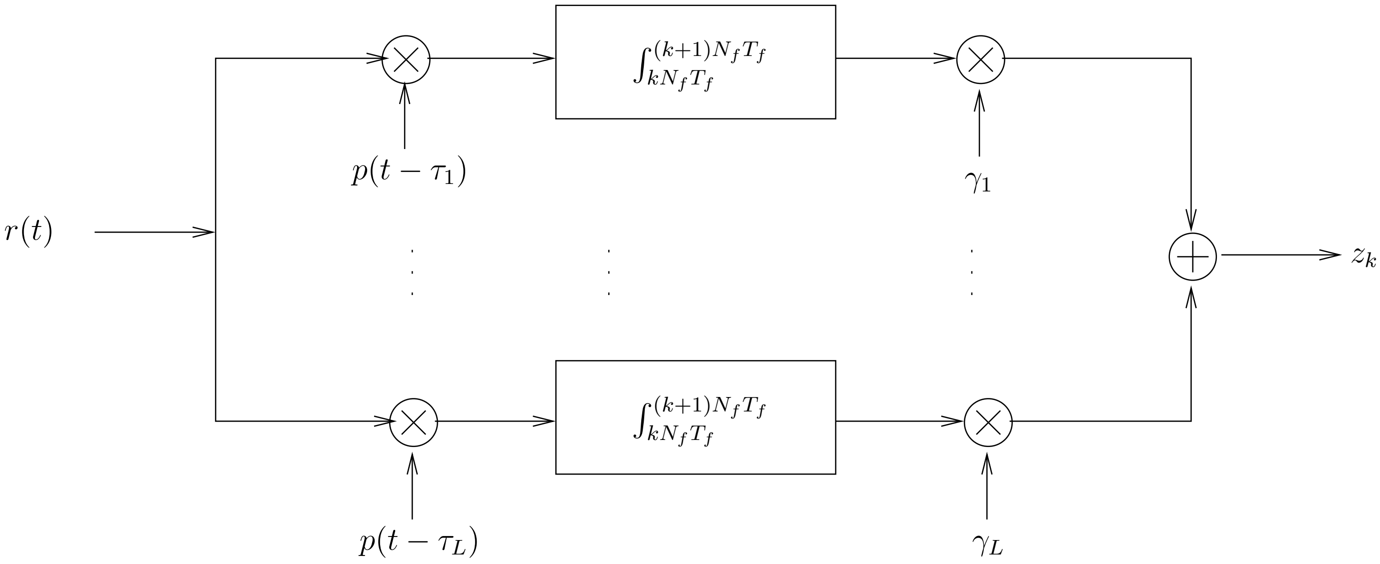
$r(t)$
$p(t-\tau_1)$
$\int_{kN_fT_f}^{(k+1)N_fT_f}$
γ_1
z_k
$\int_{kN_fT_f}^{(k+1)N_fT_f}$
$p(t-\tau_L)$
γ_L

Figure 1.10: Rake receiver with L-fingers.

Rake Receiver

In the matched filter receiver, all channel paths have to be known in order to be able to generate the optimal template waveform $v(t)$. A simpler receiver is obtained by estimating only the strongest path. In this case the decision statistic is given by

$$z_k = \int_{kN_fT_f}^{(k+1)N_fT_f} r(t)v_{1-finger}(t-kN_fT_f)dt \tag{1.16}$$

where,

$$v_{1-finger}(t) = \gamma_1(b(t-\tau_1) - b(t-\tau_1-\Delta)). \tag{1.17}$$

In the last equation, we assumed that the first path is the strongest. This is a Rake receiver with 1 finger. In general, we can use L fingers by estimating the L strongest paths. In this case, the template becomes

$$v_{K-finger}(t) = \sum_{l=1}^{L} \gamma_l(b(t-\tau_l) - b(t-\tau_l-\Delta)). \tag{1.18}$$

The optimal decision rule is the same as before.

1.3.2 Noncoherent Receivers

Noncoherent receivers are much simpler structures since they require no channel knowledge. Two types of noncoherent receivers are the main focus of research. The first is transmit reference receiver which is also known as autocorrelation receiver. The second is the energy detection receiver.

Transmit Reference Receiver

For the transmit reference receiver to work the transmitter duplicates each monopulse sent. The first monopulse remains unmodulated while the second is modulated by the data symbol send. Assuming binary PPM, the transmited signal is expressed as

$$s(t) = \sum_{i} b_{TR}(t - i2NT_f) + b_{TR}(t - i2N_fT_f - a_i\Delta) \tag{1.19}$$

where $b_{TR}(t)$ has the form

$$b_{TR}(t) = \sum_{j=0}^{N_f} p(t - j2T_f - c_jT_c). \tag{1.20}$$

Notice that the modulated monopulses are now $2T_f$ apart and an unmodulated monopulse precides each modulated. The transmit reference receiver approximates the matched filter receiver statistic by using the unmodulated monopulses as a template waveform. This is done using an analog delay line to delay the received waveform so that the unmodulated monopulse that precides each modulated monopulse reaches the multiplier synchronously with the later. We will see the transmit reference receiver in more detail in Chapter 3.

Energy Detection Receiver

The energy detector receiver is the simplest of all UWB receivers. As the name implies, it uses the energy of the received signal to detect the received symbols. The energy detector receiver cannot be used for detecting PAM modulation since the sign of the received signal cannot be determined from the received energy. Thus, energy detector receivers are used explicitly with PPM and OOK modulations. We will see the energy detector receiver in more detail in Chapter 3.

Chapter 2

Monobit Digital Receivers

2.1 Introduction

A fully digital UWB receiver must be able to sample the received signal at least at the Nyquist rate, which is twice the signal bandwidth. For UWB systems this is at least 500 MHz though, for some modulation schemes, bandwidths of several Gigahertz are required. Figure 2.1 shows a graphic representation of the most commonly used ADC converter technologies and their corresponding applications. It is evident from this figure that only pipelined ADCs are appropriate for digital implementation of UWB receivers. But even pipelined ADCs today cannot sample beyond 1 or 2 GSPS. Moreover, ADCs working at such high rate are in general very expensive and power demanding. This is a huge problem as far as implementing low-cost and low-power UWB systems such as wireless sensors. This has inspired a great amount of work around ways to overcome the ADC limitations. There are mainly three approaches that have been studied so far:

Time Interleaved Sampling: One approach [2] is to use a bank of ADCs and clock them in a manner as to take interleaved samples of the signals in order to increase the effective sampling rate of the system. This technique, however, is extremely sensitive to clock jitter and mismatches in the ADC characteristics. It also requires many ADCs thus increasing area, power consumption, cost, and design complexity. A discussion of time interleaved A/D converters can be found in [17].

Compressed Sensing: Another approach focuses on applying ideas from compressed sensing [13] to UWB receivers [7, 9, 10, 11, 12]. These receivers relies on the assumption that the received signal is sparse in some basis, i.e., only a few of the coefficients required in the signal expansion are nonzero. Under this assumption, it can be shown that the sampling rate requirements can be relaxed significantly below the Nyquist rate. From a practical point of view, compressed sensing receivers require quite a lot of analog circuitry. In particular, a bank of analog multipliers and waveform generators is needed [18, 19]. Furthermore, it is not clear that the sampling rate can be reduced enough for contemporary analog-to-digital converter technology to be feasible. If not, the ideas of time interleaved sampling and compressing could be combined but that would again lead to expensive, complex, and power hungry systems.

Low Resolution Receivers: In [14], a digital receiver based on very low resolution analog-to-digital conversion was proposed. In fact, only receivers that used a single bit per sample were considered. Receivers with more than one quantization levels were studied in [15]. In [16], a monobit digital receiver for transmitted reference was analysed. In order to remedy for the loss in performance due to quantization, the sampling rate is increased to many times the Nyquist rate (up to 12 times). If serial to parallel conversion at such high rates is possible then the system clock speed won't have to be that high.

In this chapter, we shall study 1-bit receivers or monobit receivers as they are often called in the literature.

2.2 Full-Resolution Digital Matched Filter Receiver

Before we start our study of monobit receivers, we briefly present full-resolution receivers. Two modulation types are

- binary pulse amplitude modulation,
- binary pulse position modulation.

The reader may refer to the introduction for a description of these modulations.

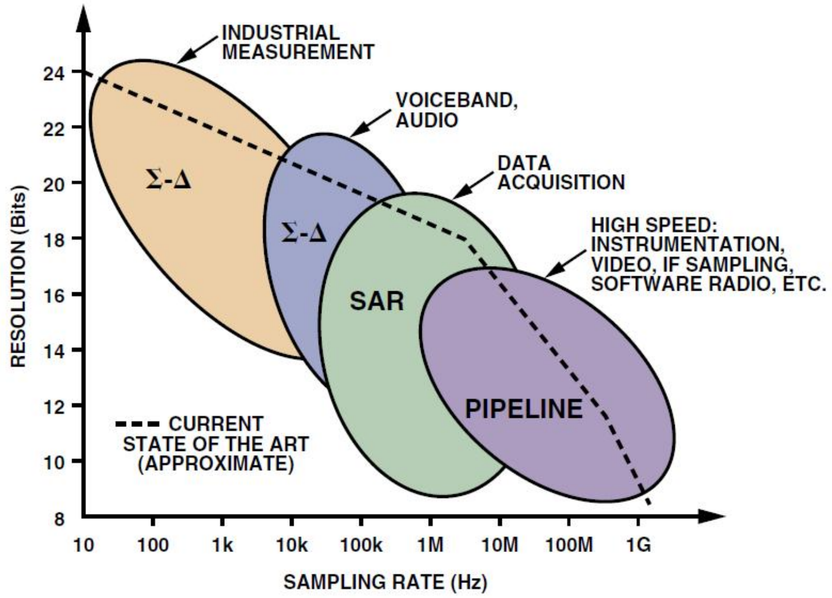

Figure 2.1: ADC architectures, applications, resolution, and sampling rates. $\Sigma-\Delta$ converters are capable of very high resolution and are capable to trade resolution for speed. Even so, the maximum sampling rate of $\Sigma-\Delta$ converters is limited to a few MSPS (Mega Samples Per Second). SAR (Successive Approximation Register) converters can achieve higher sampling speeds than $\Sigma-\Delta$ converters at resolutions of 12 to 16 bits. Still the maximum sampling rate can't exceed a few hundred MSPS. The fastest analog-to-digital converters are pipelined converters. These are capable of sampling rates up to a couple of GSPS (Giga Samples Per Second) but at much reduced resolution. Furthermore, the higher the sampling rate of a converter the more power it consumes.

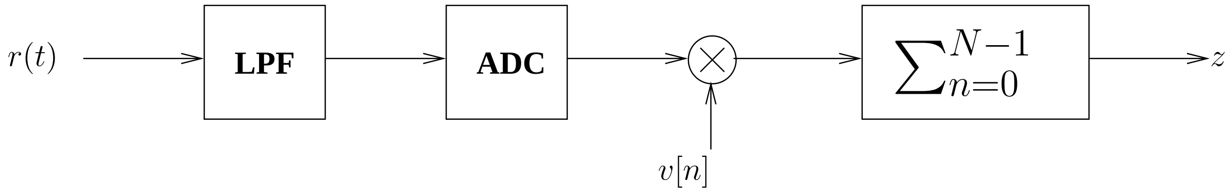

Figure 2.2: Full-Resolution Digital Matched filter receiver.

2.2.1 Binary PAM Receiver

As we have seen earlier, the transmitter sends the signal

$$s_{pam}(t) = \sum_i \alpha_i b(t - iN_f T_f) \tag{2.1}$$

where $\alpha_i \in \{+1, -1\}$ is the transmitted information symbol and $b(t)$ is the pulse train used to convey each symbol as defined in (1.4). For simplicity, we will consider only one symbol thus, the signal that the transmitter sends is

$$s_{pam}(t) = \alpha b(t). \tag{2.2}$$

Note that we ignore any ISI. This is possible if T_f is chosen sufficiently large. The received signal is expressed as

$$r_{pam}(t) = \sum_{l=1}^{L} \gamma_l s_{pam}(t - \tau_l) + w(t) = \alpha \sum_{l=1}^{L} \gamma_l b(t - \tau_l) + w(t) \tag{2.3}$$

where L is the number of channel paths, γ_l and τ_l are the attenuation and delay of the l-th path, respectively, and $w(t)$ is the thermal noise at the receiver which we assume to be white Gaussian with variance $N_0/2$. The receiver low-pass filters the received signal with an ideal low-pass filter of bandwidth B. We shall denote this filtered signal as $r'_{pam}(t)$. Thus,

$$r'_{pam}(t) = \alpha \sum_{l=1}^{L} \gamma_l b'(t - \tau_l) + w'(t) \tag{2.4}$$

where $b'(t)$ and $w'(t)$ are the low-pass filtered versions of $b(t)$ and $w(t)$ respectively. This signal is then sampled at a rate of $T_s = 1/2B$ samples per second. The resulting discrete signal can be expressed as

$$r_{pam}[n] = \alpha u[n] + w[n], n = 0, \ldots, N - 1 \tag{2.5}$$

where $N = \lceil N_f T_f / T_s \rceil$, $r_{pam}[n] = r'_{pam}(nT_s)$, $u[n] = \sum_{l=1}^{L} \gamma_l b'(nT_s - \tau_l)$ and $w[n] = w(nT_s)$. It can be proved that $w[n]$ are i.i.d Gaussian random variables with variance $\sigma_w^2 = BN_0/2$. Let

$$\mathbf{r}_{pam} = [r_{pam}[0], r_{pam}[1], \ldots, r_{pam}[N-1]] \tag{2.6}$$

$$\mathbf{u} = [u[0], u[1], \ldots, u[N-1]] \tag{2.7}$$

$$\mathbf{w} = [w[0], w[1], \ldots, w[N-1]] \,. \tag{2.8}$$

then (2.5) can be written in more compact form as

$$\mathbf{r}_{pam} = \alpha\mathbf{u} + \mathbf{w} \tag{2.9}$$

It is known from detection theory that the optimal detector of α is to project the received vector $\mathbf{r}_{pam}$ onto the vector $\mathbf{u}$. Thus the sufficient statistic for this problem is

$$z_{pam} = \mathbf{r}_{pam}^T\mathbf{u}. \tag{2.10}$$

The optimal decision rule is

$$\hat{\alpha} = \begin{cases} +1, & z_{pam} > 0 \\ -1, & z_{pam} < 0 \end{cases} \tag{2.11}$$

and the corresponding probability of error is

$$P_e = Q\left(\frac{\|\mathbf{u}\|}{\sigma_w}\right). \tag{2.12}$$

Note that $\mathbf{u}$ contains the contribution of the channel.

2.2.2 Binary PPM Receiver

In pulse position modulation, the transmitter sends the signal

$$s_{ppm}(t) = b(t - \alpha\Delta), \tag{2.13}$$

where $\alpha \in \{0, 1\}$ is the transmitted information symbol and Δ is a delay. We have concentrated to one symbol period again. The received signal is expressed as

$$r_{ppm}(t) = \sum_{l=1}^{L} \gamma_l s_{ppm}(t - \tau_l) + w(t) = \alpha \sum_{l=1}^{L} \gamma_l b(t - \alpha\Delta - \tau_l) + w(t). \tag{2.14}$$

The receiver low-pass filters the received signal with an ideal low-pass filter of bandwidth B. We denote this filtered signal as $r'_{ppm}(t)$. Thus,

$$r'_{ppm}(t) = \alpha \sum_{l=1}^{L} \gamma_l b'(t - \alpha\Delta - \tau_l) + w'(t) \tag{2.15}$$

This signal is then sampled at a rate of $T_s = 1/2B$ samples per second. The resulting discrete signal can be expressed as

$$r_{ppm}[n] = \alpha u_\alpha[n] + w[n], n = 0, \ldots, N - 1 \tag{2.16}$$

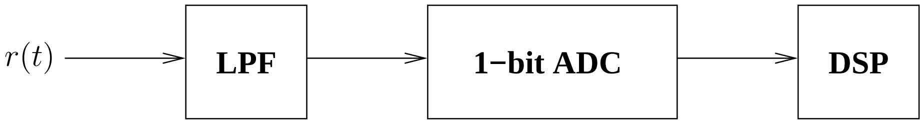

Figure 2.3: Monobit Digital Matched filter receiver.

where $r[n] = r'_{ppm}(nT_s)$, $u_\alpha[n] = \sum_{l=1}^{L} \gamma_l b'(nT_s - \alpha\Delta - \tau_l)$ and $w[n] = w(nT_s)$. $w[n]$ are i.i.d Gaussian random variables with variance $\sigma_w^2 = BN_0/2$. We define

$$\mathbf{r}_{ppm} = [r_{ppm}[0], r_{ppm}[1], \dots, r_{ppm}[N-1]] \tag{2.17}$$

$$\mathbf{u}_\alpha = [u_\alpha[0], u_\alpha[1], \dots, u_\alpha[N-1]], \alpha \in \{0,1\} \tag{2.18}$$

$$\mathbf{w} = [w[0], w[1], \dots, w[N-1]]. \tag{2.19}$$

Then, (2.16) can be written in compact form as

$$\mathbf{r}_{ppm} = \mathbf{u}_\alpha + \mathbf{w} \tag{2.20}$$

It is known from detection theory that the optimal detector of α is to project the received vector onto the vectors $\mathbf{u}_0$ and $\mathbf{u}_1$. Sufficient statistic for this decision problem is

$$z_{ppm} = \mathbf{r}_{ppm}^T(\mathbf{u}_1 - \mathbf{u}_0) \tag{2.21}$$

and the optimal decision rule is

$$\hat{\alpha} = \begin{cases} 1, & z_{ppm} > 0, \\ 0, & z_{ppm} < 0. \end{cases} \tag{2.22}$$

The corresponding probability of error is

$$P_e = Q\left(\frac{\|\mathbf{u}_1 - \mathbf{u}_0\|}{2\sigma_w}\right). \tag{2.23}$$

2.3 Monobit Matched Filter Receiver

The monobit matched filter receiver is depicted in Figure 2.3. The received signal is filtered by a low-pass filter of bandwidth B, and then sampled at a rate $T_s = 1/2B$. The filtered signal for one symbol period is described by equation (2.4) for a binary PAM signal and equation (2.15) for a binary PPM signal. The sampled signal is then quantized to one bit

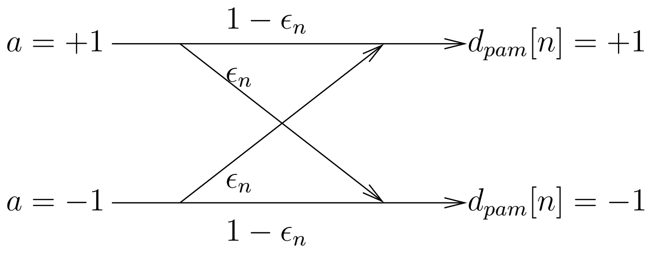

Figure 2.4: Monobit PAM receiver viewed as a binary symmetric channel.

resolution. The resulting bitstream is finally sent to a digital processing circuit. We begin by studying the optimal receiver for binary pulse amplitude modulated symbols when the receiver has perfect channel state information (CSI). Then, we will see how to estimate the channel state. Then, we will study monobit receivers for binary pulse position modulation and its corresponding channel estimation scheme.

2.3.1 Binary PAM Receiver

As we have seen earlier, the received signal is expressed as

$$r_{pam}(t) = \sum_{l=1}^{L} \gamma_l s_{pam}(t-\tau_l) + w(t) = \alpha \sum_{l=1}^{L} \gamma_l b(t-\tau_l) + w(t) \tag{2.24}$$

where L is the number of channel paths, γ_l and τ_l are the attenuation and delay of the l-th path, respectively, and $w(t)$ is the thermal noise at the receiver, which we assume to be white Gaussian with variance $N_0/2$. The receiver low-pass filters the received signal with an ideal low-pass filter of bandwidth B. We shall denote this filtered signal as $r'_{pam}(t)$. Thus,

$$r'_{pam}(t) = \alpha \sum_{l=1}^{L} \gamma_l b'(t-\tau_l) + w'(t) \tag{2.25}$$

where $b'(t)$ and $w'(t)$ are the low-pass filtered versions of $b(t)$ and $w(t)$ respectively. This signal is then sampled at a rate of $T_s = 1/2B$ samples per second and quantized to one bit resolution. The resulting discrete signal can be expressed as

$$d_{pam}[n] = \begin{cases} +1, & r'_{pam}(nT_s) > 0, \\ -1, & r'_{pam}(nT_s) \leq 0 \end{cases} \tag{2.26}$$

Define

$$\epsilon_n = Q\left(\frac{\alpha \sum_{l=1}^{L} \gamma_l b'(nT_s - \tau_l)}{2\sigma_w}\right) \tag{2.27}$$

where $Q(\cdot)$ is the Gaussian Q function. Note that ϵ_n is the probability of the n-th sample's sign being "wrong", i.e., opposite to the sign of the noiseless signal. This follows easily from (2.25). We can thus view to overall channel as a binary symmetric channel, as in Figure 2.5, with transition probabilities

$$P(d_{pam}[n]|\alpha) = \begin{cases} 1-\epsilon_n, & d_{pam}[n] = \alpha, \\ \epsilon_n, & d_{pam}[n] \neq \alpha. \end{cases} \tag{2.28}$$

Let

$$\mathbf{d}_{pam} = \left[d_{pam}[0], d_{pam}[1], \cdots, d_{pam}[N-1]\right]. \tag{2.29}$$

Assuming that α takes values $+1$ and -1 with equal probability, the optimal decision rule is the maximum likelihood. The ML decision rule of α based on $\mathbf{d}_{pam}$ is

$$\hat{\alpha} = \arg \max_{\tilde{\alpha} \in \{+1,-1\}} P(\mathbf{d}_{pam}|\tilde{\alpha}) \tag{2.30}$$

where $P(\cdot|\cdot)$ is the conditional probability density function of the random vector $\mathbf{d}_{pam}$ conditioned on α. Equivalently, we can write the former decision rule using the log-likelihood ratio as

$$\hat{\alpha} = \begin{cases} +1, & \log\left(\frac{P(\mathbf{d}_{pam}|a=+1)}{P(\mathbf{d}_{pam}|a=-1)}\right) > 0, \\ -1, & \log\left(\frac{P(\mathbf{d}_{pam}|a=+1)}{P(\mathbf{d}_{pam}|a=-1)}\right) \leq 0. \end{cases} \tag{2.31}$$

Notice that $d_{pam}[n]$ is a function of the noise sample $w[n]$ alone. Since, the noise samples are independent to each other so are $d_{pam}[n]$. Thus, the conditional density breaks up into a product of conditional densities

$$P(\mathbf{d}_{pam}|\tilde{\alpha}) = \prod_{n=0}^{N-1} P(d_{pam}[n]|\tilde{\alpha}). \tag{2.32}$$

The log-likelihood ratio takes the form

$$\log\left(\frac{P(\mathbf{d}_{pam}|a=+1)}{P(\mathbf{d}_{pam}|a=-1)}\right) = \sum_{n=0}^{N-1} \log\left(\frac{P(d_{pam}[n]|a=+1)}{P(d_{pam}[n]|a=-1)}\right). \tag{2.33}$$

If $d_{pam}[n] = +1$, then

$$\log\left(\frac{P(d_{pam}[n]|a=+1)}{P(d_{pam}[n]|a=-1)}\right) = \log\frac{1-\epsilon_n}{\epsilon_n} \tag{2.34}$$

and if $d_{pam}[n] = -1$, then

$$\log\left(\frac{P(d_{pam}[n]|a=+1)}{P(d_{pam}[n]|a=-1)}\right) = \log\frac{\epsilon_n}{1-\epsilon_n} = -\log\frac{1-\epsilon_n}{\epsilon_n}. \tag{2.35}$$

Let

$$u_{pam}[n] = \log\frac{1-\epsilon_n}{\epsilon_n},\ n = 0,\cdots,N-1. \tag{2.36}$$

and define the vector

$$\mathbf{u}_{pam} = \left[u_{pam}[0], u_{pam}[1], \ldots, u_{pam}[N-1]\right]. \tag{2.37}$$

The log-likelihood ratio can be written as

$$z_{pam}^{1bit} = \mathbf{d}_{pam}^{T}\mathbf{u}_{pam}. \tag{2.38}$$

The decision rule is

$$\hat{\alpha} = \begin{cases} +1, & z_{pam}^{1bit} > 0, \\ -1, & z_{pam}^{1bit} \leq 0. \end{cases} \tag{2.39}$$

It is interesting that the ML detector is again a linear combiner of the received samples just like in the full-resolution case. Of course, the optimal weights are now different. To the best of the author's knowledge, there is no closed form expression for the BER performance of the monobit receiver.

2.3.2 Channel Estimation

The ML receiver described in the previous section requires knowledge of the channel impulse response at the receiver. This is necessary in order to compute $\epsilon_n, n = 0,\cdots,N-1$, which are needed to calculate the weights $u_{pam}[n]$, $n = 0,\ldots,N-1$. Since we are limited to 1-bit per sample, estimating ϵ_n indirectly through the use of formula (2.27) is infeasible. Thus, practical methods for estimating ϵ_n directly are necessary. One method that has been suggested [7, 8] is to send a training signal consisting of N_t training symbols and do ML estimation of the parameters ϵ_n using the received signal. First, let $d_{pam}^{Tr}[n], n = 0,\cdots,N_{Tr}N-1$ be the signal the receiver gets when a training sequence of all ones is used. Let

$$\mathbf{d}_{pam}^{Tr} = \left[d_{pam}^{Tr}[0], d_{pam}^{Tr}[1], \cdots, d_{pam}^{Tr}[N_{Tr}N-1]\right]. \tag{2.40}$$

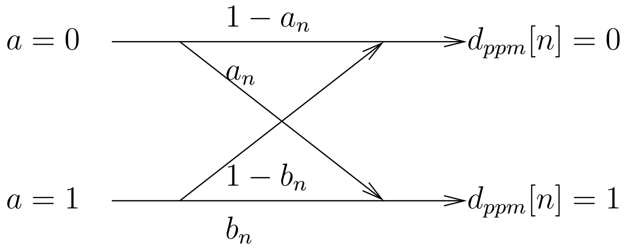

Figure 2.5: Monobit PPM receiver viewed as a binary channel.

The likelihood of $\mathbf{d}_{pam}^{Tr}$ given the parameters we want to estimate is

$$P(\mathbf{d}_{pam}^{Tr}|\epsilon_0, \cdots, \epsilon_N) = \prod_{k=0}^{N_{Tr}} \prod_{n=0}^{N-1} \epsilon_n^{(1-d_{pam}^{Tr}[n+kN])/2} (1-\epsilon_n)^{(1-d_{pam}^{Tr}[n+kN])/2}. \tag{2.41}$$

Taking the logarithm, differentiating with respect to ϵ_n and setting the derivative equal to zero, we get the ML estimator of ϵ_n

$$\hat{\epsilon}_n = 1 - \frac{1}{N_{Tr}} \sum_{k=0}^{N_{Tr}} d_{pam}^{Tr}[n+kN]. \tag{2.42}$$

We can now plug $\hat{\epsilon_n}$ into (2.36) to get an estimate of the optimal weights. Notice that this is the ML estimate $\hat{u}_{pam}[n]$ of the optimal weights based on $\mathbf{d}_{pam}^{Tr}$, i.e.,

$$\hat{u}_{pam}[n] = \log \frac{1-\hat{\epsilon}_n}{\hat{\epsilon}_n}, \; n = 0, \cdots, N-1. \tag{2.43}$$

One drawback of this method is that, when the SNR is large, ϵ_n will tend to be low and so a small estimation error on $\hat{\epsilon}_n$ will lead to a large estimation error in $u^{opt}[n]$. This might not be a practical problem since UWB systems have to operate with very low power anyway so SNR will in general not be greater than say 6 or 7 dB. Another practical problem is that N_{Tr} has to be quite large. This can be overcome by using the detected symbols as training.

2.3.3 Binary PPM Receiver

We have that the received signal is given by the expression

$$r_{ppm}(t) = \sum_{l=1}^{L} \gamma_l s_{ppm}(t - \tau_l - \alpha\Delta) + w(t) = \sum_{l=1}^{L} \gamma_l b(t - \tau_l - \alpha\Delta) + w(t) \tag{2.44}$$

where L is the number of channel paths, γ_l and τ_l are the attenuation and delay of the l-th path, respectively, and $w(t)$ is the thermal noise at the receiver, which we assume to be white Gaussian with variance $N_0/2$. The receiver low-pass filters the received signal with an ideal low-pass filter of bandwidth B. We shall denote this filtered signal as $r'_{ppm}(t)$. Thus,

$$r'_{ppm}(t) = \sum_{l=1}^{L} \gamma_l b'(t - \tau_l - \alpha\Delta) + w'(t) \tag{2.45}$$

where $b'(t)$ and $w'(t)$ are the low-pass filtered versions of $b(t)$ and $w(t)$ respectively. This signal is then sampled at a rate of $T_s = 1/2B$ samples per second and quantized to one bit resolution. The resulting discrete signal can be expressed as

$$d_{ppm}[n] = \begin{cases} 1, & r'_{ppm}(nT_s) > 0 \\ 0, & r'_{ppm}(nT_s) \leq 0 \end{cases}. \tag{2.46}$$

The derivation of the optimal binary PPM monobit receiver follows along the same lines as that of the binary PAM receiver. We can view again the overall channel from the transmitter to the receiver as a binary channel. Only now the transition probabilities are not symmetric. Define

$$a_n = Q\left(\frac{-\sum_{l=1}^{L} \gamma_l b'(nT_s - \tau_l)}{\sigma_w}\right) \tag{2.47}$$

$$b_n = Q\left(\frac{-\sum_{l=1}^{L} \gamma_l b'(nT_s - \alpha\Delta - \tau_l)}{\sigma_w}\right). \tag{2.48}$$

Notice that, a_n is the probability of receiving $d_{ppm}[n] = 1$ when $a = 0$ and b_n is the probability of receiving $d_{ppm}[n] = 1$ when $a = 1$. We can write these transition probabilities, for $n = 0, \ldots, N-1$, as

$$P(d_{ppm}[n]|\alpha = 0) = a_n^{d_{ppm}[n]}(1 - a_n)^{1-d_{ppm}[n]}, \tag{2.49}$$

$$P(d_{ppm}[n]|\alpha = 1) = b_n^{d_{ppm}[n]}(1 - b_n)^{1-d_{ppm}[n]}. \tag{2.50}$$

Define $\mathbf{d}_{ppm}$ as

$$\mathbf{d}_{ppm} = [d_{ppm}[0], d_{ppm}[1], \cdots, d_{ppm}[N-1]]\,. \tag{2.51}$$

We assume that α takes the values 1 and 0 with equal probability. Under that assumption the optimal decision rule is the ML which is

$$\hat{\alpha} = \arg \max_{\tilde{\alpha} \in \{0,1\}} P(\mathbf{d}_{ppm}|\tilde{\alpha}) \tag{2.52}$$

where $P(\cdot|\cdot)$ is the conditional probability density function of the random vector $\mathbf{d}_{ppm}$ conditioned on α. We will express the decision rule using the log-likelihood ratio again. Using the i.i.d property of the noise as before we get

$$\log\left(\frac{P(\mathbf{d}_{ppm}|a=1)}{P(\mathbf{d}_{ppm}|a=0)}\right) = \sum_{n=0}^{N-1}\log\left(\frac{P(d_{ppm}[n]|a=1)}{P(d_{ppm}[n]|a=0)}\right) \tag{2.53}$$

$$= \sum_{n=0}^{N-1}\log\left(\frac{b_n^{d_{ppm}[n]}(1-b_n)^{1-d_{ppm}[n]}}{a_n^{d_{ppm}[n]}(1-a_n)^{1-d_{ppm}[n]}}\right) \tag{2.54}$$

$$= \sum_{n=0}^{N-1} d_{ppm}[n]\log\frac{b_n}{a_n} + (1-d_{ppm}[n])\log\frac{1-b_n}{1-a_n} \tag{2.55}$$

$$= \sum_{n=0}^{N-1} d_{ppm}[n]\log\frac{b_n(1-a_n)}{a_n(1-b_n)} + \sum_{n=0}^{N-1}\log\frac{1-b_n}{1-a_n} \tag{2.56}$$

$$= \sum_{n=0}^{N-1} d_{ppm}[n]u_{ppm}[n] - \tau \tag{2.57}$$

where in the last equation we did the substitutions, for $n = 0, \ldots, N-1$,

$$u_{ppm}[n] = \log\frac{b_n(1-a_n)}{a_n(1-b_n)} \tag{2.58}$$

$$\tau = -\sum_{n=0}^{N-1}\log\frac{1-b_n}{1-a_n}. \tag{2.59}$$

Let

$$\mathbf{u}_{ppm} = \left[u_{ppm}[0], u_{ppm}[1], \cdots, u_{ppm}[N-1]\right]. \tag{2.60}$$

The log-likelihood ratio is

$$z_{ppm}^{1bit} = \mathbf{d}_{ppm}^T\mathbf{u}_{ppm} - \tau. \tag{2.61}$$

Thus, the decision rule can be written as

$$\hat{\alpha} = \begin{cases} 1, & z_{ppm}^{1bit} > \tau, \\ 0, & z_{ppm}^{1bit} \leq \tau. \end{cases} \tag{2.62}$$

2.3.4 Channel Estimation

We will describe a way to estimate a_n and b_n. In this case, because the channel is not symmetric we can't send a training signal of all ones. Instead, we will assume that the first half of the training symbols are 0 and the rest are 1. The derivation is the same as in the case of binary PAM modulation. The estimators are

$$\hat{a}_n = \frac{2}{N_{Tr}} \sum_{k=0}^{N_{Tr}/2-1} d_{ppm}^{Tr}[n + kN], \tag{2.63}$$

$$\hat{b}_n = \frac{2}{N_{Tr}} \sum_{k=0}^{N_{Tr}/2-1} d_{ppm}^{Tr}[n + kN]. \tag{2.64}$$

These estimators suffer from the same problems as the previous one.

Chapter 3

Non-coherent UWB Receivers

3.1 Introduction

Whereas low-complexity UWB transmitters are feasible, especially when considering the principle of impulse radio (IR) signaling, there are numerous of processing tasks that, as a consequence of the large signal bandwidth, make the implementation of conventional optimum receivers extremely complex, if not feasible at all.

First of all, a fully digital receiver must be able to sample the received signal at least at the Nyquist rate, which is twice the signal bandwidth. Analog-to-digital converters (ADCs) working at such a high rate are in general very expensive and power demanding. Second, to capture a sufficient amount of energy using a Rake receiver, a large number of correlators must be implemented, resulting in a very complex hardware architecture. Such receivers are further burdened with the problem of estimating the amplitude and the delay of each multipath component, which must be accomplished in general at a relatively low signal-to-noise ratio (SNR).

A coherent receiver can exploit the absolute phase information of the received carrier-modulated signal, while a noncoherent energy detection receiver can only exploit the envelope, i.e., instantaneous power, of the signal. The noncoherent receiver has a complexity advantage in the sense that no coherent carrier recovery is needed. The need for low-complexity devices with low-power consumption motivates the application of suboptimal noncoherent ultra-wideband (UWB) receivers.

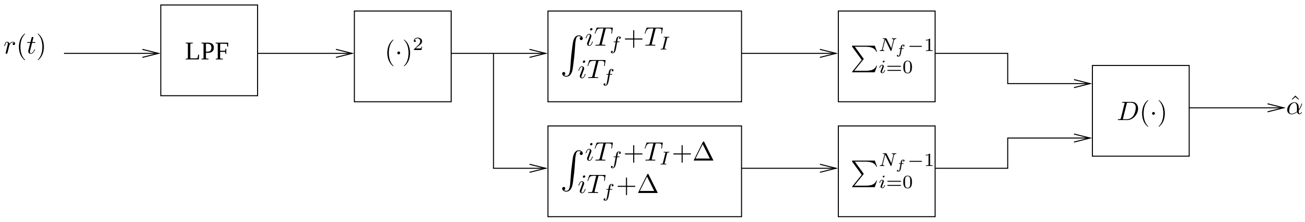

Figure 3.1: Energy Detector receiver. Although two integrators are shown the same operation can be done by a single integrator as long as the two windows do not intersect.

3.2 Energy Detector Receiver

The energy detector receiver is shown in Figure 3.1. The signal that comes from the antenna is first low-pass filtered. The resulting signal passes through a square-law device and an integrator integrates the output of the square-law device over a window to calculate the energy. In case of the ED, any information on the symbol phase is lost, therefore binary PAM is useless. Typically, on-off keying (OOK) and pulse-position modulation (PPM) are used. The integration window depends on the modulation used. In particular, two windows need to be used for PPM.

It is known that, to minimize the BER, the LPF must be matched to the transmitted waveform which is distorted by the channel impulse response. However, the channel impulse response is not known at the receiver as it would destroy the complexity advantage of the ED receiver. In our case, we will simply assume that an ideal low-pass filter is used.

We will now describe the energy detector receiver for PPM modulation more analytically. The transmitted signal for one symbol ignoring any possible ISI is

$$s_{ppm}(t) = b(t - \alpha\Delta) \tag{3.1}$$

where $b(t)$ is defined in (1.4), $\alpha \in \{0, 1\}$ is the transmitted information symbol and Δ the delay that we use to modulate the data on the waveform. The received signal is expressed as

$$r_{ppm}(t) = \sum_{l=1}^{L} \gamma_l s_{ppm}(t - \tau_l) + w(t) = \sum_{l=1}^{L} \gamma_l b(t - \alpha\Delta - \tau_l) + w(t) \tag{3.2}$$

where L is the number of channel paths, γ_l and τ_l are the attenuation and delay of the l-th path, respectively, and $w(t)$ is the thermal noise at the receiver which we assume to be white Gaussian with variance $N_0/2$. The receiver then low-pass filters the received signal

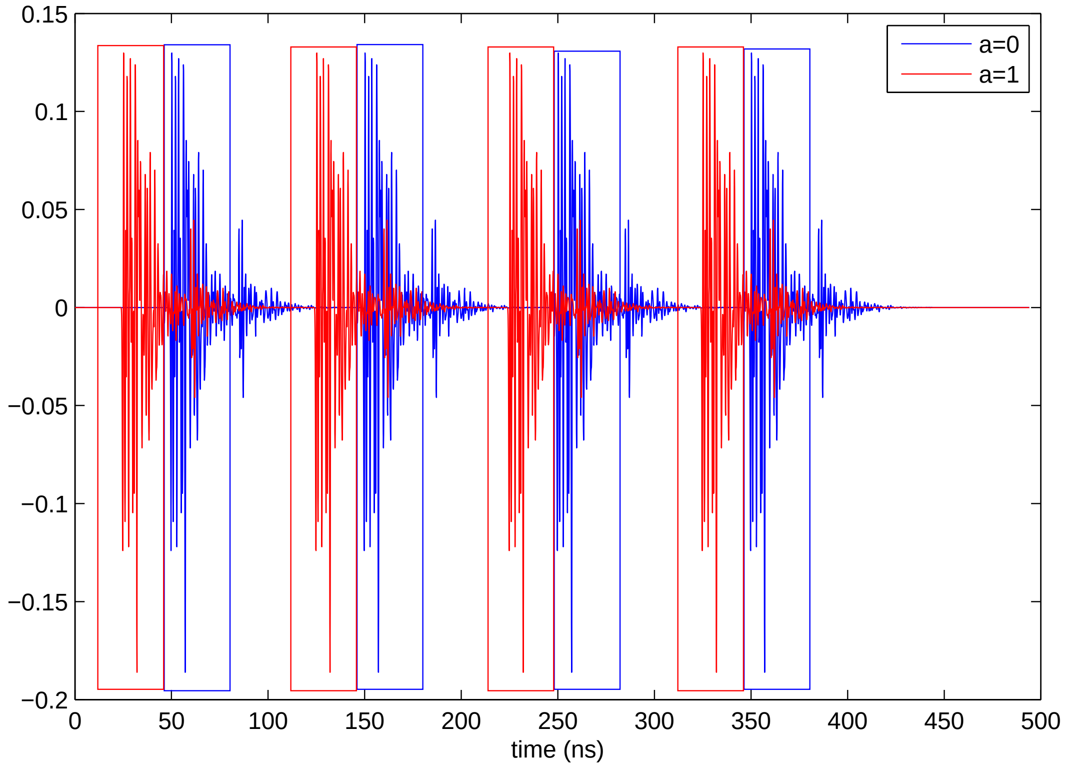

Figure 3.2: Graphical Representation of the energy detection process.

with an ideal low-pass filter of bandwidth B. We denote the filtered signal as

$$r'_{ppm}(t) = \sum_{l=1}^{L} \gamma_l b'(t - \alpha\Delta - \tau_l) + w'(t) \tag{3.3}$$

where $b'(t)$ and $w'(t)$ are the filtered version of $b(t)$ and $w(t)$ respectively. Define

$$u_\alpha(t) = \sum_{l=1}^{L} \gamma_l b'(t - \tau_l), \alpha \in \{0, 1\}. \tag{3.4}$$

It will be convenient in the rest of the analysis to write (3.3) as

$$r'_{ppm}(t) = u_\alpha(t) + w'(t). \tag{3.5}$$

Then, we pass $r'_{ppm}(t)$ through a square-law device to get $r'^2_{ppm}(t)$. An integrator then collects the energy inside two time windows that are Δ seconds apart. This is done once per frame as is shown in Figure 3.2. The position of the integration windows should be where most of the signal energy is expected to be while their length is a design parameter that the system designer must choose carefully. If we assume that the two time windows used are $[0, T_I]$ and $[\Delta, \Delta + T_I]$ then the energy collected in each at the i-th frame of a symbol is

$$E_{0,i} = \int_{iT_f}^{iT_f+T_I} r'^2(t)dt, \tag{3.6}$$

$$E_{1,i} = \int_{iT_f+\Delta}^{iT_f+\Delta+T_I} r'^2(t)dt, \tag{3.7}$$

for $i = 0, \ldots, N_f - 1.$. The total energy of all the frames is then accumulated

$$E_0 = \sum_{i=0}^{N_f} E_{0,i} \tag{3.8}$$

$$E_1 = \sum_{i=0}^{N_f} E_{1,i} \tag{3.9}$$

and compared to each other. If $E_0 > E_1$, then the receiver decides $\hat{\alpha} = 0$, while if $E_1 > E_0$ it decides $\hat{\alpha} = 1$.

3.3 Transmit Reference Receiver

The transmit reference receiver replaces the square-law device of the energy detector by a delay element and a multiplier. This is shown in Figure 3.4. The delay aligns the first of two consecutively transmitted pulses such that it can serve as a noisy template for the demodulation of the second pulse, which is PSK-modulated by the data symbol. Thus, the delay element has to delay the signal by T_f seconds. One drawback of TR is that the transmitter has to send one reference pulse per modulated pulse thus consuming only half of the total transmitted energy to send actual data.

3.3.1 TR Receiver for Binary PAM modulation

Unlike the ED receiver, the TR receiver can use binary PAM. We will describe how the TR receiver demodulates a binary PAM signal. The transmitter sends the signal

$$s_{pam}(t) = b_{TR}(t) + \alpha b_{TR}(t - T_f) \tag{3.10}$$

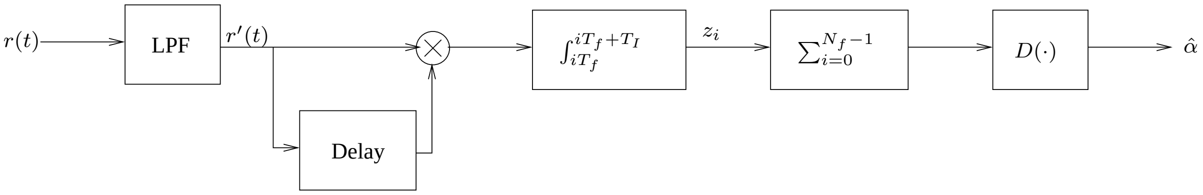

Figure 3.3: Transmit Reference receiver for binary PAM modulation.

where $\alpha \in \{+1, -1\}$ is the transmitted information symbol and $b_{TR}(t)$ is the pulse train used to convey each symbol defined as

$$b_{TR}(t) = \sum_{j=0}^{N_f} p(t - j2T_f - c_j T_c). \tag{3.11}$$

where $p(t)$ is the monopulse. The received signal is expressed as

$$\begin{aligned} r_{pam}(t) &= \sum_{l=1}^{L} \gamma_l s_{pam}(t - \tau_l) + w(t) && (3.12) \\ &= \sum_{l=1}^{L} \gamma_l b_{TR}(t - \tau_l) + \alpha \sum_{l=1}^{L} \gamma_l b_{TR}(t - T_f - \tau_l) + w(t) && (3.13) \\ &= u_{TR}(t) + \alpha u_{TR}(t - T_f) + w(t) && (3.14) \end{aligned}$$

where

$$u_{TR}(t) = \sum_{l=1}^{L} \gamma_l b_{TR}(t - \tau_l). \tag{3.15}$$

L is the number of channel paths, γ_l and τ_l are the attenuation and delay of the l-th path, respectively, and $w(t)$ is the thermal noise at the receiver which we assume to be white Gaussian with variance $N_0/2$. The receiver low-pass filters the received signal with an ideal low-pass filter of bandwidth B. We shall denote this filtered signal as $r'_{pam}(t)$

$$r'_{pam}(t) = u'_{TR}(t) + \alpha u'_{TR}(t - T_f) + w'(t) \tag{3.16}$$

where $u'_{TR}(t)$ and $w'(t)$ are the filtered versions of $u_{TR}(t)$ and $w(t)$, respectively. Then, a delay element delays the signal by T_f seconds. The delayed and the original signal are fed to a mixer to compute their product. Finally, the output of the mixer goes to an integrator that integrates the incoming signal once every frame duration inside a time window of

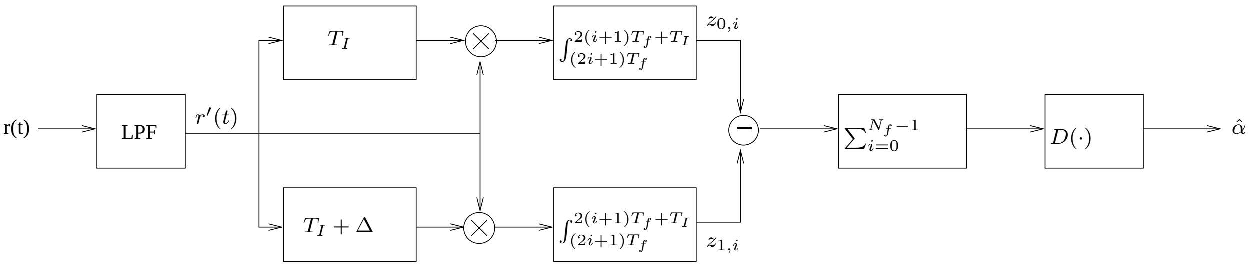

Figure 3.4: Transmit Reference receiver for binary PPM modulation.

duration T_I. The output of the integrator for the i-th frame of a symbol is

$$z_{pam,i}^{TR} = \int_{iT_f}^{iT_f+T_I} r'(t)r'(t-T_f)dt\,, i = 0, \ldots, N-1. \tag{3.17}$$

These samples are then added together and a decision is made based on the sign of the result

$$\hat{\alpha} = \begin{cases} +1, & z_{pam}^{TR} > 0, \\ -1, & z_{pam}^{TR} \leq 0, \end{cases} \tag{3.18}$$

where

$$z_{pam}^{TR} = \sum_{i=0}^{N_f-1} z_{pam,i}^{TR}. \tag{3.19}$$

3.3.2 TR Receiver for Binary PPM modulation

Next, we describe how a TR receiver demodulates a binary PPM signal. The transmitter sends the signal

$$s_{ppm}^{TR}(t) = b_{TR}(t) + b_{TR}(t - T_f - \alpha\Delta) \tag{3.20}$$

where $\alpha \in \{0, 1\}$ is the transmitted information symbol and $b_{TR}(t)$ is the pulse train used to convey each symbol defined as before by 3.11. The received signal is expressed as

$$r_{ppm}^{TR}(t) = \sum_{l=1}^{L} \gamma_l s_{ppm}^{TR}(t - \tau_l) + w(t) \tag{3.21}$$

$$= \sum_{l=1}^{L} \gamma_l b_{TR}(t - \tau_l) + \sum_{l=1}^{L} \gamma_l b_{TR}(t - T_f - \alpha\Delta - \tau_l) + w(t) \tag{3.22}$$

$$= u_{TR}(t) + u_{TR}(t - T_f - \alpha\Delta) + w(t) \tag{3.23}$$

where $u_{TR}(t)$ is defined in 3.15. The receiver low-pass filters the received signal with an ideal low-pass filter of bandwidth B. We shall denote this filtered signal as $r'^{TR}_{ppm}(t)$

$$r'^{TR}_{ppm}(t) = u'_{TR}(t) + u'_{TR}(t - T_f - \alpha\Delta) + w'(t). \tag{3.24}$$

Then, two delay elements delay the signal by T_f and $T_f + \Delta$ seconds, respectively. The delayed signals are fed to two mixers that multiply them with the original signal. The output of each mixer goes to an integrator that integrates the incoming signal in a window of duration T_I once each frame. The output of the integrators for the i-the frame is

$$z^{TR}_{0,i} = \int_{(2i+1)T_f}^{2(i+1)T_f+T_I} r'^{TR}_{ppm}(t) r'^{TR}_{ppm}(t - T_f)dt, \tag{3.25}$$

$$z^{TR}_{1,i} = \int_{(2i+1)T_f}^{2(i+1)T_f+T_I} r'^{TR}_{ppm}(t) r'^{TR}_{ppm}(t - T_f - \Delta)dt, \tag{3.26}$$

for $i = 0, \ldots, N_f - 1$.. Notice that integration takes place during the frames that carry information and not during the reference pulse. These samples are then added together and a decision is made by comparing the results

$$\hat{\alpha} = \begin{cases} 0, & z^{TR}_0 > z^{TR}_1 \\ 1, & z^{TR}_0 \leq z^{TR}_1 \end{cases} \tag{3.27}$$

where

$$z^{TR}_k = \sum_{i=0}^{N_f-1} z^{TR}_{k,i}, k \in \{0, 1\}. \tag{3.28}$$

Chapter 4

Simulations – Conclusions

4.1 Simulations

We will close this thesis with some simulation results that will give us a clearer understanding of the performance of the receivers studied in the previous chapters. Notice that the pulse shape used in the simulations was not chosen in accordance with the power spectrum regulations of the FCC. We considered an ideal system were there is no interference from other users and no power spectrum limitations.

The pulse used in the simulation is shown in Figure 1.4. The duration of the pulse is roughly 2 ns while the frame duration (remember that a frame is the time interval between consecutive pulses) was chosen to be equal to $T_f = 100$ ns. The sampling rate is chosen to be $T_s = 0.1$ ns. Finally, we have choose $N_f = 4$, i.e., four pulses are sent for each symbol so the number of samples per symbol is 4000.

Two channel models are considered

- A frequency selective additive white Gaussian noise (AWGN) channel.
- A frequency selective fading channel.

In both cases, the noise variance for a certain SNR (measured in decibels) was calculated by the formula

$$\sigma_n^2 = E_s 10^{-\mathrm{SNR}/10} \tag{4.1}$$

where E_s is the energy send per symbol i.e.

$$E_s = \int_0^{N_f T_f} b^2(t) dt. \tag{4.2}$$

The UWB channel impulse response realization are produced using the channel model developed by IEEE ([20]).

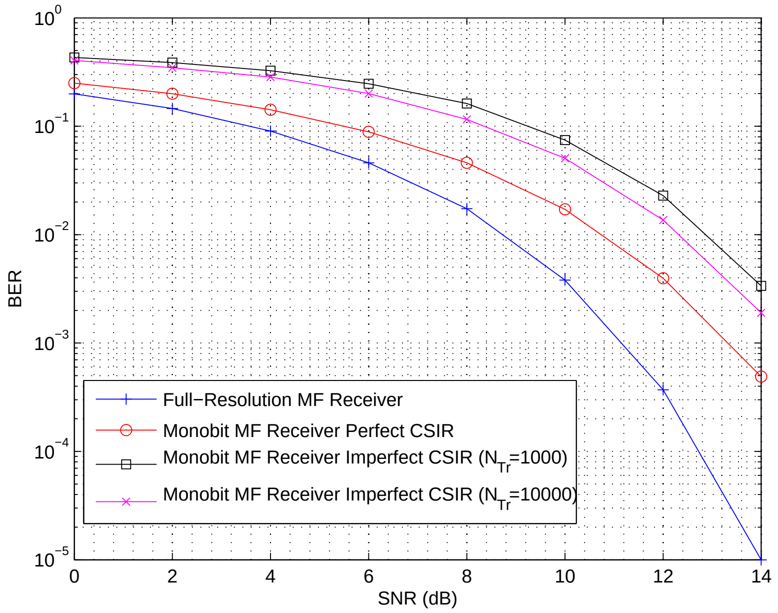

Figure 4.1: BER vs SNR performance comparisons of Monobit Digital Matched filter receiver under AWGN channel. The modulation used is binary PPM. As we can see the performance degradation due to the quantization to 1-bit resolution assuming perfect channel knowledge is only 2 dB. Channel estimation errors further degrades performance by 2 dB.

In Figure 4.1, we plot the BER versus SNR curve of a monobit binary PPM receiver for the AWGN channel. To assess the impact of channel estimation in the performance, we did one experiment in which we used $N_{Tr} = 1000$ and one in which $N_{Tr} = 10000$. For comparison, we also plot the BER vs SNR curves of the full-resolution receiver as well as the monobit receiver with perfect CSIR (Channel State Information at the Receiver). From this figure, it is evident that channel impulse response estimation is a crucial factor for the performance of monobit receivers. Figure 4.2 shows the same curves for binary PAM modulation.

Figure 4.3 plots the BER versus SNR curve of a monobit binary PAM receiver for the

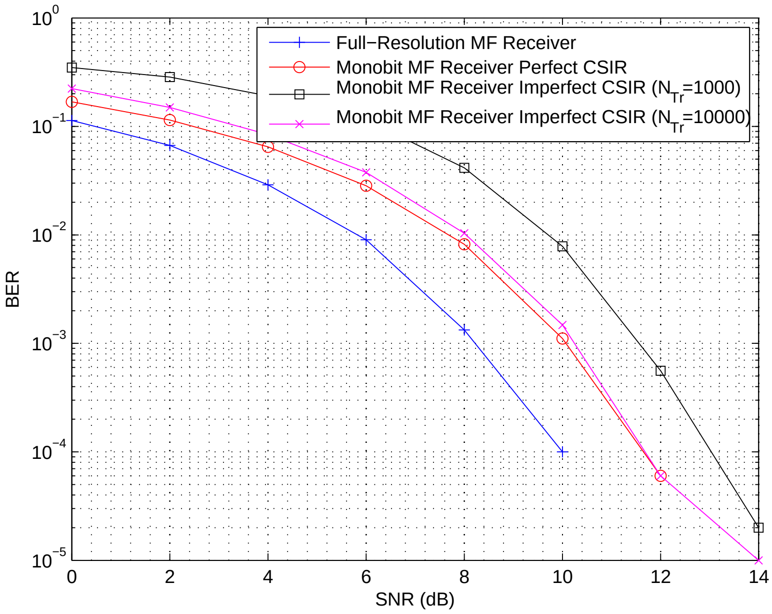

Figure 4.2: BER vs SNR performance comparisons of Monobit Digital Matched filter receiver under AWGN channel. The modulation used in this figure is binary PAM. The same channel impulse response as in the case of binary PPM is also used. The degradation of the BER due to quantization is 2 dB again as in binary PPM. In this figure we see that for $N_{TR} = 10000$ the performance is close to that of the perfect channel state information case. This is not the case for binary PPM. This means that binary PAM is less sensitive to channel estimation errors than binary PPM.

fading channel. We can see that there is a gain loss compared to the AWGN channel. Unlike narrowband fading channels, however, the BER falls exponentially with the SNR.

Figure 4.4 shows the BER performance of the transmit reference receiver for the AWGN channel. There is a huge reduction of the performance compared to the coherent matched filter receiver. Performance is in fact worse than the corresponding performance of monobit receivers even when imperfect CSI is available.

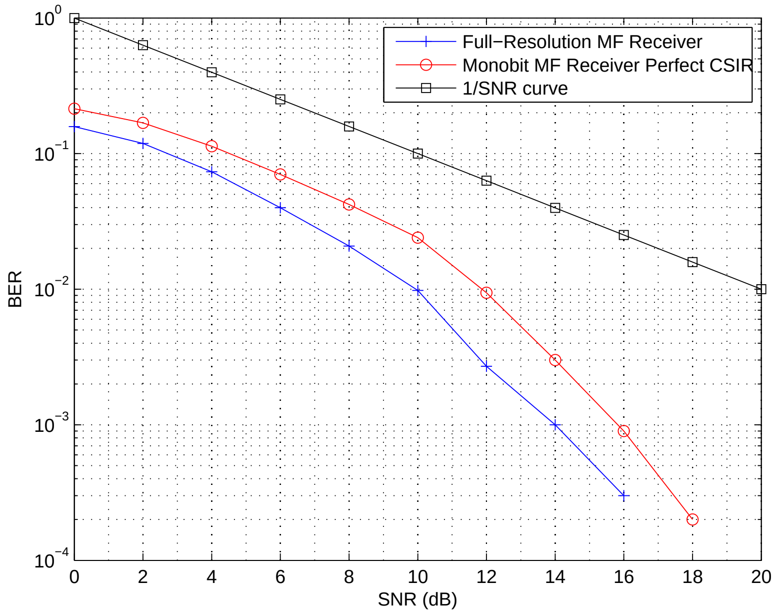

Figure 4.3: BER vs SNR performance of Monobit Digital Matched filter receiver under fading channel. The modulation used is binary PAM. We can see that the BER falls faster than the 1/SNR rate of narrowband fading channels. This is one of the most appealing features of UWB. Compared to the non fading case however we see that a bit error rate of 10^{-3} is achieved for an average SNR of 14 dB for the fading channel in contrast to only 8 dB for the non fading channel.

4.2 Conclusions

From the simulations above and the description of monobit and noncoherent receivers in the previous chapters we can draw two main conclusions

1. Monobit and noncoherent receivers have roughly the same implementation complexity. However, monobit receivers are simpler since they use less analog components such as, delay lines and mixers, than noncoherent receivers.

2. The BER performance of monobit receivers even with imperfect CSIR, due to channel estimation errors, is much better than the performance of noncoherent receivers for

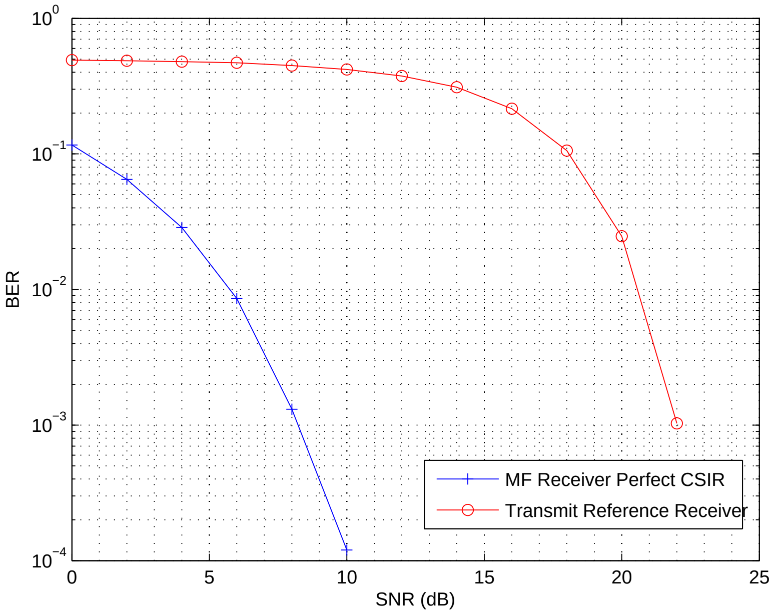

Figure 4.4: BER vs SNR performance of transmit reference receiver under AWGN channel. The modulation used is binary PAM. Note that to achieve a BER of 10^{-3} the transmit reference receiver requires an SNR of around 22 dB, 14dB more that the matched filter receiver with perfect CSI or the 13 dB for the monobit MF receiver with perfect CSI.

the same SNR. This means that less energy is required to achieve a certain BER using a monobit receiver rather than a noncoherent receiver. Thus, less interference will be caused to other users and specifically to licensed users.

Thus, monobit receivers are better for low-cost UWB applications, such as sensor networks, than noncoherent receivers.

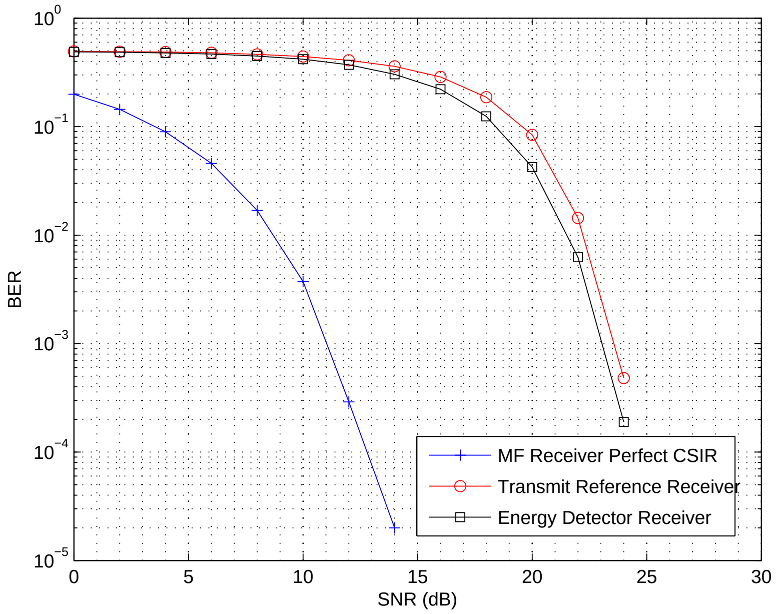

Figure 4.5: BER vs SNR performance of transmit reference and energy detector receivers under AWGN channel.

Chapter 5

Matlab code

The matlab scripts that were used to evaluate the bit error rates of the receivers that we studied are appended below.

5.1 Monobit binary PAM receiver - AWGN channel

```
close all
clear all
clc

ts = 0.1;
Tf = 100;
t_p = -Tf/2:ts:Tf/2-ts;
p = t_p.*exp(-8*t_p.^2);p = p/norm(p);

Nf = 4;
t_b = -Tf/2:ts:(Tf/2+(Nf-1)*Tf-ts);
b = kron(ones(1,Nf),p);
Es = norm(b)^2;

num_channels = 1;
cm_num = 1;  % channel model number from 1 to 4
% get channel model params based on this channel model number
[Lam,lambda,Gam,gamma,std_ln_1,std_ln_2,nlos,std_shdw] = uwb_sv_params( cm_num );
[h_ct,t_ct,t0,np] = uwb_sv_model_ct( Lam, lambda, Gam, gamma, std_ln_1, std_ln_2, nlos, ...
     std_shdw, num_channels );
[hN,N] = uwb_sv_cnvrt_ct( h_ct, t_ct, np, num_channels, ts );
if N > 1,
    h = resample(hN, 1, N);  % decimate the columns of hN by factor N
else
    h = hN;
end
% correct for 1/N scaling imposed by decimation
h = h * N;

u = conv(b,h);

numsymbols = 100000;
minSNR = 0;
maxSNR = 20;
stepSNR = 1;

for SNRdb = minSNR:stepSNR:maxSNR
    SNRdb
    errs1 = 0;
    errs2 = 0;
    errs3 = 0;
    errs4 = 0;

    sigma_n = sqrt(Es*10^(-SNRdb/10));
    e = qfunc(u/sigma_n);
    uref_opt = log((1-e)./e);

    e_est = zeros(size(uref_opt));
    Ntr = 1000;
    for k = 1:Ntr
        n = sigma_n*randn(size(u));
        r = u + n;
        e_est = e_est + (1 - sign(r))/2;
    end
    e_est = e_est/Ntr;
    uref_est = log((1-e_est)./e_est);

    e_est2 = zeros(size(uref_opt));
    Ntr = 10000;
    for k = 1:Ntr
        n = sigma_n*randn(size(u));
        r = u + n;
        e_est2 = e_est2 + (1 - sign(r))/2;
    end
    e_est2 = e_est2/Ntr;
    uref_est2 = log((1-e_est2)./e_est2);

    for ii = 1:numsymbols
        a = sign(randn);
        n = sigma_n*randn(size(u));
        r = a*u + n;
        a_hat_ml = sign(r*u');
        errs1 = errs1 + (a ~= a_hat_ml);
        a_hat_1bit = sign(sign(r)*uref_opt');
        errs2 = errs2 + (a ~= a_hat_1bit);
        a_hat_1bit_est = sign(sign(r)*uref_est');
        errs3 = errs3 + (a ~= a_hat_1bit_est);
        a_hat_1bit_est2 = sign(sign(r)*uref_est2');
        errs4 = errs4 + (a ~= a_hat_1bit_est2);
    end
    ber_ml(SNRdb-minSNR+1) = errs1/numsymbols;
    ber_1bit(SNRdb-minSNR+1) = errs2/numsymbols;
    ber_1bit_est(SNRdb-minSNR+1) = errs3/numsymbols;
    ber_1bit_est2(SNRdb-minSNR+1) = errs4/numsymbols;
end
semilogy(minSNR:stepSNR:maxSNR, ber_ml(1:stepSNR:maxSNR-minSNR+1))
hold on
semilogy(minSNR:stepSNR:maxSNR, ber_1bit(1:stepSNR:maxSNR-minSNR+1),'r')
semilogy(minSNR:stepSNR:maxSNR, ber_1bit_est(1:stepSNR:maxSNR-minSNR+1),'k' )
semilogy(minSNR:stepSNR:maxSNR, ber_1bit_est2(1:stepSNR:maxSNR-minSNR+1),'m' )
hold off
grid minor
legend('Full-Resolution␣MF␣Receiver','Monobit␣MF␣Receiver␣Perfect␣CSIR','Monobit␣MF␣Receiver␣Imperfect␣CSIR␣(N_{Tr}=1000)','Monobit␣MF␣Receiver␣Imperfect␣CSIR␣(N_{Tr}=10000)')
```

5.2 Monobit binary PPM receiver - AWGN channel

```
close all
clear all
clc

ts = 0.1;
Tf = 100;
t_p = -Tf/2:ts:Tf/2-ts;
p = t_p.*exp(-8*t_p.^2);p = p/norm(p);

Nf = 4;
b = kron(ones(1,Nf),p);
```

```
Es = norm(b)^2;

num_channels = 1;
cm_num = 1;  % channel model number from 1 to 4
% get channel model params based on this channel model number
[Lam,lambda,Gam,gamma,std_ln_1,std_ln_2,nlos,std_shdw] = uwb_sv_params( cm_num );
[h_ct,t_ct,t0,np] = uwb_sv_model_ct( Lam, lambda, Gam, gamma, std_ln_1, std_ln_2, nlos, ...
    std_shdw, num_channels );
[hN,N] = uwb_sv_cnvrt_ct( h_ct, t_ct, np, num_channels, ts );
if N > 1,
    h = resample(hN, 1, N);  % decimate the columns of hN by factor N
else
    h = hN;
end
% correct for 1/N scaling imposed by decimation
h = h * N;

uu(1,:) = conv(b,h);
N_ppm = Tf/ts/4;
uu(2,:) = [uu(1,N_ppm+1:end) uu(1,1:N_ppm)];
u = uu(2,:)-uu(1,:);

numsymbols = 100000;
minSNR = 0;
maxSNR = 40;
stepSNR = 5;

for SNRdb = minSNR:stepSNR:maxSNR
    SNRdb
    errs1 = 0;
    errs2 = 0;
    errs3 = 0;
    errs4 = 0;

    sigma_n = sqrt(Es*10^(-SNRdb/10));
    a = qfunc(-uu(1,:));
    b = qfunc(-uu(2,:));
    uref_opt = log(b.*(1-a)./(a.*(1-b)));
    t_opt = -sum(log((1-b)./(1-a)));

    a_est = zeros(size(uu(1,:)));
    Ntr = 1000;
    for k = 1:Ntr/2
```

```
        n = sigma_n*randn(size(uu(1,:)));
        r = uu(1,:) + n;
        a_est = a_est + (1 + sign(r))/2;
    end
    a_est = a_est/Ntr/2;
    b_est = zeros(size(uu(1,:)));
    Ntr = 1000;
    for k = 1:Ntr/2
        n = sigma_n*randn(size(uu(1,:)));
        r = uu(2,:) + n;
        b_est = b_est + (1 + sign(r))/2;
    end
    b_est = b_est/Ntr/2;
    uref_est = log(b_est.*(1-a_est)./(a_est.*(1-b_est)));
    t_est = -sum(log((1-b_est)./(1-a_est)));

    a_est2 = zeros(size(uu(1,:)));
    Ntr = 10000;
    for k = 1:Ntr/2
        n = sigma_n*randn(size(uu(1,:)));
        r = uu(1,:) + n;
        a_est2 = a_est2 + (1 + sign(r))/2;
    end
    a_est2 = a_est2/Ntr/2;
    b_est2 = zeros(size(uu(1,:)));
    Ntr = 1000;
    for k = 1:Ntr/2
        n = sigma_n*randn(size(uu(1,:)));
        r = uu(2,:) + n;
        b_est2 = b_est2 + (1 + sign(r))/2;
    end
    b_est2 = b_est2/Ntr/2;
    uref_est2 = log(b_est2.*(1-a_est2)./(a_est2.*(1-b_est2)));
    t_est2 = -sum(log((1-b_est2)./(1-a_est2)));

    for ii = 1:numsymbols
        a = sign(randn);
        n = sigma_n*randn(size(uu(1,:)));
        r = uu((a+1)/2+1,:) + n;

        a_hat_ml = sign(r*u');%this is correct
        errs1 = errs1 + (a ~= a_hat_ml);
        a_hat_1bit = sign(sign(r)*uref_opt'-t_opt);
        errs2 = errs2 + (a ~= a_hat_1bit);
        a_hat_1bit_est = sign(sign(r)*uref_est'-t_est);
        errs3 = errs3 + (a ~= a_hat_1bit_est);
        a_hat_1bit_est2 = sign(sign(r)*uref_est2'-t_est2);
        errs4 = errs4 + (a ~= a_hat_1bit_est2);
    end
    ber_ml(SNRdb-minSNR+1) = errs1/numsymbols;
    ber_1bit(SNRdb-minSNR+1) = errs2/numsymbols;
    ber_1bit_est(SNRdb-minSNR+1) = errs3/numsymbols;
    ber_1bit_est2(SNRdb-minSNR+1) = errs4/numsymbols;
end
semilogy(minSNR:stepSNR:maxSNR, ber_ml(1:stepSNR:maxSNR-minSNR+1))
hold on
semilogy(minSNR:stepSNR:maxSNR, ber_1bit(1:stepSNR:maxSNR-minSNR+1),'r')
semilogy(minSNR:stepSNR:maxSNR, ber_1bit_est(1:stepSNR:maxSNR-minSNR+1),'k' )
semilogy(minSNR:stepSNR:maxSNR, ber_1bit_est2(1:stepSNR:maxSNR-minSNR+1),'m' )
xlabel('SNR (dB)')
ylabel('BER')
legend('Full-Resolution MF Receiver','Monobit MF Receiver Perfect CSIR','Monobit MF Receiver Imperfect CSIR (N_{Tr}=1000)','Monobit MF Receiver Imperfect CSIR (N_{Tr}=10000)')
hold off
grid minor
```

5.3 Monobit binary PPM receiver - Fading channel

```
close all
clear all
clc

ts = 0.1;
Tf = 100;
t_p = -Tf/2:ts:Tf/2-ts;
p = t_p.*exp(-8*t_p.^2);p = p/norm(p);

Nf = 4;
b = kron(ones(1,Nf),p);
Es = norm(b)^2;

numsymbols = 10000;
minSNR = 0;
maxSNR = 21;
stepSNR = 3;

```

```
for SNRdb = minSNR:stepSNR:maxSNR
    SNRdb
    errs1 = 0;
    errs2 = 0;
    errs3 = 0;
    errs4 = 0;

    sigma_n = sqrt(Es*10^(-SNRdb/10));

    for ii = 1:numsymbols

        ii

        num_channels = 1;
        cm_num = 1;  % channel model number from 1 to 4
        % get channel model params based on this channel model number
        [Lam,lambda,Gam,gamma,std_ln_1,std_ln_2,nlos,std_shdw] = uwb_sv_params( cm_num );
        [h_ct,t_ct,t0,np] = uwb_sv_model_ct( Lam, lambda, Gam, gamma, std_ln_1, std_ln_2, nlos, ...
            std_shdw, num_channels );
        [hN,N] = uwb_sv_cnvrt_ct( h_ct, t_ct, np, num_channels, ts );
        if N > 1,
            h = resample(hN, 1, N);  % decimate the columns of hN by factor N
        else
            h = hN;
        end
        % correct for 1/N scaling imposed by decimation
        h = h * N;

        uu = [];
        uu(1,:) = conv(b,h);
        N_ppm = Tf/ts/4;
        uu(2,:) = [uu(1,N_ppm+1:end) uu(1,1:N_ppm)];
        u = uu(2,:)-uu(1,:);

        a = qfunc(-uu(1,:));
        b = qfunc(-uu(2,:));
        uref_opt = log(b.*(1-a)./(a.*(1-b)));
        t_opt = -sum(log((1-b)./(1-a)));

        a = sign(randn);
        n = sigma_n*randn(size(uu(1,:)));
        r = uu((a+1)/2+1,:) + n;

        a_hat_ml = sign(r*u');%this is correct
        errs1 = errs1 + (a ~= a_hat_ml);
        a_hat_1bit = sign(sign(r)*uref_opt'-t_opt);
        errs2 = errs2 + (a ~= a_hat_1bit);

    end
    ber_ml(SNRdb-minSNR+1) = errs1/numsymbols;
    ber_1bit(SNRdb-minSNR+1) = errs2/numsymbols;

end
semilogy(minSNR:stepSNR:maxSNR, ber_ml(1:stepSNR:maxSNR-minSNR+1))
hold on
semilogy(minSNR:stepSNR:maxSNR, ber_1bit(1:stepSNR:maxSNR-minSNR+1),'r')
xlabel('SNR (dB)')
ylabel('BER')
legend('Full-Resolution MF Receiver','Monobit MF Receiver Perfect CSIR','Monobit MF Receiver Imperfect CSIR (N_{Tr}=1000)','Monobit MF Receiver Imperfect CSIR (N_{Tr}=10000)')
hold off
grid minor
```

5.4 Transmit Referemce binary PAM receiver - AWGN channel

```
close all
clear all
clc

ts = 0.1;
Tf = 100;
t_p = -Tf/2:ts:Tf/2-ts;
p = t_p.*exp(-8*t_p.^2);p = p/norm(p);

Nf = 4;
t_b = -Tf/2:ts:(Tf/2+(Nf-1)*Tf-ts);
b = kron(ones(1,Nf),p);
Es = norm(b)^2;

num_channels = 1;
cm_num = 1;  % channel model number from 1 to 4
% get channel model params based on this channel model number
[Lam,lambda,Gam,gamma,std_ln_1,std_ln_2,nlos,std_shdw] = uwb_sv_params( cm_num );
```

```
[h_ct,t_ct,t0,np] = uwb_sv_model_ct( Lam, lambda, Gam, gamma, std_ln_1, std_ln_2, nlos, ...
    std_shdw, num_channels );
[hN,N] = uwb_sv_cnvrt_ct( h_ct, t_ct, np, num_channels, ts );
if N > 1,
    h = resample(hN, 1, N);  % decimate the columns of hN by factor N
else
    h = hN;
end
% correct for 1/N scaling imposed by decimation
h = h * N;

u = conv(b,h);

numsymbols = 100000;
minSNR = 0;
maxSNR = 30;
stepSNR = 3;

for SNRdb = minSNR:stepSNR:maxSNR
    SNRdb
    errs1 = 0;
    errs2 = 0;

    sigma_n = sqrt(Es*10^(-SNRdb/10));

    for ii = 1:numsymbols
        a = sign(randn);
        n = sigma_n*randn(size(u));
        r = a*u + n;
        a_hat_ml = sign(r*u');
        errs1 = errs1 + (a ~= a_hat_ml);

        n2 = sigma_n*randn(size(u));;
        u_tr = u + n2;
        a_hat_tr = sign(r*u_tr');
        errs2 = errs2 + (a ~= a_hat_tr);
    end
    ber_ml(SNRdb-minSNR+1) = errs1/numsymbols;
    ber_tr(SNRdb-minSNR+1) = errs2/numsymbols;
end
semilogy(minSNR:stepSNR:maxSNR, ber_ml(1:stepSNR:maxSNR-minSNR+1))
hold on
semilogy(minSNR:stepSNR:maxSNR, ber_tr(1:stepSNR:maxSNR-minSNR+1),'r')
xlabel('SNR␣(dB)')
```

```
ylabel('BER')
legend('MF␣Receiver␣Perfect␣CSIR','Transmit␣Reference␣Receiver')
hold off
grid minor
```

5.5 Noncoherent Receivers binary PPM - AWGN channel

```
close all
clear all
clc

ts = 0.1;
Tf = 100;
t_p = -Tf/2:ts:Tf/2-ts;
p = t_p.*exp(-8*t_p.^2);p = p/norm(p);
D_p = 2;

Nf = 4;
t_b = -Tf/2:ts:(Tf/2+(Nf-1)*Tf-ts);
b = kron(ones(1,Nf),p);
Es = norm(b)^2;

num_channels = 1;
cm_num = 1;  % channel model number from 1 to 4
% get channel model params based on this channel model number
[Lam,lambda,Gam,gamma,std_ln_1,std_ln_2,nlos,std_shdw] = uwb_sv_params
    ( cm_num );
[h_ct,t_ct,t0,np] = uwb_sv_model_ct( Lam, lambda, Gam, gamma, std_ln_1
    , std_ln_2, nlos, ...
    std_shdw, num_channels );
[hN,N] = uwb_sv_cnvrt_ct( h_ct, t_ct, np, num_channels, ts );
if N > 1,
    h = resample(hN, 1, N);  % decimate the columns of hN by factor N
else
    h = hN;
end
% correct for 1/N scaling imposed by decimation
h = h * N;

u(1,:) = conv(b,h);
N_ppm = Tf/ts/4;
u(2,:) = [u(1,N_ppm+1:end) u(1,1:N_ppm)];

numsymbols = 100000;
```

```
minSNR = 00;
maxSNR = 30;
stepSNR = 3;

ed_indices = [];
for ii = 1:Nf
    ed_indices = [ed_indices Tf/ts/2-D_p/ts:Tf/ts/2+D_p/ts];
end

for SNRdb = minSNR:stepSNR:maxSNR
    SNRdb
    errs1 = 0;
    errs2 = 0;
    errs3 = 0;

    sigma_n = sqrt(Es*10^(-SNRdb/10));

    for ii = 1:numsymbols
        a = sign(randn);
        n = sigma_n*randn(size(u(1,:)));
        r = u((a+1)/2+1,:) + n;

        a_hat_ml = -sign(r*(u(1,:) - u(2,:))');
        errs1 = errs1 + (a ~= a_hat_ml);

        n2 = sigma_n*randn(1,length(u(1,:))+N_ppm);;
        u_tr(1,:) = u(1,:) + n2(1:length(u(1,:)));
        u_tr(2,:) = u(2,:) + n2(N_ppm+(1:length(u(1,:))));
        a_hat_tr = -sign(r*(u_tr(1,:) - u_tr(2,:))');
        errs2 = errs2 + (a ~= a_hat_tr);

        a_hat_ed = sign(norm(r(ed_indices))^2 - norm(r(ed_indices-
            N_ppm))^2);
        errs3 = errs3 + (a ~= a_hat_ed);
    end
    ber_ml(SNRdb-minSNR+1) = errs1/numsymbols;
    ber_tr(SNRdb-minSNR+1) = errs2/numsymbols;
    ber_ed(SNRdb-minSNR+1) = errs3/numsymbols;
end
semilogy(minSNR:stepSNR:maxSNR, ber_ml(1:stepSNR:maxSNR-minSNR+1))
hold on
semilogy(minSNR:stepSNR:maxSNR, ber_tr(1:stepSNR:maxSNR-minSNR+1), 'r')
semilogy(minSNR:stepSNR:maxSNR,1-ber_ed(1:stepSNR:maxSNR-minSNR+1), 'k'
    )
xlabel('SNR␣(dB)')
```

```
ylabel('BER')
legend('MF Receiver Perfect CSIR','Transmit Reference Receiver','Energy Detector Receiver')
hold off
grid minor
```

Bibliography

[1] J. H. Reed, “An Introduction to Ultra Wideband Communication Systems,” *Prentice Hall*, 2003.

[2] M. B. Blanton, “An FPGA Software-Defined Ultra Wideband Transceiver,” *Msc Thesis, Virginia Polytechnic Institute and State University*, 2006.

[3] V. Lottici, A. D’Andrea, U. Mengali, ”Channel Estimation for Ultra-Wideband Communications,” *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 9, pp. 1638 - 1645, Dec. 2002.

[4] R. Miri, L. Zhou, P. Heydari, ”Timing synchronization in impulse-radio UWB: Trends and challenges,” *Circuits and Systems and TAISA Conference*, pp. 221 - 224, June 2008.

[5] K. Witrisal, G. Leus, G.J.M. Janssen, M. Pausini, F. Troesch, T. Zasowski, J. Romme, “Noncoherent Ultra-Wideband Systems:An overview of recent research activities,” *IEEE Signal Processing Magazine*, vol. 26, no. 4, pp. 48 - 66, July 2009.

[6] B. Schell, Y. Tsividis, “Analysis of Continuous-Time Digital Signal Processors,” *IEEE International Symposium on Circuits and Systems*, pp. 2232 - 2235, May 2007.

[7] S. Hoyos, B. M. Sadler, “Ultra-Wideband Analog-to-Digital Conversion Via Signal Expansion,” *IEEE Transactions on Vehicular Technology*, vol. 54, no. 5, pp. 1609 - 1622, Sept. 2005.

[8] S. Hoyos, B. M. Sadler, “Monobit Digital Receivers: Design, Performance, and Application to Impulse Radio,” *IEEE Transactions on Communications*, vol. 58, no. 6, pp. 1695 - 1704, June 20.

[9] J. L. Paredes, G. R. Arce, Z. Wang, "Ultra-Wideband Compressed Sensing:Channel Estimation," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 3, pp. 383 - 395, Oct. 2007.

[10] T. C.-K. Liu, X. Dong, W.-S. Lu, "Compressed Sensing Maximum Likelihood Channel Estimation for Ultra-Wideband Impulse Radio," in *Proc. IEEE International Conference on Communications*, pp. 1 - 5, June 2009.

[11] P. Zhang, Z. Hu, R. C. Qiu, B. M. Sadler "A Compressed Sensing Based Ultra-Wideband Communication System," in *Proc. IEEE International Conference on Communications*, pp. 1 - 5, June 2009.

[12] A. Oka, L. Lampe, "A Compressed Sensing Receiver for Bursty Communication with UWB Impulse Radio," in *Proc. IEEE International Conference on Ultra-Wideband*, pp. 279 - 284, Sept. 2009.

[13] E. J. Candès, M. B. Wakin, "An Introduction To Compressive Sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21 - 30, March 2008.

[14] S. Hoyos, B. M. Sadler, G R. Arce, "Monobit Digital Receivers for Ultrawideband Communications," *IEEE Transactions on Wireless Communications*, vol. 4, no. 4, pp. 1337 - 1344, July 2005.

[15] L. Ke, H. Yin, W. Gong, Z. Wang, "Finite-Resolution Digital Receiver Design for Impulse Radio Ultra-Wideband Communication," *IEEE Transactions on Wireless Communications*, vol. 7, no. 12, pp. 845 - 849, Dec. 2008.

[16] Q. Zhang, A. Nallanathan, H. K. Garg, "Monobit Digital Eigen-Based Receiver for Transmitted-Reference UWB Communications," *IEEE Transactions on Wireless Communications*, vol. 8, no. 5, pp. 2312 - 2316, May 2009.

[17] C. Vogel, H. Johansson, "Time-Interleaved Analot-To-Digital Converters: Status and Future Directions," in *International Symposium on Circuits and Systems*, pp. 3386 - 3389, May 2006.

[18] M. Mishali, Y. C. Eldar, "From Theory to Practice: Sub-Nyquist Sampling of Sparse Wideband Analog Signals," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 375 - 391, April 2010.

[19] M. Mishali, Y. C. Eldar, O. Dounaevsky, E. Shoshan, “Xampling: Analog to Digital at Sub-Nyquist Rates,” *IET Circuits, Devices & Systems*, vol. 5, no. 1, pp. 8 - 20, Jan. 2011.

[20] http://www.ieee802.org/15/pub/2003/Mar03/02490r1P802-15_SG3a-Channel-Modeling-Subcommittee-Report-Final.zip