QUADRATURE PHASE SHIFT KEYING-DIRECT SEQUENCE
SPREAD SPECTRUM-CODE DIVISION MULTIPLE ACCESS WITH
DISPARATE QUADRATURE CHIP AND DATA RATES

A thesis presented to
the faculty of
the Russ College of Engineering and Technology of
Ohio University

In partial fulfillment
of the requirements for the degree
Master of Science

Shweta S. Agarwal

March 2006

This thesis entitled

QUADRATURE PHASE SHIFT KEYING-DIRECT SEQUENCE SPREAD
SPECTRUM-CODE DIVISION MULTIPLE ACCESS WITH DISPARATE
QUADRATURE CHIP AND DATA RATES

by

SHWETA S. AGARWAL

has been approved for

the School of Electrical Engineering and Computer Science
and the Russ College of Engineering and Technology by

David W. Matolak

Associate Professor of Electrical Engineering and Computer Science

Dennis Irwin

Dean, Russ College of Engineering and Technology

Abstract

AGARWAL, SHWETA. M.S. March 2006. Electrical Engineering

Quadrature Phase Shift Keying-Direct Sequence Spread Spectrum-Code Division
Multiple Access with Disparate Quadrature Chip and Data Rates (111 pp.)

Director of Thesis: David W. Matolak

In this research, a DS-CDMA system with QPSK modulation and multiuser interference is analyzed and simulated. We investigate the BER performance for disparate data and chip rates on the I - and Q - channels for synchronous and asynchronous transmission using both orthogonal and random codes. Comparisons are made concerning the performance of the system in AWGN and Rayleigh fading channels. Power spectra for various cases are analyzed and the results show that we obtain moderate spectrum shaping and reduction in side lobes for some parameter sets. This thesis shows through using theoretical analysis and computer simulations that synchronous and asynchronous transmissions with random codes have similar performance. Our results also show that the BER is a function of the number of interfering users and processing gain, and that our disparate chip and data rate scheme presents another effective option for multi-rate transmission.

Approved:

David W. Matolak

Associate Professor of Electrical Engineering and Computer Science

Acknowledgements

The development of an individual as a scholar is not a process that can be achieved alone. The writing of a thesis is never an individual accomplishment. For this reason, I would like to acknowledge the assistance I received to achieve my goal while working on my thesis.

Primarily, I would like to thank Dr. David Matolak, my advisor, for providing me with guidance and assistance in the entire journey. He has been supportive, tolerant and lent me a patient ear at all times. He has been the guiding light that has ensured proper development of my thesis and made it the best it could be. He always provided me with priceless suggestions that I believe have considerably improved the quality of my thesis. It has been a pleasure to work with him. I am also grateful to Dr. Jeffrey Dill, Dr. Frank Van Graas, and Dr. Hans Kruse, for reviewing my thesis and providing me helpful suggestions.

I especially want to thank my father for his encouragement and belief in me. I would also take the opportunity to thank my mother, sister and brother in law, for their support. They have been a perennial source of sustenance, which helped me to pass through all those days when bogged down by small failures. I would also like to thank my friend Ganapathy Natarajan for reading and providing me critical suggestions on specific chapters and for his emotional support that enabled me to achieve my target.

Table of Contents

	Page
ABSTRACT.....	3
ACKNOWLEDGEMENTS	4
LIST OF TABLES.....	7
LIST OF FIGURES	8
CHAPTER 1:INTRODUCTION.....	10
1.1 BACKGROUND.....	10
1.2 MULTIPLE ACCESS.....	12
1.2.1 Frequency Division Multiple Access (FDMA).....	12
1.2.2 Time Division Multiple Access (TDMA).....	13
1.2.3 Code Division Multiple Access	13
1.3 QUADRATURE PHASE SHIFT KEYING SIGNALING.....	15
1.4 SPREAD SPECTRUM.....	17
1.4.1 Direct Sequence Spread Spectrum.....	18
1.4.2 Frequency Hopping Spread Spectrum	21
1.5 RAYLEIGH FADING CHANNEL	22
1.6 MULTICARRIER CDMA	25
1.7 LITERATURE REVIEW	26
1.8 THESIS OBJECTIVE	28
1.9 THESIS OUTLINE	29
CHAPTER 2:SYSTEM MODEL.....	30
2.1 SINGLE USER QPSK DS-SS.....	30
2.1.1 Single User QPSK DS-SS: I-channel Decision Statistics.....	33
2.1.2 Single User QPSK DS-SS: Q-channel Decision Statistics	34
2.2 MULTIUSER QPSK DS-SS ANALYSIS.....	35
2.2.1 Synchronous Transmission Analysis	35
2.2.2 Asynchronous Transmission Analysis.....	37
2.3 POWER SPECTRUM.....	38
CHAPTER 3:SYSTEM MODEL.....	45
3.1 SYNCHRONOUS QPSK	45
3.1.1 Analysis with Orthogonal Spreading Codes	46
3.1.2 Analysis with Random Spreading Codes: I-channel	47
3.1.3 Analysis with Random Spreading Codes: Q-channel.....	50
3.2 ASYNCHRONOUS QPSK.....	51
3.2.1 Multi-User QPSK DS-SS: I-channel Decision Statistics.....	52
3.2.2 Multi-User QPSK DS-SS: Q-channel Decision Statistics	58
3.3 PROBABILITY OF ERROR	61
3.3.1 Orthogonal Spreading Codes: Synchronous QPSK.....	63

3.3.2	Random Spreading Codes: I-Channel Synchronous QPSK.....	64
3.3.3	Probability of Error: Asynchronous QPSK.....	66
3.4	RAYLEIGH FADING.....	67
3.4.1	Single QPSK user with Rayleigh Fading.....	67
3.4.2	Multiuser QPSK with Rayleigh Fading.....	69
CHAPTER 4:SIMULATION DESCRIPTION AND RESULTS.....		76
4.1	SIMULATION TRANSMITTER DESCRIPTION.....	77
4.2	SIMULATION RESULTS	78
4.2.1	Single User.....	78
4.2.2	Synchronous QPSK	80
4.2.3	Asynchronous QPSK	85
4.2.4	BER with Varying System Load.....	89
CHAPTER 5:CONCLUSIONS AND FUTURE WORK.....		93
5.1	SUMMARY AND CONCLUSIONS.....	93
5.2	FUTURE WORK	94
REFERENCES.....		96
APPENDIX: MATLAB PROGRAMS.....		98

List of Tables

Table	Page
Table 3.1: Bit error rate for various QPSK scenarios with AWGN.....	74
Table 3.2: Bit error rate for various QPSK scenarios with Rayleigh fading	75

List of Figures

Figure	Page
Figure 1.1: Block diagram of a communication system	11
Figure 1.2: Frequency Division Multiple Access	13
Figure 1.3: Time Division Multiple Access.....	14
Figure 1.4: Code Division Multiple Access.....	14
Figure 1.5: QPSK transmitter	17
Figure 1.6: Bandwidth spreading.....	18
Figure 1.7: Basic block diagram to illustrate spread spectrum technique	20
Figure 1.8: Flat Rayleigh fading channel model.....	24
Figure 1.9: PDFs for Rayleigh fading.....	24
Figure 2.1: QPSK DS-CDMA single user transmitter block diagram.....	31
Figure 2.2: QPSK DS-SS CDMA receiver block diagram	33
Figure 2.3: Synchronous-multiuser QPSK DS-CDMA system block diagram.....	36
Figure 2.4: Asynchronous-multiuser QPSK DS-CDMA system block diagram.....	38
Figure 2.5: Power spectral density of QPSK Signal	39
Figure 2.6: Power spectrum with equal energy and $M = M_d = 1$	41
Figure 2.7: Normalized power spectrum with equal energy and $M = 4, M_d = 1$	41
Figure 2.8: Power spectrum with equal energy and $M = 4, M_d = 1$	42
Figure 2.9: Normalized power spectrum with equal energy and $M = 2, M_d = 1$	42
Figure 2.10: Normalized power spectrum with unequal energy and $M = 4, M_d = 3$	43
Figure 2.11: Normalized power spectrum with equal energy and $M = 2, M_d = 3$	43
Figure 2.12: Normalized power spectrum with unequal energy and $M = 2, M_d = 3$	44
Figure 3.1: Timing diagram for I- and Q-channel synchronous analysis	49
Figure 3.2: Timing diagram for I-channel analysis.	55
Figure 3.3: Timing diagram for Q-channel analysis.....	59
Figure 4.1: I-channel P_b vs. E_b/N_0 with single user.	79
Figure 4.2: Q-channel P_b vs. E_b/N_0 with single user.....	79
Figure 4.3: I-channel P_b vs. E_b/N_0 with multiple users and orthogonal codes.....	81
Figure 4.4: Q-channel P_b vs. E_b/N_0 with multiple users and orthogonal codes.	81
Figure 4.5: I-channel P_b vs. E_b/N_0 with 7 users and random codes.	82
Figure 4.6: Q-channel P_b vs. E_b/N_0 with 7 users and random codes.....	82
Figure 4.7: I-channel P_b vs. E_b/N_0 with 3 users and random codes.	83
Figure 4.8: Q-channel P_b vs. E_b/N_0 with 3 users and random codes.....	83
Figure 4.9: I-channel P_b vs. E_b/N_0 with 30 users and random codes.	84
Figure 4.10: Q-channel P_b vs. E_b/N_0 with 30 users and random codes.....	84
Figure 4.11: I-channel P_b vs. E_b/N_0 with 3 users and $PG^{(I)}=120$	86
Figure 4.12: Q-channel P_b vs. E_b/N_0 with 3 users and $PG^{(Q)}=10$	87
Figure 4.13: I-channel P_b vs. E_b/N_0 with 7 users and $PG^{(I)}=90$	87
Figure 4.14: Q channel P_b vs. E_b/N_0 with 7 users and $PG^{(Q)}=15$	88
Figure 4.15: I-channel P_b vs. E_b/N_0 with 3 users, $PG^{(I)}=20$ and interfering users energies uniformly distributed w.r.t. desired user energy.....	88

Figure 4.16: Q-channel P_b vs. E_b/N_0 with 3-users, $PG^{(Q)}=20$ and interfering users energies uniformly distributed w.r.t. desired user energy.....	89
Figure 4.17: I-channel BER vs. the number of users, AWGN channel.	90
Figure 4.18: I-channel BER vs. the number of users, Rayleigh channel.	91
Figure 4.19: Q-channel BER vs. the number of users, AWGN channel.	91
Figure 4.20: Q-channel BER vs. the number of users, Rayleigh channel.	92

Chapter 1: Introduction

1.1 Background

Digital Communication Systems (DCS) have become one of the most exciting areas within electrical engineering because of the ever growing demand for data communication, and because digital transmission offers data processing options and flexibility not available with analog transmission. Similar to analog communication systems, digital communication systems consist of various blocks performing specific functions. A communication system is required to transmit the information signal from source to destination. When the signal is transmitted over an electrical communication channel, because of the typically long distance between the source and destination, an unknown signal can interfere with the transmitted signal, particularly when transmission is wireless. Because of this, and due to unavoidable receiver thermal noise, the signal received may not be correct. The probability of error in the received signal and the transmission rate are generally used as performance measures of digital communications systems. The basic DCS block diagram is shown in Figure 1.1 [1] and a brief description is provided below.

Source: It is a device that generates information to be communicated by the DCS. The information source can be analog or discrete. The information should be converted into a form recognized by the communication system.

Source encoder/decoder: The output of the information source is given to the source encoder. The encoder assigns codewords to the symbols produced by the information source where codewords are typically a group of binary digits (bits). Source encoding is often “compression” encoding that removes redundancy in the source information, for more efficient transmission. At the receiver, the decoder is used to convert the binary output of the channel decoder into a symbol sequence.

Figure 1.1: Block diagram of a communication system

Channel encoder/ decoder: The output from the source encoder is given as input to the channel encoder, where some redundant bits are added to the source-encoded bit sequence in a predefined format. The decoder at the receiver uses these bits to reconstruct the information signal and reduce the effects of noise and distortion. This type of encoding is often termed channel coding, or forward error control (FEC) coding. In this case, the redundant bits serve to increase the reliability of the received signal. These extra bits carry no information, however, they are used by the decoder to detect and correct errors in the received signal.

Modulator/Demodulator: The modulator maps input digital sequences (often one's and zero's) to analog signal waveforms. Modulation “fits” the signal to the channel characteristics, and also can provide the capability to multiplex many signals for transmission (e.g., frequency division multiplexing). At the receiver, the demodulator converts the input modulated signal to the sequence of bits.

Communication Channel: The communication channel is the link between the source and the destination. The channel can be a wire, fiber optic cable or free space. In the case of the free space channel, transmission is possible using antennas. Some common problems associated with any communication channel are additive noise, signal attenuation, amplitude and phase distortion, and multipath distortion.

1.2 Multiple Access

The bandwidth available for a wireless system is limited. It is required by a wireless system to accommodate as many users as possible by efficiently sharing the limited bandwidth. Therefore, multiple access techniques are used, which allow users to simultaneously share the limited resources of bandwidth, time, etc., without performance degradation. There are three basic multiple access schemes [2]:

1. Frequency Division Multiple Access (FDMA)
2. Time Division Multiple Access (TDMA)
3. Code Division Multiple Access (CDMA)

1.2.1 Frequency Division Multiple Access (FDMA)

FDMA is the earliest multiple access technique. In this technique, the total bandwidth available to the system is divided into a number of channels and distributed among users. Each user has a finite amount of bandwidth for “permanent” use. Thus, sub-bands of frequencies are allocated to different users on a continuous time basis. Since each user has its portion of the bandwidth all the time, FDMA does not require any synchronization across different user signals. To reduce the interference between the adjacent channels, guard bands are used, which isolates the adjacent channels. The channels in FDMA are assigned only when demanded by the users. Therefore, the resource is in a sense wasted when the channel is not in use. The FDMA structure is shown in Figure 1.2.

1.2.2 Time Division Multiple Access (TDMA)

In digital systems, transmission of information is often not continuous since the users do not use the allotted bandwidth all the time. TDMA is a technique in which the entire bandwidth is available to the user only for a finite period of time. Thus, each user is given a particular time “slot,” usually periodically with some fixed period. To avoid interference between adjacent time slots, guard times are inserted between them. The TDMA structure is shown in Figure 1.3. The Global Systems for Mobile (cellular) communications (GSM) uses the TDMA technique.

1.2.3 Code Division Multiple Access

Code Division Multiple Access (CDMA), is in a sense “orthogonal” to TDMA and FDMA techniques. The structure for a CDMA system is shown in Figure 1.4.

Figure 1.2: Frequency Division Multiple Access

Figure 1.3: Time Division Multiple Access

Figure 1.4: Code Division Multiple Access

In CDMA, all the users occupy the same bandwidth at the same time; however, they are all assigned separate *codes*, which differentiate them from each other. CDMA systems utilize a spread spectrum technique. Direct Sequence Spread Spectrum (DSSS) is most commonly used for commercial CDMA. Each user in DSSS CDMA uses a codeword, which is optimally orthogonal to the codes of other users. Orthogonality may not be attainable in all circumstances. Each user data stream is modulated using a conventional modulation technique, such as BPSK or QPSK, and then this modulated signal is multiplied by the spreading code of the respective user. The receiver is required to know the code associated with the user for detection. Unlike TDMA, CDMA does not require time synchronization between the users. However, CDMA system may experience “self-jamming” which arises when the spreading codes used by different users are not perfectly orthogonal. This self-jamming is often termed multiuser interference (MUI) or multiple access interference (MAI).

1.3 Quadrature Phase Shift Keying Signaling

QPSK is a method for transmitting digital information across an analog channel in which both a cosine and sine carrier wave are varied in phase, keeping amplitude and frequency constant. In this modulation technique, two bits are transmitted in a single modulation symbol, resulting in four different symbols. The phase of the carrier takes one of the four possible values, such as 0 , $\pi/2$, π , $3\pi/2$, where each phase corresponds to a unique symbol.

The block diagram of a typical QPSK transmitter is shown in Figure 1.5. The input bit stream $d_b(t)$ is split into two bit streams, the in-phase and quadrature streams or *I*-channel and *Q*-channel bit streams. These two bit streams are then separately modulated by two carriers, which are in phase quadrature. Each modulated signal is a BPSK signal and is summed to produce a QPSK signal. At the receiver, a band pass filter is used to remove noise and adjacent channel interference. The filtered signal is then split into two parts, and each part is coherently demodulated using the in-phase and

quadrature carriers. The demodulator outputs are passed through a decision-making circuit that generates estimates of the in-phase and quadrature binary streams. These two streams are then multiplexed to reproduce the original message binary sequence.

In QPSK, two bits are transmitted in a single modulation symbol instead of one bit as for BPSK. Thus, the bandwidth efficiency of QPSK is twice that of BPSK. This is because the main lobe of the power spectral density of a QPSK signal, i.e., the null-to-null bandwidth, is equal to twice the symbol rate, which is half that of a BPSK signal. Furthermore, the bit error probability of QPSK is nearly identical to BPSK, while twice as much data can be sent in the same bandwidth. When there is no cross talk or interference between the two quadrature channels, for coherent detection, the bit error probability for QPSK is given as

$$P_b = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) \quad (1.1)$$

where E_b is the bit energy, N_0 is the one-sided noise spectral density and $Q(x)$ is defined as

$$Q(x) = \frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{x}{2}}\right) \text{ and } \operatorname{erfc}(u) = \frac{1}{\sqrt{2\pi}} \int_0^u e^{-z^2/2} dz$$

Figure 1.5: QPSK transmitter

1.4 Spread Spectrum

Spread Spectrum (SS) is a telecommunications technique in which the information signal is spread over a bandwidth considerably greater than necessary to resist jamming and other interference. This technique was initially devised for military use. There are a number of advantages of spread spectrum systems. Some of these are given as follows:

1. As the signal is spread over a large frequency band, the power spectral density is lowered and in the channel, the signal just looks like noise.
2. For narrowband jammers, spread spectrum improves anti-jamming performance.
3. Spread spectrum signals lower the probability of intercept by an unauthorized receiver
4. They allow multiple simultaneous transmissions, i.e., multiple access communications, via CDMA.

Some of the applications of spread spectrum technology include cellular telephones, wireless data transmission in wireless local area networks (WLANS) and

satellite communications. A frequency domain illustration of the spread spectrum concept is shown in Figure 1.6. There are different forms of spread spectrum, namely Direct Sequence Spread Spectrum (DS-SS), Frequency Hopping Spread Spectrum (FHSS), and time hopping. A brief explanation of DSSS and FHSS is provided below.

1.4.1 Direct Sequence Spread Spectrum

Direct sequence spread spectrum (DSSS) is a transmission technique in which a pseudorandom noise (PN) code, independent of the information data, is employed as a modulation waveform to spread the signal energy over a bandwidth much greater than the signal information bandwidth. At the receiver, the signal is despread using a synchronized replica of the pseudo-noise code. The fundamental principal of DSSS is that, in channels with narrowband noise, increasing the transmitted signal bandwidth results in an increased probability that the received information will be correct. This technique sacrifices bandwidth in order to gain signal to interference performance.

Figure 1.6: Bandwidth spreading

The pseudorandom noise sequence used for spreading the message signal is often a “noise like” signal, which is usually binary. Digital logic circuitry is typically used to generate a PN sequence, and the same circuitry may be used at the transmitter and receiver. One of the important parameters of a spread spectrum technique is the Processing Gain (PG). *PG* is defined as the ratio of the bandwidth of the spread signal to the bandwidth of the unspread data signal.

$$PG = \frac{\text{Spread Signal BW}}{\text{Unspread Signal BW}} \quad (1.2)$$

The basic characteristics of DSSS transmission are:

1. The bandwidth of the carrier modulated signal is much wider than the bandwidth of the transmitted data signal.
2. A pseudorandom sequence is used to convert the original data signal into a wideband signal.
3. The information is retrieved at the receiver by cross correlation of the wideband signal with a synchronously generated replica of the pseudorandom sequence.

The working of DS spread spectrum can be illustrated with the help of Figure 1.7.

Figure 1.7: Basic block diagram to illustrate spread spectrum technique

The input data signal $g(t)$ is spread using the PN spreading signal and the resulting signal is transmitted by modulating a carrier. At the receiver, the received signal is demodulated, then despread using the synchronized but independently generated PN sequence identical to that used at the transmitter.

$$\text{Transmitted baseband signal} \quad m(t) = g(t)c(t) \quad (1.3)$$

$$\text{Received baseband signal} \quad r(t) = m(t) + n(t) = g(t)c(t) + n(t) \quad (1.4)$$

$$\begin{aligned} \text{Despread baseband signal} \quad z(t) &= g(t)c^2(t) + n(t)c(t) \\ &= g(t) + n(t)c(t) \end{aligned} \quad (1.5)$$

where, $m(t)$ = Transmitted baseband signal

$g(t)$ = Baseband data signal

$c(t)$ = Spreading signal

$r(t)$ = Received signal

$n(t)$ = Additive White Gaussian Noise signal (AWGN)

$z(t)$ = Despread signal

In equation (3), the term $c(t)n(t)$ is the wideband spread noise. For optimal performance in AWGN, a correlator or a matched filter is used, which enables recovery of the data signal $g(t)$ from the received signal and filters out of band noise out. In the analysis, QPSK modulation is used for transmission of DS-SS. We use QPSK since, for a given bit rate and chip rate, QPSK allows larger processing gain than BPSK, and is also more spectrally efficient, as previously discussed. Most cellular DS-SS CDMA systems employ QPSK.

1.4.2 Frequency Hopping Spread Spectrum

Frequency hopping (FH) is a spread spectrum technique in which the data bits are transmitted in different frequency slots at different times. The total bandwidth of the output signal is equal to the sum of all the frequency slots, also called hops. Frequency hopping is usually pseudorandom and the sequence of hops is only known to the desired transmitter and the receiver. Unwanted receivers have to cover the complete output bandwidth to receive the frequency hop signal. Thus, in FHSS the carrier frequency hops “randomly” from one frequency to another. The symbol duration of the signal is T_s seconds and the frequency is shifted every T_c seconds. The modulation most commonly used with this technique is M-ary frequency shift keying.

In DS-SS the ability to overcome jamming is determined by the processing gain of the system. As the number of bits in the pseudorandom sequence is increased, increasing the rate of the spreading signal, the processing gain of the system increases and hence so does the bandwidth. However, there is a limitation to the physical devices that can generate the PN sequences. This limits the BW in case of DS-SS. This problem is more easily overcome in FHSS, since the instantaneous bandwidth of the FH signal is that of the information bandwidth, and carrier frequency oscillators are available that can

hop over a very wide band. In addition, FH is less affected by the near-far effect¹. This is because only a small number of frequency hops will be “blocked” by a near-by transmitter and the remaining hops can be used to recover the original data message. There are two basic types of frequency hop spread spectrum:

1. Slow frequency hopping
2. Fast frequency hopping

In case of slow FHSS $T_c \geq T_s$ and for fast FHSS $T_c < T_s$. Generally, fast FHSS gives improved performance in fading or in the presence of interfering signals.

1.5 Rayleigh Fading Channel

In wireless propagation channels, because of the obstacles and reflectors present, the transmitted signal arrives at the receiver over multiple paths. Such a phenomenon is called multipath propagation. Multipath channels are often described as having a Line-Of-Sight (LOS)—a direct connection between the transmitter and the receiver—or Non-Line-Of-Sight (NLOS), in which all the paths arrive after reflection, diffraction, or scattering. Multipath causes amplitude fluctuations, phase fluctuations, and time delay in the received signals. In many common wireless transmission scenarios, a receiver in motion relative to a transmitter may have no line-of-sight path between the antennas, which can induce multipath distortion. When the multipath signal waves received at the antenna are out of phase, a reduction of the signal strength at the receiver can occur. Rayleigh fading is often a good approximation for such realistic channel conditions. This statistical fading model arises from the Central Limit Theorem, in which a large number of approximately equal strength paths arrive at the receiver. The flat Rayleigh fading channel refers to a multiplicative distortion of the transmitted signal, i.e., $r(t)=s(t)h(t)+n(t)$, where $r(t)$ is the received waveform, $s(t)$ is the transmitted signal, $h(t)$ is the channel waveform and $n(t)$ is the noise. Rayleigh fading can therefore be

¹ A base station receives a stronger signal from a user near it than from a user near the cell boundary. This phenomenon of close users dominating the distant users is termed the *near-far effect*.

interpreted as multiple indirect paths between transmitter and receiver with no distinct dominant path. The fading channel can be represented as shown in Figure 1.8. The average signal to noise ratio at the receiver is given as, $\Gamma = E[\alpha^2]E_b/N_0$, where E_b/N_0 is the bit energy to noise ratio, and α is the Rayleigh random variable which defines amplitude variation of the signal. The Rayleigh distribution has probability density function (pdf) given by

$$p_{\alpha}(x) = \frac{x}{b_o} e^{\frac{-x^2}{2b_o}} \quad (1.6)$$

The average envelope power is $E[\alpha^2] = \bar{\Gamma} = 2b_o$. Therefore, for the squared envelope $\alpha^2(t)$ the probability density is given as the exponential pdf

$$p_{\alpha^2}(x) = \frac{1}{\bar{\Gamma}} e^{\frac{-x}{\bar{\Gamma}}} \quad (1.7)$$

A plot of the Rayleigh pdf is shown in Figure 1.9.

$$r(t) = \alpha(t)e^{-j\theta(t)}s(t) + n(t)$$

$\alpha(t)$ = gain of the channel

$\theta(t)$ = phase shift of the channel

$n(t)$ = AWGN

Figure 1.8: Flat Rayleigh fading channel model

Figure 1.9: PDFs for Rayleigh fading

1.6 *Multicarrier CDMA*

A discussion of multicarrier CDMA is provided for completeness of the topic. In the next section, we discuss the research work involved with different multicarrier CDMA systems that relates to our analysis of the disparate chip and data rate QPSK CDMA system.

There are mainly three multiple access schemes based on CDMA and OFDM² techniques namely, multicarrier CDMA (MC-CDMA), multicarrier DS CDMA (MC-DS-CDMA) and, multitone CDMA (MT-CDMA). These are also known as multicarrier CDMA schemes and are categorized as multicarrier modulation with frequency domain spreading and multicarrier modulation with time domain spreading [3]. While MC-CDMA uses frequency domain spreading, MC-DS-CDMA and MT-CDMA use time domain spreading.

In MC-CDMA, the original data stream is spread over different sub-carriers with a given spreading code in the frequency domain, i.e., each symbol of the data stream is distributed to multiple carrier frequencies, each of which corresponds to one chip of the spreading code. These chips modulate different sinusoidal sub-carriers. The MT-CDMA scheme converts the serial data into a parallel stream and spreads using a spreading code in the time domain. This technique has some advantages such as, it can support a larger number of users as compared to a DS-CDMA system (single carrier) in some cases, and it offers better resistance to frequency selective fading than MC-DS-CDMA thus providing better performance. The MC-DS-CDMA is a technique in which the data stream is serial to parallel converted to convert a high data rate stream to a low rate stream. The data symbols on each sub-channel are then spread with a user specific spreading code in the time domain, i.e., direct sequence on each channel.

² OFDM: Orthogonal Frequency Division Multiplexing is a multicarrier modulation scheme.

1.7 Literature Review

Several modifications of direct sequence code division multiple access (DS-CDMA) have been proposed for cellular and personal communication systems. The DS-CDMA technique has been attractive in 3rd generation cellular mobile systems and satellite communications. Two important requirements for application of such systems are flexibility and spectral efficiency. Some review of prior related research is provided in the following paragraphs.

Ottosson and Svensson [4] have studied the performance of different multi-rate schemes in DS-CDMA systems by considering different scenarios such as different spreading factors, mixed modulation and parallel channels. Many different modulation techniques were considered in their research namely, QAM, BPSK, QPSK and higher-order PSK modulation than QPSK (MPSK, $M > 4$). Investigation showed that the performance of QPSK is more efficient than other modulation techniques when considering mixed modulations and variable spreading ratios. Our analysis parallels this development as we study the performance of a DS-CDMA system with QPSK modulation by considering different chip and data rates on the two quadrature channels.

Matolak and Xiong have proposed a new modulation scheme called Spectrally-Shaped Generalized Multitone Direct-Sequence Spread Spectrum (S-MT-DS-SS) in [5]. This scheme has adjustable parameters on each sub carrier such as, frequency separation between sub carriers, signal amplitude, data rate, and chip rate. They showed that with variable parameters it is possible to achieve significant spectral shaping, reduced peak-to-average power ratio, (PAPR), and bit error performance similar to a conventional MT-DS-SS system. In our analysis with QPSK DS-CDMA systems we use variable parameters. Thus, it would be interesting to investigate spectral, (PAPR), and bit error performance for our system.

On one hand, DS-CDMA systems are attractive because of their characteristics which include a potential increase in capacity and robustness to multipath fading. On the other hand, DS-CDMA systems face hurdles such as degraded performance as the number of active users increase, and the near-far problem. Lupas and Verdu in [6] have presented a near-far resistant multiuser receiver such that the received signal is treated

not as noise but as the sum of all the asynchronously transmitted multiuser signals with known structure. They showed that maximum likelihood demodulation performed for each active user in an AWGN environment at the receiver gives significantly improved performance over the conventional correlator, however, the system complexity increases significantly. Also, Zou et. al. [7] in their work have studied the interference model and have performed outage analysis for DS-CDMA systems for small and large numbers of users. They have evaluated the interference power due to one interfering user by averaging the area under a pdf curve. They have used a Gamma distribution to estimate this area, and analyze system performance [7]. They show that in a multiuser CDMA system, due to handoff, the multiuser interference approaches a Gaussian random variable very fast as the number of users increases. Lee et. al. [8] in their paper have investigated a special case of QPSK PN modulation called WALSH-QPSK where the conventional PN sequence is coded with a Walsh sequence. They have analyzed the W-QPSK scheme for both coherent and non-coherent schemes, and have showed that by coding the PN sequence with Walsh codes lowers the envelope variation as compared to using conventional PN sequences. Also, Kusaka et. al., in their paper [9] have proposed a method to improve the performance characteristics of a quadrature phase shift key code-division multiple-access (CDMA/QPSK) system in a nonlinear channel, for example a satellite repeater which contains a traveling-wave tube amplifier.

Tseng and Bell [10] have proposed using a set of multiple spreading sequences for each user in a MC-CDMA system. They considered users with different spreading sequences and a different spreading sequence for each carrier of every user. They concluded that on a non-fading channel with orthogonal sequences, the multiple access interference is minimal. In addition, the performance on a Rayleigh fading channel is similar to that of using same sequence for all the users.

We know that processing gain is the ratio of spread bandwidth to the unspread bandwidth, which defines the amount of improvement in performance through spread spectrum. If the bandwidth for transmission were fixed, an increase in input data rate would decrease the processing gain, thus degrading the performance. Wong and Leung in their paper [11] show that a M -ary DSSS scheme with M different codes obtained by

shifting the same PN sequence, can increase the transmission efficiency of spread spectrum systems and also overcome the processing gain vs. data rate limitation. This technique has been named as M -ary Code Shift Keying.

Thus, we see that there has been investigation of various versions of DS-CDMA systems. However, most DS-CDMA systems to date have been proposed such that only one of the many allowable parameters is variable. That is, DS-CDMA system that have the same chip rate, or the same data rate, or the same processing gain on both the channels of a QPSK system are only considered. Future communications systems, however, should be more flexible. Our disparate chip/data rate DS-CDMA technique is a new concept and requires extensive research in terms of bit error performance and power spectrum. Thus, the logical step would be to design a system with *all* these parameters being variable, allowing even more flexibility in the system. Our work investigates this extension to system flexibility, and hence fills a gap in the literature on such systems.

1.8 Thesis Objective

The objective of this research is to examine the performance of a QPSK DS-CDMA system with disparate chip and data rates on the two quadrature channels. By mathematical analysis and simulation, the performance of the system in presence of interfering users in AWGN and Rayleigh fading channels is evaluated. Two types of transmission schemes, i.e., synchronous (with equal time delays and phase shifts for all user signals) and asynchronous (with random unequal delays and phase shifts) are considered. This research addresses the effect of disparate chip and data rates (thus different processing gains) and different user signal energies on the I - and Q -channel's performance. We provide plots of power spectra for various conditions to illustrate main lobe bandwidth, spectral efficiency, and spectrum shaping. In addition, simulation and analytical results are presented to compare the performance of the systems in situations with a single user, multiple users, the same/different data rates, and the same/different chip rates. To summarize, the following conditions or cases of disparate chip/data rate systems are considered in this work:

- Synchronous QPSK DS CDMA system with orthogonal spreading codes
- Synchronous QPSK DS CDMA system with random PN codes
- Asynchronous QPSK DS CDMA system with random PN codes
- Variable numbers of interfering users
- Variable energy for each user signal on each channel
- Analysis in AWGN and Rayleigh channels

1.9 Thesis Outline

The organization of the thesis is as follows.

Chapter 2 This chapter presents the system model for a QPSK DS-CDMA system and a brief description of the system and parameters. It also presents expressions for the signal waveforms and discusses the different issues considered in the research. In addition, power spectrum results are also shown for different cases.

Chapter 3 In this chapter, we provide analytical results for the system in the presence of AWGN and Rayleigh fading channels in terms of bit error rate (BER), for numerous system parameter variations.

Chapter 4 We discuss the findings by comparing the analytical results with the results of computer simulations. Various simulation graphs are provided for scenarios with different numbers of interfering users, different ratios of chip and data rates on the I- and Q-channels, and different processing gains. In addition, the impact on performance of the system as the number of users increase is discussed.

Chapter 5 This chapter concludes the thesis, summarizes the results and briefly presents some suggestions for the future work.

Chapter 2: System Model

In this chapter, we provide a description of the system model used for this research. Quadrature Phase Shift Keying (QPSK) Direct Sequence Spread Spectrum (DS-SS) systems with a single user and with more than one user (multi-user) are described. The transmitter and receiver block diagrams for both these cases, single- and multi-user, are provided. Also, we discuss synchronous and asynchronous transmission for the multiuser case. The chapter ends with the discussion of power spectra. Different ratios of I - and Q -channel data rate to chip rate have been considered, and example spectra for several cases are shown.

2.1 Single User QPSK DS-SS

The single user case is the simplest case we can study. The block diagram of the transmitter for this case is shown in Figure 2.1. It is seen that the input binary (bipolar NRZ) data signal $d(t)$ is serial-to-parallel converted into two bipolar NRZ signals. This generalized serial to parallel converter divides the bipolar waveform into an in-phase bit stream $I(t)$ and a quadrature-phase bit stream $Q(t)$, for the I - and Q -channels, respectively. A pseudo-noise (PN) sequence is needed for both channels, and these are denoted $C^I(t)$ and $C^Q(t)$. These are often generated separately using two independent pseudo-noise sequence generators. The product of the PN sequence and the data signal, which is the output of the multiplier, is the baseband direct sequence spread signal. These signals on each channel then modulate the carriers, resulting in signals $s^I(t)$ and $s^Q(t)$. The transmitted direct sequence QPSK signal, the sum of these two signals, is represented as $s(t)$.

The data rate of the signal $d(t)$ is R_b , and the in-phase and quadrature phase bit rates are R_s^I and R_s^Q respectively, where $R_b = R_s^I + R_s^Q$. The spreading waveforms $C^I(t)$

and $C^Q(t)$ have chip rates R_c^I and R_c^Q for the in-phase and quadrature-phase components, respectively. As noted previously, one of the figures of merit for a direct sequence spread spectrum is the Processing Gain (PG). PG is defined as the ratio of the bandwidth of the spread message signal to the bandwidth of unspread data signal.

Figure 2.1: QPSK DS-CDMA single user transmitter block diagram

For our generalized $DS/QPSK$ system, two processing gains can be defined, one for each channel (I and Q). Different chip and data rates are assumed on the I - and Q -channels such that the chip rate on the I -channel is an integer multiple of that of the Q -channel, and the data rate on the Q -channel is an integer multiple of the data rate on the I -channel. We denote the processing gain on the I -channel as PG^I , and on the Q -channel as PG^Q . The relations between the various data and chip rates are provided below.

$$R_c^{(I)} = MR_c^{(Q)} \quad (2.1)$$

$$R_s^{(Q)} = M_d R_s^{(I)} \quad (2.2)$$

where $M > 1$, and $M_d > 1$. (Note we could have also used $M > 1$ and $M_d < 1$, but our results are generalizable to allow inclusion of this case.)

$$PG^I = \frac{R_c^I}{R_s^I} \quad (2.3)$$

$$PG^Q = \frac{R_c^Q}{R_s^Q} = \frac{PG^I}{MM_d} \quad (2.4)$$

Also, it is assumed that the signal energy is different on the I - and Q -channels. The expression for the transmitted signal $s(t)$ (see Figure (2.1)) can be written as follows:

$$s^{(I)}(t) = \sqrt{\frac{2E_b^{(I)}}{T_s^{(I)}}} I(t)C^{(I)}(t) \cos \omega_c t \quad (2.5)$$

$$s^{(Q)}(t) = -\sqrt{\frac{2E_b^{(Q)}}{T_s^{(Q)}}} Q(t)C^{(Q)}(t) \sin \omega_c t \quad (2.6)$$

$$\begin{aligned} s(t) &= s^{(I)}(t) + s^{(Q)}(t) \\ &= \sqrt{\frac{2E_b^{(I)}}{T_s^{(I)}}} I(t)C^{(I)}(t) \cos \omega_c t - \sqrt{\frac{2E_b^{(Q)}}{T_s^{(Q)}}} Q(t)C^{(Q)}(t) \sin \omega_c t \end{aligned} \quad (2.7)$$

The received signal, $r(t) = s(t) + n(t)$, is degraded by noise. As is typical, the noise signal is assumed to be white and Gaussian. The received signal is applied to the local multipliers, which are supplied with the locally generated coherent carriers. Subsequent to coherent down conversion, the signal on each channel is despread by correlating with the corresponding spreading waveforms, resulting in two quadrature terms, z^I and z^Q . The two bit streams are then multiplexed to get the final output bit stream using a generalized serial to parallel converter which incorporates a decision block as well. The receiver block diagram for a single user is shown in Figure 2.2. The receiver input $r(t)$ is given by

$$r(t) = s(t) + n(t) \quad (2.8)$$

where $n(t)$ is the AWGN of zero mean and two-sided power spectral density $N_0/2$. We use coherent detection at the receiver. Since the received signals are coherently detected, the I - and Q -channel phases are known at the receiver. The decision statistics for each channel are derived separately and are shown in the following section.

Figure 2.2: QPSK DS-SS CDMA receiver block diagram

2.1.1 Single User QPSK DS-SS: I -channel Decision Statistics

For the I -channel, without loss of generality, we have the correlator output for the m^{th} bit given by

$$\begin{aligned}
z^{(I)} &= \int_m^{(m+1)T_s^{(I)}} r(t) \cos \omega_c t C^{(I)}(t) dt \\
&= \int_m^{(m+1)T_s^{(I)}} [s(t) + n(t)] \cos \omega_c t C^{(I)}(t) dt \\
&= \sqrt{\frac{2E_b^{(I)}}{T_s^{(I)}}} \int_m^{(m+1)T_s^{(I)}} [I(t)C^{(I)}(t) \cos \omega_c t - Q(t)C^{(Q)}(t) \sin \omega_c t] [\cos \omega_c t C^{(I)}(t)] + N_I
\end{aligned}$$

where N_I is the wideband noise sample. Continuing, we have

$$z^{(I)} = \sqrt{\frac{2E_b^{(I)}}{T_s^{(I)}}} \int_m^{(m+1)T_s^{(I)}} [I(t)C^{(I)2}(t) \cos^2 \omega_c t - Q(t)C^{(Q)}(t)C^{(I)}(t) \cos \omega_c t \sin \omega_c t] + N_I$$

Now, using $\int_m^{(m+1)T_s^{(I)}} \cos^2 \omega_c t = \frac{1}{2} \int_m^{(m+1)T_s^{(I)}} (1 + \cos 2\omega_c t)$ and $C^{(I)2}(t) = 1$, we obtain

$$\begin{aligned}
z^{(I)} &= \frac{1}{2} \sqrt{\frac{2E_b^{(I)}}{T_s^{(I)}}} [I(m)T_s^{(I)}] + N_I \\
&= I(m) \sqrt{\frac{E_b^{(I)}T_s^{(I)}}{2}} + N_I
\end{aligned} \tag{2.9}$$

The “cross-term” involving $C^{(Q)}(t)$ and $C^{(I)}(t)$ is easily shown to be zero when $f_c \gg R_c$ by virtue of the orthogonality of the sinusoids. The noise term N_I is zero mean, since $n(t)$ is zero mean, and its variance is $N_0 T_s^I / 4$.

2.1.2 Single User QPSK DS-SS: Q-channel Decision Statistics

Analysis on the Q -channel is performed in a similar way as on the I -channel. Therefore, the output on the Q -channel can be written as:

$$\begin{aligned}
z^{(Q)} &= \int_m^{(m+1)T_s^{(I)}} r(t) \sin \omega_c t C^{(Q)}(t) dt \\
&= \int_m^{(m+1)T_s^{(I)}} [s(t) + n(t)] \sin \omega_c t C^{(Q)}(t) dt \\
&= \frac{1}{2} \sqrt{\frac{2E_b^{(Q)}}{T_s^{(Q)}}} [Q(m)T_s^{(Q)}] + N_Q \\
z^{(Q)} &= Q(m) \sqrt{\frac{E_b^{(Q)} T_s^{(Q)}}{2}} + N_Q
\end{aligned} \tag{2.10}$$

where N_Q is independent of N_I , with the same mean and variance $N_0 T_s^{(Q)}/4$.

2.2 Multiuser QPSK DS-SS Analysis

In a multiple user system, we have K users present, where $(K-1)$ users act as interference to any given user. The quality of the information signal degrades due to presence of multiuser interference in addition to the additive white Gaussian noise. The signal energy is assumed different for all users on the I - and Q -channels. In addition, different chip and data rates are assumed on the I and Q channels. However, the chip and data rates on the $I(Q)$ -channel is the same for all users. The relations between the chip rate, data rate and processing gain of the I and Q channels are the same as shown in equations (2.1) to (2.4). Two different scenarios have been considered for analysis: synchronous and asynchronous. Both these cases are discussed in detail in the following sections.

2.2.1 Synchronous Transmission Analysis

In the synchronous case, K user signals are considered that are QPSK modulated and added along with the noise to a desired (k^{th}) signal. The k^{th} user signal is considered the desired signal and all the remaining users act as interference and are indexed with a

subscript $i \in \{1, 2, \dots, K\}$. In the synchronous case, the delay introduced by the channel on all the users is assumed to be 0, and the carrier phases for all users are identical. Without loss of generality, we assume these phases to be zero. Two different scenarios for synchronous transmission are considered: with orthogonal spreading codes and random PN codes. For a synchronous system on the AWGN channel, orthogonal codes would most likely be used, as they yield zero MUI; we study the use of random codes for completeness, and for additional validation of our computer simulations. The transmitter block diagram for a multiuser system for the synchronous case is shown in Figure 2.3. This case would represent transmission from a common point, e.g., from a cellular base station.

Figure 2.3: Synchronous-multiuser QPSK DS-CDMA system block diagram

The output of the transmitter is represented as a sum of signals of the form in (2.7). The received signal $r(t)$ is given as

$$r(t) = s_k(t) + \sum_{\substack{i=1 \\ i \neq k}}^K s_i(t) + n(t) \quad (2.11)$$

As shown in the receiver block diagram in Figure 2.2, the received signal is demodulated, multiplied by the PN sequence, and integrated to generate the I and Q channel decision variables. Coherent detection is used at the receiver so that carrier phase information is known at the receiver. Analysis of the received signal for the in-phase and quadrature-phase channels for the synchronous case with orthogonal and random codes is performed separately and is shown in Chapter 3.

2.2.2 Asynchronous Transmission Analysis

For the asynchronous case, a distortionless AWGN channel is initially considered for all users, with random delays. The channel impulse response for the i^{th} user is $h_i(t) = \delta(t - \tau_i)$. As in the synchronous case, in the analysis of the multiuser system, we consider the k^{th} user signal the desired signal and all the remaining users act as interference and are indexed with a subscript i . The k^{th} user signal has known delay $\tau_k = 0$ at the receiver and the other users have arbitrary delays $\{\tau_i\}$. The block diagram for a multiuser system for the asynchronous case is shown in Figure 2.4. Analysis for the asynchronous case is similar to that of the synchronous case. The transmitter output for the i^{th} user can be represented as

$$\begin{aligned} s_i(t - \tau_i) = & \sqrt{\frac{2E_i^{(I)}}{T_s^{(I)}}} I_i(t - \tau) C_i^{(I)}(t - \tau) \cos(\omega_c t - \phi_i) \\ & - \sqrt{\frac{2E_i^{(Q)}}{T_s^{(Q)}}} Q_i(t - \tau) C_i^{(Q)}(t - \tau) \sin(\omega_c t - \phi_i) \end{aligned} \quad (2.12)$$

where $\phi_i = \omega_c \tau_i$ is modeled as uniform on $[0, 2\pi)$ and τ_i is uniform on $[0, T_s^I]$

The received signal $r(t)$ is given as

$$r(t) = s_k(t) + \sum_{\substack{i=1 \\ i \neq k}}^K s_i(t - \tau_i) + n(t) \quad (2.13)$$

Analysis of the received signal for the in-phase and quadrature-phase channels is performed separately, and is shown in Chapter 3.

Figure 2.4: Asynchronous-multiuser QPSK DS-CDMA system block diagram

2.3 *Power Spectrum*

The power spectrum is a plot of a part of the signal's power spectral density over a given frequency range. The power spectrum does not give any spatial or phase angle information. The power spectral density of a conventional QPSK signal with rectangular pulses is shown in Figure 2.5. As seen from the figure, the power ratio between the first

side lobe and the main lobe is -13 dB and a large percentage of the signal's energy (more than 90%) is concentrated in the main lobe.

Figure 2.5: Power spectral density of QPSK Signal

In the analysis, we have considered different data rates, different chip rates and different energies for each channel. Power spectra for several different parameter sets have been plotted and are shown in the following section all for a single transmitted signal. The four example cases considered are as follows:

1. $M = M_d = 1$, Equal energy on each channel
2. $M > 1$, $M_d = 1$, Equal energy on each channel
3. $M > 1$, $M_d > 1$, Equal energy on each channel
4. $M > 1$, $M_d > 1$, Unequal energy on each channel

1. $M = M_d = 1$, Equal bit energy on each channel

In this case, the chip and the data rates on I - and Q -channel are equal, i.e., $R_c^{(I)} = R_c^{(Q)}$, $R_s^{(I)} = R_s^{(Q)}$. The power spectrum when the energy on each channel is assumed to be equal is shown in Figure 2.6; this spectrum is identical to that of Figure 2.5.

2(a). $M = 4, M_d = 1$, Equal bit energy on each channel

In this case, the I -channel chip rate is four times that of Q -channel. The data rate on both channels is equal, i.e., $R_c^{(I)} = 4R_c^{(Q)}$, $R_s^{(I)} = R_s^{(Q)}$. The power spectrum with equal bit energy on each channel without normalizing and with normalizing the power is shown in Figure 2.7 and 2.8 respectively. We normalize the power for all the other cases considered.

2(b). $M = 2, M_d = 1$, Equal bit energy on each channel

In this case, the I -channel chip rate is twice the Q -channel chip rate. The data rates on both channels are equal, i.e., $R_c^{(I)} = 2R_c^{(Q)}$, $R_s^{(I)} = R_s^{(Q)}$. The power spectrum with equal bit energies on each channel is shown in Figure 2.9.

3. $M = 4, M_d = 3$, Equal energy on each channel

The I -channel chip rate is four times that of Q -channel, and the data rate on the Q -channel is three times the data rate on I -channel, i.e., $R_c^{(I)} = 4R_c^{(Q)}$, $R_s^{(Q)} = 3R_s^{(I)}$. The power spectrum with equal bit energy on each channel is shown in Figure 2.10.

3, 4. $M = 2, M_d = 3$, Equal/Unequal energy on each channel

The I -channel chip rate is twice the Q -channel chip rate, and the data rate on the Q -channel is three times the data rate on the I -channel, i.e., $R_c^{(I)} = 2R_c^{(Q)}$, $R_s^{(Q)} = 3R_s^{(I)}$. The power spectrum with equal bit energy on each channel is shown in Figure 2.11, and with unequal bit energies on each channel is shown in Figure 2.12. The bit energy on the Q -channel is five times that on I -channel.

Figure 2.6: Power spectrum with equal energy and $M = M_d = 1$

Figure 2.7: Normalized power spectrum with equal energy and $M = 4$, $M_d = 1$

Figure 2.8: Power spectrum with equal energy and $M = 4$, $M_d = 1$

Figure 2.9: Normalized power spectrum with equal energy and $M = 2$, $M_d = 1$

Figure 2.10: Normalized power spectrum with unequal energy and $M = 4$, $M_d = 3$

Figure 2.11: Normalized power spectrum with equal energy and $M = 2$, $M_d = 3$

Figure 2.12: Normalized power spectrum with unequal energy and $M = 2$, $M_d = 3$

In Figures 2.6 - 2.12, there are three power spectra, the I -channel, Q -channel and the composite spectrum. We see that as we change the parameter M (i.e., R_c^I/R_c^Q) the spectrum changes. As seen from Figures 2.7 - 2.12, the I -channel chip rate is M times the Q -channel chip rate. The composite waveform is the resultant of the I -channel and Q -channel spectra. We see that by varying the parameters we obtain a moderate amount of spectral shaping and in some cases a reduction in the side lobes (depending on how these are defined). Additional power spectra could also be obtained by varying the parameters.

In summary, a Quadrature Phase Shift Keying Direct Sequence Spread Spectrum system model has been described in this chapter. Different schemes for which we can analyze the system performance were discussed. The single user, multiuser with synchronous transmission, and multiuser with asynchronous transmission cases were considered. The transmitter and receiver block diagrams and power spectra for several example cases were provided. In the next chapter, we determine analytically the system performance of all the cases discussed in the AWGN and Rayleigh fading channels.

Chapter 3: System Model

This chapter describes the I - and Q -channel decision statistics with multiuser interference for synchronous and asynchronous QPSK DS CDMA systems. The performance of the receiver is measured in terms of bit error probability. The chapter concludes with the discussion of error rate for single/multi user schemes for synchronous/asynchronous transmission.

As seen in Chapter 2, the received signal is demodulated and despread with the same PN sequence used at the transmitter. Despreading is completed by a correlator which results in outputs z^I and z^Q . These are the I - and the Q -channel decision statistics, required to obtain the output bit estimates on each channel. Analysis for synchronous and asynchronous cases is similar to that of the single user QPSK DSSS scenario shown in Chapter 2. The analyses for both these cases are discussed in detail in the following section.

3.1 Synchronous QPSK

For analysis, K QPSK modulated signals are assumed where the arbitrary k^{th} user signal is considered the desired signal and the remaining $K-1$ user signals act as interference to the desired signal. In the synchronous case, for the non-dispersive channels we address, the delay introduced by the channel on all the user signals is identical. The received signal equation from Chapter 2 is

$$r(t) = s_k(t) + \sum_{\substack{i=1 \\ i \neq k}}^K s_i(t) + n(t)$$

where

$$s_k(t) = \sqrt{\frac{2E_k^{(I)}}{T_s^{(I)}}} I_k(t) C_k^{(I)}(t) \cos \omega_c t - \sqrt{\frac{2E_k^{(Q)}}{T_s^{(Q)}}} Q_k(t) C_k^{(Q)}(t) \sin \omega_c t$$

for $k \in \{1, 2, \dots, K\}$.

Analysis of the received signal statistics for the in-phase and quadrature-phase channels is performed separately for two cases: orthogonal spreading codes and random spreading codes.

3.1.1 Analysis with Orthogonal Spreading Codes

The analysis for the I -channel decision statistics with orthogonal codes is shown here. With orthogonal codes, the cross correlation between the k^{th} user's spreading code and the i^{th} user's spreading code is zero. The I channel correlator output is given as

$$\begin{aligned}
 z^{(I)} &= \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} r(t) \cos \omega_c t C_k^{(I)}(t) dt \\
 &= \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} [s_k(t) + \sum_{\substack{i=1 \\ i \neq k}}^M s_i(t) + n(t)] \cos \omega_c t C_k^{(I)}(t) dt \\
 &= \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} \left(\sqrt{\frac{2E_k^{(I)}}{T_s^{(I)}}} I_k(t) C_k^{(I)}(t) \cos \omega_c t - \sqrt{\frac{2E_k^{(Q)}}{T_s^{(Q)}}} Q_k(t) C_k^{(Q)}(t) \sin \omega_c t \right) (\cos \omega_c t C_k^{(I)}(t)) dt \\
 &\quad + \sum_{\substack{i=1 \\ i \neq k}}^K \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} \left(\sqrt{\frac{2E_i^{(I)}}{T_s^{(I)}}} I_i(t) C_i^{(I)}(t) \cos(\omega_c t) \right) (\cos \omega_c t C_k^{(I)}(t)) dt \\
 &\quad - \sum_{\substack{i=1 \\ i \neq k}}^K \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} \sqrt{\frac{2E_i^{(Q)}}{T_s^{(Q)}}} Q_i(t) C_i^{(Q)}(t) \sin(\omega_c t) (\cos \omega_c t C_k^{(I)}(t)) dt + N_I
 \end{aligned} \tag{3.1}$$

where N_I is the AWGN noise sample with zero mean and variance $N_0 T_s / 4$. We simplify the above equation using the following relationships:

$$C_k^{(I)^2}(t) = 1; \quad \int_m^{(m+1)T_s^{(I)}} C_i^{(I)}(t) C_k^{(I)}(t) dt = 0$$

Then we get

$$z^I = \sqrt{\frac{E_k^{(I)} T_s^{(I)}}{2}} I_k(m) + N_I \quad (3.2)$$

which shows that by using synchronized orthogonal codes, we have no MUI. As found for the I -channel, the Q -channel correlator output can also be written as

$$z^Q = \sqrt{\frac{E_k^{(Q)} T_s^{(Q)}}{2}} Q_k(m) + N_Q \quad (3.3)$$

3.1.2 Analysis with Random Spreading Codes: I -channel

The analysis for the I -channel decision statistics is now shown. The I channel correlator output is given by

$$\begin{aligned} z^{(I)} &= \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} r(t) \cos \omega_c t C_k^{(I)}(t) dt \\ &= \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} [s_k(t) + \sum_{\substack{i=1 \\ i \neq k}}^M s_i(t) + n(t)] \cos \omega_c t C_k^{(I)}(t) dt \\ &= \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} \left(\sqrt{\frac{2E_k^{(I)}}{T_s^{(I)}}} I_k(t) C_k^{(I)}(t) \cos \omega_c t \right) (\cos \omega_c t C_k^{(I)}(t)) dt \\ &\quad - \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} \left(\sqrt{\frac{2E_k^{(Q)}}{T_s^{(Q)}}} Q_k(t) C_k^{(Q)}(t) \sin \omega_c t \right) (\cos \omega_c t C_k^{(I)}(t)) dt \\ &\quad + \sum_{\substack{i=1 \\ i \neq k}}^K \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} \left(\sqrt{\frac{2E_i^{(I)}}{T_s^{(I)}}} I_i(t) C_i^{(I)}(t) \cos(\omega_c t) \right) (\cos \omega_c t C_k^{(I)}(t)) dt \\ &\quad - \sum_{\substack{i=1 \\ i \neq k}}^K \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} \left(\sqrt{\frac{2E_i^{(Q)}}{T_s^{(Q)}}} Q_i(t) C_i^{(Q)}(t) \sin(\omega_c t) \right) (\cos \omega_c t C_k^{(I)}(t)) dt + N_I \end{aligned} \quad (3.4)$$

where N_I is the AWGN noise sample with zero mean and variance $N_0 T_s / 4$. In (3.4), the “cross” terms are an integration of the product of a sine and a cosine of identical frequency and phase; hence the integration of this product over a chip duration is 0. (Procedurally, we actually break the symbol-duration integral into a sum of chip-duration integrals, and all terms except the sinusoids are taken outside the integral.) Hence,

$$z^{(I)} = \frac{1}{2} \left[\sqrt{\frac{2E_k^{(I)}}{T_s^{(I)}}} (I_k(m) T_s^{(I)}) + \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{2E_i^{(I)}}{T_s^{(I)}}} \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} I_i(t) C_i^{(I)}(t) C_k^{(I)}(t) dt \right] + N_I \quad (3.5)$$

From (3.5) it is seen that, $z^{(I)}$ can be expressed as a sum of three terms: $z^{(I)} = \text{Desired Signal} + \text{AWGN} + \text{MUI}$. To continue, we represent the MUI term as

$$X^{(I)} = \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(I)}}{2T_s^{(I)}}} \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} [I_i(t) C_i^{(I)}(t) C_k^{(I)}(t)] dt$$

Figure 3.1 shows the timing diagram for a single symbol on both the I - and Q - channels. Referring to Figure 3.1(a) for the integration of the chip sequence on the I - and Q - channels over one symbol period, we can express the MUI term as in (3.6).

Figure 3.1: Timing diagram for I- and Q-channel synchronous analysis

$$X^{(I)} = \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(I)} T_s^{(I)}}{2}} \left(I_i(m) \frac{1}{PG^I} \sum_{n=0}^{PG^I-1} C_k^{(I)}(n) C_i^{(I)}(n) \right) \quad (3.6)$$

Thus, using (3.6) in (3.5) we have

$$\begin{aligned} z^I &= \sqrt{\frac{E_k^{(I)} T_s^{(I)}}{2}} I_k(m) + \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(I)} T_s^{(I)}}{2}} \left(I_i(m) \frac{1}{PG^I} \sum_{n=0}^{PG^I-1} C_k^{(I)}(n) C_i^{(I)}(n) \right) + N_I \\ &= \sqrt{\frac{E_k^{(I)} T_s^{(I)}}{2}} I_k(m) + \sum_{\substack{i=1 \\ \neq k}}^K I_i(m) \sqrt{\frac{E_i^{(I)} T_s^{(I)}}{2}} \rho_{k,i}^{(I)} + N_I \end{aligned} \quad (3.7)$$

where $\rho_{k,i}^{(I)}$ is the cross correlation between user k 's and user i 's I -channel spreading codes and $0 \leq |\rho_{k,i}^{(I)}| \leq 1$; $0 \leq |\rho_{k,i}^{(Q)}| \leq 1$. We see that, as the cross correlation increases, the multiuser interference increases.

3.1.3 Analysis with Random Spreading Codes: Q -channel

As found for the I -channel, the Q -channel correlator output can be written as

$$z^{(Q)} = \sqrt{\frac{E_k^{(Q)}}{2T_s^{(Q)}}} Q_k(m) T_s^{(Q)} + \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(Q)}}{2T_s^{(Q)}}} \int_{mT_s^{(Q)}}^{(m+1)T_s^{(Q)}} Q_i(t) C_i^{(Q)}(t) C_k^{(Q)}(t) dt + N_Q \quad (3.8)$$

We let the MUI term here be

$$\begin{aligned} Y^Q &= \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(Q)}}{2T_s^{(Q)}}} \int_{mT_s^{(Q)}}^{(m+1)T_s^{(Q)}} Q_i(t) C_i^{(Q)}(t) C_k^{(Q)}(t) dt \\ &= \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(Q)} T_s^{(Q)}}{2}} \left[Q_i(m) \frac{1}{PG^{(Q)}} \sum_{n=0}^{PG^{(Q)}-1} C_k^{(Q)}(n) C_i^{(Q)}(n) \right] \end{aligned}$$

Hence,

$$\begin{aligned} z^Q &= \sqrt{\frac{E_k^{(Q)} T_s^{(Q)}}{2}} Q_k(m) + \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(Q)} T_s^{(Q)}}{2}} \left[Q_i(m) \frac{1}{PG^{(Q)}} \sum_{n=0}^{PG^{(Q)}-1} C_k^{(Q)}(n) C_i^{(Q)}(n) \right] + N_Q \\ &= \sqrt{\frac{E_k^{(Q)} T_s^{(Q)}}{2}} Q_k(m) + \sum_{\substack{i=1 \\ \neq k}}^K Q_i(m) \sqrt{\frac{E_i^{(Q)} T_s^{(Q)}}{2}} \rho_{k,i}^{(Q)} + N_Q \end{aligned} \quad (3.9)$$

where $\rho_{k,i}^{(Q)}$ is the cross correlation between user k 's and user i 's Q -channel spreading codes. Thus, the I - and Q -channel output for synchronous case are given as follows:

$$\begin{aligned} z^I &= \sqrt{\frac{E_k^{(I)} T_s^{(I)}}{2}} I_k(m) + \sum_{\substack{i=1 \\ \neq k}}^K I_i(m) \sqrt{\frac{E_i^{(I)} T_s^{(I)}}{2}} \rho_{k,i}^{(I)} + N_I \\ z^Q &= \sqrt{\frac{E_k^{(Q)} T_s^{(Q)}}{2}} Q_k(m) + \sum_{\substack{i=1 \\ \neq k}}^K Q_i(m) \sqrt{\frac{E_i^{(Q)} T_s^{(Q)}}{2}} \rho_{k,i}^{(Q)} + N_Q \end{aligned} \quad (3.10)$$

where $\rho_{k,i}^{(I)}$ & $\rho_{k,i}^{(Q)}$ are the cross correlation between the desired user's spreading code and the i^{th} interfering user's code on the I - and Q -channel, respectively. These correlations are given by

$$\begin{aligned}\rho_{k,i}^{(I)} &= \frac{1}{PG^I} \sum_{n=0}^{PG^I-1} C_k^{(I)}(n) C_i^{(I)}(n) \\ \rho_{k,i}^{(Q)} &= \frac{1}{PG^Q} \sum_{n=0}^{PG^Q-1} C_k^{(Q)}(n) C_i^{(Q)}(n)\end{aligned}\tag{3.11}$$

3.2 Asynchronous QPSK

For the asynchronous case, a distortionless channel is considered for all users, where all user signals undergo random delays. Without loss of generality, the k^{th} user (desired user) signal is assumed to have zero delay and all other users have arbitrary delays. The delay for user i is denoted τ_i , and without loss of generality, we consider the delays to be uniformly distributed in $[0, T_s]$. The analysis for the asynchronous case is similar to that for the synchronous case. The received signal $r(t)$, from Chapter 2, is given as

$$r(t) = s_k(t) + \sum_{\substack{i=1 \\ \neq k}}^K s_i(t - \tau_i) + n(t)$$

where the arbitrary i^{th} signal is defined as

$$\begin{aligned}s_i(t - \tau_i) &= \sqrt{\frac{2E_i^{(I)}}{T_s^{(I)}}} I_i(t - \tau) C_i^{(I)}(t - \tau) \cos(\omega_c t - \phi_i) \\ &\quad - \sqrt{\frac{2E_i^{(Q)}}{T_s^{(Q)}}} Q_i(t - \tau) C_i^{(Q)}(t - \tau) \sin(\omega_c t - \phi_i)\end{aligned}$$

and $\phi_i = \omega_c \tau_i$ is modeled as uniform on $[0, 2\pi)$. The analysis of the correlator outputs for the in-phase and quadrature-phase channels is done separately and is shown in the following sections.

3.2.1 Multi-User QPSK DS-SS: I-channel Decision Statistics

The analysis for the I -channel decision statistics for the multiuser case is shown. The I -channel correlator output is given by substituting (2.13) into the correlator equation as in section 2.1.1.

$$\begin{aligned}
 z^{(I)} &= \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} r(t) \cos \omega_c t C_k^{(I)}(t) dt \\
 &= \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} [s_k(t) + \sum_{\substack{i=1 \\ i \neq k}}^K s_i(t - \tau_i) + n(t)] \cos \omega_c t C_k^{(I)}(t) dt \\
 &= \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} \left(\sqrt{\frac{2E_k^{(I)}}{T_s^{(I)}}} I_k(t) C_k^{(I)}(t) \cos \omega_c t - \sqrt{\frac{2E_k^{(Q)}}{T_s^{(Q)}}} Q_k(t) C_k^{(Q)}(t) \sin \omega_c t \right) (\cos \omega_c t C_k^{(I)}(t)) dt \\
 &\quad + \sum_{\substack{i=1 \\ i \neq k}}^K \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} \left(\sqrt{\frac{2E_i^{(I)}}{T_s^{(I)}}} I_i(t - \tau_i) C_i^{(I)}(t - \tau_i) \cos(\omega_c t - \phi_i) \right) (\cos \omega_c t C_k^{(I)}(t)) dt \\
 &\quad - \sum_{\substack{i=1 \\ i \neq k}}^K \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} \sqrt{\frac{2E_i^{(Q)}}{T_s^{(Q)}}} Q_i(t - \tau_i) C_i^{(Q)}(t - \tau_i) \sin(\omega_c t - \phi_i) (\cos \omega_c t C_k^{(I)}(t)) dt + N_I
 \end{aligned}$$

where N_I is again the AWGN noise sample. Continuing, we have

$$\begin{aligned}
 z^{(I)} &= \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} \left(\sqrt{\frac{2E_k^{(I)}}{T_s^{(I)}}} I_k(t) C_k^{(I)}(t) \cos^2 \omega_c t - \sqrt{\frac{2E_k^{(Q)}}{T_s^{(Q)}}} Q_k(t) C_k^{(Q)}(t) C_k^{(I)}(t) \cos \omega_c t \sin \omega_c t \right) dt \\
 &\quad + \sum_{\substack{i=1 \\ i \neq k}}^K \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} \sqrt{\frac{2E_i^{(I)}}{T_s^{(I)}}} I_i(t - \tau_i) C_i^{(I)}(t - \tau_i) C_k^{(I)}(t) \cos(\omega_c t - \phi_i) \cos \omega_c t dt \\
 &\quad - \sum_{\substack{i=1 \\ i \neq k}}^K \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} \sqrt{\frac{2E_i^{(Q)}}{T_s^{(Q)}}} Q_i(t - \tau_i) C_i^{(Q)}(t - \tau_i) C_k^{(I)}(t) \sin(\omega_c t - \phi_i) \cos \omega_c t dt + N_I
 \end{aligned}$$

Then, employing trigonometric identities, we obtain

$$\begin{aligned}
z^{(I)} = & \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} \left[\frac{1}{2} \sqrt{\frac{2E_k^{(I)}}{T_s^{(I)}}} I_k(t) (1 + \cos 2\omega_c t) dt \right. \\
& + \frac{1}{2} \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{2E_i^{(I)}}{T_s^{(I)}}} \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} I_i(t - \tau_i) C_i^{(I)}(t - \tau_i) C_k^{(I)}(t) (\cos(2\omega_c t - \phi_i) + \cos \phi_i) dt \\
& \left. - \frac{1}{2} \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{2E_i^{(Q)}}{T_s^{(Q)}}} \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} Q_i(t - \tau_i) C_i^{(Q)}(t - \tau_i) C_k^{(I)}(t) (\sin(2\omega_c t - \phi_i) - \sin \phi_i) dt + N_I \right]
\end{aligned}$$

and the double-frequency terms are filtered out, hence we have

$$\begin{aligned}
z^{(I)} = & \frac{1}{2} \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} \sqrt{\frac{2E_k^{(I)}}{T_s^{(I)}}} I_k(t) dt + \frac{1}{2} \sum_{\substack{i=1 \\ \neq k}}^K \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} \sqrt{\frac{2E_i^{(I)}}{T_s^{(I)}}} I_i(t - \tau_i) C_i^{(I)}(t - \tau_i) C_k^{(I)}(t) \cos \phi_i dt \\
& + \frac{1}{2} \sum_{\substack{i=1 \\ \neq k}}^K \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} \sqrt{\frac{2E_i^{(Q)}}{T_s^{(Q)}}} Q_i(t - \tau_i) C_i^{(Q)}(t - \tau_i) C_k^{(I)}(t) \sin \phi_i dt + N_I \\
z^{(I)} = & \frac{1}{2} \sqrt{\frac{2E_k^{(I)}}{T_s^{(I)}}} (I_k(m)T_s^{(I)}) + \frac{1}{2} \sum_{\substack{i=1 \\ \neq k}}^K \cos \phi_i \sqrt{\frac{2E_i^{(I)}}{T_s^{(I)}}} \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} I_i(t - \tau_i) C_i^{(I)}(t - \tau_i) C_k^{(I)}(t) dt \\
& + \frac{1}{2} \sum_{\substack{i=1 \\ \neq k}}^K \sin \phi_i \sqrt{\frac{2E_i^{(Q)}}{T_s^{(Q)}}} \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} Q_i(t - \tau_i) C_i^{(Q)}(t - \tau_i) C_k^{(I)}(t) dt + N_I
\end{aligned} \tag{3.12}$$

From (3.10) it is seen that as with the synchronous case, $z^{(I)}$ can be expressed as a sum of three terms.

$$z^{(I)} = \text{Desired Signal} + \text{AWGN} + \text{MUI}$$

To simplify the discussion and further analysis of (3.10), we let

$$X^{(I)} = \sum_{\substack{i=1 \\ \neq k}}^K \cos \phi_i \sqrt{\frac{E_i^{(I)}}{2T_s^{(I)}}} \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} [I_i(t - \tau_i) C_i^{(I)}(t - \tau_i) C_k^{(I)}(t)] dt$$

and

$$Y^{(I)} = \sum_{\substack{i=1 \\ \neq k}}^K \sin \phi_i \sqrt{\frac{E_b^Q}{2T_s^{(Q)}}} \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} [Q_i(t - \tau_i) C_i^{(Q)}(t - \tau_i) C_k^{(I)}(t)] dt$$

Figure 3.2 shows the timing diagram for the I -channel asynchronous case analysis. We use this diagram for the integration of the chip sequence on the I - and Q -channels. We let L_i equal the number of chips by which user i 's signal is shifted, such that $\tau_i = L_i T_c^{(I)} + \varepsilon_i$. For simplicity, we assume that $\varepsilon_i = 0$ and without loss of generality we can assume that $L_i < PG^I$. Therefore, the I -to- I -channel MUI term is

$$\begin{aligned} X^{(I)} &= \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(I)}}{2T_s^{(I)}}} \int_{mT_s^{(I)}}^{(m+1)T_s^{(I)}} I_i(t - \tau) C_i^{(I)}(t - \tau) C_k^{(I)}(t) \cos \phi_i dt \\ &= \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(I)}}{2T_s^{(I)}}} \left[\int_{mT_s^{(I)}}^{\tau_i} I_i(m-1) C_i^{(I)}(t - \tau) C_k^{(I)}(t) \cos \phi_i dt + \int_{\tau_i}^{(m+1)T_s^{(I)}} I_i(m) C_i^{(I)}(t - \tau) C_k^{(I)}(t) \cos \phi_i dt \right] \\ &= \sum_{\substack{i=1 \\ \neq k}}^K \cos \phi_i \sqrt{\frac{E_i^{(I)}}{2T_s^{(I)}}} \left[I_i(m-1) \int_{mT_s^{(I)}}^{\tau_i} C_i^{(I)}(t - \tau) C_k^{(I)}(t) dt + I_i(m) \int_{\tau_i}^{(m+1)T_s^{(I)}} C_i^{(I)}(t - \tau) C_k^{(I)}(t) dt \right] \end{aligned} \quad (3.13)$$

Figure 3.2: Timing diagram for I-channel analysis.

From the above timing diagram (Figure 3.2), we obtain the following summation forms for the partial cross correlation terms within $X^{(l)}$.

$$\int_0^{\tau_i} C_i^{(l)}(t-\tau) C_k^{(l)}(t) dt = \int_0^{L_i T_{ci}^{(l)}} C_i^{(l)}(t-\tau) C_k^{(l)}(t) dt = \sum_{n=0}^{L_i-1} C_k^{(l)}(n) C_i^{(l)}(n-L_i) \quad (3.14)$$

$$\int_{\tau_i}^{T_s^{(l)}} C_i^{(l)}(t-\tau) C_k^{(l)}(t) dt = \int_{L_i T_{ci}^{(l)}}^{PG^{(l)}-1} C_i^{(l)}(t-\tau) C_k^{(l)}(t) dt = \sum_{n=L_i}^{PG^{(l)}-1} C_k^{(l)}(n) C_i^{(l)}(n-L_i) \quad (3.15)$$

This then results in the following expression for the I -to- I channel MUI term

$$X^{(l)} = \sum_{\substack{i=1 \\ i \neq k}}^K \sqrt{\frac{E_i^{(l)} T_s^{(l)}}{2}} \frac{\cos \phi_i}{PG^{(l)}} (X_1^{(l)} + X_2^{(l)}) \quad (3.16)$$

where

$$\begin{aligned}
 X_1^{(I)} &= I_i(m-1) \sum_{n=0}^{L_i-1} C_k^{(I)}(mPG^{(I)} + n) C_i^{(I)}(mPG^{(I)} + n - L_i) \\
 X_2^{(I)} &= I_i(m) \sum_{n=L_i}^{PG^{(I)}-1} C_k^{(I)}(mPG^{(I)} + n) C_i^{(I)}(mPG^{(I)} + n - L_i)
 \end{aligned} \tag{3.17}$$

For calculating $Y^{(I)}$, we must take into account the different chip and data rates on the I and Q channels. The pertinent relations we need are as follows:

$$R_c^{(I)} = MR_c^{(Q)}, R_s^{(Q)} = M_d R_s^{(I)}.$$

In the above diagram, $L_i = 3, M = 4, M_d = 2$. With the aid of the timing diagram, and these relationships, we solve the integral and obtain

$$Y^{(I)} = \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{M_d} \sqrt{\frac{E_i^{(Q)} T_s^{(Q)}}{2}} \sin \phi_i \left[\begin{aligned} & Q_i(m-1) C_i^{(Q)}(-1) \sum_{n=0}^{L_i-1} C_k^{(I)}(n) \\ & + Q_i(m) \left(C_i^{(Q)}(0) \sum_{n=L_i}^{L_i+M-1} C_k^{(I)}(n) + C_i^{(Q)}(1) \sum_{n=L_i+M}^{L_i+2M-1} C_k^{(I)}(n) \right) \\ & + Q_i(m+1) \left(C_i^{(Q)}(2) \sum_{n=L_i+2M}^{L_i+3M-1} C_k^{(I)}(n) + C_i^{(Q)}(3) \sum_{n=L_i+3M}^{PG^{(I)}-1} C_k^{(I)}(n) \right) \end{aligned} \right]$$

Modifying the above equation by incorporating different data/chip rates explicitly gives

$$Y^{(I)} = \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(Q)} T_s^{(I)}}{2}} \frac{\sqrt{M_d} \sin \phi_i}{PG^{(I)}} [Y_1^{(I)} + Y_2^{(I)} + Y_3^{(I)} + Y_4^{(I)} + Y_5^{(I)}] \tag{3.18}$$

where each of the terms in (3.18) is defined as follows

$$\begin{aligned}
Y_1^{(I)} &= \sum_{m=0}^{M_d-1} Q_i(-A+m) \sum_{s=0}^{PG^Q-1} C_i^{(Q)}(s+mPG^Q) \sum_{n=0}^{M-1} C_k^{(I)}(sM+n) \\
Y_2^{(I)} &= \sum_{m=-A-1}^{-A-1} Q_i(m) \sum_{s=0}^{B-1} C_i^{(Q)}((m+1)PG^Q-1-s) \sum_{n=0}^{M-1} C_k^{(I)}((L_i-1)+(1+s)M-n) \\
Y_3^{(I)} &= \sum_{m=-A-1}^{-A-1} Q_i(m) C_i^{(Q)}((m+1)PG^Q-B-3) \sum_{n=0}^{D-1} C_k^{(I)}(n) \\
Y_4^{(I)} &= \sum_{m=M_d-A}^{M_d-A} Q_i(m) \sum_{s=0}^{-B-1} C_i^{(Q)}(mPG^Q+s) \sum_{n=0}^{M-1} C_k^{(I)}(sM+L_i) \\
Y_5^{(I)} &= \sum_{m=M_d-A}^{M_d-A} Q_i(m) C_i^{(Q)}(mPG^Q-B) \sum_{n=0}^{-D} C_k^{(I)}(n+mMPG^Q-BM)
\end{aligned} \tag{3.19}$$

and we have defined the following terms:

$$W = \left\lfloor \frac{L_i}{MPG^Q} \right\rfloor, A = \text{floor}(W), B = \text{floor}\left(\frac{\text{mod}(W)}{M}\right), D = \text{mod}\left(\frac{\text{mod}(W)}{M}\right)$$

Therefore, the expression for $z^{(I)}$ is

$$\begin{aligned}
z^{(I)} &= \sqrt{\frac{E_k^{(I)} T_s^{(I)}}{2}} I_k(m) + \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(I)} T_s^{(I)}}{2}} \frac{\cos \phi_i}{PG^I} (X_1^{(I)} + X_2^{(I)}) \\
&\quad + \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(Q)} T_s^{(I)}}{2}} \frac{\sqrt{M_d} \sin \phi_i}{PG^I} [Y_1^{(I)} + Y_2^{(I)} + Y_3^{(I)} + Y_4^{(I)} + Y_5^{(I)}] + N_I
\end{aligned} \tag{3.20}$$

where the X's and Y's are given in (3.17) and (3.19).

3.2.2 Multi-User QPSK DS-SS: Q-channel Decision Statistics

Similar to the analysis for the I -channel, the Q -channel output can be written as

$$\begin{aligned}
 z^{(Q)} = & \sqrt{\frac{E_k^{(Q)}}{2T_s^{(Q)}}} Q_k(m) T_s^{(Q)} + \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(Q)}}{2T_s^{(Q)}}} \int_{mT_s^{(Q)}}^{(m+1)T_s^{(Q)}} Q_i(t - \tau_i) C_i^{(Q)}(t - \tau_i) C_k^{(Q)}(t) \cos \phi_i dt \\
 & + \sqrt{\frac{E_i^{(I)}}{2T_s^{(I)}}} \sum_{\substack{i=1 \\ \neq k}}^K \int_{mT_s^{(Q)}}^{(m+1)T_s^{(Q)}} I_i(t - \tau_i) C_k^{(Q)}(t) C_i^{(I)}(t - \tau_i) \sin \phi_i dt + N_Q
 \end{aligned} \tag{3.21}$$

Here we let

$$X^Q = \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(Q)}}{2T_s^{(Q)}}} \int_{mT_s^{(Q)}}^{(m+1)T_s^{(Q)}} Q_i(t - \tau_i) C_i^{(Q)}(t - \tau_i) C_k^{(Q)}(t) \cos \phi_i dt$$

$$\text{and } Y^Q = \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(I)}}{2T_s^{(I)}}} \int_{mT_s^{(Q)}}^{(m+1)T_s^{(Q)}} I_i(t - \tau_i) C_k^{(Q)}(t) C_i^{(I)}(t - \tau_i) \sin \phi_i dt$$

and get,

$$X^Q = \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(Q)} T_s^{(Q)}}{2}} \frac{\cos \phi_i}{PG^Q} [X_1^{(Q)} + X_2^{(Q)}] \tag{3.22}$$

where

$$\begin{aligned}
 X_1^{(Q)} &= Q_i(m-1) \sum_{n=0}^{L_q-1} C_k^{(Q)}(mPG^{(Q)} + n) C_i^{(Q)}(mPG^{(Q)} + n - L_q) \\
 X_2^{(Q)} &= Q_i(m) \sum_{n=L_q}^{PG^{(Q)}-1} C_k^{(Q)}(mPG^{(Q)} + n) C_i^{(Q)}(mPG^{(Q)} + n - L_q)
 \end{aligned} \tag{3.23}$$

Here again, L_q is the number of chips by which the i^{th} user signal is shifted such that $\tau_q = L_q T_c^{(Q)} + \varepsilon_q$. Again, for simplicity we let $\varepsilon_q = 0$, and without loss of generality assume that $L_q < PG^Q$,

$$Y^Q = \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(I)}}{2T_s^{(I)}}} \int_{mT_s^{(Q)}}^{(m+1)T_s^{(Q)}} I_i(t-\tau) C_k^{(Q)}(t) C_i^{(I)}(t-\tau) \sin \phi_i dt$$

Integrating the product of the chips on I and Q over one symbol period, making use of Figure 3.3, and using, $T_s^{(Q)} = T_s^{(I)} / M_d$ gives

$$Y^Q = \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{1}{M_d}} \sqrt{\frac{E_i^{(I)} T_s^{(Q)}}{2}} \frac{\sin \phi_i}{PG^Q} [Y_1^{(Q)} + Y_2^{(Q)}] \quad (3.24)$$

where

$$\begin{aligned} Y_1^{(Q)} &= I_i \left(\text{ceil} \left(\frac{m}{M_d} \right) - 1 \right) \sum_{s=0}^{L_q-1} C_k^{(Q)}(mPG^{(Q)} + s) \sum_{n=0}^{M-1} C_i^{(I)}(sM + n - L_q M) \\ Y_2^{(Q)} &= I_i \left(\text{floor} \left(\frac{m}{M_d} \right) \right) \sum_{s=L_q}^{PG^{(Q)}-1} C_k^{(Q)}(mPG^{(Q)} + s) \sum_{n=0}^{M-1} C_i^{(I)}(sM + n - L_q M) \end{aligned} \quad (3.25)$$

Figure 3.3: Timing diagram for Q-channel analysis.

The final expression for the Q -channel decision statistic is then as follows:

$$\begin{aligned}
 z^Q = & \sqrt{\frac{E_k^{(Q)} T_s^{(Q)}}{2}} Q_k(m) + \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(Q)} T_s^{(Q)}}{2}} \frac{\cos \phi_i}{PG^Q} [X_1^{(Q)} + X_2^{(Q)}] \\
 & + \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{1}{M_d}} \sqrt{\frac{E_i^{(I)} T_s^{(Q)}}{2}} \frac{\sin \phi_i}{PG^Q} [Y_1^{(Q)} + Y_2^{(Q)}] + N_Q
 \end{aligned} \tag{3.26}$$

where the X 's and Y 's are defined in (3.23) and (3.25).

Summarizing, the I - and Q - channel statistics for the asynchronous case are

$$\begin{aligned}
 z^I = & \sqrt{\frac{E_k^{(I)} T_s^{(I)}}{2}} I_k(m) + \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(I)} T_s^{(I)}}{2}} \frac{\cos \phi_i}{PG^I} (X_1^{(I)} + X_2^{(I)}) \\
 & + \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(Q)} T_s^{(I)}}{2}} \frac{\sqrt{M_d} \sin \phi_i}{PG^I} [Y_1^{(I)} + Y_2^{(I)} + Y_3^{(I)} + Y_4^{(I)} + Y_5^{(I)}] + N_I \\
 z^Q = & \sqrt{\frac{E_k^{(Q)} T_s^{(Q)}}{2}} Q_k(m) + \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{E_i^{(Q)} T_s^{(Q)}}{2}} \frac{\cos \phi_i}{PG^Q} [X_1^{(Q)} + X_2^{(Q)}] \\
 & + \sum_{\substack{i=1 \\ \neq k}}^K \sqrt{\frac{1}{M_d}} \sqrt{\frac{E_i^{(I)} T_s^{(Q)}}{2}} \frac{\sin \phi_i}{PG^Q} [Y_1^{(Q)} + Y_2^{(Q)}] + N_Q
 \end{aligned}$$

where,

$$\begin{aligned}
X_1^{(I)} &= I_i(m-1) \sum_{n=0}^{L_i-1} C_k^{(I)}(mPG^{(I)} + n) C_i^{(I)}(mPG^{(I)} + n - L_i) \\
X_2^{(I)} &= I_i(m) \sum_{n=L_i}^{PG^I-1} C_k^{(I)}(mPG^{(I)} + n) C_i^{(I)}(mPG^{(I)} + n - L_i) \\
Y_1^{(I)} &= \sum_{m=0}^{M_d-1} Q_i(-A+m) \sum_{s=0}^{PG^Q-1} C_i^{(Q)}(s+mPG^Q) \sum_{n=0}^{M-1} C_k^{(I)}(sM+n) \\
Y_2^{(I)} &= \sum_{m=-A-1}^{-A-1} Q_i(m) \sum_{s=0}^{B-1} C_i^{(Q)}((m+1)PG^Q - 1 - s) \sum_{n=0}^{M-1} C_k^{(I)}((L_i-1) + (1+s)M - n) \\
Y_3^{(I)} &= \sum_{m=-A-1}^{-A-1} Q_i(m) C_i^{(Q)}((m+1)PG^Q - B - 3) \sum_{n=0}^{D-1} C_k^{(I)}(n) \\
Y_4^{(I)} &= \sum_{m=M_d-A}^{M_d-A} Q_i(m) \sum_{s=0}^{-B-1} C_i^{(Q)}(mPG^Q + s) \sum_{n=0}^{M-1} C_k^{(I)}(sM + L_i) \\
Y_5^{(I)} &= \sum_{m=M_d-A}^{M_d-A} Q_i(m) C_i^{(Q)}(mPG^Q - B) \sum_{n=0}^{-D} C_k^{(I)}(n + mMPG^Q - BM) \\
X_1^{(Q)} &= Q_i(m-1) \sum_{n=0}^{L_q-1} C_k^{(Q)}(mPG^{(Q)} + n) C_i^{(Q)}(mPG^{(Q)} + n - L_q) \\
X_2^{(Q)} &= Q_i(m) \sum_{n=L_q}^{PG^{(Q)}-1} C_k^{(Q)}(mPG^{(Q)} + n) C_i^{(Q)}(mPG^{(Q)} + n - L_q) \\
Y_1^{(Q)} &= I_i \left(\text{ceil} \left(\frac{m}{M_d} \right) - 1 \right) \sum_{s=0}^{L_q-1} C_k^{(Q)}(mPG^{(Q)} + s) \sum_{n=0}^{M-1} C_i^{(I)}(sM + n - L_q M) \\
Y_1^{(Q)} &= I_i \left(\text{floor} \left(\frac{m}{M_d} \right) \right) \sum_{s=L_q}^{PG^{(Q)}-1} C_k^{(Q)}(mPG^{(Q)} + s) \sum_{n=0}^{M-1} C_i^{(I)}(sM + n - L_q M) \\
W &= \left(\frac{L_i}{MPG^Q} \right), \quad A = \text{floor}(Z), \quad B = \text{floor} \left(\frac{\text{mod}(Z)}{M} \right), \quad D = \text{mod} \left(\frac{\text{mod}(Z)}{M} \right)
\end{aligned}$$

3.3 Probability of Error

In the absence of noise, the baseband signal voltage for a BPSK system can take one of two values, $\pm V$. In an ideal case, if the signal is greater than 0, the output is considered V , and if the signal is less than 0 it is assumed $-V$. However, in the presence of noise, this distinction between $\pm V$ becomes more difficult. There is a non-zero

probability that the signal plus noise will be below 0, and will be decided as -V, even though a +1 was transmitted. In such conditions, a bit error has occurred. The probability that a bit error occurs is referred as the bit error ratio (BER). The BER is also often called a probability of bit error P_b .

The performance of the receiver is usually measured in terms of the probability of error (P_b), and the receiver is said to be optimum if it has the minimum P_b . Thus, probability of error denotes the measure of quality of a received signal. The lower the error probability, the better is the signal received. For example, $P_b = 10^{-4}$ means that on average, one error occurs in every 10,000 bits. The bit error probability for QPSK system is given as [1].

$$P_b = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) \quad (3.27)$$

For a multiuser QPSK DS CDMA system with the same data and chip rates on the I and Q channels, for any given user, the probability of error is calculated using

$$P_b = Q\left(\sqrt{\frac{\mu_k^2}{\sigma_{AWGN}^2 + MUI}}\right) \quad (3.28)$$

where $\mu^2 = [E(Z_k^I | I_k(m) = 1)]^2$, $MUI = Var[(Z_k^I - N_I | I_k(m) = 1)]$, which in (3.28) is approximated as Gaussian, and $\sigma^2 = \text{noise variance}$. We also note that $Var(Z_k^I | I_k(m) = 1) = \sigma_{AWGN}^2 + MUI$

The general expression for the multiuser BER is given in (3.29), where we have assumed equal received signal energies, and equal PG^I and PG^Q , and $R_S^I = R_S^Q$ for all users and β is a constant that depends on the alignment of the chip and the carrier phases.

$$P_b = Q \left(\sqrt{\frac{2E_b}{N_0 + 2(K-1)E_b/\beta N}} \right) \quad (3.29)$$

In (3.29), K = number of users, and N =processing gain.

For asynchronous transmission with random spreading codes, and random carrier phases $\beta = 3$, and the BER expression is:

$$P_b = Q \left(\sqrt{\frac{2E_b}{N_0 + 2(K-1)E_b/3N}} \right)$$

As seen from this equation, the BER is dependent on the processing gain and the number of users. The expression for the bit error probability, with different processing gains on the I - and Q -channels, in the presence of multiuser interference, is shown in the following section. The analysis is done for both synchronous and asynchronous QPSK systems.

3.3.1 *Orthogonal Spreading Codes: Synchronous QPSK*

To compute the probability of error on the I - and Q -channels, we need to calculate the means and variances of the correlator output terms. The mean and variance for equations (3.2) and (3.3) is required. Also, the probability of error on individual channels can be estimated using equation (3.28). For zero-mean random data, we have,

$$\begin{aligned} \mu^2 &= E \left[(z^I | I_k(m) = 1) \right]^2 = \frac{E_k^{(I)} T_s^{(I)}}{2} \text{ and,} \\ \text{Var}(z^I | I_k(m) = 1) &= \sigma_{AWGN}^2 \end{aligned} \quad (3.30)$$

This yields,

$$P_b^I = Q \left(\sqrt{\frac{\frac{E_k^{(I)} T_s^{(I)}}{2}}{\frac{N_0 T_s^I}{4}}} \right) = Q \left(\sqrt{\frac{2E_k^{(I)}}{N_0}} \right) \quad (3.31)$$

A similar method can be used to determine the probability of error for the Q -channel, with an analogous result

$$P_b^Q = Q \left(\sqrt{\frac{2E_k^{(Q)}}{N_0}} \right) \quad (3.32)$$

These results are those for conventional, single-user QPSK, resulting from the code orthogonality, which yields zero MUI.

3.3.2 *Random Spreading Codes: I-Channel Synchronous QPSK*

To compute the probability of error on the I - and Q -channels, we calculate the means and variances of the correlator output terms. The mean and variance for equations (3.7) and (3.9) is required. Also, the probability of error on individual channels can be estimated using equation (3.28).

For zero-mean random data and zero-mean random spreading codes, we have,

$$\begin{aligned} \mu^2 &= E \left[\left(z^I \mid I_k(m) = 1 \right)^2 \right] = \frac{E_k^{(I)} T_s^{(I)}}{2} \text{ and,} \\ \text{Var} \left(z^I \mid I_k(m) = 1 \right) &= \sigma_{AWGN}^2 + MUI \\ &= \frac{N_0 T_s^{(I)}}{4} + \sum_{\substack{i=1 \\ i \neq k}}^K \left[\left(\frac{E_i^{(I)} T_s^{(I)}}{2PG^I} \right) \right] \end{aligned} \quad (3.33)$$

This yields,

$$P_b^I = Q \left(\sqrt{\frac{\frac{E_k^{(I)} T_s^{(I)}}{2}}{\frac{N_0 T_s^I}{4} + \sum_{\substack{i=1 \\ i \neq k}}^K \left[\frac{E_i^{(I)} T_s^{(I)}}{2 P G^I} \right]}} \right) \quad (3.34)$$

$$P_b^I = Q \left(\sqrt{\frac{2 E_k^{(I)}}{N_0 + \sum_{\substack{i=1 \\ i \neq k}}^K \left[\frac{2 E_i^{(I)}}{P G^I} \right]}} \right)$$

A similar method can be used to determine the probability of error for the Q -channel, with an analogous result

$$P_b^Q = Q \left(\sqrt{\frac{\frac{E_k^{(Q)} T_s^{(Q)}}{2}}{\frac{N_0 T_s^Q}{4} + \sum_{\substack{i=1 \\ i \neq k}}^K \left[\frac{E_i^{(Q)} T_s^{(Q)}}{2 P G^Q} \right]}} \right) \quad (3.35)$$

$$= Q \left(\sqrt{\frac{2 E_k^{(Q)}}{N_0 + \sum_{\substack{i=1 \\ i \neq k}}^K \left[\frac{2 E_i^{(Q)}}{P G^Q} \right]}} \right)$$

Note that for equal user energies, since all I channels and all Q channels have the same data rate and chip rate, (3.34) and (3.35) are in the form of (3.29) with $\beta = 1$.

3.3.3 Probability of Error: Asynchronous QPSK

The probability of error analysis for the asynchronous case is similar to the synchronous case analysis. The mean and variance are calculated for equations (3.20) and (3.26), and the probability of error on individual channels is calculated using equation (3.28). The conditional I channel decision statistic mean and variance are

$$\mu^2 = \frac{E_k^{(I)} T_s^{(I)}}{2}$$

$$MUI = \sum_{\substack{i=1 \\ \neq k}}^K \left[\frac{1}{PG^I} \left(\frac{E_i^{(I)} T_s^{(I)}}{2} \right) \cos^2 \phi_i + \frac{M_d}{PG^I} \left(\frac{E_i^{(Q)} T_s^{(I)}}{2} \right) \sin^2 \phi_i \right]$$

which yields

$$P_b^I = Q \left(\sqrt{\frac{\frac{E_k^{(I)} T_s^{(I)}}{2}}{\frac{N_0 T_s^I}{4} + \frac{1}{3} \sum_{\substack{i=1 \\ \neq k}}^K \left[\frac{1}{PG^I} \left(\frac{E_i^{(I)} T_s^{(I)}}{2} \right) \cos^2 \phi_i + \frac{M_d}{PG^I} \left(\frac{E_i^{(Q)} T_s^{(I)}}{2} \right) \sin^2 \phi_i \right]}} \right) \quad (3.36)$$

$$= Q \left(\sqrt{\frac{2E_k^{(I)}}{N_0 + \frac{2}{3} \sum_{\substack{i=1 \\ \neq k}}^K \left[\frac{E_i^{(I)} \cos^2 \phi_i}{PG^I} + \frac{M_d E_i^{(Q)} \sin^2 \phi_i}{PG^I} \right]}} \right)$$

As in the synchronous case, the probability of error for the Q -channel is determined in a similar manner. Equation (3.28) is used to calculate P_b on the Q -channel. The mean and variance are

$$\mu^2 = \frac{E_k^{(I)} T_s^{(I)}}{2}$$

$$MUI = \sum_{\substack{i=1 \\ \neq k}}^K \left[\frac{1}{PG^Q} \left(\frac{E_i^{(Q)} T_s^{(Q)}}{2} \right) \cos^2 \phi_i + \frac{M}{PG^Q} \left(\frac{E_i^{(I)} T_s^{(Q)}}{2} \right) \sin^2 \phi_i \right]$$

and the resulting probability of bit error expression is

$$P_b^Q = Q \left(\sqrt{\frac{\frac{E_k^{(Q)} T_s^{(Q)}}{2}}{\frac{N_0 T_s^Q}{4} + \frac{1}{3} \sum_{\substack{i=1 \\ \neq k}}^K \left[\frac{1}{PG^Q} \left(\frac{E_i^{(Q)} T_s^{(Q)}}{2} \right) \cos^2 \phi_i + \left(\frac{M}{PG^Q} \right) \left(\frac{E_i^{(I)} T_s^{(Q)}}{2} \right) \sin^2 \phi_i \right]}} \right) \quad (3.37)$$

$$= Q \left(\sqrt{\frac{2E_k^{(Q)}}{N_0 + \frac{2}{3} \sum_{\substack{i=1 \\ \neq k}}^K \left[E_i^{(Q)} \frac{\cos^2 \phi_i}{PG^Q} + E_i^{(I)} M \frac{\sin^2 \phi_i}{PG^Q} \right]}} \right)$$

3.4 Rayleigh Fading

In this section we determine the performance of our disparate chip rate QPSK scheme in a fading channel. The transmitted signal passes through a slowly fading channel before it is received at the transmitter. We determine the QPSK performance for single user and multiuser (synchronous and asynchronous) systems in the presence of a Rayleigh fading in this section.

3.4.1 Single QPSK user with Rayleigh Fading

The probability of error in a flat fading channel for a single user signal that undergoes Rayleigh fading during transmission can be calculated using (3.38). To calculate the final average probability of error, we integrate the product of probability density function (pdf) of the Rayleigh fading channel amplitude with the conditional BER expression for a single QPSK user in AWGN for a specific signal to noise ratio.

$$P_{avg} = \int_0^{\infty} P_e(X) p(X) dX \quad (3.38)$$

where $P_e(X)$ = probability of error for QPSK at a specific $\frac{E_b}{N_0}$

$p(X)$ = Rayleigh fading pdf

The probability of bit error for QPSK and the probability density function for the square of the Rayleigh random variable is given below.

$$P_b = Q\left(\sqrt{\frac{2\alpha^2 E_b}{N_0}}\right)$$

$$p(X) = \frac{1}{\bar{\Gamma}} e^{-\frac{x}{\bar{\Gamma}}} \quad x \geq 0$$

where, $\Gamma = \alpha^2 \frac{E_b}{N_0}$ and $\bar{\Gamma} = E\left[\alpha^2 \frac{E_b}{N_0}\right] = E[\alpha^2] \frac{E_b}{N_0} = \bar{\alpha}^2 \frac{E_b}{N_0}$

Therefore,

$$\begin{aligned} P_{avg} &= \int_0^{\infty} P_b(X) p(X) dX \\ &= \frac{1}{\bar{\Gamma}} \int_0^{\infty} Q(\sqrt{2\Gamma}) e^{-\frac{x}{\bar{\Gamma}}} \end{aligned}$$

Solving the integral using $Q(\sqrt{2\Gamma}) = \frac{1}{2} \text{erfc} \sqrt{\Gamma}$ and employing other well known

algebraic expressions [9], we obtain

$$P_{avg} = \frac{1}{2} \left[1 - \sqrt{\frac{\bar{\Gamma}}{1 + \bar{\Gamma}}} \right] \quad (3.39)$$

3.4.2 Multiuser QPSK with Rayleigh Fading

The analysis to calculate the probability of bit error for multiuser QPSK signals undergoing Rayleigh fading during transmission is similar to the single user analysis. The same method is used for both the synchronous and asynchronous scheme. The average probability of error can be calculated by integrating the product of the Rayleigh pdf and the conditional BER for the multiuser QPSK in AWGN for a specific signal to noise ratio using (3.32).

a. Synchronous Transmission with Rayleigh Fading

The analysis for the error probability for the synchronous case using orthogonal codes is similar to that of single user and is given as

$$P_{avg} = \frac{1}{2} \left[1 - \sqrt{\frac{\bar{\Gamma}}{1 + \bar{\Gamma}}} \right]$$

where, for the I -channel, $\Gamma = \alpha^2 \frac{E_k^{(I)}}{N_0}$ and for the Q -channel, $\Gamma = \alpha^2 \frac{E_k^{(Q)}}{N_0}$.

The probability of error for random spreading code, synchronous transmission with multiple users in an AWGN channel is given by (3.33). We use (3.33) and (3.38) to calculate the average BER. The conditional BER for multiuser synchronous QPSK is

$$P_b^I = Q \left(\sqrt{\frac{2\alpha^2 E_k^{(I)}}{N_0 + \sum_{\substack{i=1 \\ i \neq k}}^K \left[\frac{2\alpha^2 E_i^{(I)}}{PG^I} \right]}} \right)$$

Since synchronous and transmitted from a single source, all the α 's are the same.

Therefore, we have

$$\begin{aligned} P_{avg} &= \int_0^\infty P_b(X) p(X) dX \\ &= \frac{1}{\bar{\Gamma}} \int_0^\infty \mathcal{Q}(\sqrt{2\Gamma}) e^{-\frac{x}{\bar{\Gamma}}} \end{aligned}$$

where, in this multiuser case, the SNIR is defined as follows:

$$\Gamma = \frac{\alpha^2 E_k^{(I)}}{N_0 + \sum_{\substack{i=1 \\ \neq k}}^K \left[\frac{2\alpha^2 E_i^{(I)}}{PG^I} \right]}$$

Here, the MUI term is assumed to be Gaussian. Again, the integral can be solved using $\mathcal{Q}(\sqrt{2x})$ and other well-known algebraic expressions [9]. Doing this, we obtain

$$P_{avg} = \frac{1}{2} \left[1 - \sqrt{\frac{\bar{\Gamma}}{1 + \bar{\Gamma}}} \right]$$

$$\text{where, } \bar{\Gamma} = E \left[\frac{\alpha^2 E_k^{(I)}}{N_0 + \sum_{\substack{i=1 \\ \neq k}}^K \left[\frac{2\alpha^2 E_i^{(I)}}{PG^I} \right]} \right]$$

The above equation is of the form

$$E \left(\frac{\alpha^2 A}{B + \alpha^2 C} \right) \text{ for constant values of A, B and C.}$$

Let this be $E(f(\alpha))$. For $A, B, C \geq 0$, $f(\alpha)$ is of the “convex down” form. Jensen’s inequality states that for a convex down function $E(f(\alpha)) \leq f(E(\alpha))$, thus

$E\left(\frac{\alpha^2 A}{B + \alpha^2 C}\right) \leq \frac{(E(\alpha))^2}{B + C(E(\alpha))^2} < \frac{E(\alpha^2)}{B + CE(\alpha^2)}$ for our normalization [12]. For simplicity,

as an approximation, we use the mean values $E[\alpha^2] = 1$. Thus, we note that a lower

bound on $\bar{\Gamma}$ provides an upper bound on P_{avg} . Hence, we get

$$P_{avg}^I = \frac{1}{2} \left[1 - \sqrt{\frac{E_k^{(I)}}{E_k^{(I)} + N_0 + \sum_{i=1, i \neq k}^K \left[\frac{2E_i^{(I)}}{PG^I} \right]}} \right] \quad (3.40)$$

Similarly, the average BER on the Q -channel is given as

$$P_{avg}^Q = \frac{1}{2} \left[1 - \sqrt{\frac{E_k^{(Q)}}{E_k^{(Q)} + N_0 + \sum_{i=1, i \neq k}^K \left[\frac{2E_i^{(Q)}}{PG^Q} \right]}} \right] \quad (3.41)$$

b. Asynchronous Transmission with Rayleigh Fading

The probability of error for asynchronous multiuser QPSK in AWGN channel is given by (3.36). Equations (3.36) and (3.38) are used to calculate the average BER. The conditional BER for multiuser synchronous QPSK is

$$P_b^I = Q \left(\sqrt{\frac{2\alpha^2 E_k^{(I)}}{N_0 + \frac{2}{3}\alpha^2 \sum_{i=1, i \neq k}^K \left[\frac{E_i^{(I)} \cos^2 \phi_i}{PG^I} + \frac{M_d E_i^{(Q)} \sin^2 \phi_i}{PG^I} \right]}} \right)$$

Again using Jensen's inequality yields a lower bound to $\bar{\Gamma}$. Therefore,

$$\begin{aligned} P_{avg} &= \int_0^\infty P_b(X) p(X) dX \\ &= \frac{1}{\bar{\Gamma}} \int_0^\infty Q(\sqrt{2\Gamma}) e^{-\frac{x}{\bar{\Gamma}}} \end{aligned}$$

where in this multiuser asynchronous case, the SNIR is defined as:

$$\Gamma = \frac{\alpha^2 E_k^{(I)}}{N_0 + \frac{2}{3} \alpha^2 \sum_{\substack{i=1 \\ \neq k}}^K \left[\frac{E_i^{(I)} \cos^2 \phi_i}{PG^I} + \frac{M_d E_i^{(Q)} \sin^2 \phi_i}{PG^I} \right]}$$

Again, the MUI term is assumed to be Gaussian. Solving the above integral yields the result

$$P_{avg} = \frac{1}{2} \left[1 - \sqrt{\frac{\bar{\Gamma}}{1 + \bar{\Gamma}}} \right]$$

$$\text{where } \bar{\Gamma} = E \left[\frac{\alpha^2 E_k^{(I)}}{N_0 + \frac{2}{3} \alpha^2 \sum_{\substack{i=1 \\ \neq k}}^K \left[\frac{E_i^{(I)} \cos^2 \phi_i}{PG^I} + \frac{M_d E_i^{(Q)} \sin^2 \phi_i}{PG^I} \right]} \right]$$

Again for simplicity we use Jensen's inequality and $E[\alpha^2] = 1$. Hence, we get

$$P_{avg}^I = \frac{1}{2} \left[1 - \sqrt{\frac{E_k^{(I)}}{E_k^{(I)} + N_0 + \frac{2}{3} \sum_{\substack{i=1 \\ \neq k}}^K \left[\frac{E_i^{(I)} \cos^2 \phi_i}{PG^I} + \frac{M_d E_i^{(Q)} \sin^2 \phi_i}{(PG^I)} \right]} \right]} \quad (3.42)$$

Similarly, for the average BER on the Q -channel we obtain

$$P_{avg}^Q = \frac{1}{2} \left[1 - \sqrt{\frac{E_k^{(Q)}}{E_k^{(Q)} + N_0 + \frac{2}{3} \sum_{\substack{i=1 \\ i \neq k}}^K \left[E_i^{(Q)} \frac{\cos^2 \phi_i}{PG^Q} + E_i^{(I)} M \frac{\sin^2 \phi_i}{PG^Q} \right]}} \right] \quad (3.43)$$

We summarize the results of probability of error for QPSK for the different cases obtained in this chapter. In Chapter 4, we corroborate these analytical results with computer simulations. The results are given in Tables 3.1 and 3.2. The different scenarios considered are the following:

1. Single user QPSK with AWGN
2. Synchronous multiuser QPSK with AWGN
3. Asynchronous multiuser QPSK with AWGN
4. Single user QPSK with Rayleigh fading
5. Synchronous multiuser QPSK with Rayleigh fading
6. Asynchronous multiuser QPSK with Rayleigh fading

Table 3.1: Bit error rate for various QPSK scenarios with AWGN

Scenario	BER Expression
Single user QPSK with AWGN	$P_b = Q\left(\sqrt{\frac{2E_b}{N_0}}\right)$
Synchronous multiuser QPSK I-channel with AWGN	$P_b^I = Q\left(\sqrt{\frac{2E_k^{(I)}}{N_0 + \sum_{i=1, i \neq k}^K \left[\frac{2E_i^{(I)}}{PG^I}\right]}}\right)$
Synchronous multiuser QPSK Q-channel with AWGN	$P_b^Q = Q\left(\sqrt{\frac{2E_k^{(Q)}}{N_0 + \sum_{i=1, i \neq k}^K \left[\frac{2E_i^{(Q)}}{PG^Q}\right]}}\right)$
Asynchronous multiuser QPSK I-channel with AWGN	$P_b^I = Q\left(\sqrt{\frac{2E_k^{(I)}}{N_0 + \frac{2}{3} \sum_{i=1, i \neq k}^K \left[\frac{E_i^{(I)} \cos^2 \phi_i}{PG^I} + \frac{M_d E_i^{(Q)} \sin^2 \phi_i}{PG^I}\right]}}\right)$
Asynchronous multiuser QPSK Q-channel with AWGN	$P_b^Q = Q\left(\sqrt{\frac{2E_k^{(Q)}}{N_0 + \frac{2}{3} \sum_{i=1, i \neq k}^K \left[E_i^{(Q)} \frac{\cos^2 \phi_i}{PG^Q} + E_i^{(I)} M \frac{\sin^2 \phi_i}{PG^Q}\right]}}\right)$

Table 3.2: Bit error rate for various QPSK scenarios with Rayleigh fading

Scenario	BER Expression
Single user QPSK with Rayleigh fading	$P_{avg} = \frac{1}{2} \left[1 - \sqrt{\frac{E_b}{E_b + N_0}} \right]$
Synchronous multiuser QPSK I- channel with Rayleigh fading	$P_{avg}^I = \frac{1}{2} \left[1 - \sqrt{\frac{E_k^{(I)}}{E_k^{(I)} + N_0 + \sum_{i=1, i \neq k}^K \left[\frac{2E_i^{(I)}}{PG^I} \right]}} \right]$
Synchronous multiuser QPSK Q- channel with Rayleigh fading	$P_{avg}^Q = \frac{1}{2} \left[1 - \sqrt{\frac{E_k^{(Q)}}{E_k^{(Q)} + N_0 + \sum_{i=1, i \neq k}^K \left[\frac{2E_i^{(Q)}}{PG^Q} \right]}} \right]$
Asynchronous multiuser QPSK I- channel with Rayleigh fading	$P_{avg}^I = \frac{1}{2} \left[1 - \sqrt{\frac{E_k^{(I)}}{E_k^{(I)} + N_0 + \frac{2}{3} \sum_{i=1, i \neq k}^K \left[\frac{E_i^{(I)} \cos^2 \phi_i}{PG^I} + \frac{M_d E_i^{(Q)} \sin^2 \phi_i}{(PG^I)} \right]}} \right]$
Asynchronous multiuser QPSK Q- channel with Rayleigh fading	$P_{avg}^Q = \frac{1}{2} \left[1 - \sqrt{\frac{E_k^{(Q)}}{E_k^{(Q)} + N_0 + \frac{2}{3} \sum_{i=1, i \neq k}^K \left[E_i^{(Q)} \frac{\cos^2 \phi_i}{PG^Q} + E_i^{(I)} M \frac{\sin^2 \phi_i}{PG^Q} \right]}} \right]$

Chapter 4: Simulation Description and Results

In this research, a QPSK DS-CDMA simulation model was developed. The aim of development of this model was primarily to study the system and to evaluate its performance in an AWGN and Rayleigh fading channel, and compare results with analytical results. In addition, the simulation provides a platform for future study of system characteristics and an aid in the design of practical applications. As noted, the performance of a digital system can be evaluated in terms of bit error probability. The simulation model is developed in *MATLAB* and is based on the *Monte Carlo* method, which utilizes random numbers to perform the simulation. The simulation computes an estimate for the bit error probability for *I*- and *Q*-channels of a QPSK system.

The simulation is performed for a QPSK system in both AWGN and Rayleigh fading channels. Various different conditions were simulated. Simulate performance of the system with a single user and with multiple users was obtained. A synchronous QPSK system was simulated with both orthogonal and random spreading codes. Synchronous QPSK with random spreading codes was also simulated to observe the system performance due to interference of multiple signals which is not present with perfect orthogonal codes.

In addition, an asynchronous QPSK system with random delays for all users was simulated. The processing gain is assumed different on each channel (i.e., different on *I* and *Q* of any user signal); all users employ the same *I* and *Q*-channel processing gains. The spreading codes are also different on the *I* and *Q* channels for all the users. The spreading codes are assumed to employ rectangular shaped pulses. Different processing gains and spreading codes help us to understand the performance of the system under different conditions. In addition, the data rate on the *I* and *Q* channels is different and user selectable. In simulation, we have considered short spreading codes, i.e., we use the same spreading code on each symbol in a sequence.

The input parameters to the program are defined below. These parameters are user selectable.

- M : Ratio of chip rate on I -channel to chip rate on Q -channel
 M_d : Ratio of data rate on Q -channel to data rate on I -channel
 PGI : Processing gain on I -channel
 K : Number of users
 E_b/N_0 Range

4.1 ***Simulation Transmitter Description***

The random binary data source is generated using the *rand* function of MATLAB, which generates uniform random numbers between 0 and 1 with equal probability. The binary source data is then converted into an antipodal sequence where numbers between 0 and 0.5 are considered as 1, else -1. The input sequence is then serial to parallel converted depending on the input parameter M_d . The spreading code generator is also a random binary generator similar to the source generator (except for the orthogonal code case) and generates random ± 1 vectors of length equal to the processing gain (depending upon I/Q channel) for each source bit. A series of similar steps is performed for all the users. Different bit energies can be selected for all the users. Depending upon the synchronous or asynchronous case, the user signals have random chip delays and a uniformly distributed random phase. For the synchronous orthogonal code case, the *hadamard* function of MATLAB is used to generate orthogonal codes. The output of all the user signal generators is added to the desired user output and transmitted through the channel. The channel can be AWGN or Rayleigh.

We use a conventional receiver for the detection, which consists of a correlator and a comparator. The received signal is despread by multiplying the received signal with the desired user's spreading code, and then integrating (summing) over a symbol time period and threshold decisions are made on the bits. The bit error calculations are performed by comparing these received bit estimates with the transmitted bits for each channel. A bit error is recorded if the difference of the transmitted and received bits is

not zero. The bit error performance of the system for a fixed E_b/N_0 with varying number of interfering users is also calculated. The program files are attached in Appendix-A for reference.

4.2 *Simulation Results*

In this section, we present the results of the MATLAB simulations. The system described in Chapter 2 was simulated as explained in the previous section. The simulation provides the results for performance of the QPSK DS-CDMA in the presence of AWGN and Rayleigh fading channel for single user and multiuser scenarios. The analytical expressions obtained for the bit error rate in Chapter 3 are used to compare with the simulation results. For the multiuser case, synchronous transmission with both orthogonal and random codes is discussed. Also, results for asynchronous transmission with random codes and different energies for each user signal are also shown. Lastly, the BER curves as a function of the number of users (K) are provided.

4.2.1 *Single User*

A single user situation is simulated and results are shown in Figures 4.1 and 4.2 for the I - and Q -channel, respectively. We have used equal data and chip rates on both channels and the processing gain on each channel is 20. The performance of the system in both AWGN and Rayleigh channels is shown. We see that the BER curve is same as that for conventional QPSK in both the AWGN and Rayleigh channels. This simulation result confirms the validity of the simulation.

Figure 4.1: I-channel P_b vs. E_b/N_0 with single user.

Figure 4.2: Q-channel P_b vs. E_b/N_0 with single user.

4.2.2 Synchronous QPSK

In this section, we provide simulation results for synchronous QPSK with orthogonal and random codes. Figure 4.3 and 4.4 show results of synchronous transmission with orthogonal codes. We have simulated a scenario where the chip rate on the I -channel is twice that of Q -channel, and the data rate on Q is three times that of I -channel. There are seven users in this scheme, where six users act as interference to the desired user. Despite the presence of multiple user signals, we see that the BER performance is the same as QPSK in AWGN. Thus, the performance of the system with orthogonal codes does not degrade with multiuser interference, as expected from analysis.

A similar scenario is simulated for synchronous QPSK with random codes. The results are shown in Figures 4.5 and 4.6 for the I - and Q - channel, respectively. Here, we see that the performance of the system does degrade due to multiuser interference. Also, we note that the performance on the Q -channel is worse than the I -channel. This is due to the fact that the Q -channel is relatively more highly loaded than is I -channel, since $PG^I > PG^Q$. Lastly, we observe that the analytical and simulation results are in good agreement for AWGN and Rayleigh channels (Note that the Rayleigh P_b is upper bound, as described in previous chapter).

Several other random code synchronous cases were simulated with varying different parameters. Figures 4.7-4.10 show some of the results. We see from these figures, that as expected, the performance of the system degrades as the number of users, and MUI, increases. In both cases, the performance of the I -channel is better than Q because PG^Q is lower than PG^I , and the load is relatively higher. In addition, in both the cases, the analysis and simulation results agree.

Figure 4.3: I-channel P_b vs. E_b/N_0 with multiple users and orthogonal codes.

Figure 4.4: Q-channel P_b vs. E_b/N_0 with multiple users and orthogonal codes.

Figure 4.5: I-channel P_b vs. E_b/N_0 with 7 users and random codes.

Figure 4.6: Q-channel P_b vs. E_b/N_0 with 7 users and random codes.

Figure 4.7: I-channel P_b vs. E_b/N_0 with 3 users and random codes.

Figure 4.8: Q-channel P_b vs. E_b/N_0 with 3 users and random codes.

Figure 4.9: I-channel P_b vs. E_b/N_0 with 30 users and random codes.

Figure 4.10: Q-channel P_b vs. E_b/N_0 with 30 users and random codes.

4.2.3 Asynchronous QPSK

This section discusses the results obtained for the asynchronous QPSK simulation with random codes. Results for three different cases are shown and discussed below.

- i. In the first case, we have considered three users (2-interfering users), chip rate on I -channel thrice that of the Q -channel ($M = 3$), and data rate on the Q -channel is four times that on the I -channel ($M_d = 4$). The results for the I - and Q -channel are shown in Figures 4.11 and 4.12, respectively. We see from the figures that the Q -channel performance is again worse than the I -channel (as expected). We also observe, as expected, that the system has degraded performance in the Rayleigh fading channel. Lastly, the analytical and simulation results agree for both AWGN and Rayleigh channels.
- ii. In the second case, we consider seven users with $R_c^I = 2R_c^Q$ and $R_s^Q = 3R_s^I$ ($M=2$, $M_d =3$). These results are shown in Figures 4.13 and 4.14. We compare these results with the results for synchronous transmission with random codes as shown in Figure 4.5 and 4.6. We observe that with same parameters, the performance of the system with asynchronous transmission and random codes is similar to the synchronous system with random codes. We also note that since delays in our asynchronous scheme were restricted to be an integer number of chips, this result is as expected. If delays were permitted to be arbitrary real numbers, the asynchronous random-code system performance would be slightly better [1].
- iii. In the third case, the E_b/N_0 of each interfering user is uniformly varied with reference to the desired user's SNR. Specifically, the desired user's SNR is selected as the reference $(E_b/N_0)_{REF}$ and the SNR of the i^{th} user is obtained as $(E_b/N_0)_{REF} + U_i$, where U_i is uniformly distributed in the interval $[-2.5, 2.5]$ and has zero mean. This corresponds to a maximum deviation of E_b/N_0 of approximately ± 4 dB from that of the reference user (a maximum "near-far ratio" of 4 dB). We analyzed the system performance and the analytical and simulation results for this scenario are shown in Figures 4.15

and 4.16. We have considered three users and equal chip and data rates on the I -channel and Q -channels. The performance on both the channels due to different user bit energy can be seen from the figures where for any particular E_b/N_0 on the abscissa (the desired signal SNR), all users have a different E_b/N_0 value. We observe that the I and Q channel bit error rates look nearly identical and analysis is in good agreement with simulations.

Figure 4.11: I-channel P_b vs. E_b/N_0 with 3 users and $PG^{(1)}=120$.

Figure 4.12: Q-channel P_b vs. E_b/N_0 with 3 users and $PG^{(Q)}=10$.

Figure 4.13: I-channel P_b vs. E_b/N_0 with 7 users and $PG^{(I)}=90$.

Figure 4.14: Q channel P_b vs. E_b/N_0 with 7 users and $PG^{(Q)} = 15$.

Figure 4.15: I-channel P_b vs. E_b/N_0 with 3 users, $PG^{(I)} = 20$ and interfering users energies uniformly distributed w.r.t. desired user energy.

Figure 4.16: Q-channel P_b vs. E_b/N_0 with 3-users, $PG^{(Q)}=20$ and interfering users energies uniformly distributed w.r.t. desired user energy.

4.2.4 BER with Varying System Load

In this section, the results obtained for the system performance as the load factor K/N increases for asynchronous case is discussed. We obtain bit error probability values for a constant the E_b/N_0 by varying the number of users. Error probability results for different numbers of users (1, 3, 5, 7, 15 and, 30) are plotted for E_b/N_0 of 2 dB and 5 dB for both AWGN and Rayleigh channels. The results for the I -channel are shown in Figure 4.17 and 4.18 and the results for the Q -channel are shown in Figures 4.19 and 4.20. From all these plots, we observe that the performance of the system degrades linearly with an increase in the number of users on the I -channel in both AWGN and Rayleigh channels. However, on the Q -channel the performance degradation is much faster than on the I -channel as $PG^Q < PG^I$. We also observe some variations in the simulation curves when the number of users is small; this is due to an insufficient number

of trials used for averaging. Thus, we conclude that with a larger processing gain, the degradation of the system performance with increasing load is not as much as with lower processing gain, as expected.

Figure 4.17: I-channel BER vs. the number of users, AWGN channel.

Figure 4.18: I-channel BER vs. the number of users, Rayleigh channel.

Figure 4.19: Q-channel BER vs. the number of users, AWGN channel.

Figure 4.20: Q-channel BER vs. the number of users, Rayleigh channel.

We summarize the observations made in this chapter. There are five main observations:

1. The analytical and simulation results agree, validating both.
2. A synchronous QPSK system performs better with orthogonal codes than with random codes due to minimal or zero multiuser interference.
3. Performance of a QPSK DS CDMA system with synchronous transmission using random codes is similar to that for asynchronous transmission with random codes.
4. The bit error rate increases as the number of user increases.
5. The rate of increase of error probability is inversely proportional to the processing gain.

Chapter 5: Conclusions and Future Work

In this chapter, we summarize the whole thesis. We recap in this chapter the research performed in this thesis and the conclusions obtained. In addition, topics for future research are also suggested.

5.1 Summary and Conclusions

In this research, we have examined the performance of a QPSK DS-CDMA system with disparate chip and data rates on the two quadrature channels. The effect of disparate chip and data rates (thus different processing gains) on the I - and Q -channels and different user signal energies was analyzed. The performance of the system in the presence of interfering users in AWGN and Rayleigh fading channels was evaluated both by analysis and by simulation. Two types of transmission schemes, i.e., synchronous (with equal time delays and phase shifts for all user signals) and asynchronous (with random unequal delays and phase shifts) were considered. Analysis for the synchronous case was performed using orthogonal and random codes to estimate the effect on system performance due to interfering users. Simulation and analytical results were presented for a single user case, multiple user cases, same/different data rate cases, and same/different chip rate cases. In addition, plots for power spectra were provided to illustrate main lobe bandwidth, spectral efficiency, and the moderate amount of spectrum shaping attainable with our technique. Excellent agreement between the simulation results and the analytical expressions were found. The main findings of our work are:

- 1) Our system differs from other systems in that it provides another method for obtaining variable/selectable data rates, chip rate and hence processing gains.
- 2) We achieve a moderate amount of spectral shaping and in some cases a reduction in the side lobes by varying the parameters. The power spectrum of the composite

waveform (the resultant of the I -channel and Q -channel spectra) can be varied by varying the parameters (M and M_d).

- 3) The performance of a synchronous QPSK system with orthogonal codes is better than that of a synchronous system with random codes due to minimal or zero multiuser interference. Thus, we observed and illustrated the importance of code selection.
- 4) Synchronous transmission using random codes offers system performance similar to asynchronous transmission with random codes.
- 5) Bit error rate (BER) is a function of the load, i.e., the number of interfering users in the system. It was observed that the bit error rate increases as the number of user increases and this rate of increase of error probability is inversely proportional to the processing gain.

5.2 *Future Work*

In this section we provide a few suggestions for future extension of this work.

- 1) First, in this research we have considered different data and chip rates on the I - and Q - channels for any given user. However, all the users have same data/chip rate on their I - channels and Q -channels. A likely extension of our work would be to consider different data and chip rates on I - and Q - channels of all the users. This would help us to observe the system performance with different parameters for each user, and would also constitute a system where all user signals do not have the same bandwidth.
- 2) Second, we have assumed that the chip delays are such that they are aligned at the chip boundaries. That is, the number of chips by which the user i 's signal is shifted is integer multiple of the chip period.

$$\tau_i = L_i T_c^{(l)} + \varepsilon_i \quad \text{where } \varepsilon_i = 0.$$

This case represents the worst case. However, it would be appropriate to consider chip alignments not always at the chip boundary with $\varepsilon_i \neq 0$. We would expect asynchronous system performance to improve slightly in this case, as has been observed in conventional QPSK DS-SS CDMA schemes.

- 3) Design explicitly *for* spectral shaping by selection of appropriate values of M, M_d, E_b^I , and E_b^Q .

References

- [1] B. Sklar, *Digital Communications: Fundamentals and Applications*, 2nd ed., Prentice-Hall, Upper Saddle River, New Jersey, 1995.
- [2] T. S. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed., Prentice Hall, New York, December 2001.
- [3] S. Hara, R. Prasad, "Overview of Multicarrier CDMA," *IEEE Communications Magazine*, vol. 35, no. 12, pp. 126-133, December 1997.
- [4] T. Ottosson and A. Svensson, "Performance of Different Multi-rate Schemes in DS/CDMA Systems," *Proceedings of NRS Seminar on Radio Communication Networks*, Linköping, Sweden, October 1994, pp. 15-18.
- [5] D. W. Matolak, W. Xiong, "Spectrally Shaped Generalized Multitone Direct-Sequence Spread Spectrum," submitted to *IEEE Transactions on Vehicular Technology*, Feb. 2004.
- [6] R. Lupas and S. Verdù, "Near-far resistance of multiuser detectors in asynchronous channels," *IEEE Transactions on Communications*, vol. 38, pp. 496-508, April 1990.
- [7] J. Zou, V. K. Bhargava, and Q. Wang, "Reverse Link Interference Modeling and Outage Analysis of DS/CDMA Cellular Systems," *Wireless Personal Communications*, vol. 2, pp. 189-215, Oct. 1995.
- [8] D. Lee, H. Lee, and L. B. Milstein, "Direct Sequence Spread Spectrum Walsh-QPSK Modulation," *IEEE Transaction on Communications*, vol. 46, no. 9, pp. 1227-1232, Sept. 1998.
- [9] H. Kusaka, M. Kominami, K. Matsumoto, and Y. Tashiro, "Performance improvement of CDMA/QPSK systems with nonlinear channel using decision feedback equalizers" *Proceedings of IEEE 1998 Conference on Spread Spectrum Techniques and Applications*, vol. 3, pp. 812 – 817, 1998.
- [10] S. Tseng and M. R. Bell, "Asynchronous Multicarrier DS-CDMA Using Mutually Orthogonal Complementary Sets of Sequences," *IEEE Transactions on Communications*, vol. 48, no. 1, pp. 53-59, January 2000.

- [11] A. Y Wong and V. C. M. Leung, "Code-phase-shift keying: a power and bandwidth efficient spread spectrum signaling technique for wireless local area network applications" *Proceedings of IEEE 1997 Canadian Conference on Electrical and Computer Engineering*, vol. 2, pp. 478 – 481, 1997.
- [12] A. N. Shiryayev, *Probability*, Springer-Verlag, New York, NY, 1984.
- [13] S. Parkvall, "Variability of user performance in cellular DS-CDMA – long vs short spreading sequences" *Proceedings of IEEE 1998 Conference on Spread Spectrum Techniques and Applications*, vol. 3, pp. 681 – 685, 1998.
- [14] A. Ahlén, M. Sternad, C. Tidestav, and M. Nicoli, "Research on multi-user detectors for cdma based cellular systems" Signals and Systems Group, Uppsala University. Retrieved from <http://www.signal.uu.se/Research/rmultiuser.html>
- [15] J.G. Proakis, *Digital Communications*, 3rd ed., McGraw-Hill, New York, 1995.
- [16] R. L. Peterson, R. E. Ziemer, D. E. Borth, *Introduction to Spread Spectrum Communications*, Prentice Hall, Upper Saddle River, New Jersey, 1995.
- [17] T. Luo, S. Paupathy, and E. S. Sousa, "Interference Control and Chip Waveform Design in Multirate DS-CDMA Communication Systems", *IEEE Transactions on Wireless Communications*, vol. 1, no. 1, pp. 56-66, Jan 2002.

Appendix: MATLAB Programs

```

%-----
% Asynchronous QPSK DS CDMA
% In this program we evaluate the BER for Multiuser QPSK system
% with varying M, Md, PGI, and PGQ for AWGN and Rayleigh Channel
%-----

clear all; close all; clc;
Eb=100;
Nav=1;
EbN0_min = 0 ; EbN0_inc = 1; EbN0_max = 8;
EbN0=[EbN0_min: EbN0_inc: EbN0_max]; % EbN0 db vector
EbN0W=10.^(EbN0/10); % EbN0 in value
PbA1=erfc(sqrt(EbN0W))/2; % BER analytical
Pb_ray = 0.5 * (1-sqrt(EbN0W./(1+EbN0W)));
Ne=100; % number of errors to count
Nb1=ceil(Ne./PbA1); % Number of bits for constant error bits
blk = 100;
Nb = blkdiv(Nb1,blk)
PGI=90; % I-channel processing gain
M=2; % Rci/Rcq
Md=3; % Rsq/Rsi
Rsi=1; %input('enter integer value of Rsi: ');
users= input('enter total number of users: ');
PGQ = round(PGI/(M*Md)); % Q-channel processing gain

%Generate energy for each user and for I and Q channel
val=Eb/2;
Ebi(1,:)=energy(Eb, val, users)
Ebq(1,:)=energy(Eb, val, users)
PbS = zeros(1, length(EbN0));PbSi = zeros(1, length(EbN0));PbSq = zeros(1, length(EbN0));
ray_PbS = zeros(1, length(EbN0));ray_PbSi = zeros(1, length(EbN0));ray_PbSq = zeros(1,
length(EbN0));
for aver = 1:Nav
    inc=0;aver
    for counter= EbN0_min: EbN0_inc: EbN0_max;
        pack; inc=inc+1
        no_blk = Nb(1,inc)/blk;
        PbSt2=0; PbSti2=0; PbStq2=0; ray_PbSt2=0; ray_PbSti2=0; ray_PbStq2=0;
        for count_blk = 1:1:no_blk
            b=zeros((users), blk);
            PbSt=0; PbSti=0; PbStq=0;
            ray_PbSt=0; ray_PbSti=0; ray_PbStq=0;
            Lvi=((blk*PGI)/(Md+1));
            Lvq=((blk*Md*PGQ)/(Md+1));

```

```

deni = (lcm(Lvi,Lvq))/Lvi;
denq = (lcm(Lvi,Lvq))/Lvq;
bios=zeros((users), Lvi);
bqos=zeros((users), Lvq);
binary_bi=zeros((users), (blk/(Md+1)));
binary_bq=zeros((users), (blk*Md/(Md+1)));
side_transmit = zeros((users), Lvi*deni);
transmit=0; bi = binary_bi; bq = binary_bq;
chipios = bios; chipqos = bqos; xbi = bios; x bq = bqos; st = bios; ang_transmit = bios;
for MUI = 1:1:(users)
    chipi(MUI,:) = 2*BINO1((PGI),0.5)-1;
    chipq(MUI,:) = 2*BINO1((PGQ),0.5)-1;
    b(MUI,:) = BINO1(blk,0.5);
    temp5 = zeros(MUI, Lvi); temp6 = zeros(MUI, Lvq);
    k=size(b(MUI,:));
    temp1=1; temp2=1;
    for count=1:1:k(2);
        if(mod(count,(Md+1))~= 1)
            binary_bq(MUI, temp2)=b(MUI, count); temp2=temp2+1;
        else
            binary_bi(MUI, temp1)=b(MUI, count); temp1=temp1+1;
        end
    end
    bi(MUI, :) = 1-2*binary_bi(MUI, :);
    bq(MUI, :) = 1-2*binary_bq(MUI, :);
    bios(MUI,:) = OverN(bi(MUI,:),PGI);
    bqos(MUI,:) = OverN(bq(MUI,:),PGQ);
    chipios(MUI,:) = repmat(chipi(MUI,:),1,(Lvi/PGI));
    chipqos(MUI,:) = repmat(chipq(MUI,:),1,(Lvq/PGQ));
    xbi(MUI,:) = chipios(MUI,:).*bios(MUI,:);
    x bq(MUI,:) = chipqos(MUI,:).*bqos(MUI,:);
    Li = delay(users, PGI,M); % Calculate the shift
    Lq = delay1(users, PGQ,Md); % Calculate the shift
    xbi(MUI, :) = shift(xbi(MUI, :), Li(1,MUI)); % Shift input vector if not the desired user
    x bq(MUI, :) = shift(x bq(MUI, :), Lq(1,MUI)); % Shift input vector if not the desired user
    x bios(MUI,:) = repmat(xbi(MUI,:),1,deni);
    x bqos(MUI,:) = repmat(x bq(MUI,:),1,denq);
    st(MUI,:) = x bios(MUI,:)+j*x bqos(MUI,:);
    clear bi bq bios bqos chipios chipqos temp1 temp2 xbi x bq x bios x bqos
    ang_transmit(MUI,:) = angle(st(MUI,:));
    if MUI == 1
        side_transmit(MUI,:) = sqrt(2*Ebi(1,MUI))*((cos(ang_transmit(MUI,:)))/(sqrt(PGI))) +
            j*sqrt(2*Ebq(1,MUI))*((sin(ang_transmit(MUI,:)))/(sqrt(PGQ)));
    else
        phi = 2*pi*rand(1,1);
        side_transmit(MUI,:) = sqrt(2*Ebi(1,MUI))/(sqrt(PGI))*[cos(ang_transmit(MUI,:))*cos(phi)+sin(a
            ng_transmit(MUI,:))*sin(phi)]+j*sqrt(2*Ebq(1,MUI))/(sqrt(PGQ))*[sin(ang_transmit(MUI,:))*co
            s(phi)- cos(ang_transmit(MUI,:))*sin(phi)];
    end
end

```

```

        transmit = transmit + side_transmit(MUI, : );
    end
    noise=randn(1,Lvi*deni)*sqrt(Eb/2/(10^(counter/10)))+j*(randn(1,Lvq*denq)*sqrt(Eb/2/(10^(counter/10))));
    rayi1=rayleigh1(Lvi/PGI,0);          rayi1 = rayi1(1:Lvi/PGI);
    rayq1=rayleigh1(Lvq/PGQ,0);          rayq1 = rayq1(1:Lvq/PGQ);
    Rayi1=OverN(rayi1,PGI); Rayi1 = repmat(Rayi1, 1, deni);
    Rayq1=OverN(rayq1,PGQ); Rayq1 = repmat(Rayq1, 1, denq);
    Txil=Rayi1.*real(transmit);          Txq1=Rayq1.*imag(transmit);
    ray_transmit = Txil + j*Txq1 ;        ray_rec= ray_transmit+noise;
    rec= transmit+noise;
    clear side_transmit %ang_transmit noise
    ray_bit_i = zeros(deni,(Lvi/PGI));    ray_bit_q = zeros(denq,(Lvq/PGQ));
    bit_i = zeros(deni,(Lvi/PGI));        bit_q = zeros(denq,(Lvq/PGQ));
    ray_rec_real = (reshape(real(ray_rec), Lvi,deni))* (sqrt(PGI/(2*Eb)));
    rec_real = (reshape(real(rec), Lvi,deni))* (sqrt(PGI/(2*Eb)));
    ray_rec_imag = (reshape(imag(ray_rec), Lvq,denq))* (sqrt(PGQ/(2*Eb)));
    rec_imag = (reshape(imag(rec), Lvq,denq))* (sqrt(PGQ/(2*Eb)));
    rec_length= (deni*denq)*((Lvi/PGI) + (Lvq/PGQ));
    ray_newbit_i = demodulate(ray_rec_real,PGI, chipi);
    newbit_i = demodulate(rec_real,PGI, chipi);
    ray_newbit_q = demodulate(ray_rec_imag,PGQ, chipq);
    newbit_q = demodulate(rec_imag,PGQ, chipq);
    ray_newbiti=repmat(ray_newbit_i,1,denq);
    newbiti=repmat(newbit_i,1,denq);
    ray_newbitq=repmat(ray_newbit_q,1,deni);
    newbitq=repmat(newbit_q,1,deni);
    clear newbit_i newbit_q ray_newbit_i ray_newbit_q
    ray_output_bit = combine(ray_newbiti, ray_newbitq, Md);
    output_bit = combine(newbiti, newbitq, Md);
    inputbit = repmat(b(1,:),1,(deni*denq));
    inputbitI = repmat(binary_bi(1,:),1,(deni*denq));
    inputbitQ = repmat(binary_bq(1,:),1,(deni*denq));
    ray_output_error=abs(inputbit-ray_output_bit);
    output_error=abs(inputbit-output_bit);
    ray_error_i=abs(inputbitI-ray_newbiti);
    error_i=abs(inputbitI-newbiti);
    ray_error_q=abs(inputbitQ-ray_newbitq);
    error_q=abs(inputbitQ-newbitq);
    ray_PbSt=sum(ray_output_error)/((deni*denq)*(blk));
    PbSt=sum(output_error)/((deni*denq)*(blk));
    ray_PbSti=sum(ray_error_i)/((deni*denq)*(Lvi/PGI));
    PbSti=sum(error_i)/((deni*denq)*(Lvi/PGI));
    ray_PbStq=sum(ray_error_q)/((deni*denq)*(Lvq/PGQ));
    PbStq=sum(error_q)/((deni*denq)*(Lvq/PGQ));
    ray_PbSt2=ray_PbSt2 + ray_PbSt; ray_PbSti2=ray_PbSti2 + ray_PbSti;
    ray_PbStq2=ray_PbStq2 + ray_PbStq;
    PbSt2=PbSt2 + PbSt; PbSti2=PbSti2 + PbSti; PbStq2=PbStq2 + PbStq;
end;

```

```

ray_PbS_inc(1,inc)=ray_PbSt2/no_blk; ray_PbSi_inc(1,inc)=ray_PbSti2/no_blk;
ray_PbSq_inc(1,inc)=ray_PbStq2/no_blk;
PbS_inc(1,inc)=PbSt2/no_blk; PbSi_inc(1,inc)=PbSti2/no_blk;
PbSq_inc(1,inc)=PbStq2/no_blk;
end;
ray_PbS = ray_PbS + ray_PbS_inc; ray_PbSi = ray_PbSi + ray_PbSi_inc;
ray_PbSq = ray_PbSq + ray_PbSq_inc;
PbS = PbS + PbS_inc; PbSi = PbSi + PbSi_inc; PbSq = PbSq + PbSq_inc;
end
PbS = PbS/Nav; PbSi = PbSi/Nav; PbSq = PbSq/Nav;
ray_PbS = ray_PbS/Nav; ray_PbSi = ray_PbSi/Nav; ray_PbSq = ray_PbSq/Nav;

% Analytical expression for MUI
N0 = Eb./EbN0W;
mul_veci = (1/2)*(2)*(1/PGI*sum(Ebi(2:1:length(Ebi)))+1/PGQ*sum(Ebq(2:1:length(Ebq))));
mul_vecq = (1/2)*(2)*(1/PGQ*sum(Ebq(2:1:length(Ebq)))+1/PGI*sum(Ebi(2:1:length(Ebi))));
raymul_veci =
(1/3)*(2)*(1/PGI*sum(Ebi(2:1:length(Ebi)))+Md/(PGI)*sum(Ebq(2:1:length(Ebq))));
raymul_vecq =
(1/3)*(2)*(1/PGQ*sum(Ebq(2:1:length(Ebq)))+M/PGQ*sum(Ebi(2:1:length(Ebi))));
for countN = 1:1:length(N0)
    PbIA(1,countN) = 0.5*erfc(sqrt((2*Ebi(1,1))/(2*(N0(1,countN)+(raymul_veci))));
    PbQA(1,countN) = 0.5*erfc(sqrt((2*Ebq(1,1))/(2*(N0(1,countN)+ raymul_vecq))));
    ray_PbIA(1,countN) = 0.5*(1-sqrt((Ebi(1,1))/(Ebi(1,1)+(N0(1,countN)+(raymul_veci))));
    ray_PbQA(1,countN) = 0.5*(1-sqrt((Ebq(1,1))/(Ebq(1,1)+(N0(1,countN)+ raymul_vecq))));
end
figure(1)
semilogy(EbN0,PbA1,'-', EbN0,PbIA,'-x', EbN0,PbSi,'--', EbN0,Pb_ray,'-
*',EbN0,ray_PbIA,'-+',EbN0,ray_PbSi,'-o', 'Linewidth',2);grid;
xlabel('E_b/N_0, dB'); ylabel('P_b: Bit error rate')
title('P_b vs. E_b/N_0 for I-channel DS-SS QPSK')
legend('Single User AWGN','MU AWGN Anal', 'MU AWGN Sim','Single User Rayleigh
Anal','MU Rayleigh Anal','MU Rayleigh Sim');
figure(2)
semilogy(EbN0,PbA1,'-', EbN0,PbQA,'-x', EbN0,PbSq,'--', EbN0,Pb_ray,'-
*',EbN0,ray_PbQA,'-+',EbN0,ray_PbSq,'-o', 'Linewidth',2);grid;
xlabel('E_b/N_0, dB'); ylabel('P_b: Bit error rate')
title('P_b vs. E_b/N_0 for Q-channel DS-SS QPSK')
legend('Single User AWGN','MU AWGN Anal', 'MU AWGN Sim','Single User Rayleigh
Anal','MU Rayleigh Anal','MU Rayleigh Sim');

```

```

%-----
% Synchronous QPSK DS CDMA
% In this program we evaluate the BER for Multiuser QPSK system
% with varying M, Md, PGI, and PGQ for AWGN and Rayleigh Channel
%-----
clear all; close all; clc;
Eb=100;
Nav=3;
EbN0_min = 0 ; EbN0_inc = 1; EbN0_max = 8;
EbN0=[EbN0_min: EbN0_inc: EbN0_max]; % EbN0 db vector
EbN0W=10.^(EbN0/10); % EbN0 in value
PbA1=erfc(sqrt(EbN0W))/2; % BER analytical
Pb_ray = 0.5 * (1-sqrt(EbN0W./(1+EbN0W)));
Ne=100; % number of errors to count
Nb1=ceil(Ne./PbA1); % Number of bits for constant error bits
blk = 100;
Nb = blkdiv(Nb1,blk);
PGI= 96; % I-channel processing gain
M=2; % Rci/Rcq
Md=3; % Rsq/Rsi
users= 7;
PGQ = round(PGI/(M*Md)); % Q-channel processing gain
Rsq = Md * Rsi;
%Generate energy for each user and for I and Q channel
val=0;
Ebi(1,:)=energy(Eb, val, users); Ebq(1,:)=energy(Eb, val, users);
PbS = zeros(1, length(EbN0));PbSi = zeros(1, length(EbN0));PbSq = zeros(1, length(EbN0));
ray_PbS = zeros(1, length(EbN0));ray_PbSi = zeros(1, length(EbN0));ray_PbSq = zeros(1,
length(EbN0));
for aver = 1:Nav
    inc=0;aver
    for counter= EbN0_min: EbN0_inc: EbN0_max;
        pack; inc=inc+1
        no_blk = Nb(1,inc)/blk;
        PbSt2=0; PbSti2=0; PbStq2=0;
        ray_PbSt2=0; ray_PbSti2=0; ray_PbStq2=0;
        for count_blk = 1:1:no_blk
            b=zeros((users), blk);
            PbSt=0; PbSti=0; PbStq=0;
            ray_PbSt=0; ray_PbSti=0; ray_PbStq=0;
            Lvi=((blk*PGI)/(Md+1));
            Lvq=((blk*Md*PGQ)/(Md+1));
            deni = (lcm(Lvi,Lvq))/Lvi;
            denq = (lcm(Lvi,Lvq))/Lvq;
            bios=zeros((users), Lvi);
            bqos=zeros((users), Lvq);
            binary_bi=zeros((users), (blk/(Md+1)));
            binary_bq=zeros((users), (blk*Md/(Md+1)));

```

```

side_transmit = zeros((users), Lvi*deni);
transmit=0; bi = binary_bi; bq = binary_bq;
chipios = bios; chipqos = bqos; xbi = bios; x bq = bqos; st = bios; ang_transmit = bios;
for MUI = 1:1:(users)
    temp1 = hadamard(PGI);
    temp2 = hadamard(PGQ);
    chipi(MUI,:)= temp1(MUI+1,:);
    chipq(MUI,:)= temp2(MUI+1,:);
    %For random codes use the following:
    %chipi(MUI,: )=2*BINO1((PGI),0.5)-1;
    %chipq(MUI,: )=2*BINO1((PGQ),0.5)-1;
    b(MUI,:)=BIN01(blk,0.5);
    temp5 = zeros(MUI, Lvi); temp6 = zeros(MUI, Lvq);
    k=size(b(MUI,:));
    temp1=1; temp2=1;
    for count=1:1:k(2);
        if(mod(count,(Md+1))~= 1)
            binary_bq(MUI, temp2)=b(MUI, count); temp2=temp2+1;
        else
            binary_bi(MUI, temp1)=b(MUI, count); temp1=temp1+1;
        end
    end
    bi(MUI, :)=1-2*binary_bi(MUI, :);
    bq(MUI, :)=1-2*binary_bq(MUI, :);
    bios(MUI,:)=OverN(bi(MUI,:),PGI);
    bqos(MUI,:)=OverN(bq(MUI,:),PGQ);
    chipios(MUI,:)=repmat(chipi(MUI,:),1,(Lvi/PGI));
    chipqos(MUI,:)=repmat(chipq(MUI,:),1,(Lvq/PGQ));
    xbi(MUI,:)=chipios(MUI,:).*bios(MUI,:);
    x bq(MUI,:)=chipqos(MUI,:).*bqos(MUI,:);
    Li = zeros(1,users-1); Lq = zeros(1,users-1);
    Li = [0 Li]; Lq = [0 Lq];
    xbi(MUI, :) = shift(xbi(MUI, :), Li(1,MUI));    % Shift input vector if not the desired
user
    x bq(MUI, :) = shift(x bq(MUI, :), Lq(1,MUI));    % Shift input vector if not the desired
user
    x bios(MUI,:) = repmat(xbi(MUI,:),1,deni);
    x bqos(MUI,:) = repmat(x bq(MUI,:),1,denq);
    st(MUI,:) = x bios(MUI,:)+j*x bqos(MUI,:);
    clear bi bq bios bqos chipios chipqos temp1 temp2 xbi x bq x bios x bqos
    ang_transmit(MUI,:)= angle(st(MUI,:));
    ang_transmit(MUI,:)= angle(st(MUI,:));
    side_transmit(MUI,:)=sqrt(2*Ebi(1,MUI))*((cos(ang_transmit(MUI,:)))/(sqrt(PGI))) +
        j*sqrt(2*Ebq(1,MUI))*((sin(ang_transmit(MUI,:)))/(sqrt(PGQ)));
    transmit = transmit + side_transmit(MUI, : );
end
noise=randn(1,Lvi*deni)*sqrt(Eb/2/(10^(counter/10)))+j*(randn(1,Lvq*denq)*
    sqrt(Eb/2/(10^(counter/10))));
rayi1=rayleigh1(Lvi/PGI,0);

```

```

rayi1 = rayi1(1:Lvi/PGI);
rayq1=rayleigh1(Lvq/PGQ,0);
rayq1 = rayq1(1:Lvq/PGQ);
Rayi1=OverN(rayi1,PGI); Rayi1 = repmat(Rayi1, 1, deni);
Rayq1=OverN(rayq1,PGQ); Rayq1 = repmat(Rayq1, 1, denq);
Tx11=Rayi1.*real(transmit);
Txq1=Rayq1.*imag(transmit);
ray_transmit = Tx11 + j*Txq1 ;
ray_rec= ray_transmit+noise;
rec= transmit+noise;
clear side_transmit ang_transmit noise
ray_bit_i = zeros(deni,(Lvi/PGI));
ray_bit_q = zeros(denq,(Lvq/PGQ));
bit_i = zeros(deni,(Lvi/PGI));
bit_q = zeros(denq,(Lvq/PGQ));
ray_rec_real1 = (reshape(real(ray_rec), Lvi,deni))* (sqrt(PGI/(2*Eb)));
rec_real1 = (reshape(real(rec), Lvi,deni))* (sqrt(PGI/(2*Eb)));
ray_rec_imagq = (reshape(imag(ray_rec), Lvq,denq))* (sqrt(PGQ/(2*Eb)));
rec_imagq = (reshape(imag(rec), Lvq,denq))* (sqrt(PGQ/(2*Eb)));
rec_length= (deni*denq)*((Lvi/PGI) + (Lvq/PGQ));
ray_newbit_i = demodulate(ray_rec_real1,PGI, chipi);
newbit_i = demodulate(rec_real1,PGI, chipi);
ray_newbit_q = demodulate(ray_rec_imagq,PGQ, chipq);
newbit_q = demodulate(rec_imagq,PGQ, chipq);
ray_newbiti=repmat(ray_newbit_i,1,denq);
newbiti=repmat(newbit_i,1,denq);
ray_newbitq=repmat(ray_newbit_q,1,deni);
newbitq=repmat(newbit_q,1,deni);
clear newbit_i newbit_q ray_newbit_i ray_newbit_q
ray_output_bit = combine(ray_newbiti, ray_newbitq, Md);
output_bit = combine(newbiti, newbitq, Md);
inputbit = repmat(b(1,:),1,(deni*denq));
inputbitI = repmat(binary_bi(1,:),1,(deni*denq));
inputbitQ = repmat(binary_bq(1,:),1,(deni*denq));
ray_output_error=abs(inputbit-ray_output_bit);
output_error=abs(inputbit-output_bit);
ray_error_i=abs(inputbitI-ray_newbiti);
error_i=abs(inputbitI-newbiti);
ray_error_q=abs(inputbitQ-ray_newbitq);
error_q=abs(inputbitQ-newbitq);
ray_PbSt=sum(ray_output_error)/((deni*denq)*(blk));
PbSt=sum(output_error)/((deni*denq)*(blk));
ray_PbSti=sum(ray_error_i)/((deni*denq)*(Lvi/PGI));
PbSti=sum(error_i)/((deni*denq)*(Lvi/PGI));
ray_PbStq=sum(ray_error_q)/((deni*denq)*(Lvq/PGQ));
PbStq=sum(error_q)/((deni*denq)*(Lvq/PGQ));
ray_PbSt2=ray_PbSt2 + ray_PbSt; ray_PbSti2=ray_PbSti2 + ray_PbSti;
ray_PbStq2=ray_PbStq2 + ray_PbStq;
PbSt2=PbSt2 + PbSt; PbSti2=PbSti2 + PbSti; PbStq2=PbStq2 + PbStq;

```



```

end;
ray_PbS_inc(1,inc)=ray_PbSt2/no_blk; ray_PbSi_inc(1,inc)=ray_PbSti2/no_blk;
ray_PbSq_inc(1,inc)=ray_PbStq2/no_blk;
PbS_inc(1,inc)=PbSt2/no_blk; PbSi_inc(1,inc)=PbSti2/no_blk;
PbSq_inc(1,inc)=PbStq2/no_blk;
toc
end;
ray_PbS = ray_PbS + ray_PbS_inc; ray_PbSi = ray_PbSi + ray_PbSi_inc;
ray_PbSq = ray_PbSq + ray_PbSq_inc;
PbS = PbS + PbS_inc; PbSi = PbSi + PbSi_inc; PbSq = PbSq + PbSq_inc;
end
PbS = PbS/Nav; PbSi = PbSi/Nav; PbSq = PbSq/Nav;
ray_PbS = ray_PbS/Nav; ray_PbSi = ray_PbSi/Nav; ray_PbSq = ray_PbSq/Nav;

% Analytical expression for MUI
N0 = Eb./EbN0W;
mul_veci = (2/PGI)*sum(Ebi(2:1:length(Ebi)));;
mul_vecq = (2/PGQ)*sum(Ebq(2:1:length(Ebq)));;
for countN = 1:1:length(N0)
    PbIA(1,countN) = 0.5*erfc(sqrt((2*Ebi(1,1))/(2*(N0(1,countN)))));
    PbQA(1,countN) = 0.5*erfc(sqrt((2*Ebq(1,1))/(2*(N0(1,countN)))));
    ray_PbIA(1,countN) = 0.5*(1-sqrt((Ebi(1,1))/(Ebi(1,1)+(N0(1,countN)))));
    ray_PbQA(1,countN) = 0.5*(1-sqrt((Ebq(1,1))/(Ebq(1,1)+(N0(1,countN)))));
    %With Random Codes use the following
    %ray_PbIA(1,countN) = 0.5*(1-sqrt((Ebi(1,1))/(Ebi(1,1)+(N0(1,countN)+mul_veci))));
    %ray_PbQA(1,countN) = 0.5*(1-sqrt((Ebq(1,1))/(Ebq(1,1)+(N0(1,countN)+mul_vecq))));
end
figure(1)
semilogy(EbN0,PbA1,'-', EbN0,PbSi,'-o', EbN0,Pb_ray,'-*',EbN0,ray_PbSi,'-p',
'Linewidth',2);grid;
xlabel('E_b/N_0, dB'); ylabel('P_b: Bit error rate')
title('P_b vs. E_b/N_0 for I-channel DS-SS QPSK')
legend('Anal AWGN','MU AWGN Sim','Rayleigh Anal','MU Rayleigh Sim');
figure(2)
semilogy(EbN0,PbA1,'-', EbN0,PbSq,'-o', EbN0,Pb_ray,'-*',EbN0,ray_PbSq,'-
p','Linewidth',2);grid;
xlabel('E_b/N_0, dB'); ylabel('P_b: Bit error rate')
title('P_b vs. E_b/N_0 for Q-channel DS-SS QPSK')
legend('Anal AWGN','MU AWGN Sim','Rayleigh Anal','MU Rayleigh Sim');

% -----
% Program for Power Spectrum
% -----
clear all; close all; clc;
Eb=100;Ebic = 100; Ebqc = 100;Nav=5;
blk = 100;
Nb1 = 10000;

```

```

Nb = blkdiv(Nb1,blk);
PGI= 60; % I-channel processing gain
M=2; % Rci/Rcq
Md=3; % Rsq/Rsi such that (Md+1) is div by 100 & Md*M is
div by PGI.
users= 1; % total number of users
PGQ = round(PGI/(M*Md)); % Q-channel processing gain

%Generate energy for each user and for I and Q channel
Ebi(1,:) = energy(Eb, Eb/2, users); %mean = Eb, and std dev = Eb/2
Ebq(1,:) = 7.39 * Ebi(1,:);

ss1 =1000;fsTs=4;val=20;
vspecta=zeros(Nav,ss1);vspect1a=zeros(Nav,ss1);
vspect2a=zeros(Nav,ss1); % initialize spectrum vector for v
ff=0:fsTs/(val*ss1):fsTs/val - fsTs/(val*ss1); % frequency vector for plotting
no_blk = Nb/blk;
for aver = 1:1:Nav
    i=0; aver
    for count_blk = 1:1:no_blk
        i = i+1;
        b=zeros((users), blk);
        PbSt=0; PbSti=0; PbStq=0;
        Lvi=((blk*PGI)/(Md+1));
        Lvq=((blk*Md*PGQ)/(Md+1));
        deni = (lcm(Lvi,Lvq))/Lvi;
        denq = (lcm(Lvi,Lvq))/Lvq;
        bios=zeros((users), Lvi);
        bqos=zeros((users), Lvq);
        binary_bi=zeros((users), (blk/(Md+1)));
        binary_bq=zeros((users), (blk*Md/(Md+1)));
        side_transmit = zeros((users), Lvi*deni*fsTs);
        transmit = zeros(1, length(side_transmit));
        bi = binary_bi; bq = binary_bq;
        chipios = bios; chipqos = bqos; xbi = bios; x bq = bqos;
        st = zeros(users,length(bios)*fsTs); bios; ang_transmit = st;
        for MUI = 1:1:(users)
            chipi(MUI,:)=2*BINO1((PGI),0.5)-1;
            chipq(MUI,:)=2*BINO1((PGQ),0.5)-1;
            b(MUI,:)=BIN01(blk,0.5);
            temp5 = zeros(MUI, Lvi);
            temp6 = zeros(MUI, Lvq);
            k=size(b(MUI,:));
            temp1=1; temp2=1;
            for count=1:1:k(2);
                if(mod(count,(Md+1))~= 1)
                    binary_bq(MUI, temp2)=b(MUI, count); temp2=temp2+1;
                else
                    binary_bi(MUI, temp1)=b(MUI, count); temp1=temp1+1;

```

```

        end
    end
    bi(MUI, :)=1-2*binary_bi(MUI, :);
    bq(MUI, :)=1-2*binary_bq(MUI, :);
    bios(MUI,:)=OverN(bi(MUI,:),PGI);
    bqos(MUI,:)=OverN(bq(MUI,:),PGQ);
    chipios(MUI,:)=repmat(chipi(MUI,:),1,(Lvi/PGI));
    chipqos(MUI,:)=repmat(chipq(MUI,:),1,(Lvq/PGQ));
    xbi(MUI,:)=chipios(MUI,:).*bios(MUI,:);
    x bq(MUI,:)=chipqos(MUI,:).*bqos(MUI,:);
    Li = delay(users, PGI);
    Lq = delay(users, PGQ);
    xbi(MUI, :) = shift(xbi(MUI, :), Li(1,MUI));
    x bq(MUI, :) = shift(x bq(MUI, :), Lq(1,MUI));
    xbios(MUI,:) = repmat(xbi(MUI,:),1,1);
    x bqos(MUI,:) = repmat(x bq(MUI,:),1,1);
    Is(MUI,:)=OverN(xbios,deni*fsTs);
    Qs(MUI,:)=OverN(x bqos,denq*fsTs);
    st(MUI,:) = Is(MUI,:)+j*Qs(MUI,:);
    clear bi bq bios bqos chipios chipqos temp1 temp2 xbi x bq xbios x bqos
    ang_transmit(MUI,:)= angle(st(MUI,:));
    if MUI == 1
        side_transmit(MUI,:)=sqrt(2*Ebi(1,MUI))*((cos(ang_transmit(MUI,:)))/(sqrt(PGI))) +
...
        j*sqrt(2*Ebq(1,MUI))*((sin(ang_transmit(MUI,:)))/(sqrt(PGQ)));
    else
        side_transmit(MUI,:)=exp(-
2*j*pi*rand(1,1))*[sqrt(2*Ebi(1,MUI))*((cos(ang_transmit(MUI,:)))/(sqrt(PGI)))+...
        j*sqrt(2*Ebq(1,MUI))*((sin(ang_transmit(MUI,:)))/(sqrt(PGQ)))];
    end
    transmit = transmit + side_transmit(MUI, : );
    vspecta(aver, :)=vspecta(aver, :)+abs(fft(transmit,ss1)).^2;
    vspect1a(aver, :)=vspect1a(aver, :)+abs(fft(sqrt(2)*real(transmit),ss1)).^2;
    vspect2a(aver, :)=vspect2a(aver, :)+abs(fft(sqrt(2)*imag(transmit),ss1)).^2;
end
end
end
vspect = sum(vspecta,1)/Nav; vspect1 = sum(vspect1a,1)/Nav; vspect2 = sum(vspect2a,1)/Nav;
t_vspectc=10*log10(vspect/max(vspect));      x = fliplr(t_vspectc(2:length(t_vspectc))); vspectc
= [x t_vspectc];
t_vspectc1=10*log10(vspect1/max(vspect1));    x1 = fliplr(t_vspectc1(2:length(t_vspectc1)));
vspectc1 = [x1 t_vspectc1];
t_vspectc2=10*log10(vspect2/max(vspect2));    x2 = fliplr(t_vspectc2(2:length(t_vspectc2)));
vspectc2 = [x2 t_vspectc2];
ffx = fliplr(ff(2:length(ff))); ff1 = [-ffx ff];
% Plot all results
rs = ff1*val;
figure(1)
plot(ff1,vspectc,'-r',ff1,vspectc1,'-b',ff1,vspectc2,'-k','linewidth',1);

```

```

title('Power Spectrum, dB');xlabel('Symbol Rate');grid;
legend('Composite', 'I-Channel', 'Q-Channel');
axis([-3/val 3/val -70 10])

```

%Functions Used in the program

```

% -----
% Function BIN01.m generates a random binary vector x, with elements in set {0,1}
% Probability of a 1 is an input parameter p0, and length of x is N.
% Syntax y=BIN01(N,p0), where 0<= p0 <=1
% -----

```

```

function xb = BIN01(N,p0)
xb=(rand(1,N) < p0);

```

```

% -----
% Function OverN oversamples input vector x by N
% thus, for example, if N=3,
% x=[x(1) x(2) ... x(M)] becomes y=[x(1) x(1) x(1) x(2) x(2) x(2) ... x(M) x(M) x(M)]
% not yet generalized for matrices
% -----

```

```

function y=OverN(x,N)
Lx=length(x);
y=zeros(1,N*Lx);
for kk=1:Lx
    for jj=1:N
        y((kk-1)*N+jj)=x(kk);
    end
end

```

% initialize oversampled vector y
% loop to create y

```

% -----
% This function makes the input vector divisible by the block length
% Input: Block length (N); Input vector Nb
% -----

```

```

function y = blkdiv(x,N)
for count = 1:1:length(x)
    a=mod(x(count),N);
    if a~=0
        b = N-a;
        x(count) = x(count)+b;    y(count) = x(count);
    else
        y(count) = x(count);
    end
end
end

```

```

% -----
% This function sees if the input bit length is equally
% divisible into I and Q channel according to the specified Md
% Input parameters: Nb length and value of ratio of
% Q-channel data rate to I-channel data rate
% -----
function y = eveN(x,Md)
if(mod(x,(Md+1))~=0)
    while(mod(x,(Md+1))~=0)
        x= x-1;
    end
end
y=x;

% -----
% This function shifts the full vector by the number of bits specified
% Input: Vector to be shifted, Length of bits to be shifted,
% maximum length of input vector
% -----
function y = shift(x, Nshift)
Nmax = length(x);
temp5(1, :) = x(1,:);
for count = 1:1:Nmax
    if (count+Nshift)>Nmax
        x(1, count) = temp5(1, count+Nshift-Nmax);
    else
        x(1, count) = temp5(1, count+Nshift);
    end
end
y(1,:)=x(1,:);

% -----
% This function combines different vectors to form a single vector
% Md is the ratio of Q-channel data rate to I channel data rate
% -----
function y = combine(newbiti, newbitq, Md)
temp3=1; temp4=1;
R = length(newbiti)+length(newbitq);
for count=1:1:R
    if (mod(count,(Md+1))~=1)
        output_bit(count)=newbitq(temp4); temp4=temp4+1;
    else
        output_bit(count)=newbiti(temp3); temp3=temp3+1;
    end
end
y(1,:)=output_bit(1,:);

```

```

% -----
% This function generates an array by shifting the input signal
% for multuser scenario
% Input: Number of users, PG
% -----
function y = delay(users, PG, M)
L = zeros(1,users-1);
for countN = 1:1:users-1
    value= PG;
    L(1,countN)=value;
    while (L(1,countN) >= value) | (rem(L(1,countN),M)~=0)
        L(1,countN) = round(100*rand(1,1));
    end
end
y = [0 L];

```

```

% -----
% function Demodulate for demodulating the QPSK signals
% -----
function y=demodulate(rec_vector, PG, chip);
siz = size(rec_vector);
Lv = siz(1);
den = siz(2);
for count = 1:1:den
    rec_reshape=reshape(rec_vector(:,count), PG, (Lv/PG));
    despread=rec_reshape'*chip(1,:);
    bit(count,:)=mod(round(angle(despread)/pi),2);
    for counter2 = 1:1:(Lv/PG)
        newbit(1,(count-1)*(Lv/PG)+counter2) = bit(count,counter2);
    end
end
y = newbit;
clear count counter2

```

```

% -----
% Rayleigh Function
% -----
function [r, phi] = rayleigh1(Ns,u)
j=sqrt(-1);
% SELECT ONE OF THE FILTERS BELOW
if u == 0; % no filter, for memoryless fading
    aa=[1]; bb=[1];
end
if u == 1

```

```

% Filter #1, fdT=0.01 (or, fd/fs=0.01)
    aa=[1 -1.9346 0.9376]; bb=[3.772 -4.7 3.772]*1e-3;
end
if u == 2
% Filter #2, fdT=0.1 (or, fd/fs=0.1)
    aa=[1 -1.295 0.5296]; bb=[5.765 0.1064 5.765]*1e-2;
end
% generate the Gaussian i.i.d. r.v.'s, w/100 "extra" samples for
% initialization
Iin = randn(1,Ns+100);
Qin = randn(1,Ns+100);
% pass the input components through the filters
Iout = filter(bb,aa,Iin);
Iout = Iout(101:Ns+100);
Qout = filter(bb,aa,Qin);
Qout = Qout(101:Ns+100);
phi = angle(Iout + j*Qout);
r = sqrt(Iout.^2 + Qout.^2);
% normalize for E[r^2]=1
rms = sqrt( mean( r.*r ) );
r = r/rms;
% Compute phase vector
% Compute Rayleigh magnitude vector

```