PERFORMANCE ANALYSIS OF OPTICAL CDMA AND WDMA FOR BROADCAST LAN'S

A
Major Project Report

Submitted to Faculty of Technology University of Delhi towards the Partial Fulfillment of the Requirement for Award of the Degree of

MASTER OF ENGINEERING
IN
ELECTRONICS & COMMUNICATION ENGINEERING

By
Nishant Dubey
(University Roll No. 3102)

Under the Guidance of
Prof. M. Kulkarni

Session:2003-05
Department of Electronics & Communication Engineering
Delhi College of Engineering
University of Delhi,
Delhi-110042

CERTIFICATE

This is to certify that the work entitled " **Performance Analysis of Optical CDMA And WDMA For Broadcast LAN's**" has been carried out by **Nishant Dubey** under my supervision in partial fulfillment of the requirement for the degree of **Master of Engineering** in **Electronics and Communication Engg**. of **University of Delhi** during the session 2003-05 at **Delhi College of Engineering, Delhi.**

Mr. M. Kulkarni	**Prof. A.Bhattacharyya**
Asst. Professor	**(Head of Department)**
Deptt. Of Information Technology	**Deptt. of Elec. & Comm.Engg**
Delhi College of Engineering	**Delhi College of Engineering**
Delhi	**Delhi**

ACKNOWLEDGEMENT

It is my pleasure to acknowledge and express my deep sense of gratitude towards my teacher and guide, **Mr. M. Kulkarni**, Asst. Professor in Deptt. of Information Technology, Delhi College of Engineering, who supervised the work, reported in this Major thesis report. He was always kind, co operative and helped me whenever I needed.

I would also like to thank our HOD Electronics and Communication Engg. **Prof. A. Bhattacharyya** for his support.

I am thankful to all my friends who have co-operated me in completing the project.

Nishant Dubey
Roll No. 3102
Elec. & Comm. Engg

Table Of Contents

ABSTRACT 8
CHAPTER 1 9
Introduction 9
  1.1 Background 10
  1.2 Purpose 10
  1.3 Method 11
  1.4 Limitations 12
CHAPTER 2 13
Fiber optic communication 13
  2.1 FO-CDMA 16
  2.2 Optical Orthogonal Codes 19
  2.3 Upper Bound on number of codes 22
  2.4 OCDMA COMMUNICATION SYSTEM 23
CHAPTER 3 29
Prime Codes 29
  3.1 Carrier Hopping Prime Codes 31
CHAPTER 4 33
2-Dimensional Codes 33
  4.1 Hybrid wavelength Hopping/Time spreading schemes 33
  4.2 Generation of Prime Hop sequences 33
  4.3 Properties of EQC/Prime Sequences 36
  4.4 Generation of hybrid code 37
  4.5 Comparison Of Systems 39
  4.6 Receiver for the eqc/prime code 41
CHAPTER 5 43
Beat Noise and Its Impact on System Performance 43
  5.1 Analysis Of 2-D Scheme 44
CHAPTER 6 52
Performance Analysis of OCDMA and WDMA 52
  6.1 Introduction 52
  6.2 Analytical Framework 53
  6.3 Multiple-Access Schemes 55
  A. Optical CDMA 56
  B. Wavelength-Division Multiple Access (WDMA) 58
CHAPTER 7 59
Implementation Tool-MATLAB 59
  7.1 The MATLAB System 60
  7.2 Tool box Used 61
Chapter 8 63

Conclusions..63
Appendix 64
   Source Code For The Programs...64
   Results...77
   Graphical Representation of the Result 89
   REFERENCES 98

List Of Figures

Figure 2-1:A FO-CDMA System 15
Figure 2-2:superposition of codeword 18
Figure 4-1: construction of the hybrid sequences for p=5 & i=1 35
Figure 5-1: Frequency composition of data & cross talk signal at the instant of autocorrelation peak 46
Figure 5:Plot For User 1 89
Figure 6:Plot For User 2 90
Figure 7:Plot For User 3 91
Figure 8:Plot For User 4 92
Figure 9: Graph for beat with 5 users sending 5 bits 93
Figure 10 AWGN with 5 users each sending 5 bits 94
Figure 11:comparison of different noises 95
Figure 12: Performance with beat noise for p=13 96
Figure 13:Graph for utilisation 97

List Of Tables

Table 2-1:codeword length n for cardinality |C| and weight w 22
Table 4-1:Table for eqc prime sequences for p=5.................................38
Table 4-2:Table showing different wavelengths used for transmitting different chips..39
Table 4-3:comparison between different sequences................................41

ABSTRACT

The performance of optical code-division multipleacces (CDMA) systems with wavelength-hopping/time-spreading codes is compared to that of a wavelength-division multiple-access (WDMA) system. The multiple-access techniques are applied in a time-slotted broadcast local area network. The utilization, defined as the throughput per unit of time-domain bandwidth expansion, and packet delay are used as metrics of performance. The performance of OCDMA was analyzed in the presence of awgn and Beat Noise. A comparison of degradation in the performance of the system due to presence of these noises has been made.

It was found that the impact of beat noise on the system performance cannot be neglected though it is not the major cause of performance degradation.

CHAPTER 1

Introduction

Telecommunication systems and networks of the future are expected to provide a variety of integrated narrowband and broadband services to customers. To satisfy the future needs, these integrated networks require substantial increase in the throughput as well as the range of bandwidth supported. Conventional networks using bandwidth limited media, such as twisted pair and coaxial cables, will not be able to integrate these broadband services sufficiently.

The advanced developments in fiber optics for the past few decades have made possible the use of optical fibers as transmission media or channels in modern communication Systems. Ever since the mid 90’s optical fibers have been used for point to point communication at a very high speed. Often the optical fiber offers much higher speed than the speed of electronic signal processing at both ends of the fiber. So to be able to take the full advantage of the speed in optical fibers one of the basics concepts in fiber optic communication is the idea of allowing several users to transmit data simultaneously over the communication channel. This is called multiple access.

There are several techniques to provide multiple access and one of them is fiber optic-code division multiple access (FO-CDMA). In FO-CDMA each user is assigned one or more binary signature sequence, so called codewords. The data to be sent is mapped onto the codewords and the different users codewords are ”mixed” together and sent over the channel. At the receiver end a decoder, which is individual for each user, compares the incoming sequence with stored copies of the codewords to be able to extract the information bits.

A set of codes that are suitable for FO-CDMA are optical orthogonal code (OOC) and was first introduced by Salehi. These have the desired property that it should be easy to extract data with the desired codeword in the presence of other users codewords.

1.1 Background

The traditional encoding process for FO-CDMA is to send a codeword or the equal amount of zeros depending on the information bit being 1 or 0. This means that to be able to transmit one information bit we are sending a whole codeword and if the codewords are long the rate could be rather low. However this is often acceptable because of the high speed in the fiber optic channel. But as the speed of electronic signal processing increases it becomes more important to have a high rate in the optical channel. This can be achieved by an effective mapping of the information bits onto the codewords in the encoding process.

1.2 Purpose

Among the various multiplexing schemes considered for local area network (LAN's), Code Division Multiple access (CDMA) techniques are recently receiving substantial attention. This is due to development of large bandwidth a fiber-optic communication channel, which are advantageous over traditional networking. The study of OOCs has been motivated by an application in optical code division multiple access. There have been many efforts to take the full advantage of fiber-optic signal processing techniques to establish an all-optical CDMA communication system since CDMA was first applied to the optical domain in the mid-1980s. Many users transmit information over common wide band channel.

To take advantage of the high speed in an optic fiber, one of the basic concept in fiber optic communication is to allow several users to simultaneously transmit data over the channel. One technique that provides multiple access is fiber optic-code division multiple access (FO-CDMA). In FO-CDMA each user is assigned one or more signature sequences called codewords, which are subsets of a type of optical orthogonal code (OOC). The channel input/output consists of the superposition of several users codewords and at the receiver end an optical correlator extracts the information.

There are numerous advantage in using 2-D (wavelength time) codes in these optical CDMA systems. In addition to having favorable cross-correlation and autocorrelation characteristics two dimensional codes allow a large number of users to be supported with much less time spreading than a pure time domain(1D) code having same cardinality.

Given that each station in an optical CDMA network with 2-D codes can produce a number of wavelengths. Wavelength division Multiple Access (WDMA) cold be implemented as an alternative without any major hardware upgrade. Optical CDMA has the advantage that the number of active users can be much larger than the number of available wavelengths; however the cost is bandwidth expansion and a non zero BER (even when all physical noise is neglected) due to interference between users.WDMA does not require bandwidth expansion and does not suffer from multiple access interference but the number of simultaneous users is limited by the number of available wavelengths.

1.3 Method

The model has been implemented in Matlab because of its easy handling with vectors and matrices. For the encoding and decoding process the encoder and decoder in a Matlab function, which are called by a
function that simulates transmitting and receiving.

For the simulation a suitable OOC has been chosen and each blocks are simulated separately.

1.4 Limitations

In the real case there can be interference from noise in the fiber optic channel. In this report all physical noise (such as shot, thermal and beat noise) is neglected and it focus only on the beat noise. This is an approximation because the inclusion of physical noise may have affect on the system performance. However the purpose of this work is to investigate the performance with respect to beat noise.

There are more or less complex decoding techniques that can use all users codewords to statistically decide whether or not a codeword has been sent. In the simulation environment created for this work the simplest decoder has been used, namely only comparison with the users own codewords.

Another limitation in the model is the absence of an external error correction method, such as a forward error correction (FEC) scheme. This might decrease the system performance but on the other hand the result of this model can be used for decision of what (FEC) scheme is necessary.

CHAPTER 2

Fiber optic communication

Traditional fiber optic communication systems use either TDMA or WDMA schemes to allocate bandwidth among multiple users. Unfortunately, both present significant drawbacks in local area systems requiring large numbers of users. In a TDMA system, the total system throughput is limited by the product of the number of users and their respective transmission rates, since only one user can transmit at a time. For instance, if 100 users wish to transmit at 1 gigabit per second, at a minimum the communication hardware would need to be capable of sustaining a throughput of 100 gigabits per second, a data rate that would strain even the highest performance optical networking equipment. In addition, TDMA systems show significant latency penalties because of the coordination required to coordinate and grant requests for time slots from users by the central node. Unlike TDMA, a WDMA system allows each user to transmit at the peak speed of the network hardware since each channel is transmitted on a single wavelength of light. A WDMA system could easily support a bandwidth of one terabit per second, ideal for the needs of a local area network. Unfortunately, it is difficult to construct a WDMA system for a dynamic set of multiple users because of the significant amount of coordination among the nodes required for successful operation. To build a WDMA network with a dynamic user base, control channels and collision detection schemes would need to be implemented that would waste significant bandwidth. Fortunately, an alternative to TDMA and WDMA networking schemes, optical CDMA communication systems, require neither the time nor the frequency management systems. Optical CDMA can operate asynchronously, without centralized control, and it does not suffer from packet collisions. As a result, optical CDMA systems have lower latencies than TDMA or WDMA. Furthermore, since time and frequency (or wavelength) slots do not need to be allocated to each individual user, significant performance gains can be achieved through multiplexing. Also, TDMA and WDMA systems are limited by hardware because of the slot allocation requirements. In contrast, CDMA systems are only limited the tolerated bit error rate relationship to the number of users. Along with this Traditional multiple access approaches such as frequency division, time division, or demand assignment require elaborate network synchronization at high speed (often optical speed), and frequent conversion from optical domain to electrical domain. The process of optical to electrical conversion and electrical to optical conversion limits how much fiber bandwidth can be used because of limit speed of electrical signal processor. Therefore, a desirable feature of optical communications would be the ability to perform signal-processing functions optically so that the signal conversion from optical to electrical would be done only when desired.

Code division multiple access (CDMA) is a technique that are used for several communication systems. Other techniques for multiple access are Frequency Division Multiple Access (FDMA) or Time Division Multiple Access (TDMA) but one of the main advantages of CDMA is its flexibility. When a FDMA system has a user in every frequency channel or a TDMA system has a user in each time slot the systems are full and no more users are allowed to use the channel. In CDMA the maximum number of users are not fixed but allowing more users to access the channel means lower rates for active users. In a CDMA system you can control the potential number of active users versus the rate and this makes the system flexible. The figure below shows a FO-CDMA network with i pairs of transmitters and receivers.

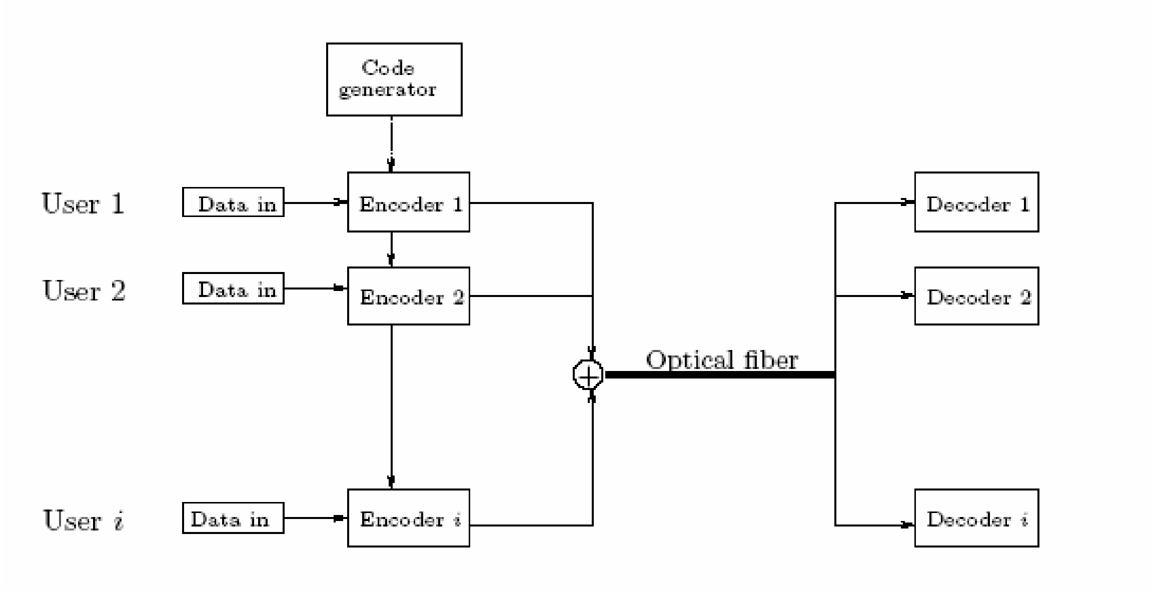


Figure 2-1:A FO-CDMA System

In this chapter we will discuss how CDMA is used today in fiber optic communication and the orthogonal codes needed for this technique to work.

2.1 FO-CDMA

Fiber optic code-division multiple access (FO-CDMA) is one technique to allow several users to transmit simultaneously over the same optical fiber.

A FO-CDMA system can, for each user, be described by a data source, containing the data that will be sent, followed by an encoder and then a laser that maps the signal from electrical form to an optical pulse sequence. At the receiver end an optical correlator is used to extract the encoded data.

In a FO-CDMA system it is common that each user is assigned one signature sequence called codeword. Each bit of information data is encoded by the signature sequence consisting of a number of shorter bits called chips. When this sequence is sent it represent that a user with that unique signature has sent the information bit ’1’. If the information bit is ’0’ it simply means that we send the corresponding length of zeros i.e. no light pulses during that interval.

All users encoded data are then added together chip by chip and the result, which is called the superposition, are sent over the channel. If a light pulse represent the binary bit 1 (mark that this is a chip and not an information bit) and the absence of a light pulse represent the binary bit 0 the superposition mechanism has the following properties.

$$0 + 1 = 1 + 0 = 1 + 1 = 1$$

$$0+0=0 \tag{2.1}$$

In this report we will consider the system to be chip synchronized, that is all users chip durations (pulse widths) are of equal lengths and when creating the superposition of chips they overlap each other precisely. Of course the users can be delayed relative each other. Figure 2.2 on the next page shows an example of how the bits are encoded and the superposition of the users codewords that are sent over the channel. Here user 1 is delayed with 3 chips relative user 2.

The individual receiver, consisting of optical correlators, continuously observes the superposition of all incoming pulse transmissions and recovers the data from the corresponding transmitter. This is done by correlation between the incoming signal and stored copies of that users unique sequence. The correlator will give a peak if the incoming stream of optical pulses contains the unique sequence and the presence of other users will be considered as noise. The decoding process is accomplished by using optical correlation.

If a user is assigned one codeword of length n it means that n chips will carry one information bit. However some users require higher rate and are therefore assigned more codewords.

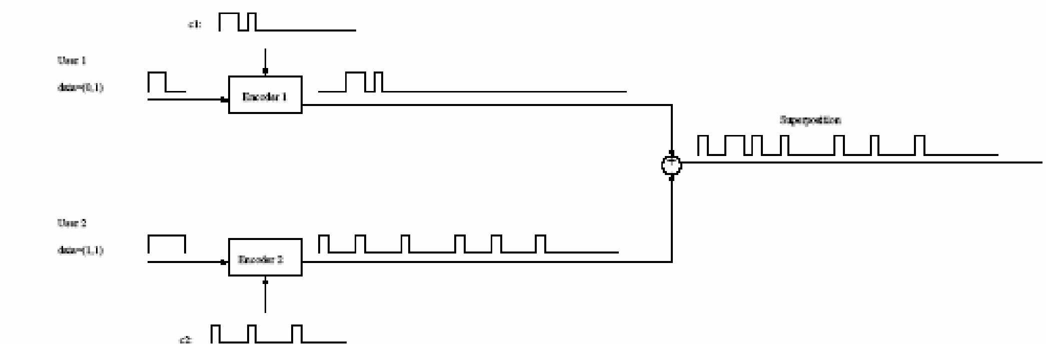


Figure 2-2:superposition of codeword

A prerequisite to successfully recover the data is that the signature sequences comes from a family of orthogonal codes. These codes are presented in the next section and have special properties for correlation between codewords and if the correlation is kept low it reduces the crosstalk between users. One of the main goals of a FO-CDMA system is that as many users as possible should be able to access the channel simultaneously and this with a low probability of error. However if we want to produce |C| number of codewords from an OOC with weight w (the number of ones each codeword has) and lowest possible correlation between the codewords, this require a codeword length n which is approximately.

$$n \geq |C|w(w-1) \tag{2.4}$$

This tells us that if we want to produce more codewords of the same weight this means we get larger codewords. Because one codeword sent over the channel corresponds to one information bit we get a lower rate.

Therefore methods to develop codes with the desired properties are an essential part of FO-CDMA. Optical orthogonal codes was first introduced by Salehi, and these are presented in the next section.

2.2 Optical Orthogonal Codes

Central to any successful code division multiple access scheme, whether electrical or optical, is the choice of high rate sequences, namely, the signature sequences, on which the information database of different user is mapped. In CDMA, many asynchronous users occupy the same channel simultaneously. A desired user's receiver must be able to extract its signature sequence in the presence of other users' signature sequences. Therefore, a set of signature sequences that are distinguishable from time shifted versions of themselves and for which any two such signature sequences are easily distinguishable from (a possible time sifted version of) each other is needed. The codes with these properties are called optical orthogonal codes.

An (n, w, λ_a, λ_c) optical orthogonal codes C is a family of (0,1) sequences o0f length n and the weight w, which satisfy the following properties.

- *The autocorrelation property:*

$$\sum_{t=0}^{n-1} x_t x_{t+\tau} \le \lambda_a$$

for any $x \in C$ and any integer τ, $0<\tau<n$.

- The cross correlation property:

$$\sum_{t=0}^{n-1} x_t y_{t+\tau} \le \lambda_c$$

for any $x \neq y \in C$ and any integer τ.

Where the correlations are periodic correlation, i.e., the subscripts are reduced modulo n whenever necessary. In short, the autocorrelation of each sequence in optical orthogonal code (OOC) exhibits the thumb back shape, and the cross correlation between any two sequences remains low throw out. Since each sequence x has weight w, the autocorrelation equals a w when τ equals to zero. The numbers λ_a and λ_c are called the auto and cross- correlation constraints. The (0,1) sequences of an optical orthogonal code are called its code words. The size of an optical orthogonal code, denoted by |C| , is the number of code words in it. Throughout this project, we require λ_a, $\lambda_c < = w$ to avoid triviality.

Cyclic shifts of code words of an optical orthogonal code do not affect its correlation properties. Let C be (n, w, λ_a, λ_c) code and let C' be derived from c by shifting an arbitrary subset of code words by an arbitrary amount (different code words may be shifted by different amounts). The C' is still an (n, w, λ_a, λ_c) code. We don not make a distinction between code that can be obtained form each other by cyclic shifts.

The codewords can be cyclic shifted and therefore $\oplus$ denotes modulo n so the correlation becomes periodic correlation. λ_a and λ_c are often called the correlation constraints and from now on we will only discuss $\lambda_a = \lambda_c = \lambda = 1$, which is the lowest possible correlation, and the notation for an OOC will become (n,w, 1).

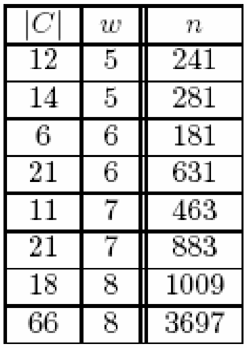
It is desirable to have a large OOC. For a given set of values of n, w, λ_a, and λ_c.The largest possible size of (n, w, λ_a, λ_c) optical orthogonal code is denoted by φ (n, w, λ_a, λ_c). An optical orthogonal code having the maximum size is said to be optimal.

Fiber optic code-division multiple access (FO-CDMA) is one technique to allow several users to transmit simultaneously over the same optical fiber. A FO-CDMA System can, for each user, be described by a data source, containing the data that will be sent, followed by an encoder and then a laser that maps the signal from electrical form to an optical pulse sequence. At the receiver end an optical correlator is used to extract the encoded data.

Often an OOC is denoted by the positions where C has ones. As an example consider the code C = {c1, c2} = {110100000000000, 100010000100000}, which is an (15, 3, 1) OOC with two codewords. This can be written as C ={{0, 1, 3}15, {0, 4, 9}15}.

What the correlation constraint (λ= 1) also say is that no distances between the positions of ones in the code (C) are repeated. For example the distance between the first and the second one in c1 is 1, but as we look at two consecutive c1's (cyclic shifted) the distance 14 will also be taken into consideration. In the code above the distances are {1, 14, 3, 12, 2, 13, 4, 11, 9, 6, 5, 10}. The distances {14, 12, 13, 11, 6, 10} are due to cyclic shifts of the codewords (when we look at two consecutive codewords). We see here that no distance is repeated and therefore the code is an OOC with λ= 1. If λ= 2 it means that any distance between the positions of ones cannot be repeated more than once.

A desirable property of a code is that it should be as large as possible, i.e. contain as many codewords as possible. This is to enable more users to access the channel. An OOC is said to be optimal if it has the maximum cardinality for a given n, w, λ. As mentioned in the previous section the codeword length n increases with increasing cardinality and weight (2.4). In the table below some optimal constructions are presented and it shows what codeword length n is required to construct a code with given size |C| and weight w.



$\|C\|$	w	n
12	5	241
14	5	281
6	6	181
21	6	631
11	7	463
21	7	883
18	8	1009
66	8	3697

Table 2-1:codeword length n for cardinality |C| and weight w

When the codewords are sent without any separation between them the correlator looks at cyclic shifted versions of the codewords. The cyclic shifts must be taken into consideration when producing an OOC and the correlation must be taken modulo n. But if the codewords are separated with at least the codeword length the correlator only look at acyclic shifted codewords. Therefore one can produce codes that only take acyclic shifts into consideration and we call these codes acyclic optical orthogonal codes (AOOC).

2.3 Upper Bound on number of codes

For designing any CDMA code, it is necessary to know the bound on the number of codes, as the number of users is directly related to the number of codes generated (each user is given one unique code). The upper bounds on the maximum size of an optical orthogonal code(n,w, λ) are obtained from the related result in the coding theory. The upper bound on the number of codes is given by as:

$$\phi(n, w, \lambda) \leq \left[\frac{1}{w} \left[\frac{n-1}{w-1} \left[\frac{n-2}{w-2} \left[\cdots\cdots\cdots \left[\frac{n-\lambda}{w-\lambda} \right] \right] \right] \right] \right]$$

This immediately leads to the upper bound for φ (n, w, 1)

$$\phi(n, w, 1) \leq \frac{n-1}{w(w-1)}$$

2.4 OCDMA COMMUNICATION SYSTEM

A typical FO-CDMA (fiber-optic code division multiple access) communication system is best represented by an information data source followed by a laser when the information is in electrical signal form, and an optical encoder that maps each bit of the output information into a very high rate optical sequence, that is then coupled into the signal mode fiber channel. At the receiver end of the FO-CDMA, the optical pulse sequence would be compared to a stored replica of itself (co-relation process) and to a threshold level at the comparator for the data a recovery .

In FO-CDMA there are n such transmitter and receiver pairs (users). Fig shows one such network in a star configuration. The set of FO-CDMA optical pulse sequences essentially becomes a set of address codes or signature sequences for the network.

To send information from user j to user k, the address code for receiver k is impressed upon the data by the encoder at the jth node. One of the primary goals of FO-CDMA is to extract data with the desired optical pulse sequence in the presence of all other users optical pulse sequences. We therefore need to design a sequence that satisfies two conditions namely.

- Each sequence can easily be distinguished from a shifted version itself and.
- Each sequence can be easily distinguished from (a possibly shifted version of) every other sequence in the other set.

A new class of sequences for fiber optic signal processing, namely, the “ optical orthogonal course that satisfy the above two conditions is introduced. The study of OOC’s has been motivated by an application in optical code division multiple access. As shown in figure 2 many users are transmitting information over a wide band optical channel. The objective is to design an efficient system, with available implementation technology, to allow the users to share the common channel. The traditional multiple access approaches such as frequency division, time division, and collision detection or demand assignment require elaborate network synchronization at high speed (often optical speed), and frequent conversions between the optical domain and electronic domain. These requirements limit the efficiencies of such an optical multiple access system. However, by employing a code division multiple access system with optical orthogonal codes, we are able to simplify greatly the complexity of the system to implement it with available technology, and achieve potentially higher transmission efficiency.

Let an (n, w, 1) optical orthogonal code C with M code words (i.e. w sets) be used. The system can accommodate m transmitters simultaneously. Each transmitter is assigned a w – set from C. (here, we use the set theoretical notation of OOC's). At a transmitter every information bit is encoded into a frame of n optical chips in the following ways. (a chips is an optical time slot which can assume one of two values: ON or OFF) let the assigned w set for a particular transmitter be S = $\{s_1 , s_2 .... s_w\}$. Assume the information bit is one. In the corresponding frame, which consists of n optical chips, photon pulses (i.e., ON signals) are send at exactly the s_1 th, s_2 th, …, and s_wth chips. In other n-w chips, no photon pulses (i.e., OFF signals) are sent.

All M users are allowed to transmit at any time. There is no network synchronization is required. At the receiving end, correlation type decoders are used to separate the transmitted signals. The decoder consists of a bank of M tapped delay lines, one for each code words. The delay tabs on a particular line exactly match the signature sequences.

Each tabbed delay line effectively calculates the correlations of the received waveforms with its signature sequence. Because of the properties of optical orthogonal codes the new correlation between different signature sequences is low, thus the delay line output is high only when the intended transmitter's information bit is one. The transmitted information is extracted by thresholding the correlator output.

This optical code division multiple access system can be easily implemented. The tabbed delay line correlator is readily available. Little or no electronic-optical domain conversion is required. There are no synchronization requirements in the network. Although transmitter affects bandwidth expansion, the simplicity and flexibility of the system concepts enables us to pump optical pulses at a much faster chip rates then otherwise possible. The overall system throughput efficiency can be much improved.

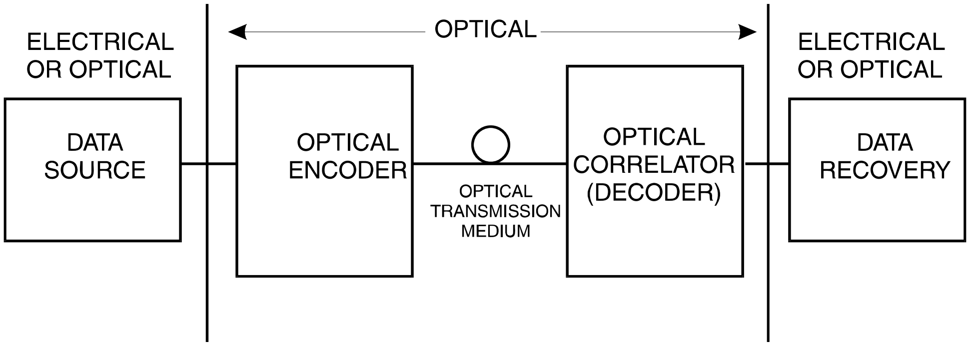


Figure 2.3:A fiber-optic communication system using optical encoder and decoder (correlator)

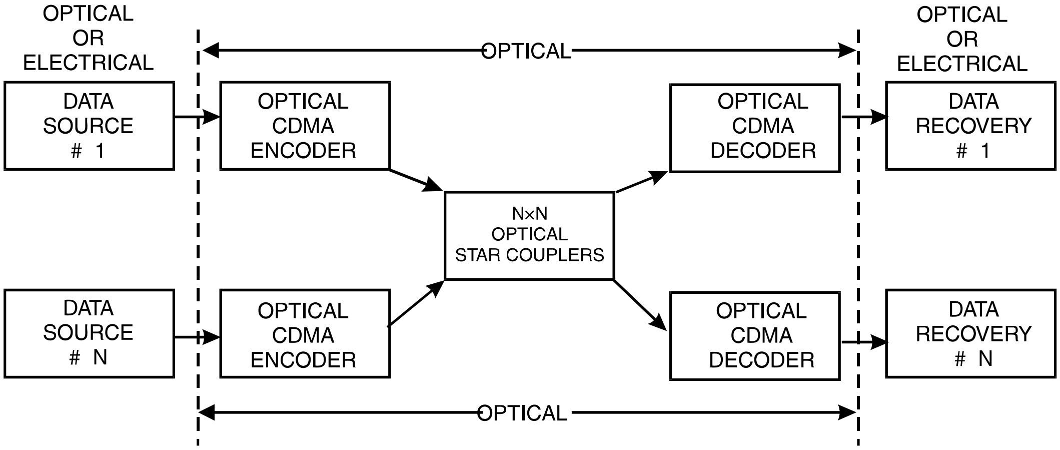


Figure 02.4:A schematic diagram of an optical code division multiple-access communication system with an all-optical encoder and decoder in a star configuration

.

2.5 Implementation of system

Let an (n, w, 1) Optical Orthogonal Code C with N codeword (i.e. w-sets) be used. The system can accommodate N transmitters simultaneously. Each receiver is assigned a w-set from C.At the transmitter end; every information bit is coded into a frame of n-optical chips. Particular transmission be S= {S_1, S2 ,…..S_w }. in the data sequence a ONE is replaced by a corresponding frame, which consists of n Optical chips and photonic pulses (i.e. ON signal)are sent exactly the S_1th,S_2th,…..,S_wth chips. Thus the transmitter uses the code word set as signature sequence. On the other hand if the information bits ZERO, no photonic pulses are sent in the corresponding frame, i.e., all OFF signals are sent. All M users are allowed to transmit at any time. There is no network synchronization required. At the receiving end, correlator type decoders are used to separate the transmitted signal. The decoder consist of a back of N tapped delay-line, one for each codeword. The delay taps on a particular line exactly match the signature sequence. Each tapped delay-line effectively calculates the correlation of the received waveform with its signature sequence. Because of the properties of the Optical Orthogonal Codes, the cross-correlation between signature sequences is low. Thus the delay-line output is high only when the intended transmitter's information bit is high. The transmitter information is extracted by thresholding the correlator output. The Optical Code Division Multiple Access can be easily implemented. The tapped delay line correlator is readily available. Also, as little or no electrical-optical or optical-electrical domain conversion is required, high-speed transmission rates can be achieved. There is no synchronization requirement in the network. The simplicity and flexibility of the system concept enables us to pump optical pulses at much faster rates than otherwise possible. Thus the overall system efficiency is improved.

CHAPTER 3

Prime Codes

The antecedent version of prime codes was first introduced by Cooper and Nettleton in 1978 for cellular mobile communication systems that utilized a frequency hopping spread spectrum technique. The purpose of designing such a code was to support asynchronous transmissions from mobile units to base stations by using signaling waveforms that had uniformly small cross correlation functions for any relative time shift. In 1981, Titlebaum introduced a time frequency ho code, which was later called the (original) prime code, for applications in the areas of coherant multiuser radar and asynchronous FH/SS communications. The structure of original prime code was based on the theory of linear congruence. Two years later, Shaar and Davis introduced prime sequences for CDMA optical networks. To exploit the vast bandwidth offered by optical fibers, Pruncal and his students demonstrated in 1986 the first all optical encoding and decoding hardware which employed fiber optic delay lines for generation and correlation of the binary (0,1) sequences of the original prime code.

The original prime code is equivalent, under column permutations, to extended cyclic Reed Solomon code. The number of binary code sequences in the original prime code over GF(p), a prime field, is p. The minimum Hamming distance of the prime code is p-1 and the code is a kind of maximum distance separable cyclic code. Extensions to larger set of code sequences can be obtained by using quadratic congruence

Prime codes are a family of optical address codes that are suitable for optical fiber code-division multiple access (CDMA) systems using all-optical processing.

However, the code weight *w* of a prime code is always fixed to the maximum number of users *N*. This means that, once *N* becomes large (e.g. in a system with many users), the resulting cost and optical power losses of all-optical CDMA encoder/decoder can be high. In turn, this may prevent the use of integrated optics from implementing such encoders and decoders.

Optical code-division multiple access (CDMA) techniques have been studied for potential use in future all-optical communication systems, because CDMA allows many users to simultaneously access the channel with zero waiting time provided that the destinations are different.

Incoherent optical CDMA (OCDMA) systems have attracted more attention than their coherent OCDMA counterparts, because the former have a far lower complexity and require simpler devices/components than the latter. For example, an incoherent OCDMA encoder or decoder with a `parallel' configuration simply comprises optical delay lines and a passive optical power splitter and combiner, resulting in an ultrahigh processing speed beyond 100 Gbit s^{-1} . So far, *prime codes* and *optical orthogonal codes* (OOCs) have been widely studied for incoherent OCDMA applications . It is known that OOCs have better properties of auto- and cross-correlation than prime codes. However, much more complex algorithms are required to generate and correlate the OOCs than the prime codes. Since the generation of prime codes is extremely simple (i.e. based on modulo-multiplications), this in turn can greatly reduce the processing time in an OCDMA system to guarantee real-time high-speed data communications. Moreover, a prime code can be divided into *P* equal-length subsequences of which each contains only one pulse, where *P* is a prime number. Consequently, these characteristics make prime codes very adequate for OCDMA systems that use a single optical tunable delay line as an encoder to achieve a fast reconfiguration time at each transmitter. However, a major problem associated with prime codes is that their code weight w is always fixed to the number of codewords (i.e. code size) and must be a prime number P. To accommodate more users in an OCDMA system, a larger P is required, so is the code weight w. Since all-optical CDMA encoders and decoders for prime codes use a `parallel' configuration, the resulting optical power losses and complexity of an encoder or decoder would be high if w becomes large. For example, the power loss of an all-parallel encoder (or decoder) is as high as 35.4 dB if $P = 59$, and the required number of optical delay lines per encoder (or decoder) is equal to $w = P = 59$. In this case, the encoders and decoders are also bulky, which may prevent their implementation by using integrated optics.

As reported in, the bit error rate (BER) of incoherent OCDMA systems using a prime code is decreased with increasing w. When P is large enough (e.g., $P \geqslant 41$ for OCDMA systems without optical hard-limiting), the BER becomes very low even if all the users simultaneously transmit data in the system.

3.1 Carrier Hopping Prime Codes

Family of prime codes has many members such as the original prime code, extended prime code, synchronized prime code, 2^n prime code generalized prime code, carrier hopping prime code, multilength carrier hopping prime code, we shall be discussing carrier hopping prime code in detail.

The prime codes mentioned above other than carrier hopping prime code use same frequency or wavelength. Although the use of multiple carriers increases system complexity, this approach adds coding flexibility and, more importantly, improves code performance. In optical systems, wavelength hopping is usually integrated. The prime hop codes have every pulse in each binary code sequence of the original prime code encoded in distinct wavelength, resulting in a class of optical orthogonal codes with the autocorrelation sidelobes of zero and cross correlation sidelobes of at most one. For a given prime number p, p different wavelengths were used, and the prime hop code had length, weight, and cardinality of p^2, p and p(p-1), respectively. By utilizing p wavelengths, the code provides a factor of p more code sequences than the original prime code.

CHAPTER 4

2-Dimensional Codes

4.1 Hybrid wavelength Hopping/Time spreading schemes.

In incoherent optical Code Division Multiple (CDMA) each bit of information is transmitted as a sequence of pulse called chips. This procedure is also referred as Time spreading. The information is retrieved at the receiver by correlation process. In order to increase the throughput(decrease the cross correlation) & to increase the cardinality an integration between WDM & CDMA systems through the amalgamation of the time spreading pattern in incoherent optical CDMA and a wavelength hopping scheme is done.

In this scheme the pulses within code are transmitted at different wavelengths causing the transmitter to hop between various wavelengths in accordance to a certain algorithm, hence the hybrid technique is labeled time spreading/wavelength hopping

4.2 Generation of Prime Hop sequences

We are using the same prime number p for generation of the spreading and hopping patterns, meaning that there are exactly the same numbers of wavelengths in the hopping pattern as the pulses to be colored in the spreading pattern. The system is therefore labeled symmetric which is a special case of more general asymmetric system where the prime number used for spreading & coloring the waveforms are different i.e, *ps* & *ph* respectively.

The cross correlation characteristics of prime hop sequences albeit very attractive, are not so well defined in case of general asymmetric system. Here critical factor is the correlation between sequences with same spreading but different hopping patterns(corresponding to the autocorrelation of the original time spread sequences) which in general can be much higher than 2,there by calling upon a careful selection of hopping pattern.

The most elegant way of sequence generation makes use of a tunable laser with fast tuning rate which could pose problems in high speed transmission system.

Let the prime number chosen be p. the code sequence is divided into p blocks of length p and only one pulse is placed within each block. The position of each pulse in the block is determined by certain algorithm. In case of prime sequences a linear congruent operator is employed to position the pulse within the block

. $a_{ij}= [i.j] \quad i=0, 1,2,\ldots\ldots\ldots,p-1$

$j=0, 1, 2,\ldots\ldots\ldots, p-1$

Where [.] represents modulo p operation

p is a prime number ,

i represents the number of the sequence within the family of sequence(i=0 is reserved for the first sequence, i=1 for 2nd & so on)

j represents the block number

Hence the prime algorithm produces p sequences of length p^2 . As an example for the case i=1 & p=5,j takes on all possible values in sequential order one for each block giving the code sequence

10000 01000 00100 00010 00001.

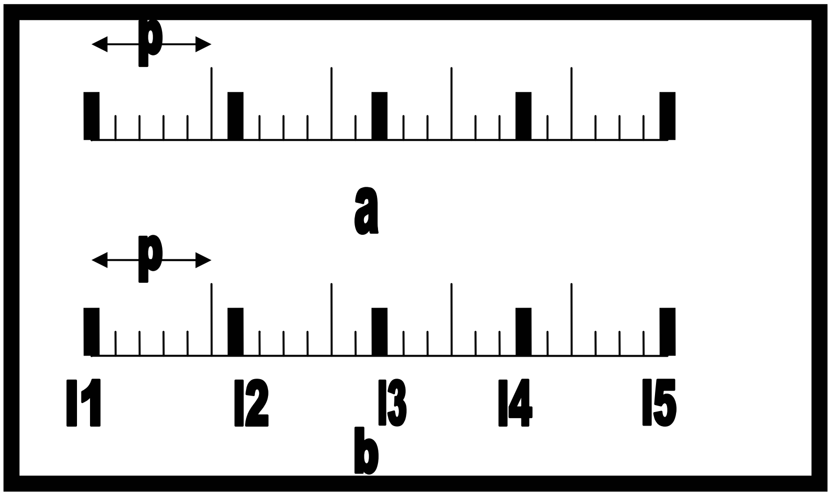


Figure 4-1: construction of the hybrid sequences for p=5 & i=1

a) time spreading (prime sequences)

b) hybrid prime hop sequences (l1,l2,l3,l4,l5 are the wavelengths)

Note that there are 5 blocks of length of length 5 chips each block containing only one "one" (light pulse) the position of which is determined by the above equation therefore the code length has 25 chips. Since the original bit is transmitted as a sequence of chips this concept is known as time spreading.

The same algorithm can be used for generating the hopping pattern as well, using the same prime number p. In that case, there are p wavelengths available for coloring the spreading pattern, exactly as there are p pulses in the code sequence, and therefore the system is referred to as symmetric prime hop sequences

When integrating time spreading and wavelength hopping every pulse In the code sequence is transmitted at different wavelengths therefore causing the transmitter to hop between different wavelengths during transmission .Hence the system is labeled time spreading /wavelength hopping. When using the prime algorithm for both spreading & hopping the resulting system is labeled Prime hop sequence.

Another algorithm that exists for implementing the wavelength hopping/time spreading is using ***extended quadratic congruency*** (eqc) for time spreading while using linear congruency for time spreading. Using eqc for time spreading can truly lead to a random column selection function increasing at the same time the number of sequences in the family(sequence cardinality) and preserving excellent correlation properties.

4.3 Properties of EQC/Prime Sequences

The extended quadratic sequences can be generated using the following placement operator based on Quadratic congruencies.

$$a_{ij}= [i.j. (j+1)/2] \qquad i=1, 2,3,\ldots,p-1$$
$$j=0, 1, 2,\ldots, p-1$$

Where [.] represents modulo p operation

The algorithm given by above equation determines the place of pulse within a block of length 2p-1 so that the sequence length is p(2p-1) and there are p-1 sequences in the family . The construction of the sequence for a given I and for all j (one for each block) proceeds in a completely similar fashion as the one described for prime sequences.

The main difference is that

- the block size is increased from p to 2p-1 and
- the 2nd algorithm unlike the the 1st algorithm does not produce distinct numbers i.e, some of the placement positions are the same.

This could be noted from the figure below where an EQC/prime sequence with p=5 and i=1 is depicted. Note that block size is 2p-1=9 and that the first and fifth block and the second and fourth blocks have pulses placed on a same position within the block . Because of the increased size

- The autocorrelation peaks are utmost one,
- The cross-correlation peaks are utmost two.

These characteristics facilitate the integration with hopping patterns in general asymmetric system. Consequently families with many more sequences can be implemented for the hopping pattern, resulting in increased number of hybrid sequences in the family and, therefore ,possible stations in the network ;similarly a truly column selection method can be employed. For simplicity the introductory analysis will be confined to symmetric systems ,and will subsequently be generalized to asymmetric systems.

4.4 Generation of hybrid code

One possible algorithm for the hopping pattern could be the prime algorithm. When integrating extended quadratic congruencies for spreading and prime sequences for hopping every pulse in the spreading pattern is colored the prime hopping pattern. The process can best be understood from the following table where,

- firstly, the matrix representation for the EQC sequences generated using the 2nd equation and prime sequences generated using the 1st equation are given on the left for p=5 the time spread sequence are also given under the column eqc/prime where the position of a pulse within a block of length 2p-1 is determined by the matrix representation for eqc sequences.

- Finally every pulse in the eqc/spread column can be colored using the matrix for prime sequences, shown in the column eqc/prime.

The first sequence results from the integration of first eqc-spread sequence and the first row in the matrix for prime sequences. Equally integration of the first sequence in the first column and all other rows in the matrix for prime sequences as well as integration of all the other sequences in the eqc-spread with the first row in the matrix gives a valid eqc/prime sequence (not shown in the table). Therefore since there are p-1 spreading patterns and p-1 hopping patterns the number of sequences in the family is $(p-1)^2$

eqc	**eqc-spread**				
5131 5	100000000	010000000	000100000	010000000	100000000
5212 5	100000000	001000000	100000000	001000000	100000000
5343 5	100000000	000100000	000010000	000100000	100000000
5424 5	100000000	000010000	001000000	000010000	100000000

Table 4-1:Table for eqc prime sequences for p=5

prime	**eqc/prime**				
5 1 2 3 4	500000000	010 0 0 0 0 0 0	000 2 0 0 0 0 0	030 0 0 0 0 0 0	400000000
5 2 4 1 3	500000000	002 0 0 0 0 0 0	040 0 0 0 0 0 0	001 0 0 0 0 0 0	300000000
5 3 1 4 2	500000000	000 3 0 0 0 0 0	000 0 1 0 0 0 0	000 4 0 0 0 0 0	200000000
5 4 3 2 1	500000000	000 4 0 0 0 0 0	003 0 0 0 0 0 0	000 0 2 0 0 0 0	100000000

Table 04-2:Table showing different wavelengths used for transmitting different chips

4.5 Comparison Of Systems

The performance comparison of systems using different codes can be done on the basis of the autocorrelation and cross correlation exhibited by them. The autocorrelation peak will result whenever the coinciding pulses match in color i.e, have the same wavelength. Since a particular color in *overcolored* system occur only once in the pattern, every pulse will match with only one pulse and that will occur at zero time shift, when the sequences are in phase, giving an autocorrelation peak. In an *undercolored* system, employing wavelength reuse will cause autocorrelation side lobes to appear since a wavelength will match with a corresponding wavelength more than once and outside the zero time shift as well(albeit not very often),meaning that certain side lobes will be present but of limited strength. In general, having an autocorrelation function with zero or low strength side lobes facilitates greatly the operation of the system since there is no need for synchronization between the receiver & transmitter and is one of the major prerequisites for optical CDMA systems.

Investigating the cross correlation properties *eqc/prime* systems: potential limitation is the case of same spreading/different hopping at the zero time shift, corresponding to the autocorrelation peak of the original eqc sequences used for time spreading(for the other cases the cross correlation of the time spread eqc sequences is at most 2).

Next, a comparison between the *eqc/prime* sequences and sequences proposed for incoherent CDMA is presented. For that purpose, the correlation characteristics of two families of hybrid sequences (prime-hop and eqc/prime) are compared with the characteristics of optical orthogonal codes(OOC) and eqc sequences.

The comparison is made on the basis of equal spreading meaning that the code length is roughly the same(hence the weights are different); similarly, for fair comparison OOC , prime ,and eqc sequences are assumed to be multiplexed at N_w different wavelengths meaning that total signal bandwidths are roughly the same. The results are represented in the table below. Out of the sequences already proposed for incoherent CDMA OOC have the best correlation properties-however, there are very few codes in the family of OOC as can be seen from the fact that of length N=961 comprises 17 members but at the expense of having very low Hamming weight and therefore low signal power. Prime sequences have very good cross correlation properties but very poor autocorrelation side lobes and from that point of view eqc would seem to be better choice where excellent correlation properties are attained at the expense of roughly doubling the code length. Hybrid sequences, however, have superior correlation properties & much larger cardinality, this specialty being true for eqc/prime which, as evident from the table, can support massive no. of stations at the expense of having slightly worse cross correlation and twice the size of the prime hop sequences(for a same weight).

sequence	**length**	**weight**	**cardinality**	**A**	**C**
OCC	N=961	h=8	N_w.(N-1)/(h-1)=527	1	1
Prime	p^2=961	p=31	N_w p=961	p-1=30	2
Eqc	P(2p-1=1035)	p=23	N_w (p-1)=682	1	2
prime hop	p^2=961	p=31	(N_w -1)p=930	0	1
eqc/prime	P(2p-1)=1035	p=23	N_w(N_w-1)(p-1)=20460	0	2

Table 4-3:comparison between different sequences

In the table above

- A represents maximum autocorrelation side lobe
- C is the maximum Cross correlation peak

4.6 Receiver for the eqc/prime code

The reception of the eqc/prime sequences can be done using correlation receiver. The encoded data from different user is transmitted using the common channel. The multiplexing of the data message is done by chip wise ORing of the pulses for each wavelength. Each user is assigned the same unique code for both the transmitter and the receiver. When a data is received its correlation with the signature sequence is calculated, and then compared with the threshold value(estimated on the basis of autocorrelation & cross correlation properties). If the sequence is intended for that particular receiver an autocorrelation peak will occur and the information will be received otherwise the information will appear as cross correlation and, due to the characteristics of the coding scheme, will be neglected.

CHAPTER 5

Beat Noise and Its Impact on System Performance

TWO-DIMENSIONAL (2-D) wavelength hopping/time spreading code division multiple access (CDMA) systems have recently been proposed as a candidate for optical local area networks (LAN's). There are number of advantages accruing from using 2-D coding: the crosscorrelation is reduced, the autocorrelation sidelobes are nonexistent, the cardinality of the code family is greatly increased and, increased security.

There can be many algorithms for performing the coding in both dimensions each producing distinct 2-D code sequences. We shall confine our analysis to asymmetric *prime-hop* sequences where the algorithms for coding in both dimensions are based on linear congruencies In the time domain, the code sequence is divided into p_s blocks of length (p_s is a prime number) each block comprises one pulse whose position is determined by the linear congruence operator. Hence, there are p_s pulses in a code sequence of length p_s^2 . We use wavelength domain coding to color the pulses, again based on the linear congruence operator, this time the operation being performed modulo p_h (where p_h is again prime number representing the number of available wavelengths and $p_h >= p_s$). Since $p_h >= p_s$ there are more wavelengths than pulses to be colored and therefore a column selection function (ideally purely random) needs to be deployed. The family has $p_s (p_h -1)$ codes of length p_s^2 , each code having Hamming weight p_s; the crosscorrelation is at most one with no autocorrelation sidelobes are non existent.

Practical implementation of such systems incorporate tunable laser sources (or laser arrays) and tunable **fiber delay-lines** to select the pulse wavelength and time position. At the receiver, a passive network of matched delay lines and corresponding **optical filters** are employed. The signal at the photodetector is a superposition of pulses forming the autocorrelation plus pulses due to the multiple user interference (crosscorrelation). Due to square law detection there will be beating between pulses at the same wavelength giving rise to beat noise. The purpose of this manuscript is to assess the strength of this process and its impact on the probability of error.

5.1 Analysis Of 2-D Scheme

Assume a 2-D hybrid CDMA system employing asymmetric *prime-hop* sequences, generated using laser sources with chirp free external modulation (tunable lasers or laser arrays), modulation depth is infinite, all signals have same amplitude shift keying (ASK) nonreturn-to-zero (NRZ) modulation format and the polarizations are aligned (worst case scenario). For notational convenience, wavelengths appearing in the desired user

code sequence are labeled λ_1 through λ_{p_s}. Each bit is detected by correlating the received signal with the desired user's code and comparing the output to a pre-determined threshold. The receiver comprises a network of **fiber delays and optical filters** matched to the desired user's code, so that during the time of thresholding (the autocorrelation peak) pulses are visible on all p_s wavelengths (one per wavelength) as represented in fig under "data signal." Pulses from other users contribute

to crosstalk signal as illustrated in Fig. 1 (at most one pulse per interferer because the cross correlation at any time is at most one). The factors k_i i are random variables$(0 \leq k_i \leq k$ **where k is the number of interferers**), measuring the strength of the interference at each particular wavelength λ_i, that is, the number of pulses at wavelength λ_i with $\sum_{i=1}^{p_s} k_i = k$, the total number of crosstalk pulses. Hence, the composite optical field at the autocorrelation output is:

$$\begin{aligned} E(t) \propto & \sum_{i=1}^{p_s} \sqrt{P_d(t)} \exp(j(\omega_{di} t + \Phi_{di}(t))) \\ & + \sum_{i=1}^{p_s} \sum_{j=1}^{k_i} \sqrt{P_c(t)} \\ & \cdot \exp(j(\omega_{cij}(t - \tau_{ij}) + \Phi_{cij}(t - \tau_{ij}))) \end{aligned} \tag{1}$$

where $P_d(t)$ and $P_c(t)$ are the instantaneous optical powers at thresholding; ω_{di} is the optical frequency of the data pulse corresponding to λ_i; ω_{cij} ; is the frequency of the crosstalk pulse at λ_i originating from the jth user; $\Phi_{di}(t)$ and $\Phi_{cij}(t)$ are laser phase noises assumed to be mutually independent Gaussian distributed Wiener–Levy stochastic processes [3]; and τ_{ij} denotes a propagation transit delay of the crosstalk pulse from the user j at λ_i at relative to the data pulse at λ_i. It is also assumed that all the data and crosstalk pulses have same instantaneous optical power $P_d(t)$ and $P_c(t)$ respectively (the near–far field problem is neglected).

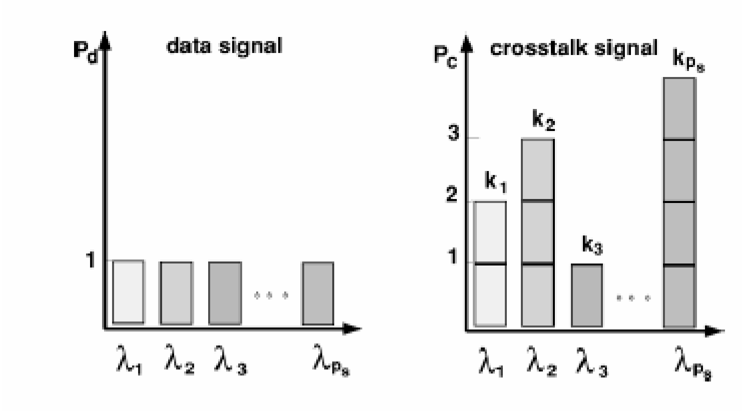


Figure 5-1: Frequency composition of data & cross talk signal at the instant of autocorrelation peak

Assuming that the separation between optical frequencies
is much greater than the photodetector bandwidth B, i.e., for all j and all i $\neq l$, $(\omega_{di} - \omega_{dl})/2\pi$ >>B , and $(\omega_{cij} - \omega_{clj})/2\pi$ >>B, all crossterms in the square of (1) are assumed to be filtered out of the detected signal.We assume that frequencies have been stabilized so that $\omega_{di} = \omega_{cij} = \omega_i$, the photocurrent at the output of the square-law detector is

$$\begin{aligned} i/\mathcal{R} \propto\, & p_s P_d(t) + k P_c(t) + 2 \sum_{i=1}^{p_s} \sum_{j=1}^{k_i} \sqrt{P_d(t)}\sqrt{P_c(t)} \\ & \cdot \cos(\Phi_{cij}(t-\tau_{ij}) - \Phi_{di}(t) - \omega_i \tau_{ij}) \\ & + 2 \sum_{i=1}^{p_s} \sum_{j=1}^{k_i - 1} \sum_{l=j+1}^{k_i} \sqrt{P_c(t)}\sqrt{P_c(t)} \\ & \cdot \cos(\Phi_{cij}(t-\tau_{ij}) - \Phi_{cil}(t-\tau_{il}) - \omega_i \tau_{ij} + \omega_i \tau_{il}) \end{aligned} \quad (2)$$

where $\mathcal{R}$ is the photodiode responsivity assumed to be equal to 1. The first factor in (2) represents the useful autocorrelation peak which is p_s units high. The second term represents the addition in power from the different interferers. When the desired user transmits an information "1," the interference is constructive. When the desired user's bit is an information "0," the interference term can lead to errors, and is the predominant source of noise in incoherent systems. The third term in (2) represents the beating between the **pulses** forming the autocorrelation peak and the **pulses** from the crosstalk signal. Each pulse at wavelength λ_i from the data signal k_i will beat with the pulses from the crosstalk signal. Hence the total sum of the beating terms must equal the total number of interfering terms, i.e. k. And finally, the last term represents beating among the interferer's **pulses**. At a particular wavelength λ_i there are k_i **pulses** beating against each other. The third and fourth terms in (2) can be positive or negative, and are dubbed signal-interferer and interferer- interferer beat noise, respectively.

Tractable analysis requires that the beating terms be statistically independent. The variance of the beating terms can be calculated using the two-pronged probability density function (*pdf*) [2] obtained under the assumption that the phases are random variables uniformly distributed over $[-\pi, \pi]$, giving the following electrical signal-to-noise ratio for bit detection at the instance of the autocorrelation peak:

$$\mathrm{SNR}_1 = \frac{(p_s P_d + kP_c - p_s P_d D)^2}{2kP_d P_c + 2P_c P_c \sum_{i=1}^{p_s} \binom{k_i}{2}}$$

$$\mathrm{SNR}_0 = \frac{(p_s P_d D - kP_c)^2}{2P_c P_c \sum_{i=1}^{p_s} \binom{k_i}{2}} \qquad (3)$$

where $p_s P_d D$ is the threshold level and SNR1 and SNR0 represent the signal-to-noise ratio in case of information “1” and “0” respectively. D is the threshold level (i.eD=1/2., if the threshold is set at half the signal amplitude) [2]. Note that the signal-interferer beat noise vanishes in SNR0 as there is no energy in positive systems when “0” is transmitted, and assuming long correlation times, such that the phases act like fixed random variables over the pulse widths.

Observe that the SNR incorporates the distribution vector $\{k_i\}$ whose components are random variables. Given that k **pulses** due to the interferers fall at the photodetector, they will be distributed over λ_i according to the k_i factors. The components of the distribution vector we model $\{k_1, k_2, \cdots, k_{p_s}\}$ as obeying the multinomial distribution with equal probabilities $P_i = 1/p_s$ Averaging over all possible distribution vectors $\{k_i\}$, the average number of interferer-interferer terms can be easily shown to be

$$\left\langle \sum_{i=1}^{p_s} \binom{k_i}{2} \right\rangle = \sum_{i} \frac{p_s!}{\prod_{l} R(k_l)!} \cdot \frac{k!}{p_s^k \cdot \prod_{l=1}^{p_s} (k_l)!} \cdot \sum_{l=1}^{p_s} \binom{k_l}{2}$$
$$= \frac{1}{p_s}\binom{k}{2} \qquad (4)$$

where $R(k_l)!$ is the number of repetition times of element k_l in vector k , and the product is taken over l for which k_l are distinct. This result is not surprising, as one would expect that the number of interferer-interferer terms is reduced by the amount of deployed wavelengths in the code sequence.

In order to calculate the probability of error, we approximate the pdf of the noise to be Gaussian via central limit arguments The only remaining unknown factor is k (3), i.e., the number of **pulses** from the interferers falling on the photodetector at the autocorrelation peak. The parameter k is dependent on the code structure and its characterization is complicated in the case of asymmetric system as code sequences do not share the same set of wavelengths. Per [1], the average number of wavelengths common to any pair of asymmetric *prime-hop* sequences is estimated as

$$\langle \mu_\lambda \rangle = \frac{1}{\binom{P_h}{p_s}} \left\{ \binom{p_h - 1}{p_s - 1} \frac{(p_s - 1)(p_s - 2) + (p_h - 2)}{p_h - 2} + \binom{p_h - 1}{p_s} \frac{p_s(p_s - 1)}{p_h - 2} \right\} \qquad (5)$$

which gives the average number of received wavelengths due to a single interferer. These contributions are uniformly distributed over all p_s^2 **pulses**. In the case of a symmetric system, i.e., $p_h=p_s$, the average number of common wavelengths is p_s , since all code sequences share the same set of wavelengths. We condition

a) on the event that i users out of the possible k-1 (k simultaneous users) are sending “1” (binomial distribution with probability) and

b) on the event that among i ones there are j **pulses** deposited at the time of interrogating the output of the autocorrelator at the wavelengths forming the autocorrelation peak (binomial distribution with probability $\langle\mu_\lambda\rangle/p_s^2)$. The total probability of error is

$$P_e = \sum_{i=1}^{K-1} \binom{K-1}{i} 2^{-(K-1)} \cdot \sum_{j=1}^{i} \binom{i}{j} \left(\frac{\langle\mu_\lambda\rangle}{p_s^2}\right)^j \cdot \left(1-\frac{\langle\mu_\lambda\rangle}{p_s^2}\right)^{i-j} \frac{1}{2} \left\{ Q\left(\frac{p_s P_d D - jP_c}{\sqrt{2P_c P_c \binom{j}{2} \frac{1}{p_s}}}\right) + Q\left(\frac{p_s P_d + jP_c - p_s P_d D}{\sqrt{2jP_d P_c + 2P_c P_c \binom{j}{2} \frac{1}{p_s}}}\right) \right\} \qquad (6)$$

where

$$Q(x) = \frac{1}{2\pi} \int_x^\infty \exp(-y^2/2)\, dy \qquad (7)$$

Consider the limiting case of negligible beat noise, i.e., all noise sources are neglected giving infinite SNR. For positive signals the Q function contributes zero in (6). When the signal becomes negative (this is the case when “0” is transmitted and j becomes greater than the threshold $\theta = p_s P_d D$) the function Q is 1. Hence, the total probability of error becomes

$$P_e = \frac{1}{2} \sum_{i=1}^{K-1} \binom{K-1}{i} 2^{-(K-1)} \cdot \sum_{j=\theta}^{i} \binom{i}{j} \left(\frac{\langle \mu_\lambda \rangle}{p_s^2}\right)^{j} \left(1 - \frac{\langle \mu_\lambda \rangle}{p_s^2}\right)^{i-j} \qquad (8)$$

which, unlike (6), captures only the effect of the code structure and the multiple user interference due to the cross correlation.

CHAPTER 6

Performance Analysis of OCDMA and WDMA

The performance of optical code-division multipleaccess (CDMA) systems with wavelength-hopping/time-spreading codes is compared to that of a wavelength-division multiple-access (WDMA) system. The multiple-access techniques are applied in a time-slotted broadcast local area network. The utilization, defined as the throughput per unit of time-domain bandwidth expansion, and packet delay are used as metrics of performance. When more than seven wavelengths are available, optical CDMA systems using

asymmetric prime-hop codes and all-optical signal processing are shown to have higher peak utilization and lower corresponding delay than aWDMAsystem with the same number of wavelengths. When the encoders/decoders operate at the chip rate, the utilization of opticalCDMAexceeds that ofWDMAat high offered loads; however, the peak utilization of the WDMA system is still superior.

6.1 Introduction

OPTICAL code-division multiple access (CDMA) is a method of sharing the bandwidth of optical fiber among a number of active users in a broadcast local area network.

There are numerous advantages in using 2-D (wavelength-time) codes in these optical CDMA systems. In addition to having favorable cross-correlation and autocorrelation characteristics, two-dimensional codes allow a large number of users to be supported with much less time-spreading than a pure time-domain (1-D) code having the same cardinality.

Given that each station in an optical CDMA network with 2-D codes can produce a number of wavelengths, wavelength-division multiple access (WDMA) could be implemented as an alternative without a major hardware upgrade. Optical CDMA has the advantage that the number of active users can be much larger than the number of available wavelengths; however, the cost is bandwidth expansion and a nonzero bit-error rate (even when all physical noise is neglected) due to interference between the users. WDMA does not require bandwidth expansion and does not suffer from multiple-access interference but the number of simultaneous users is limited by the number of available wavelengths.

In the analysis that follows, the system performance of 2-D OCDMA is compared to the performance of a WDMA system with the same number of wavelengths.

Typically, the transmitter and receiver structures of an optical CDMA station employ optical components to perform high-speed signal processing. Optical CDMA encoders built with tunable Bragg gratings or tapped delay lines have been demonstrated. These optical encoders allow the electronics of each station to operate at the user data rate, rather than the chip rate; however, if the stations implement either a passive-correlator or a high-speed chip-level detector, the receiver electronics

must operate at the chip rate. Therefore, in this manuscript we consider optical CDMA systems that operate both at the data rate (denoted *all-optical* CDMA) and at the chip rate (termed *chip-rate* CDMA).

6.2 Analytical Framework

The synchronous, random-access, packet broadcast network has been considered. Users begin transmissions on common clock instances and the length of a slot corresponds to a packet of length L bits. With a suitable multiple-channel multiple-access scheme (WDMA or CDMA), a number of packets from different sources can be transmitted over the optical fiber in a single slot. We denote the number of simultaneous packets on the channel during a slot interval by m.

Due to multiple-access interference, some of the packets will arrive at the receiver with bit errors. We let P_B (m) be the probability of a bit error when there are m simultaneous transmissions on the channel. The form of P_B (m) will depend upon the type and parameters of the specific multiple-access scheme. The probability of receiving a packet without errors when m simultaneous transmissions are on the channel is given by

$$P_C(m) = [1 - P_B(m)]^L . \qquad (1)$$

With suitable error-detection capability, the receiver can determine if one or more errors have occurred in a packet. All packets with errors are dropped by the receiver. For simplicity, we neglect the overhead required for this error-detection. In a broadcast network, the sender can independently determine the success or failure of the transmission and schedule the packet for retransmission after a random delay.

Let M be a random variable that represents the number of simultaneous transmissions in a time slot. The conditional distribution of the number of successfully received packets S is then

$$P[S = s|M = m] = \binom{m}{s} P_C^s(m)[1 - P_C(m)]^{m-s}. \qquad (2)$$

The steady-state throughput β can be shown to equal

$$\beta = E[S] = E\left[E[S|M]\right] = \sum_{m=1}^{\infty} m P_C(m) f_M(m) \tag{3}$$

where $f_M(m)$ is the steady-state probability distribution of composite arrivals (new and retransmitted packets).

We assume that the composite arrival distribution is Poissonian with arrival rate λ

$$f_M(m) = \frac{(\lambda T)^m}{m!} e^{-\lambda T}. \tag{4}$$

This choice of arrival distribution corresponds to an infinite user population. Defining $\gamma \triangleq \lambda T$ to be the offered load (average number of attempted transmissions per time slot), the throughput becomes

$$\beta = e^{-\gamma} \sum_{m=1}^{\infty} m P_C(m) \frac{\gamma^m}{m!}. \tag{5}$$

The delay, measured in the average number of retransmissions per packet, can be shown in this case [6] to equal

$$d = \frac{\gamma}{\beta - 1}. \tag{6}$$

6.3 Multiple-Access Schemes

We now have a framework that describes a simple, general, local-area network. The characteristics of both optical CDMA and WDMA can be represented through the error-free packet probability P_C (m).

A. Optical CDMA

We confine our analysis to optical CDMA systems that employ the *asymmetric prime-hop sequences*. The 2-D codes in this family have low cross-correlations, nonexistent autocorrelation sidelobes and large cardinalities. The performance of different code families has not been considered in this letter; however, the sparseness (low ratio of lit to unlit chips) of the asymmetric prime-hop sequences is typical of many other optical CDMA code families.
It is therefore believed that the general trends in the results of this letter should apply well to other two-dimensional optical CDMA codes.

The bit-error rate of asymmetrical prime-hop sequences is given by

$$P_B(m) = \frac{1}{2}\sum_{i=1}^{m-1}\binom{m-1}{i}2^{-(m-1)} \cdot \sum_{j=\theta}^{i}\binom{i}{j}\left(\frac{\langle\mu_\lambda\rangle}{p_s^2}\right)\left(1-\frac{\langle\mu_\lambda\rangle}{p_s^2}\right)^{i-j} \qquad (7)$$

where p_s (prime) is the number of time pulses per wavelength,
$p_h > p_s$ (prime) is the number of available wavelengths, $\theta = p_s$

is the threshold and

$$\langle \mu_\lambda \rangle = \frac{1}{\binom{p_h}{p_s}} \left[\binom{p_h - 1}{p_s - 1} \frac{(p_s - 1)(p_s - 2) + (p_h - 2)}{p_h - 2} + \binom{p_h - 1}{ps} \frac{p_s(p_s - 1)}{p_h - 2} \right] \quad (8)$$

is the average number of wavelengths common to any pair of
asymmetric prime-hop sequences.

In deriving this error probability, all sources of physical noise were neglected; only the impact of multiple-access interference on the bit-error rate was included. This is a good approximation when the signal-to-noise ratio is large; however, in a power-limited regime, all-optical and chip-rate receivers structures have different BER performance. Therefore, the results presented below could be considerably altered in the power-limited case.

The asymmetric prime-hop sequences have a length in the time domain of p_s^2 chips and can support a maximum of p_s (p_h -1) simultaneous users. Without optical signal processing, an optical CDMA transmitter using a 2-D asymmetric prime-hop sequence requires an electronic processing speed of p_s^2 times the per-user data rate.

In terms of the analytical framework discussed in earlier Sections, the probability of receiving a correct packet is

$$P_C(m) = \begin{cases} [1 - P_B(m)]^L, & 0 \leq m \leq p_s(p_h - 1) \\ 0, & p_s(p_h - 1) < m < \infty. \end{cases} \quad (9)$$

The probability given in (9) represents an upper bound; there is the implicit assumption that no two users transmit with the same code during the same slot.

B. Wavelength-Division Multiple Access (WDMA)

For the WDMA system, we neglect all physical noise and channel impairments. With p_h available wavelengths, the probability of receiving a correct bit is

$$P_C(m) = \begin{cases} 1, & 0 \leq m \leq p_h \\ 0, & p_h < m < \infty. \end{cases} \quad (10)$$

Again, the probability given in (10) is an upper bound since we neglect the fact that two or more users may choose to transmit on the same wavelength during a slot. Accounting for the influence of more complex medium-access control protocols (with reservation or collision-detection, for example) on the error probabilities (9) and (10) is an area that merits further investigation.

CHAPTER 7

Implementation Tool-MATLAB

MATLAB® is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. Typical uses include

- Math and computation
- Algorithm development
- Data acquisition Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics Application development, including graphical user interface building

MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allows us to solve many technical computing problems, especially those with matrix and vector formulations, in a fraction of the time it would take to write a program in a scalar non interactive language such as C or Fortran.

The name MATLAB stands for matrix laboratory. MATLAB was originally written to provide easy access to matrix software developed by the LINPACK and EISPACK projects. Today, MATLAB engines incorporate the LAPACK and BLAS libraries, embedding the state of the art in software for matrix computation. MATLAB has evolved over a period of years with input from many users. In university environments, it is the standard instructional tool for introductory and advanced courses in mathematics, engineering, and science. In industry, MATLAB is the tool of choice for high-productivity research, development, and analysis.

MATLAB features a family of add-on application-specific solutions called toolboxes. Very important to most users of MATLAB, toolboxes allow us to learn and apply specialized technology. Toolboxes are comprehensive collections of MATLAB functions (M-files) that extend the MATLAB environment to solve particular classes of problems. Areas in which toolboxes are available include signal processing, control systems, neural networks, fuzzy logic, wavelets, simulation, and many others.

7.1 The MATLAB System.

The MATLAB system consists of five main parts:

Development Environment. This is the set of tools and facilities that help you use MATLAB functions and files. Many of these tools are graphical user interfaces. It includes the MATLAB desktop and Command Window, a command history, an editor and debugger, and browsers for viewing help, the workspace, files, and the search path.

The MATLAB Mathematical Function Library. This is a vast collection of computational algorithms ranging from elementary functions, like sum, sine, cosine, and complex arithmetic, to more sophisticated functions like matrix inverse, matrix eigenvalues, Bessel functions, and fast Fourier transforms.

The MATLAB Language. This is a high-level matrix/array language with control flow statements, functions, data structures, input/output, and object-oriented programming features. It allows both "programming in the small" to rapidly create quick and dirty throw-away programs, and "programming in the large" to create large and complex application programs.

Graphics. MATLAB has extensive facilities for displaying vectors and matrices as graphs, as well as annotating and printing these graphs. It includes high-level functions for two-dimensional and three-dimensional data visualization, image processing, animation, and presentation graphics. It also includes low-level functions that allow you to fully customize the appearance of graphics as well as to build complete graphical user interfaces on your MATLAB applications.

The MATLAB Application Program Interface (API). This is a library that allows you to write C and Fortran programs that interact with MATLAB. It includes facilities for calling routines from MATLAB (dynamic linking), calling MATLAB as a computational engine, and for reading and writing MAT-files.

7.2 Tool box Used

Communication Toolbox

The Communications Toolbox extends the MATLAB® technical computing environment with functions, plots, and a graphical user interface for exploring, designing, analyzing, and simulating algorithms for the physical layer of communication systems. The toolbox helps you create algorithms for commercial and defense wireless or wireline systems.

The key features of the toolbox are

- Functions for designing the physical layer of communications links, including source coding, channel coding, interleaving, modulation, channel models, and equalization

- Plots such as eye diagrams and constellations for visualizing communications signals
- Graphical user interface for comparing the bit error rate of your system with a wide variety of proven analytical results
- Galois field data type for building communications algorithms

Chapter 8

Conclusions

The study was done by varying the number of users using the channel simultaneously and was found that the system performance is seriously degraded by the presence of beat noise. Effect of AWGN was also studied and was found that it also degrades the system performance to a great extent.

The performance comparison of OCDMA and WDMA, on the basis of utilization shows that OCDMA has better performance I certain regime of operations.

The results for the above observations has been included in the Appendix.

Appendix

Source Code For The Programs

```
%this function adds awgn noise to the modulated signal at the receiver

%*************************************************************************
%ich            :input channel carrying modulated data
%ich1          : data with added noise
%p             : no of wavelengths
%no_of_users: the no of users accessing the system simultaneously
%user_data    : the data transmitted by the users
%no_of_bits    : the no of information bit for each user
%spow          : signal power
%sr             : symol rate
%br             :bit rate which is chip rate in our case
%ber            :bit error rate
%ebn0          :signal to noise ratio
%*************************************************************************

clear all;
close all;
clc;

p=input('enter the no. of wavelengths,p:');
no_of_bits=input('enter the no. of information bit for each user:');
ebn0=3;
sr=2*10^12;
br=sr*p*(2*p-1);
nloop=10;%no. of simulations
no_of_users=1;
ber=zeros(1,nloop);
yy=zeros(1,15);
for jj=1:15
   for ii=1:nloop
   user_data=randn(no_of_users,no_of_bits)>0.5;
   ich=prime_encoder1(p,no_of_users,user_data,no_of_bits);
   store=mux1(p,no_of_users,user_data);

%calculation of signal power
```

```
spow=sum(rot90(ich.^2))/no_of_bits;

%calclation of attenuation
attn=sqrt(spow*(0.5*sr/br*10^(-ebn0/10)));

ich1=awgnnoise(ich,attn);% add white gaussian noise

%calculate the correlation of the contaminated received signal with the
%copy of actual user code at receiving end

for i=1:no_of_users
   for n=0:no_of_bits-1
         coeff=0;
         for j=1:p
            y=cross1(ich1(i,(n*p*(2*p-1)+1):(n+1)*p*(2*p-1)),store(j,:),p,j);
            coeff=coeff+y;
         end
         if (coeff>(p-1))
            output(i,(n+1))=1;
         else
            output(i,(n+1))=0;
         end
   end
end
output

   noe=sum(sum(abs(user_data-output)));
   nod=jj*no_of_bits*p*(2*p-1)*10^9;
   fprintf('%d\t%e\n',ii,noe/nod);
   ber(ii)=noe/nod;
end
ber
yy(jj)=mean(ber);
no_of_users=no_of_users+1;
end
yy

figure();
%for i=1:nloop
semilogy(yy);
xlabel('no of users');
ylabel('ber');
```

```
xlim([1 15]);
awgn.m
------------------------------------------------------------------

function [iout]=awgnnoise(idata,attn)
%function to add awgn

%*********************************************************************

%idata  :input channel data
%iout   :output channnel data
%attn   :attenuation level caused by Eb/N0

%*********************************************************************

v=length(idata);
h=length(attn);

iout=zeros(h,v);

for i=1:h
   iout(i,:)=idata(i,:)+randn(1,v)*attn(i);
end
**********************************************************************


cross1.m
------------------------------------------------------------------
function [y]=cross1(a,b,p,j)

n=p*(2*p-1);

y=0;

for k=0:n-1

   for i=1:n

      u=rem(i+k,n);

      if u==0

         u=n;

      end

      if (a(1)>1)
```

```
         if (b(u)~=0)
            c=a-j;
            if (c(i)>0 & c(i)<1)
               prod=c(i)+1;
            else
               prod=0;
            end
         else
            prod=0;
         end
      else
         if b(u)~=0
            prod=a(i);
         else
            prod=0;
         end
      end
      y=y+prod;
   end
end
```

- -

mux1.m

- -

```
%***********************************************************************
%function to multiplex the data transmitted by different users
%the output is the different wavelengths carrying pulses
%***********************************************************************
```

```
function [store]=mux1(p,no_of_users,user_data)

userdata=ones(no_of_users,1);
data=zeros(no_of_users,p*(2*p-1));
data=prime_encoder1(p,no_of_users,userdata);
store=zeros(p,p*(2*p-1));
for i=1:p
   x=i;
   tr_data=zeros(1,p*(2*p-1));
   for j=1:no_of_users
        for k=1:p*(2*p-1)
            if (data(j,k)==x)
                tr_data(k)=1;
            end


        end
    end
        store(i,:)=i*tr_data;
 end
```

----

prime_enoder1.m

```
function [encoded_data1]=prime_encoder1(p,no_of_users,user_data,no_of_bits)
if (nargin==3)
    no_of_bits=1;
end
encoded_data=zeros(no_of_users,no_of_bits*p*(2*p-1));
```

```
orig_code=zeros((p-1)^2,p*(2*p-1));
% get the code values from testcode & store in orig_code
orig_code=testcode(p);
% define a matrix to store the encoded data transmitted by the user
user_code=zeros(1,p*(2*p-1));
% check if no. of users exceeds the max. user lmt if yes then display the error
msg
store=zeros(no_of_users,p*(2*p-1));
store1=zeros(size(store));
if (no_of_users>(p-1)^2)
   disp('No. of users xceeds the limit');
else
   for i=1:no_of_users
      user_code(1,:)=orig_code(i,:);
      store(i,:)=user_code(1,:);
   end
end
%encode the msg data
x=[1:no_of_bits*p*(2*p-1)];
for i=1:no_of_users
%set counter
   n=0;
   for j=1:no_of_bits
      if user_data(i,j)==1
         encoded_data(i,(n*p*(2*p-1)+1):(n+1)*p*(2*p-1))=store(i,:);
         n=n+1;
      else
         if user_data(i,j)==0
```

```
            encoded_data(i,(n*p*(2*p-1)+1):(n+1)*p*(2*p-1))=0;
            n=n+1;
         end
      end
   end
end
[M N]=size(encoded_data);
encoded_data1=zeros(M,N);
encoded_data1=encoded_data;
```

---

testcode.m

```
%this function calculates the chip position to place a pulse using eqc and
%then each pulse is colored according to the prime algorithm hence the code
eqc/prime
function [code1]=testcode(p)
a=zeros(p-1);
c=zeros(p-1);
% calculate the chip position to place the pulse using eqc
% matrix a represents the chip position within a block
% matrix c represents which wavelength to be used to color the chips 1
represents 1st wavelength,
% 2 represents 2nd wavelength & so on
for i=1:p-1
   for j=1:p-1
      x=mod((j*(j+1))/2,p);
      a(i,j)=mod(i*x,p);
```

```
        c(i,j)=mod(i*j,p);
    end
end
code=zeros((p-1)^2,((2*p-1)*p)-1);
i=1;
t=1;
% generation of matrix representing the code
for l=1:p-1
    for i=1:p-1
    j=1;
    for m=1:p-1
        k=m*(2*p-1)+a(l,m);
        code(t,k)=c(i,j);
        j=j+1;
    end
    t=t+1;
end
end
d=p*ones((p-1)^2,1);
code1=[d code];
********************************************************************************
programs to add beat noise to the signal
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

%this function adds awgn noise to the modulated signal at the receiver

%**************************************************************************
%ich       :input channel carrying modulated data
%ich1       :data with added noise
```

```
%p          :no of wavelengths
%no_of_users:the no of users accessing the system simultaneously
%user_data  :the data transmitted by the users
%no_of_bits :the no of information bit for each user
%spow       :signal power
%sr         :symol rate
%br         :bit rate which is chip rate in our case
%ber        :bit error rate
%ebn0       :signal to noise ratio
%**********************************************************************

clear all;
close all;
clc;

p=input('enter the no. of wavelengths,p:');
no_of_bits=input('enter the no. of information bit for each user:');
ebn0=3;
sr=10^9;
br=sr*p*(2*p-1);
nloop=4;%no. of simulations
no_of_users=1;
ber=zeros(1,nloop);
yy=zeros(1,15);


for jj=1:15
   userdata=ones(no_of_users,1);
```

```
usercode=prime_encoder2(p,no_of_users,userdata);
  yy(jj)=((jj+p/5)^2*10^-9);
  for ii=1:nloop
    user_data=randn(no_of_users,no_of_bits)>0.5
    ich=prime_encoder2(p,no_of_users,user_data,no_of_bits);
    ich1=mux2(ich,p);
    ich2=beatnoise1(ich1);
    for j=1:no_of_users
      for n=0:no_of_bits-1
        coeff=0;
        for i=1:p
          y=cross2(ich2(i,(n*p*(2*p-1)+1):(n+1)*p*(2*p-1)),usercode(j,:),p,j);
          coeff=coeff+y;
        end
        if (coeff>(p-1))
          output(j,(n+1))=1;
        else
          output(j,(n+1))=0;
        end
      end
    end
    output

    noe=sum(sum(abs(user_data-output)));
    nod=jj*no_of_bits*p*(2*p-1)*10^9;
    fprintf('%d\t%e\n',ii,noe/nod);
    ber(ii)=noe/nod;
  end
```

```
    ber
    yy(jj)=mean(ber)+yy(jj);
     xx(jj)=log(yy(jj));
    no_of_users=no_of_users+1;
end
figure();
plot(xx);
xlabel('no of users');
ylabel('ber');
 xlim([0 15]);
```

----

beatnoise1.m

```
% **********************************************************************
%this function is to define the characterstics of beat noise
%this function takes the trnsmitted wavelengts as input, and then calculates
%the effect of interfering wavelengts. The wavelengths having pulses at a
%position tend to beat with other pulses and accordingly contaminate the
%signal. The magnitude of this interfering pulse is determined by the
%matrix which gives the coefficient of interference. The contaminated
%signal is output to store.

% ***********************************************************************

function [ich2]=beatnoise1(ich1)

%**********************************************************************
% interference   coefficient matrix determines to what extent a wavelength
```

```
%               will distort the other
% p              No. of wavelength
% M               No. of chips
%********************************************************************************

[p M]=size(ich1);
interference=rand(p);
ich2=ich1;
for i=1:p
   for j=1:p
      if i==j
         interference(i,j)=1;
      end
   end
end
for ii=1:p
   for kk=1:p-1
      remainder=rem(ii+kk,p);
      if remainder==0
         tt=p;
      else
         tt=remainder;
      end
      for jj=1:M
         if(ich1(tt,jj)~=0)
           ich2(ii,jj)=ich2(ii,jj)+interference(ii,tt);
         end
      end
```

```
    end
end
```

----

cross2.m

```
function [y]=cross2(a,b,p,j)
% close all;
% clear all;
% clc;
% a=[1 0 1]
% b=[0 1 0]
n=p*(2*p-1);
% n=3;
y=0;
for k=0:n-1
    for i=1:n
        u=rem(i+k,n);
        if u==0
            u=n;
        end
        if (b(u)~=0)
            if (a(i)~=0)
                c(i)=a(i)-i;
                if (c(i)>0 & c(i)<1)
                    prod=c(i)+1;
                else
                    if (c(i)>1 | c(i)==1)
                        prod=1;
                    else
```

```
                  if (i==1)
                     if (c(i)<0 & c(i)>-1)
                        prod=a(i);
                     else
                        prod=1;
                     end
                  else
                     if (c(i)<0 & c(i)>-1)
                        prod=1;
                     else
                        if (c(i)<-1 | c(i)==-1)
                           prod=a(i);
                        end
                     end
                  end
               end
            end
         else
            prod=0;
         end
      else
         prod=0;
      end
      y=y+prod;
   end
```

Results

RESULTS FOR THE ENCODED AND RECOVERED DATA FOR PRIME NO. 5

```
enter the value of p:5
enter the no.of users:5
enter the no.of bits:1

user_data =

     1
     1
     1
     1
     0


code1 =

  Columns 1 through 14

     5     0     0     0     0     0     0     0     0     0     1     0     0     0
     5     0     0     0     0     0     0     0     0     0     2     0     0     0
     5     0     0     0     0     0     0     0     0     0     3     0     0     0
     5     0     0     0     0     0     0     0     0     0     4     0     0     0
     5     0     0     0     0     0     0     0     0     0     0     1     0     0
     5     0     0     0     0     0     0     0     0     0     0     2     0     0
     5     0     0     0     0     0     0     0     0     0     0     3     0     0
     5     0     0     0     0     0     0     0     0     0     0     4     0     0
     5     0     0     0     0     0     0     0     0     0     0     0     1     0
     5     0     0     0     0     0     0     0     0     0     0     0     2     0
     5     0     0     0     0     0     0     0     0     0     0     0     3     0
     5     0     0     0     0     0     0     0     0     0     0     0     4     0
     5     0     0     0     0     0     0     0     0     0     0     0     0     1
     5     0     0     0     0     0     0     0     0     0     0     0     0     2
     5     0     0     0     0     0     0     0     0     0     0     0     0     3
     5     0     0     0     0     0     0     0     0     0     0     0     0     4

  Columns 15 through 28

     0     0     0     0     0     0     0     2     0     0     0     0     0     0
     0     0     0     0     0     0     0     4     0     0     0     0     0     0
     0     0     0     0     0     0     0     1     0     0     0     0     0     0
     0     0     0     0     0     0     0     3     0     0     0     0     0     0
     0     0     0     0     0     2     0     0     0     0     0     0     0     0
     0     0     0     0     0     4     0     0     0     0     0     0     0     0
     0     0     0     0     0     1     0     0     0     0     0     0     0     0
     0     0     0     0     0     3     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     2     0     0     0     0     0
     0     0     0     0     0     0     0     0     4     0     0     0     0     0
```

```
     0     0     0     0     0     0     0     0     1     0     0     0     0     0
     0     0     0     0     0     0     0     0     3     0     0     0     0     0
     0     0     0     0     0     0     2     0     0     0     0     0     0     0
     0     0     0     0     0     0     4     0     0     0     0     0     0     0
     0     0     0     0     0     0     1     0     0     0     0     0     0     0
     0     0     0     0     0     0     3     0     0     0     0     0     0     0

  Columns 29 through 42

     3     0     0     0     0     0     0     0     4     0     0     0     0     0
     1     0     0     0     0     0     0     0     3     0     0     0     0     0
     4     0     0     0     0     0     0     0     2     0     0     0     0     0
     2     0     0     0     0     0     0     0     1     0     0     0     0     0
     0     3     0     0     0     0     0     0     4     0     0     0     0     0
     0     1     0     0     0     0     0     0     3     0     0     0     0     0
     0     4     0     0     0     0     0     0     2     0     0     0     0     0
     0     2     0     0     0     0     0     0     1     0     0     0     0     0
     0     0     3     0     0     0     0     0     4     0     0     0     0     0
     0     0     1     0     0     0     0     0     3     0     0     0     0     0
     0     0     4     0     0     0     0     0     2     0     0     0     0     0
     0     0     2     0     0     0     0     0     1     0     0     0     0     0
     0     0     0     3     0     0     0     0     4     0     0     0     0     0
     0     0     0     1     0     0     0     0     3     0     0     0     0     0
     0     0     0     4     0     0     0     0     2     0     0     0     0     0
     0     0     0     2     0     0     0     0     1     0     0     0     0     0

  Columns 43 through 45

     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0

code for user 1 is
user_code =

  Columns 1 through 14
```

```
   5   0   0   0   0   0   0   0   0   0   1   0   0   0

  Columns 15 through 28

   0   0   0   0   0   0   0   2   0   0   0   0   0   0

  Columns 29 through 42

   3   0   0   0   0   0   0   0   4   0   0   0   0   0

  Columns 43 through 45

   0   0   0

code for user 2 is
user_code =

  Columns 1 through 14

   5   0   0   0   0   0   0   0   0   0   2   0   0   0

  Columns 15 through 28

   0   0   0   0   0   0   0   4   0   0   0   0   0   0

  Columns 29 through 42

   1   0   0   0   0   0   0   0   3   0   0   0   0   0

  Columns 43 through 45

   0   0   0

code for user 3 is
user_code =

  Columns 1 through 14

   5   0   0   0   0   0   0   0   0   0   3   0   0   0

  Columns 15 through 28

   0   0   0   0   0   0   0   1   0   0   0   0   0   0

  Columns 29 through 42

   4   0   0   0   0   0   0   0   2   0   0   0   0   0

  Columns 43 through 45
```

```
     0     0     0

code for user 4 is
user_code =

  Columns 1 through 14

     5     0     0     0     0     0     0     0     0     0     4     0     0     0

  Columns 15 through 28

     0     0     0     0     0     0     0     3     0     0     0     0     0     0

  Columns 29 through 42

     2     0     0     0     0     0     0     0     1     0     0     0     0     0

  Columns 43 through 45

     0     0     0

code for user 5 is
user_code =

  Columns 1 through 14

     5     0     0     0     0     0     0     0     0     0     0     1     0     0

  Columns 15 through 28

     0     0     0     0     0     2     0     0     0     0     0     0     0     0

  Columns 29 through 42

     0     3     0     0     0     0     0     0     4     0     0     0     0     0

  Columns 43 through 45

     0     0     0

encoded data for user 1 is
ans =

  Columns 1 through 14

     5     0     0     0     0     0     0     0     0     0     1     0     0     0

  Columns 15 through 28

     0     0     0     0     0     0     0     2     0     0     0     0     0     0
```

```
  Columns 29 through 42

    3     0     0     0     0     0     0     0     4     0     0     0     0     0

  Columns 43 through 45

    0     0     0

encoded data for user 2 is
ans =

  Columns 1 through 14

    5     0     0     0     0     0     0     0     0     0     2     0     0     0

  Columns 15 through 28

    0     0     0     0     0     0     0     4     0     0     0     0     0     0

  Columns 29 through 42

    1     0     0     0     0     0     0     0     3     0     0     0     0     0

  Columns 43 through 45

    0     0     0

encoded data for user 3 is
ans =

  Columns 1 through 14

    5     0     0     0     0     0     0     0     0     0     3     0     0     0

  Columns 15 through 28

    0     0     0     0     0     0     0     1     0     0     0     0     0     0

  Columns 29 through 42

    4     0     0     0     0     0     0     0     2     0     0     0     0     0

  Columns 43 through 45

    0     0     0

encoded data for user 4 is
ans =
```

```
  Columns 1 through 14

    5    0    0    0    0    0    0    0    0    0    4    0    0    0

  Columns 15 through 28

    0    0    0    0    0    0    0    3    0    0    0    0    0    0

  Columns 29 through 42

    2    0    0    0    0    0    0    0    1    0    0    0    0    0

  Columns 43 through 45

    0    0    0

encoded data for user 5 is
ans =

  Columns 1 through 14

    0    0    0    0    0    0    0    0    0    0    0    0    0    0

  Columns 15 through 28

    0    0    0    0    0    0    0    0    0    0    0    0    0    0

  Columns 29 through 42

    0    0    0    0    0    0    0    0    0    0    0    0    0    0

  Columns 43 through 45

    0    0    0


data =

  Columns 1 through 14

    5    0    0    0    0    0    0    0    0    0    1    0    0    0
    5    0    0    0    0    0    0    0    0    0    2    0    0    0
    5    0    0    0    0    0    0    0    0    0    3    0    0    0
    5    0    0    0    0    0    0    0    0    0    4    0    0    0
    0    0    0    0    0    0    0    0    0    0    0    0    0    0

  Columns 15 through 28

    0    0    0    0    0    0    0    2    0    0    0    0    0    0
    0    0    0    0    0    0    0    4    0    0    0    0    0    0
```

```
     0     0     0     0     0     0     0     1     0     0     0     0     0     0
     0     0     0     0     0     0     0     3     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0     0     0     0     0

  Columns 29 through 42

     3     0     0     0     0     0     0     0     4     0     0     0     0     0
     1     0     0     0     0     0     0     0     3     0     0     0     0     0
     4     0     0     0     0     0     0     0     2     0     0     0     0     0
     2     0     0     0     0     0     0     0     1     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0     0     0     0     0

  Columns 43 through 45

     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0

information carried by wavelength 1 is
ans =

  Columns 1 through 14

     0     0     0     0     0     0     0     0     0     0     1     0     0     0

  Columns 15 through 28

     0     0     0     0     0     0     0     1     0     0     0     0     0     0

  Columns 29 through 42

     1     0     0     0     0     0     0     0     1     0     0     0     0     0

  Columns 43 through 45

     0     0     0

information carried by wavelength 2 is
ans =

  Columns 1 through 14

     0     0     0     0     0     0     0     0     0     0     2     0     0     0

  Columns 15 through 28

     0     0     0     0     0     0     0     2     0     0     0     0     0     0
```

```
  Columns 29 through 42

     2     0     0     0     0     0     0     0     2     0     0     0     0     0

  Columns 43 through 45

     0     0     0

information carried by wavelength 3 is
ans =

  Columns 1 through 14

     0     0     0     0     0     0     0     0     0     0     3     0     0     0

  Columns 15 through 28

     0     0     0     0     0     0     0     3     0     0     0     0     0     0

  Columns 29 through 42

     3     0     0     0     0     0     0     0     3     0     0     0     0     0

  Columns 43 through 45

     0     0     0

information carried by wavelength 4 is
ans =

  Columns 1 through 14

     0     0     0     0     0     0     0     0     0     0     4     0     0     0

  Columns 15 through 28

     0     0     0     0     0     0     0     4     0     0     0     0     0     0

  Columns 29 through 42

     4     0     0     0     0     0     0     0     4     0     0     0     0     0

  Columns 43 through 45

     0     0     0

information carried by wavelength 5 is
ans =

  Columns 1 through 14
```

```
     5     0     0     0     0     0     0     0     0     0     0     0     0     0

  Columns 15 through 28

     0     0     0     0     0     0     0     0     0     0     0     0     0     0

  Columns 29 through 42

     0     0     0     0     0     0     0     0     0     0     0     0     0     0

  Columns 43 through 45

     0     0     0


store =

  Columns 1 through 14

     0     0     0     0     0     0     0     0     0     0     1     0     0     0
     0     0     0     0     0     0     0     0     0     0     2     0     0     0
     0     0     0     0     0     0     0     0     0     0     3     0     0     0
     0     0     0     0     0     0     0     0     0     0     4     0     0     0
     5     0     0     0     0     0     0     0     0     0     0     0     0     0

  Columns 15 through 28

     0     0     0     0     0     0     0     1     0     0     0     0     0     0
     0     0     0     0     0     0     0     2     0     0     0     0     0     0
     0     0     0     0     0     0     0     3     0     0     0     0     0     0
     0     0     0     0     0     0     0     4     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0     0     0     0     0

  Columns 29 through 42

     1     0     0     0     0     0     0     0     1     0     0     0     0     0
     2     0     0     0     0     0     0     0     2     0     0     0     0     0
     3     0     0     0     0     0     0     0     3     0     0     0     0     0
     4     0     0     0     0     0     0     0     4     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0     0     0     0     0

  Columns 43 through 45

     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
```

```
code1 =

  Columns 1 through 14

    5   0   0   0   0   0   0   0   0   0   1   0   0   0
    5   0   0   0   0   0   0   0   0   0   2   0   0   0
    5   0   0   0   0   0   0   0   0   0   3   0   0   0
    5   0   0   0   0   0   0   0   0   0   4   0   0   0
    5   0   0   0   0   0   0   0   0   0   0   1   0   0
    5   0   0   0   0   0   0   0   0   0   0   2   0   0
    5   0   0   0   0   0   0   0   0   0   0   3   0   0
    5   0   0   0   0   0   0   0   0   0   0   4   0   0
    5   0   0   0   0   0   0   0   0   0   0   0   1   0
    5   0   0   0   0   0   0   0   0   0   0   0   2   0
    5   0   0   0   0   0   0   0   0   0   0   0   3   0
    5   0   0   0   0   0   0   0   0   0   0   0   4   0
    5   0   0   0   0   0   0   0   0   0   0   0   0   1
    5   0   0   0   0   0   0   0   0   0   0   0   0   2
    5   0   0   0   0   0   0   0   0   0   0   0   0   3
    5   0   0   0   0   0   0   0   0   0   0   0   0   4

  Columns 15 through 28

    0   0   0   0   0   0   0   2   0   0   0   0   0   0
    0   0   0   0   0   0   0   4   0   0   0   0   0   0
    0   0   0   0   0   0   0   1   0   0   0   0   0   0
    0   0   0   0   0   0   0   3   0   0   0   0   0   0
    0   0   0   0   0   2   0   0   0   0   0   0   0   0
    0   0   0   0   0   4   0   0   0   0   0   0   0   0
    0   0   0   0   0   1   0   0   0   0   0   0   0   0
    0   0   0   0   0   3   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   2   0   0   0   0   0
    0   0   0   0   0   0   0   0   4   0   0   0   0   0
    0   0   0   0   0   0   0   0   1   0   0   0   0   0
    0   0   0   0   0   0   0   0   3   0   0   0   0   0
    0   0   0   0   0   0   2   0   0   0   0   0   0   0
    0   0   0   0   0   0   4   0   0   0   0   0   0   0
    0   0   0   0   0   0   1   0   0   0   0   0   0   0
    0   0   0   0   0   0   3   0   0   0   0   0   0   0

  Columns 29 through 42

    3   0   0   0   0   0   0   0   4   0   0   0   0   0
    1   0   0   0   0   0   0   0   3   0   0   0   0   0
    4   0   0   0   0   0   0   0   2   0   0   0   0   0
    2   0   0   0   0   0   0   0   1   0   0   0   0   0
    0   3   0   0   0   0   0   0   4   0   0   0   0   0
    0   1   0   0   0   0   0   0   3   0   0   0   0   0
    0   4   0   0   0   0   0   0   2   0   0   0   0   0
```

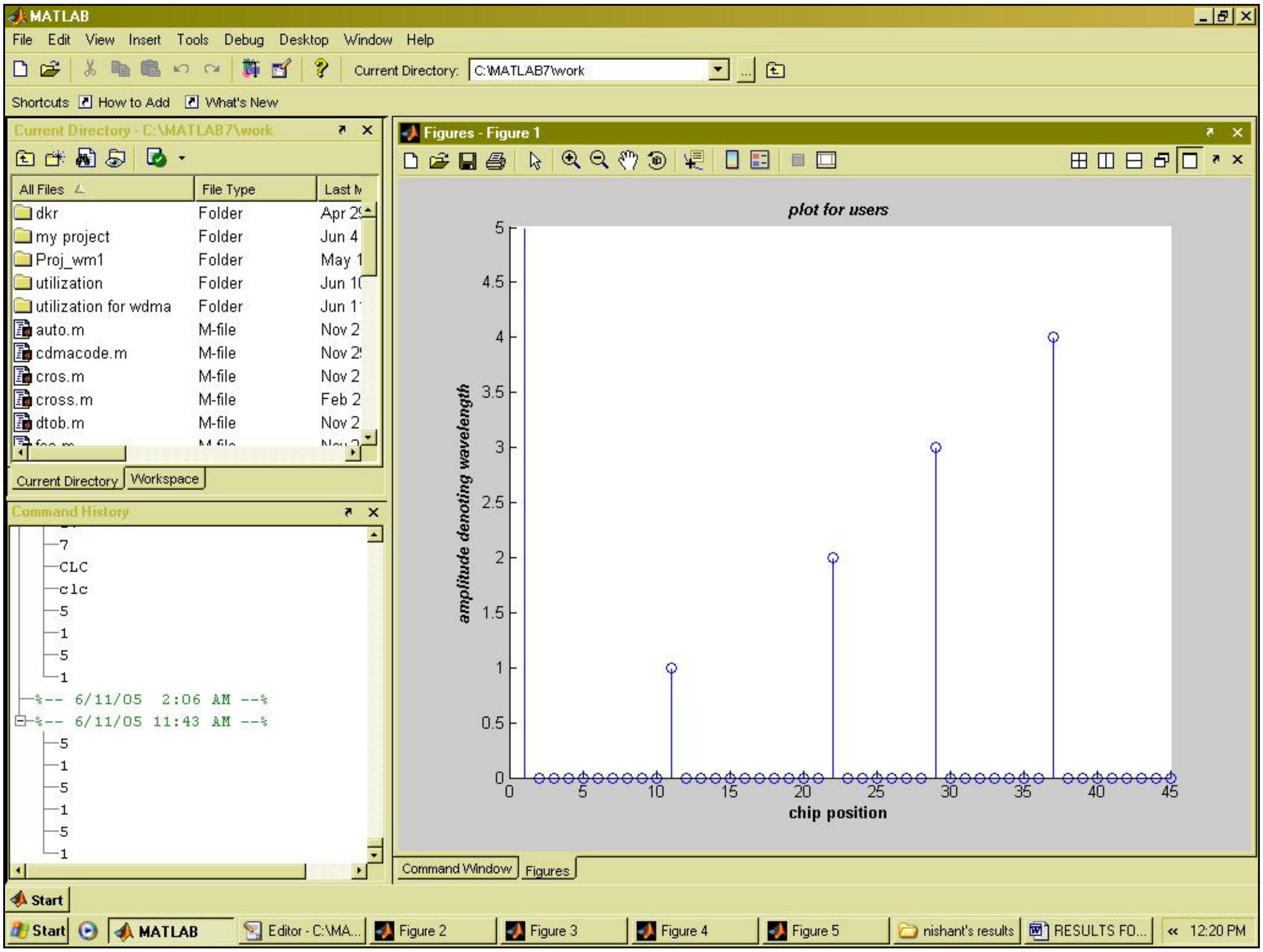
```
    0    2    0    0    0    0    0    0    1    0    0    0    0    0
    0    0    3    0    0    0    0    0    4    0    0    0    0    0
    0    0    1    0    0    0    0    0    3    0    0    0    0    0
    0    0    4    0    0    0    0    0    2    0    0    0    0    0
    0    0    2    0    0    0    0    0    1    0    0    0    0    0
    0    0    0    3    0    0    0    0    4    0    0    0    0    0
    0    0    0    1    0    0    0    0    3    0    0    0    0    0
    0    0    0    4    0    0    0    0    2    0    0    0    0    0
    0    0    0    2    0    0    0    0    1    0    0    0    0    0

  Columns 43 through 45

    0    0    0
    0    0    0
    0    0    0
    0    0    0
    0    0    0
    0    0    0
    0    0    0
    0    0    0
    0    0    0
    0    0    0
    0    0    0
    0    0    0
    0    0    0
    0    0    0
    0    0    0
    0    0    0

the data transmitted by user 1 is 1
the data transmitted by user 2 is 1
the data transmitted by user 3 is 1
the data transmitted by user 4 is 1
the data transmitted by user 5 is 0
```

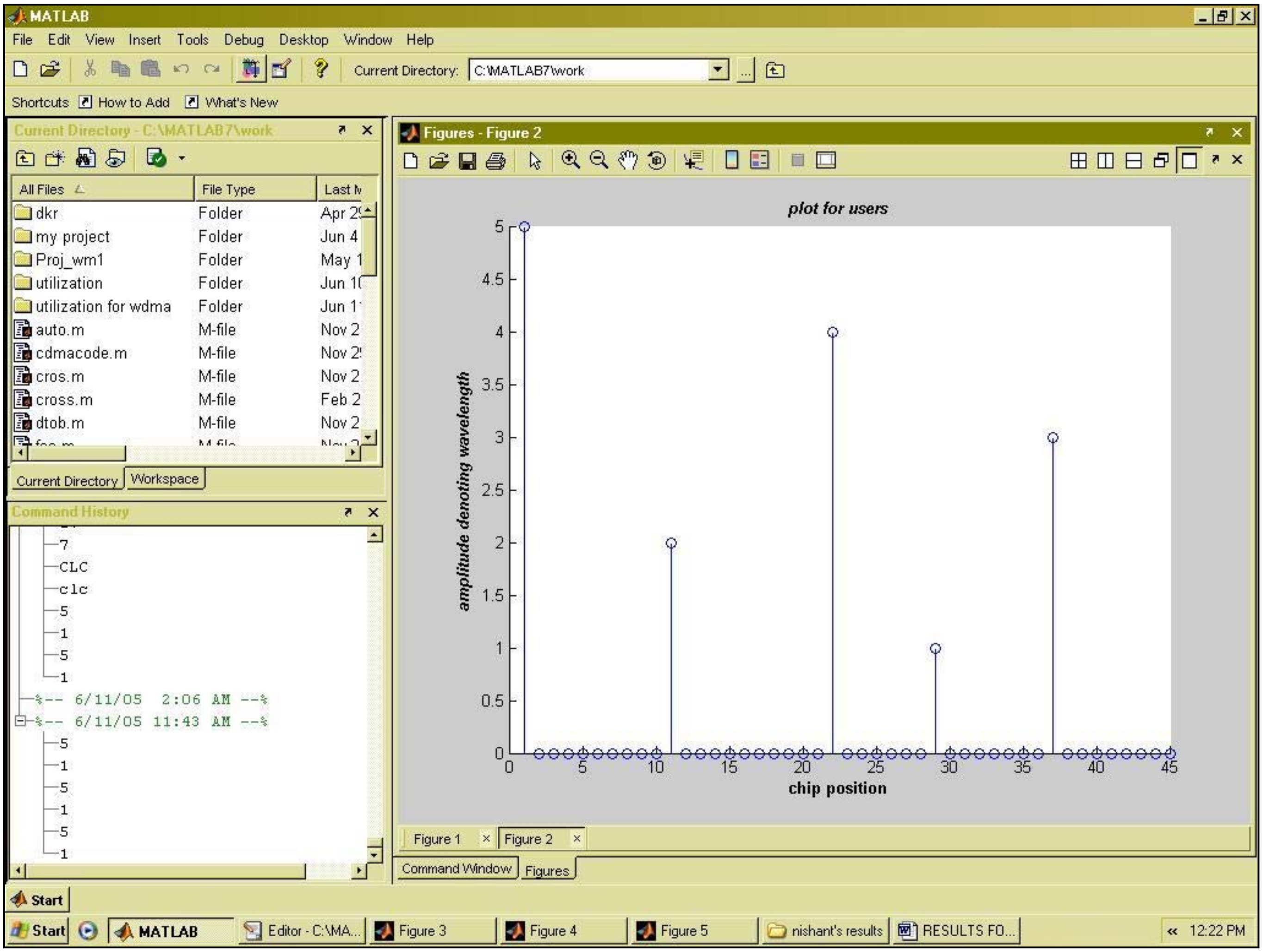
Graphical Representation of the Result



All Files	File Type	Last M
dkr	Folder	Apr 2
my project	Folder	Jun 4
Proj_wm1	Folder	May 1
utilization	Folder	Jun 1
utilization for wdma	Folder	Jun 1
auto.m	M-file	Nov 2
cdmacode.m	M-file	Nov 2
cros.m	M-file	Nov 2
cross.m	M-file	Feb 2
dtob.m	M-file	Nov 2

```
7
CLC
clc
5
1
5
1
%-- 6/11/05  2:06 AM --%
%-- 6/11/05 11:43 AM --%
5
1
5
1
5
1
```

Figure 2:Plot For User 1

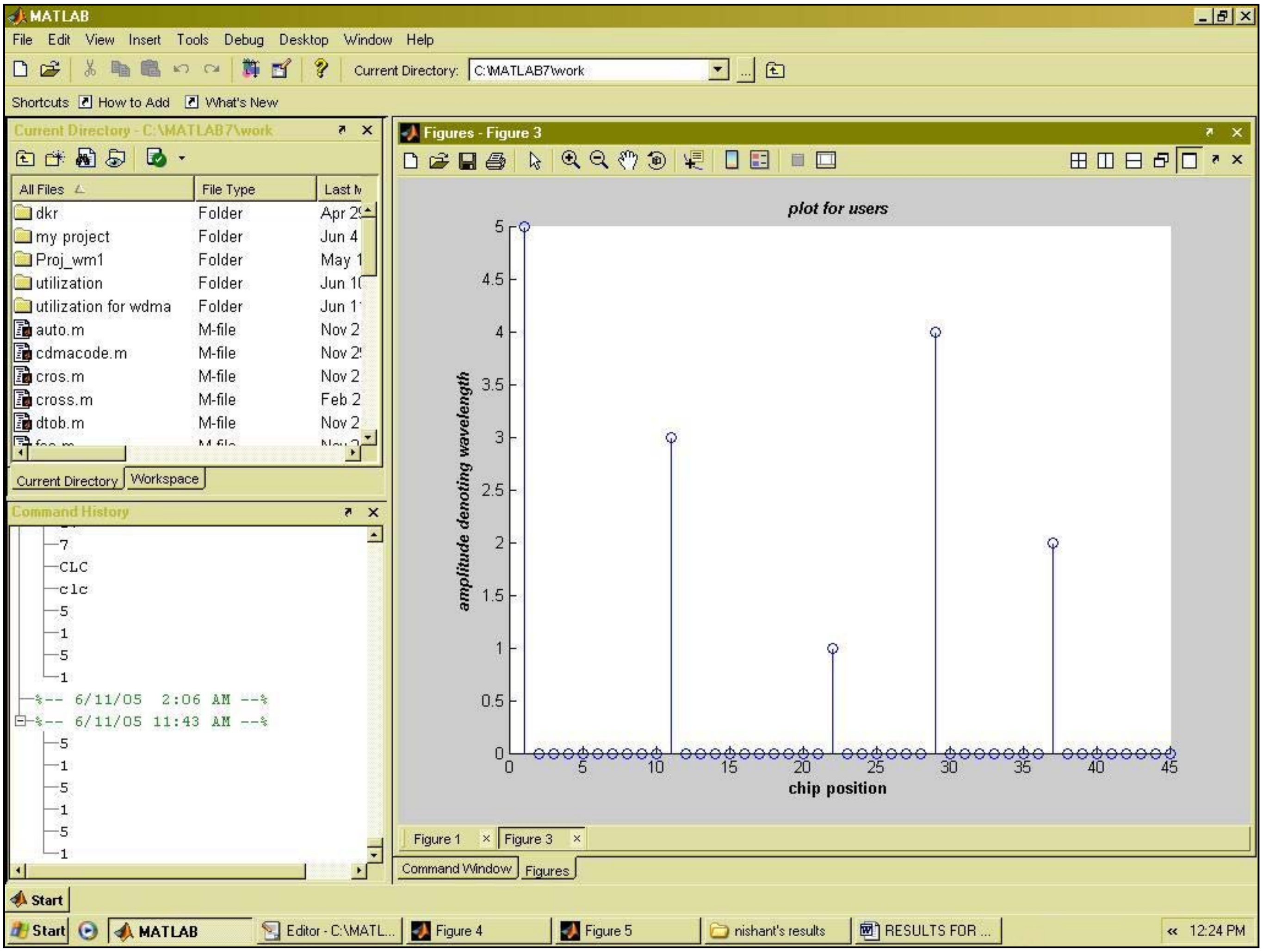


Current Directory: C:\MATLAB7\work

All Files	File Type	Last M
dkr	Folder	Apr 2
my project	Folder	Jun 4
Proj_wm1	Folder	May 1
utilization	Folder	Jun 1
utilization for wdma	Folder	Jun 1
auto.m	M-file	Nov 2
cdmacode.m	M-file	Nov 2
cros.m	M-file	Nov 2
cross.m	M-file	Feb 2
dtob.m	M-file	Nov 2

```
7
CLC
clc
5
1
5
1
%-- 6/11/05  2:06 AM --%
%-- 6/11/05 11:43 AM --%
5
1
5
1
5
1
```

Figure 3:Plot For User 2

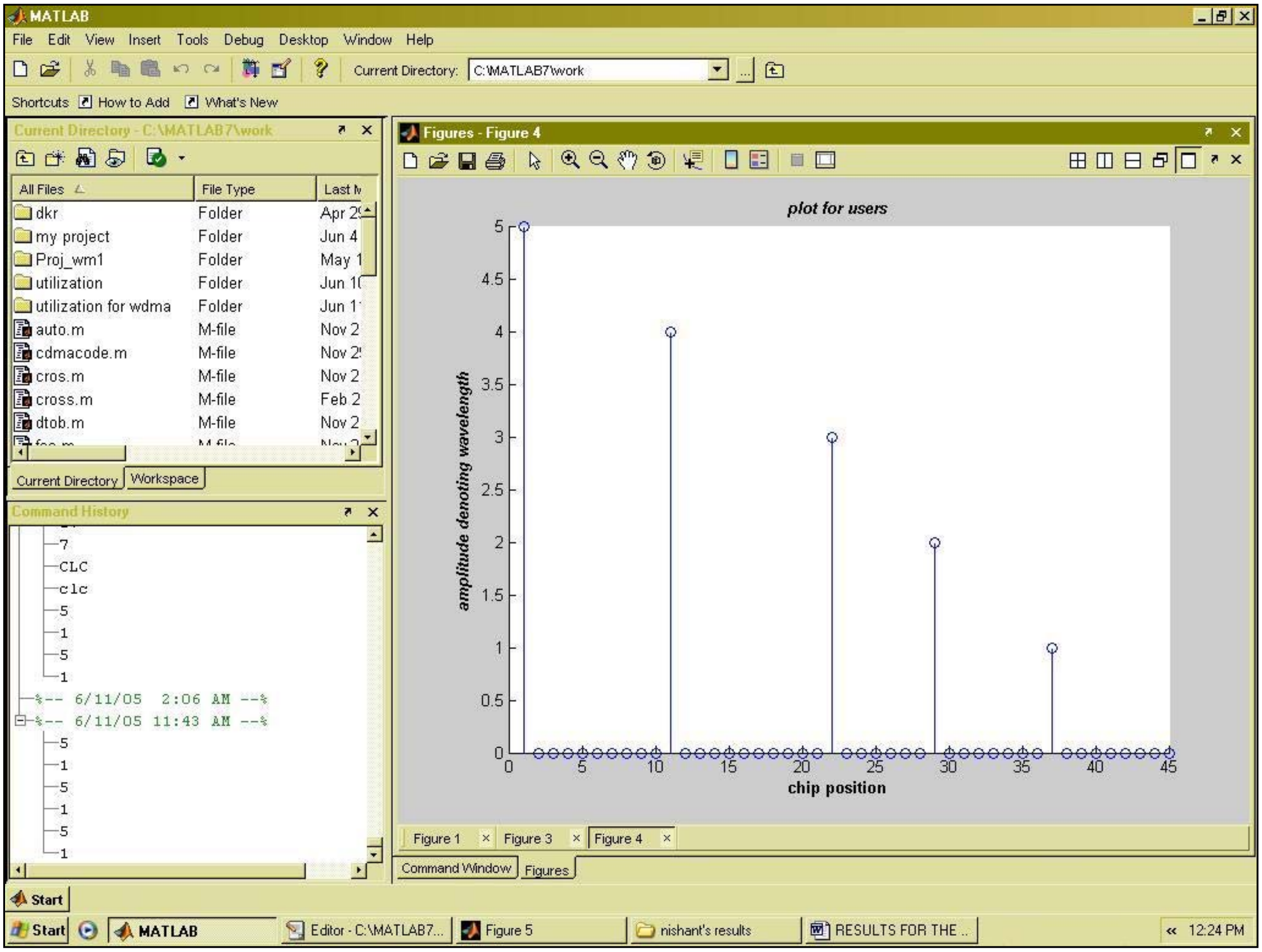


Current Directory: C:\MATLAB7\work

All Files	File Type	Last M
dkr	Folder	Apr 2
my project	Folder	Jun 4
Proj_wm1	Folder	May 1
utilization	Folder	Jun 1
utilization for wdma	Folder	Jun 1
auto.m	M-file	Nov 2
cdmacode.m	M-file	Nov 2
cros.m	M-file	Nov 2
cross.m	M-file	Feb 2
dtob.m	M-file	Nov 2

```
7
CLC
clc
5
1
5
1
%-- 6/11/05  2:06 AM --%
%-- 6/11/05 11:43 AM --%
5
1
5
1
5
1
```

Figure 4:Plot For User 3

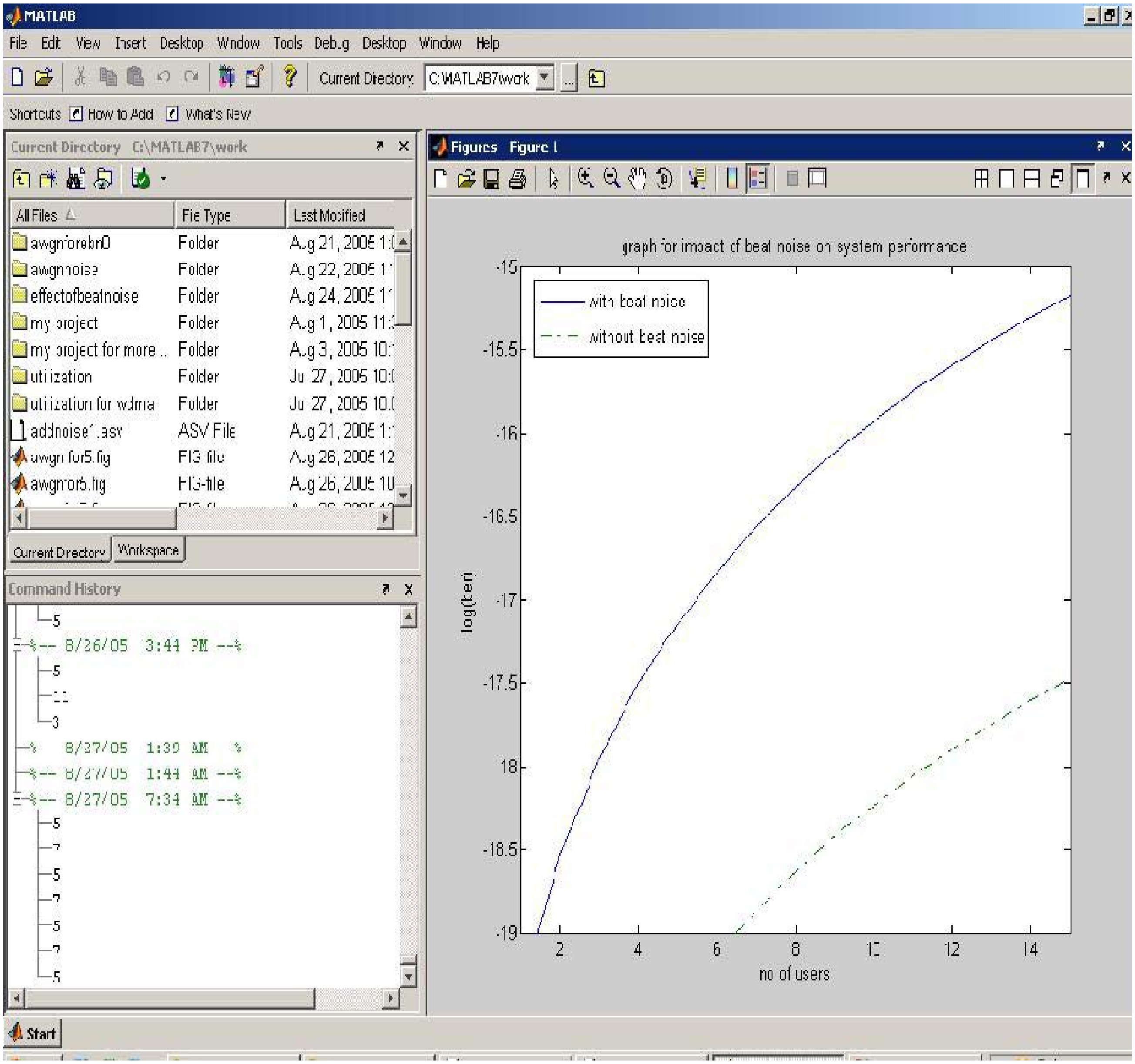


Current Directory: C:\MATLAB7\work

All Files	File Type	Last M
dkr	Folder	Apr 2
my project	Folder	Jun 4
Proj_wm1	Folder	May 1
utilization	Folder	Jun 1
utilization for wdma	Folder	Jun 1
auto.m	M-file	Nov 2
cdmacode.m	M-file	Nov 2
cros.m	M-file	Nov 2
cross.m	M-file	Feb 2
dtob.m	M-file	Nov 2

```
7
CLC
clc
5
1
5
1
%-- 6/11/05  2:06 AM --%
%-- 6/11/05 11:43 AM --%
5
1
5
1
5
1
```

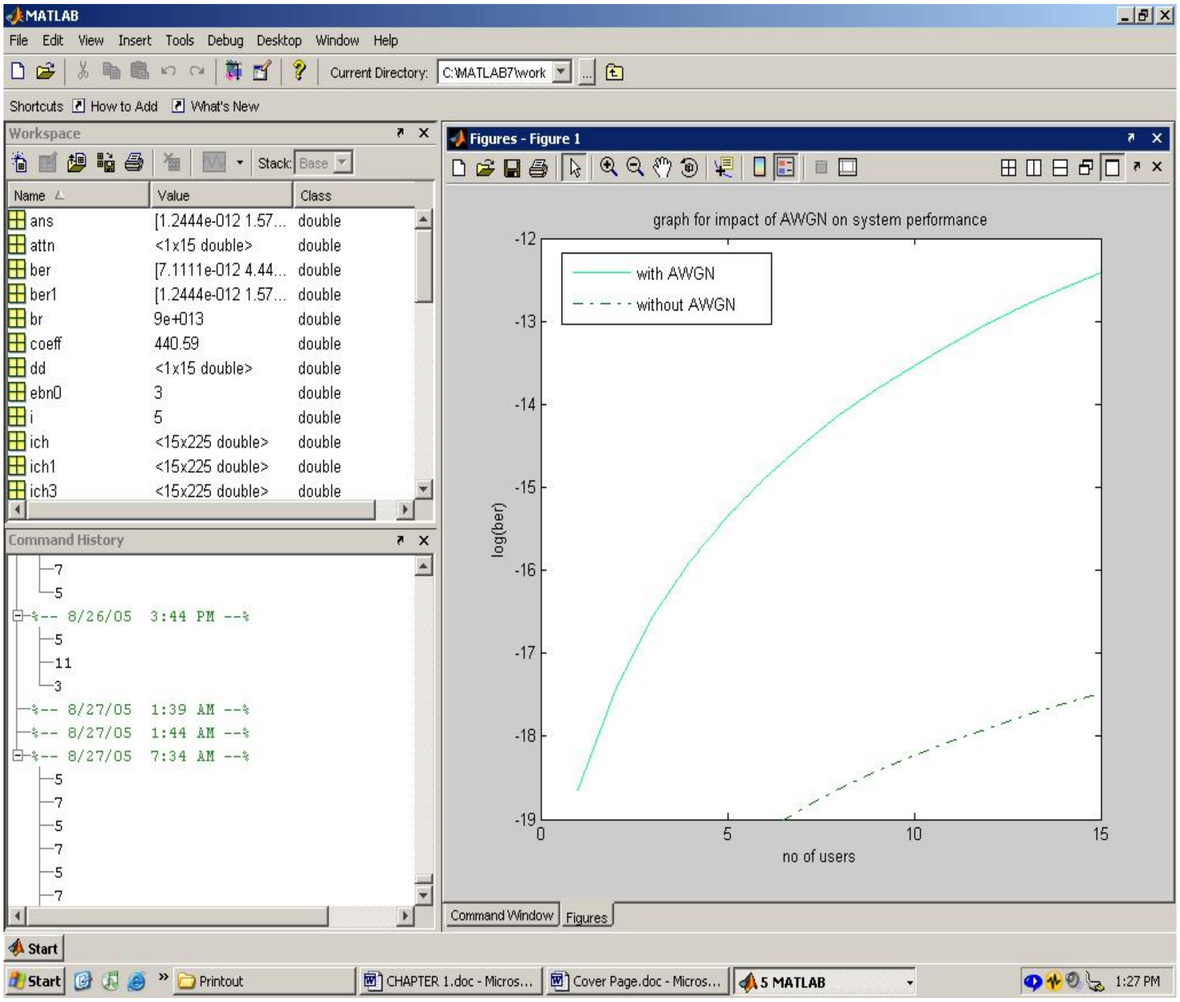
Figure 5:Plot For User 4



All Files	File Type	Last Modified
awgnforebn0	Folder	Aug 21, 2005 1
awgnnoise	Folder	Aug 22, 2005 1
effectofbeatnoise	Folder	Aug 24, 2005 1
my project	Folder	Aug 1, 2005 11
my project for more ..	Folder	Aug 3, 2005 10
utilization	Folder	Jul 27, 2005 10
utilization for wdma	Folder	Jul 27, 2005 10
addnoise1.asv	ASV File	Aug 21, 2005 1
awgnfor5.fig	FIG file	Aug 26, 2005 12
awgnfor5.fig	FIG-file	Aug 26, 2005 10

```
  └─5
⊟─%-- 8/26/05  3:44 PM --%
  ├─5
  ├─11
  └─3
├─%   8/27/05  1:30 AM   %
├─%-- 8/27/05  1:44 AM --%
⊟─%-- 8/27/05  7:34 AM --%
  ├─5
  ├─7
  ├─5
  ├─7
  ├─5
  ├─7
  └─5
```

Figure 6: Graph for beat with 5 users sending 5 bits

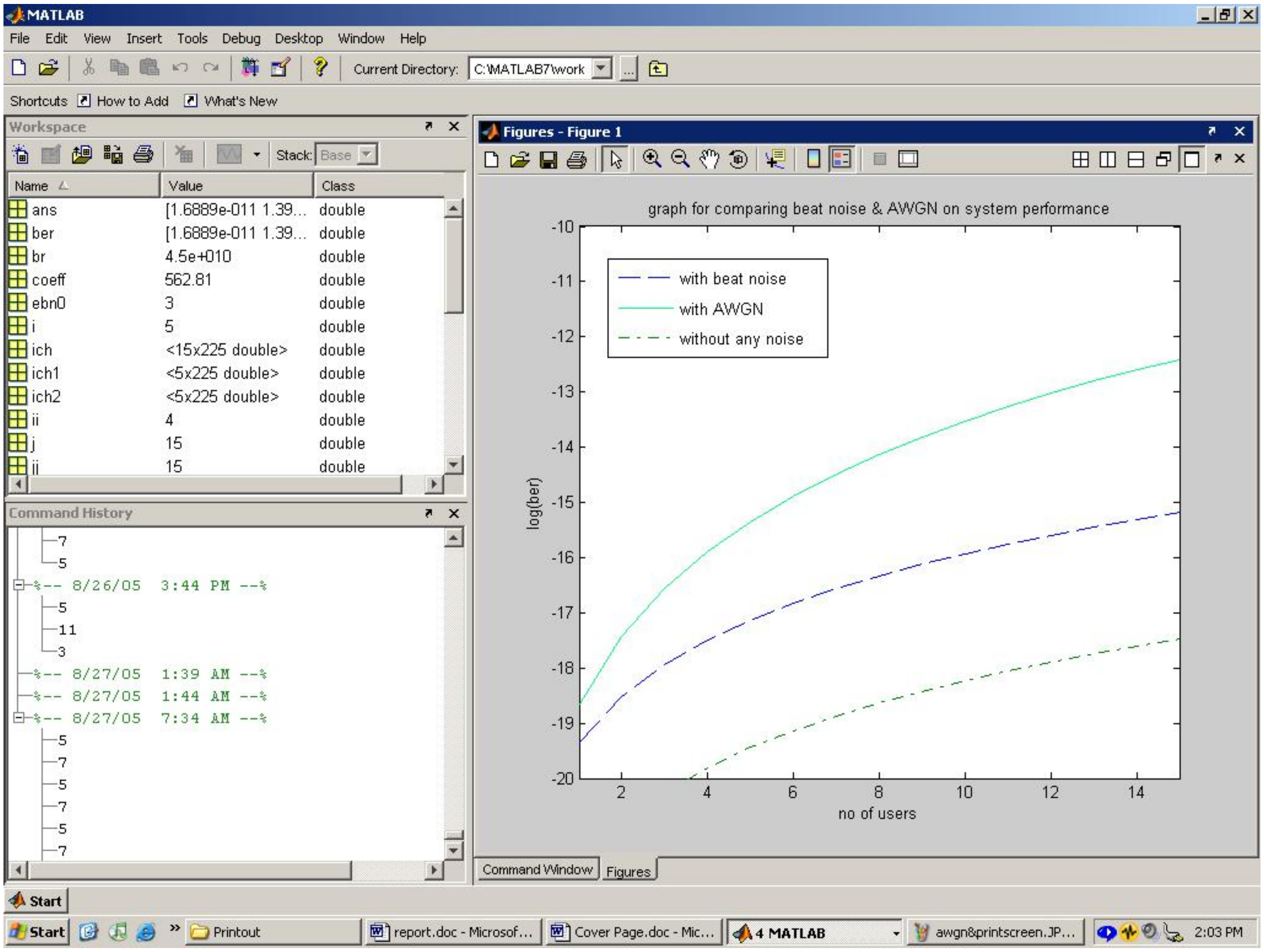


Current Directory: C:\MATLAB7\work

Name	Value	Class
ans	[1.2444e-012 1.57...	double
attn	<1x15 double>	double
ber	[7.1111e-012 4.44...	double
ber1	[1.2444e-012 1.57...	double
br	9e+013	double
coeff	440.59	double
dd	<1x15 double>	double
ebn0	3	double
i	5	double
ich	<15x225 double>	double
ich1	<15x225 double>	double
ich3	<15x225 double>	double

```
   7
   5
%-- 8/26/05  3:44 PM --%
   5
   11
   3
%-- 8/27/05  1:39 AM --%
%-- 8/27/05  1:44 AM --%
%-- 8/27/05  7:34 AM --%
   5
   7
   5
   7
   5
   7
```

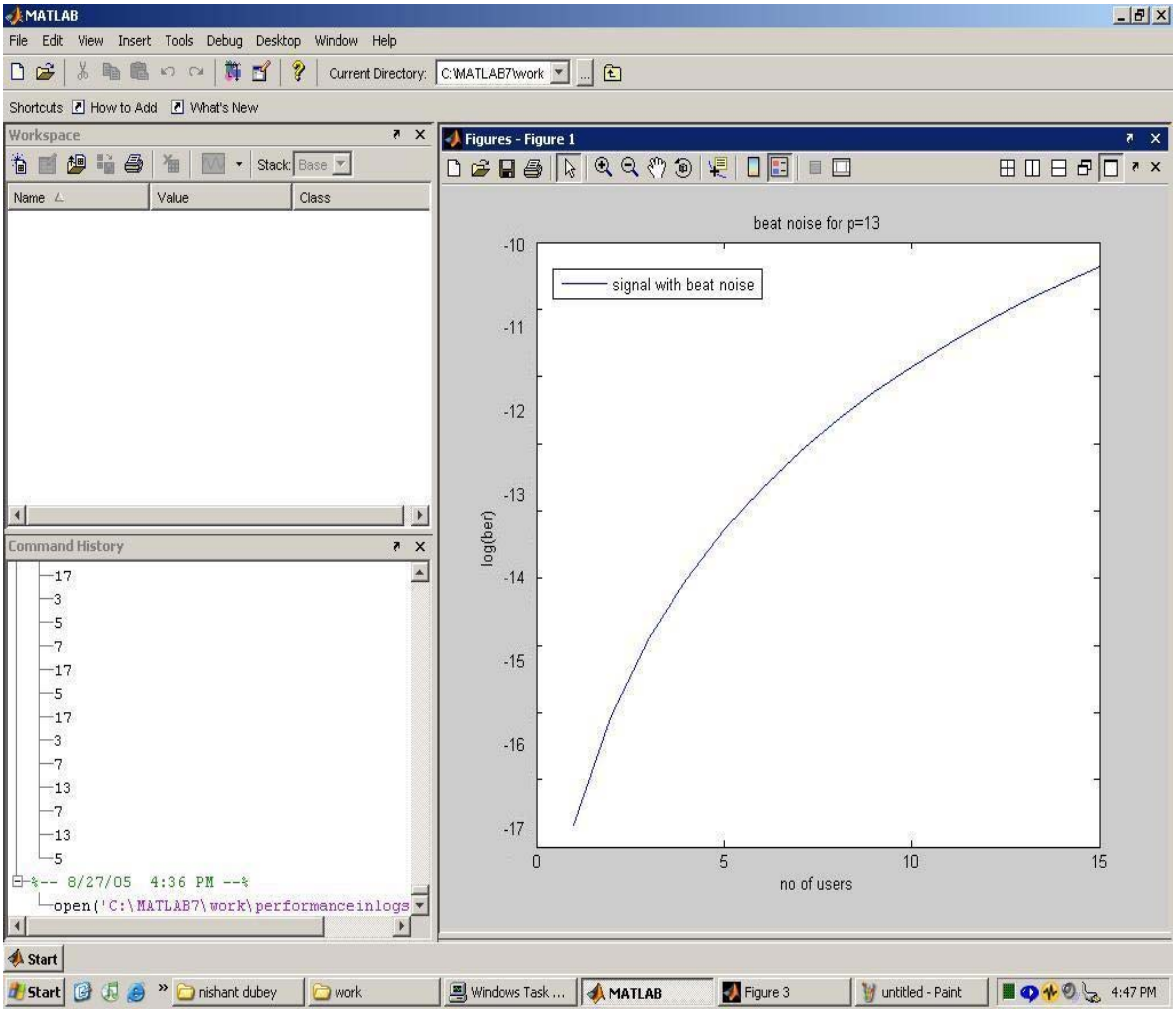
Figure 7 AWGN with 5 users each sending 5 bits



Name	Value	Class
ans	[1.6889e-011 1.39...	double
ber	[1.6889e-011 1.39...	double
br	4.5e+010	double
coeff	562.81	double
ebn0	3	double
i	5	double
ich	<15x225 double>	double
ich1	<5x225 double>	double
ich2	<5x225 double>	double
ii	4	double
j	15	double
jj	15	double

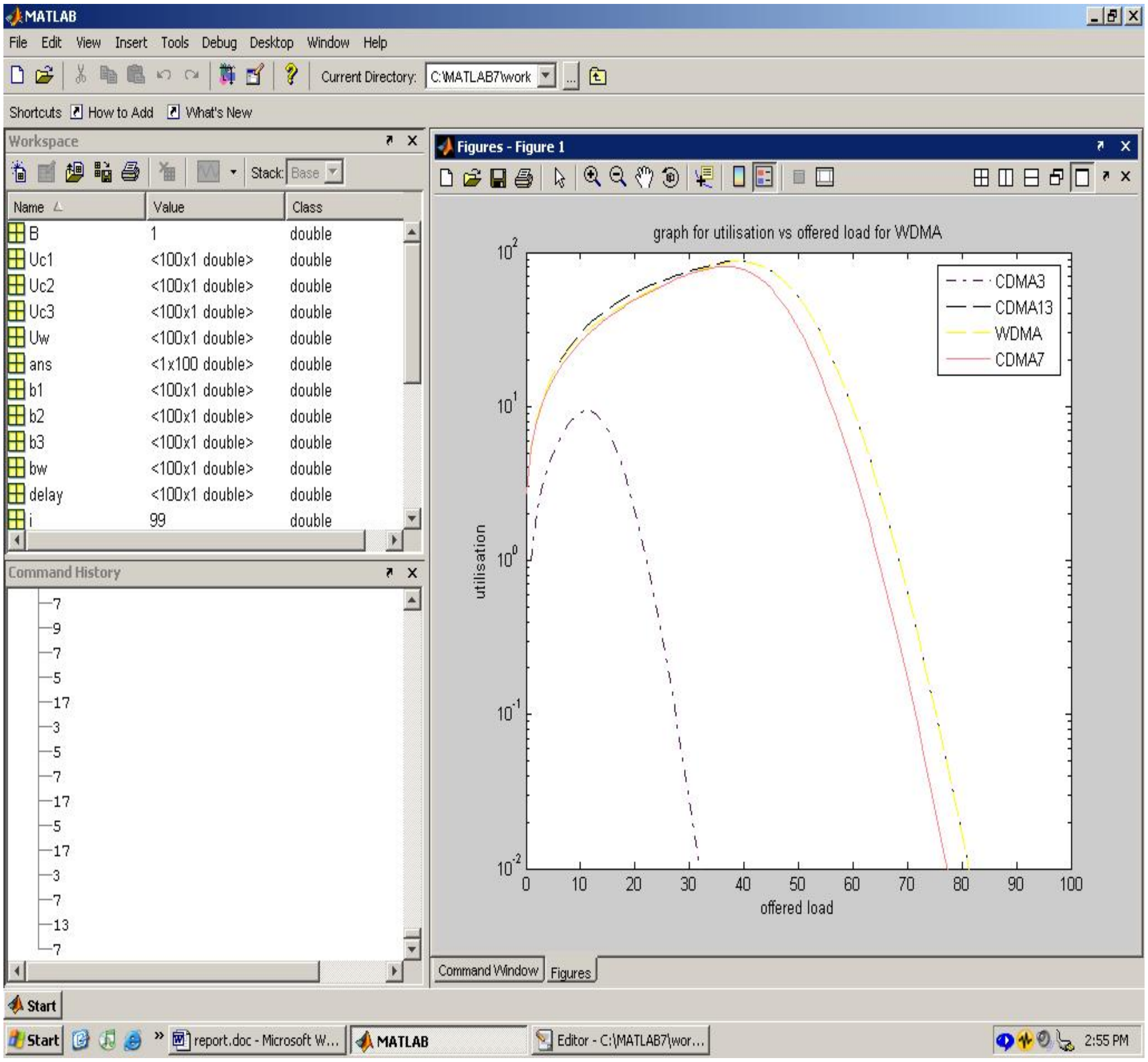
```
   7
   5
%-- 8/26/05  3:44 PM --%
   5
   11
   3
%-- 8/27/05  1:39 AM --%
%-- 8/27/05  1:44 AM --%
%-- 8/27/05  7:34 AM --%
   5
   7
   5
   7
   5
   7
```

Figure 8:comparison of different noises



```
  -17
  -3
  -5
  -7
  -17
  -5
  -17
  -3
  -7
  -13
  -7
  -13
  -5
%-- 8/27/05  4:36 PM --%
  open('C:\MATLAB7\work\performanceinlogs
```

Figure 9: Performance with beat noise for p=13



Name	Value	Class
B	1	double
Uc1	<100x1 double>	double
Uc2	<100x1 double>	double
Uc3	<100x1 double>	double
Uw	<100x1 double>	double
ans	<1x100 double>	double
b1	<100x1 double>	double
b2	<100x1 double>	double
b3	<100x1 double>	double
bw	<100x1 double>	double
delay	<100x1 double>	double
i	99	double

```
—7
—9
—7
—5
—17
—3
—5
—7
—17
—5
—17
—3
—7
—13
—7
```

Figure 10:Graph for utilisation

REFERENCES

- J.A. Salehi, “Code division multiple access techniques in optical fiber networks-part I: Fundamental principles,” *IEEE Trans. Commun.,* vol. 37, pp. 824-833, Aug. 1989.

- F.R.K. Chunkg. J.A. Salehi, V.K. Wei, “Optical orthogonal codes: Design, analysis and applications,” *IEEE Trans. Inform. Theory.* Vol. 35, pp. 595-604, 1989.

- S. Zahedi and J.A. Salehi, “Analytical comparison of various fiber-optic CDMA receiver structures,” *J. Lightwave Technol.,* vol. 18, pp.1718-1827, Dec. 2000.

- Wing. C. Kwong and Guu-Chang Yang,”Prime Codes with Application to CDMA Optical and Wireless Network.

- L.Tancevski and Ivan Andonovic,”Hybrid Wavelength Hopping /Time Spreading Schemes for Use in Massive Optical Network With Increased Security,”J. *Lightwave Technology,*vol 14, NO.12, DEC 1996.

- L.Tancevski and L.A Rusch “Impact Of Beat Noise On The Performance Of 2-D OCDMA Systems,”IEEE Comm. Letters ,vol. 4,No. 8 August 2000

- Andrew Stok and Edward H. Sargent ”System Performance Comparison Of OCDMA And WDMA In Broadcast LAN” IEEE Comm. Letters vol 6. No.9 sept 2002

- D.Raychaudhari," Performance Analysis of Random Access Packet Switched CDMA Systems" IEEE transactions on Comm. Vol. com29,No.6 june 1981