IMPLEMENTATION OF MIMO-OFDM TO REDUCE BIT ERROR RATE

Submitted in partial fulfillment of the requirements
of the degree of

Bachelor of Engineering

By

SHAIKH SADAF SHAKIL	12ET08
SIDDIQUI MISBA JAMATALI	12ET11
ANSARI ZUBER KHURSHID	12ET14
SAYYED MD. SHANAWAZ	11ET48

Supervisor:
Asst. Prof. Mujib A. Tamboli

Department of Electronics and Telecommunication Engineering
Anjuman-I-Islam's Kalsekar Technical Campus,
New Panvel

MUMBAI UNIVERSITY

2015-2016

Project Report Approval for B.E

This project report entitled ***IMPLEMENTATION OF MIMO-OFDM TO REDUCE BIT ERROR RATE*** by ***SHAIKH SADAF , SIDDIQUI MISBA , ANSARI ZUBER AHAMAD*** *and* ***SAYYED SHANAWAZ*** is approved for the degree of ***Bachelor of Engineering***.

Examiners:

1.___________________________________.

2.___________________________________.

Supervisor:

Asst. Prof. MUJIB A. TAMBOLI

H.O.D(EXTC):

Asst. Prof. MUJIB A. TAMBOLI

Date:

Place:

DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

1.__________________________________

SHAIKH SADAF SHAKIL.

12ET08

2.__________________________________

SIDDIQUI MISBA JAMATALI

12ET11

3.__________________________________

ANSARI ZUBER KHURSHID

12ET14

4.._________________________________

SAYYED MD. SHANAWAZ

11ET48

Date:

Place:

ACKNOWLEDGEMENT

We appreciate the beauty of a rainbow, but never do we think that we need both the sun and the rain to make its colors appear. Similarly, this project work is the fruit of many such unseen hands. It’s those small inputs from different people that have lent a helping to our project.

I take this opportunity to express my profound gratitude and deep regards to my guide **Asst.Prof. MUJIB.A.TAMBOLI** for his exemplary guidance, monitoring and constant encouragement throughout the course of this project work.

I also take this opportunity to express a deep sense of gratitude to **Asst.Prof. MUJIB A.TAMBOLI**, HOD of E.X.T.C. Dept. for his cordial support, valuable information and guidance, which helped me in completing this task through various stages.

I am obliged to staff members of **AIKTC**, for the valuable information provided by them in their respective fields. I am grateful for their cooperation during the period of my project work.

MIMO-OFDM

1: ABSTRACT

Multiple Input Multiple Output (MIMO) systems use more than one antenna at both ends of the communication link. Over the past decade, the use of MIMO system has rapidly gained popularity due to its enhanced performance capabilities of improved Reliability, Spatial Diversity Gain and Spatial Multiplexing Gain. Orthogonal Frequency Division Multiplexing (OFDM) is one of the best digital modulation schemes, where signal is divided into number of narrow band signals to obtain spectrum efficiency and minimizing the Inter Symbol Interference (ISI). Thus, combining MIMO and OFDM technologies will improve spectral efficiency, Link reliability, spectral gain and data rate.

The main aim of MIMO-OFDM systems is to combine OFDM technology with the techniques of MIMO systems. The growing demand of multimedia services and the growth of Internet related contents lead to increasing interest to high speed communications. OFDM is a type of multichannel modulation that divides a given channel into many parallel sub channels or subcarriers, so that multiple symbols are sent in parallel.

The proposed MIMO-OFDM system has distinguished advantages over the conventional SISO , SIMO and MISO systems and this is being implemented in 4G cellular and in various other emerging communication technologies.

TABLE OF CONTENT

Project approval for B.E ____________ I

Declaration ____________ II

Acknowledgement ____________ III

1. Abstract ____________ 6
2. Introduction ____________ 8

   2.1 Types of antenna

   2.2 Why MIMO??
3. Literature Survey ____________ 12

   3.1 New transmission scheme for mimo-ofdm system

   3.2 MIMO-OFDM wireless systems: basics, perspectives, and challenges

   3.3 Evaluation of BER for AWGN, Rayleigh and Rician Fading Channels under Various Modulation Schemes

   3.4 BER Comparison of Rayleigh Fading, Rician Fading and AWGN Channel using Chaotic Communication based MIMO-OFDM System
4. Problem statement ____________ 15
5. Methodology ____________ 15
6. System requirement ____________ 16
7. Design and working ____________ 17

   7.1 Binary format

   7.2 Convolution encoder and decoder

   7.3 General block interleaver/de-interleaver

   7.4 QAM modulation and demodulation

7.5 Pilot signal

7.6 ADC and DAC

7.7 Serial to parallel and parallel to serial converter

7.8 Add cyclic prefix

7.9 FFT and IFFT

7.10 OFDM transmitter and receiver

7.11 Symbol detection

8. Software and Hardware ______ 38

9. Working ______ 46

10. Results and discussion ______ 52

11. Future scope ______ 53

12. Conclusion ______ 54

13. Reference ______ 55

14. Appendix ______ 56

2:INTRODUCTION

Wireless communication involves the transfer of data without the use of wires. The earliest uses of wireless technology were very limited and data transferred was minimal, hence the available spectrum was sufficient. Communication technology is growing day by day, and system devices are reducing in size as well as being capable of increased processing power. With the growth in wireless communication, customers are demanding more improved and attractive applications, due to which there is a need for improvement in capacity of the available wireless spectrum. Many major technologies have been developed to meet user requirements. 4G/WiMAX technology will provide high data rate and capacity.

Designing the future wireless system with multiple input and output is emerging in a high rate. Multiple Transmit and receive antenna are now widely used to form Multiple Input Multiple Output (MIMO) system used in wireless communications offers various benefits such as higher capacity (bits/s/Hz) through spatial multiplexing scheme and better transmission quality (Bit Error Rate, outage) through transmit diversity scheme(Space Time Block Coding).

The wireless communication devices must have very high spectrum efficiency and the capacity of overcoming the channel fading in the environment of multi-path channel. In wireless telecommunications, multipath is the propagation phenomenon that results in radio signals' reaching the receiving antenna by two or more paths. Causes of multipath include atmospheric ducting, ionosphere reflection and refraction, and reflection from water bodies and terrestrial objects such as mountains and buildings. The effects of multipath include constructive and destructive interference, and phase shifting of the signal. Since the shape of the signal conveys the information being transmitted, the receiver will make mistakes when demodulating the signal's information. If the delays caused by multipath are great enough, bit errors in the packet will occur. The receiver won't be able to distinguish the symbols and interpret the corresponding bits correctly. This leads to an error in the symbol decoding. It is very difficult to match these requirements using the traditional modulation technique including MIMO & Orthogonal Frequency Division Multiplexing (OFDM); however the hybrid MIMO-OFDM system can meet these requirements.

MIMO-OFDM

Hybridization of MIMO-OFDM system is a combination of MIMO and OFDM technologies. MIMO is an antenna technology which uses multiple antennas at both the receiver and transmitter side. OFDM is the one of the best digital modulation techniques which splits the signal into several narrow band channels to obtain spectral efficiency. Some of the features of 4G technologies are supporting multimedia, video streaming, internet and other broadband services.

MIMO systems take advantage of the multiple signals to improve the quality and reliability of the transmitted information signal as the information in wireless channels is mainly affected by multipath fading. Multipath results in the multiple copies of the transmitted information at the receiver with some delays. OFDM uses the spectrum very efficiently by overlapping the sub carriers. It increases the data rate, reduces the ISI (Inter Symbol Interference) and utilizes the spectrum very effectively which is required for transmission of video and other multimedia messages.

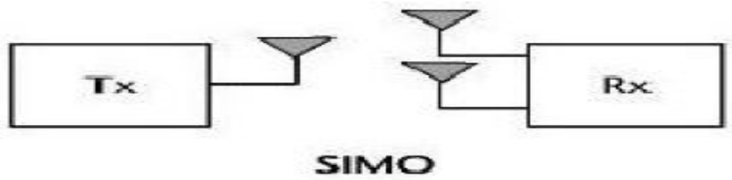
Types of Antenna

SISO

Single-Input Single-Output is the classical method in wireless communication and the most common antenna configuration, using one antenna at transmitter and one at the receiver. It is used in radio, TV broadcast and in technology as WiFi, Bluetooth.

SIMO

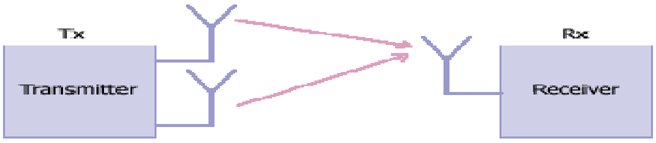
Single-Input Multiple-Output is the system using one antenna at transmitter and multiple antennas at the receiver. It provides receiver diversity which receive the strongest signal from several transmit antennas. Generally, it is used in Uplink environment.



1x2 SIMO system

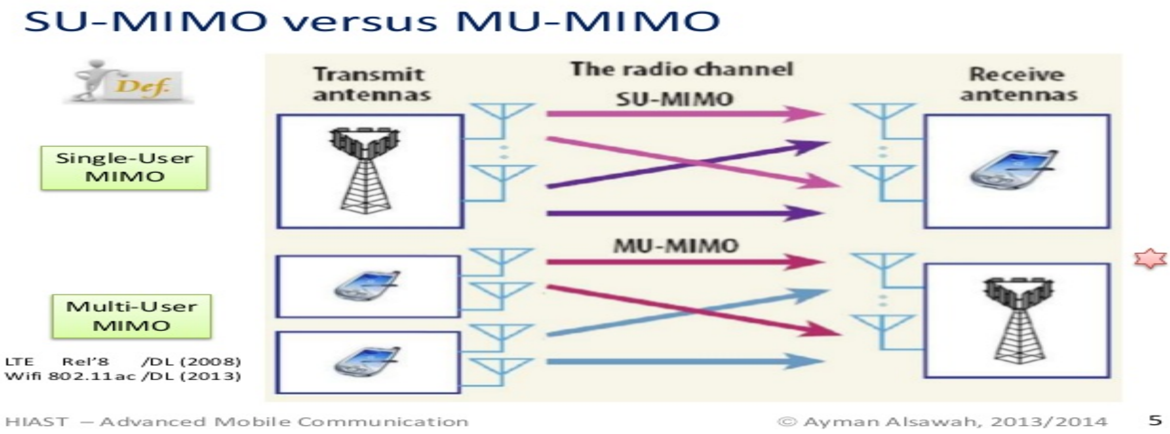
MISO

In Multiple-Input Single-Output two or more number of antennas are used in the transmitter and one antenna at the receiver. It provides transmit diversity because of multiple antenna at a transmitter side. MISO technology has applications in WLAN, MAN and digital television (DTV). Commonly, it is used in downlink scenarios.



MIMO-OFDM

MIMO:- Multiple-Input Multiple-Output uses multiple antennas at both sides which provides transmit diversity and receiver diversity. It's applicable in every kind of networks like PAN, LAN, WLAN, WAN, MAN. MIMO system can be applied in different ways to receive either a diversity gain, capacity gain or to overcome signal fading.



WHY MIMO?

The typical aspirations of a system designer are high data rate, low bit error rate, low power consumption, low cost and easy implement ability. The MIMO system ensures us very high data rates even more than 1Gbps while minimizing the bit error rate. By Shanon's theorem the rate of transmission is always less than or equal to the capacity. Practically it is less than the capacity. The capacity depends on the bandwidth of the channel and SNR of the channel. Both the bandwidth and signal to noise ratio are characteristics of the channel. The SNR can be improved either by reducing noise power or by increasing signal power. Reduction in noise power is not possible while increase in signal power requires more power for transmission which should be avoided for a good design. The improvement of bandwidth is not possible. However there are techniques like OFDM (orthogonal frequency division multiplexing) which assure us efficient use of the channel i.e. spectral efficiency. But however the use of multiple antennas at the transmitter and at the receiver that is use of MIMO meets the ongoing requirements in 4G. The bit error rate in MIMO is very less as compared to conventional SISO systems.

3:LITERATURE SURVEY

NEW TRANSMISSION SCHEME FOR MIMO-OFDM SYSTEM

This contribution introduces a new transmission scheme for multiple-input multiple-output (MIMO) orthogonal frequency division multiplexing (OFDM) systems. The new scheme is efficient and suitable especially for symmetric channels such as the link between two base stations or between two antennas on radio beam transmission. The principle is based on the estimation of channel parameters of a pilot data send by the receiver to the transmitter. Then, the transmitter codes the transmitted signal using the estimated channel parameters to adapt the signal to the channel variations. Conducted Monte-Carlo simulation results show that the proposed scheme has better performance, in terms of bandwidth efficiency and complexity, compared to the conventional MIMO-OFDM scheme methods in the case of a symmetric channel.

MIMO-OFDM WIRELESS SYSTEMS: BASICS, PERSPECTIVES, AND CHALLENGES

Multiple-input multiple-output (MIMO) wireless technology in combination with orthogonal frequency division multiplexing (MIMOOFDM) is an attractive air-interface solution for next-generation wireless local area networks (WLANs), wireless metropolitan area networks (WMANs), and fourth-generation mobile cellular wireless systems. This article provides an overview of the basics of MIMO-OFDM technology and focuses on space-frequency signaling, receiver design, multiuser systems, and hardware implementation aspects. We conclude with a discussion of relevant open areas for further research.

Evaluation of BER for AWGN, Rayleigh and Rician Fading Channels under Various Modulation Schemes

Several transmission modes are defined in IEEE 802.11 a/b/g WLAN standards. A very few transmission modes are considering for IEEE 802.11 a/b/g in physical layer parameters and wireless channel characteristics. In this paper, we evaluated the performance of available transmission modes in IEEE 802.11b. However, the performance analysis can be done Straight forward using the evaluation of IEEE 802.11b. The performance of transmission modes are evaluated by calculating the probability of Bit Error Rate (BER) versus the Signal Noise Ratio (SNR) under the frequently used three wireless channel models (AWGN, Rayleigh and Rician). We consider the data modulation and data rate to analyze the performance that is BER vs. SNR. We also consider multipath received signals. The simulation results had shown the performance of transmission
modes under different channel models and the number of antennas. Based on simulation results, we observed that some transmission modes are not efficient in IEEE 802.11b. The evaluation of performance confirms the increase in the coverage area of the physical layer in the 802.11b WLAN devices.

BER Comparison of Rayleigh Fading, Rician Fading and AWGN Channel using Chaotic Communication based MIMO-OFDM System

This paper proposes a technique which uses chaotic communication system combined with adaptive beam forming, for secure communications and to improve the system performance by mitigating interference. For secure communications, chaotic sequences are used. Many chaotic communication systems have been proposed, but most of them show a poor performance under realistic channel conditions (i.e. noise and multipath fading).This paper proposes a wireless communication structure based on two coupled chaotic systems. In order to enhance the error-rate performance of MIMO-OFDM system, adaptive beam forming is used. Evaluation and comparison of the performances of MIMO- OFDM system in the AWGN (Additive White

Gaussian Noise) channel, Rician fading channel and the Rayleigh fading channel are provided. Results are verified and analyzed for two cases, one when adaptive beam forming is used in the proposed system and second when adaptive beam forming is not used in the proposed system. Computer simulations are done to verify the performance of the proposed approach. A simulation tool with a Graphical User Interface (GUI) which implements these algorithms is also developed to provide ease in the execution.

Implementation of OFDM Transmitter and Receiver Using FPGA

Orthogonal Frequency Division Multiplexing (OFDM) is the most promising modulation technique. It has been adopted by most wireless and wired communication standards. The idea is to utilize a number of carriers, spread regularly over a frequency band, in such a way so that the available bandwidth is utilized to maximal efficiency. The objective of this paper is to carry out an efficient implementation of the OFDM system (i.e. transmitter and receiver) using “Field Programmable Gate Array (FPGA)” and find the result by simulating all the blocks used in proposed project by using QuartusII & Modelsim simulation tool.

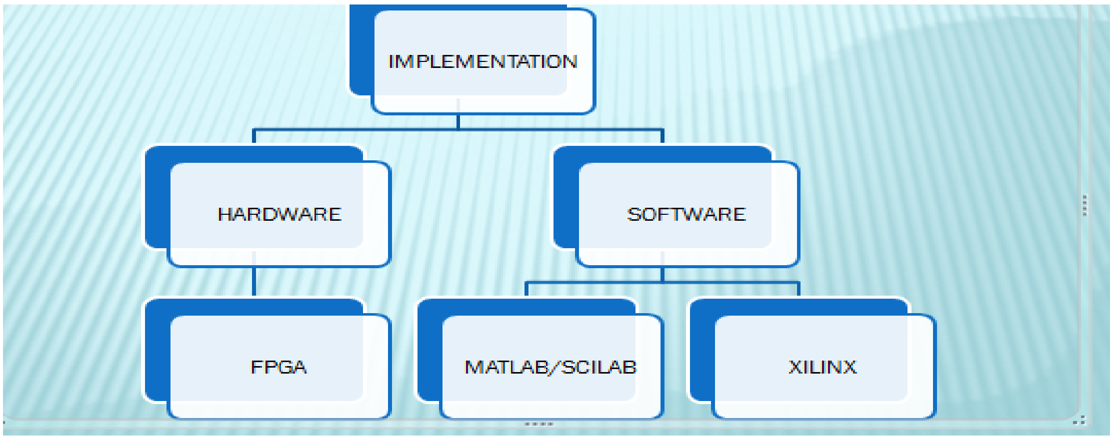
MIMO-OFDM

PROBLEM STATEMENT

The available spectrum is fixed but the demand for high data rate and high reliability is growing day by day to suite the more improved and attractive applications. In this project, we implement and optimize the 4x4 MIMO-OFDM transmitter and receiver for different embedded platform. The idea behind developing the MIMO-OFDM transmitter is to get the advantage of both MIMO and the OFDM technology in achieving higher data rate, spectral efficiency and reliability**.** To reduce the bit error rate of MIMO-OFDM from 10^{-2} to 10^{-6} using 4x4 array.

METHODOLOGY

Wireless communication using multiple-input multiple-output (MIMO) system enables increased spectral efficiency for a given total transmit power. The capacity is increased by introducing additional spatial channels that are exploited by using space-time coding. Multiple antennas are used at both the source (transmitter) and the destination (receiver) side. The antennas at each end of the communications system are combined to minimize errors and optimize data speed.



MIMO-OFDM

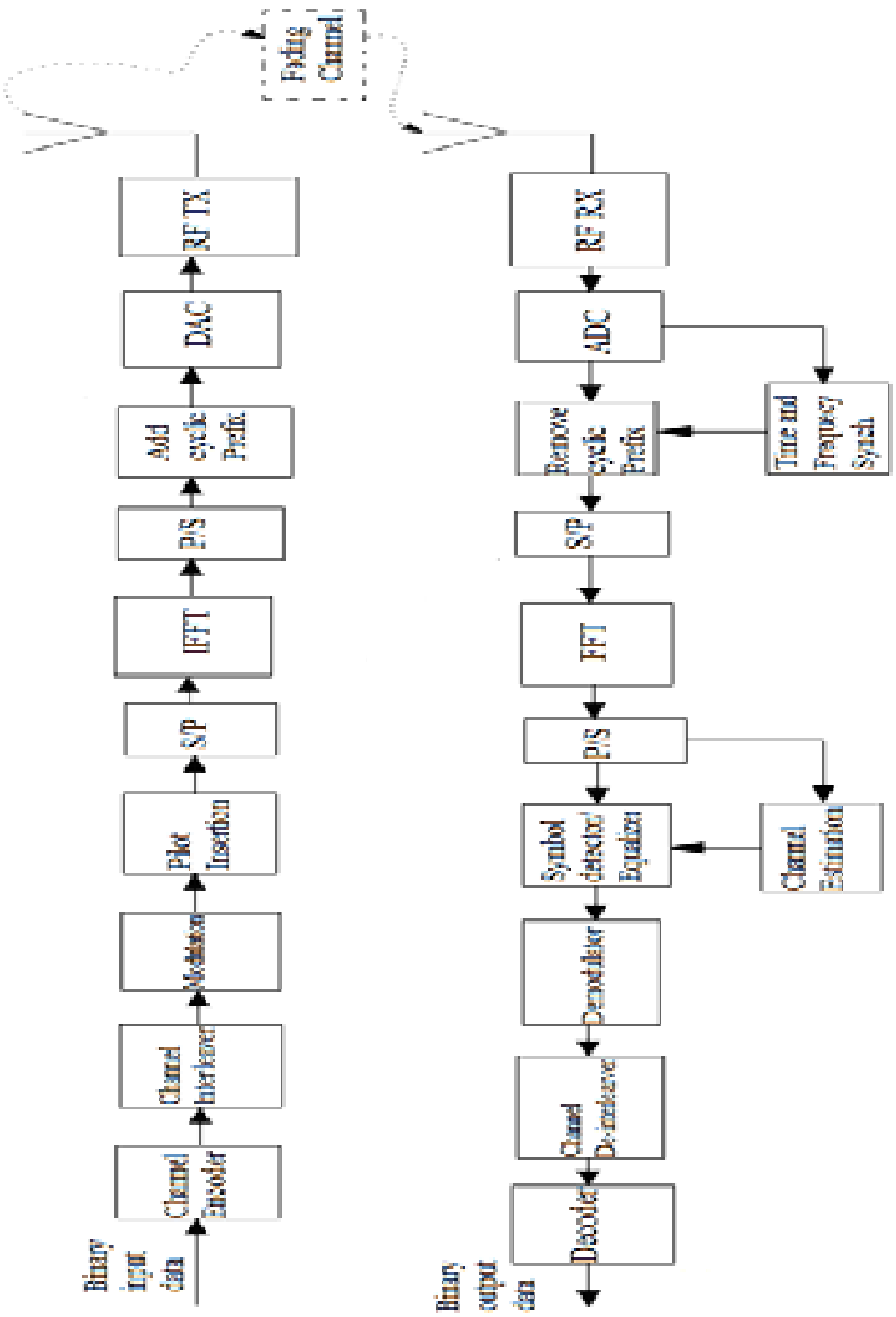
SYSTEM REQUIREMENTS

The following are the system requirement specifications arrived for different subsystems of the system.

• MIMO system typically consists of *m* transmitting antennas and *n* receiving antennas.
• In this system channel is assumed to be a static channel and that the channel is known perfectly at the receiver for all the systems being developed.
• OFDM system generally consists of different sub blocks namely FFT block, Cyclic prefix and parallel to serial conversion.
• Both spatial diversity and spatial multiplexing techniques should be supported.Channel is assumed to be a static channel and is known perfectly at the receiver for all the systems being developed.

5:DESIGN AND WORKING

BLOCK DIAGRAM

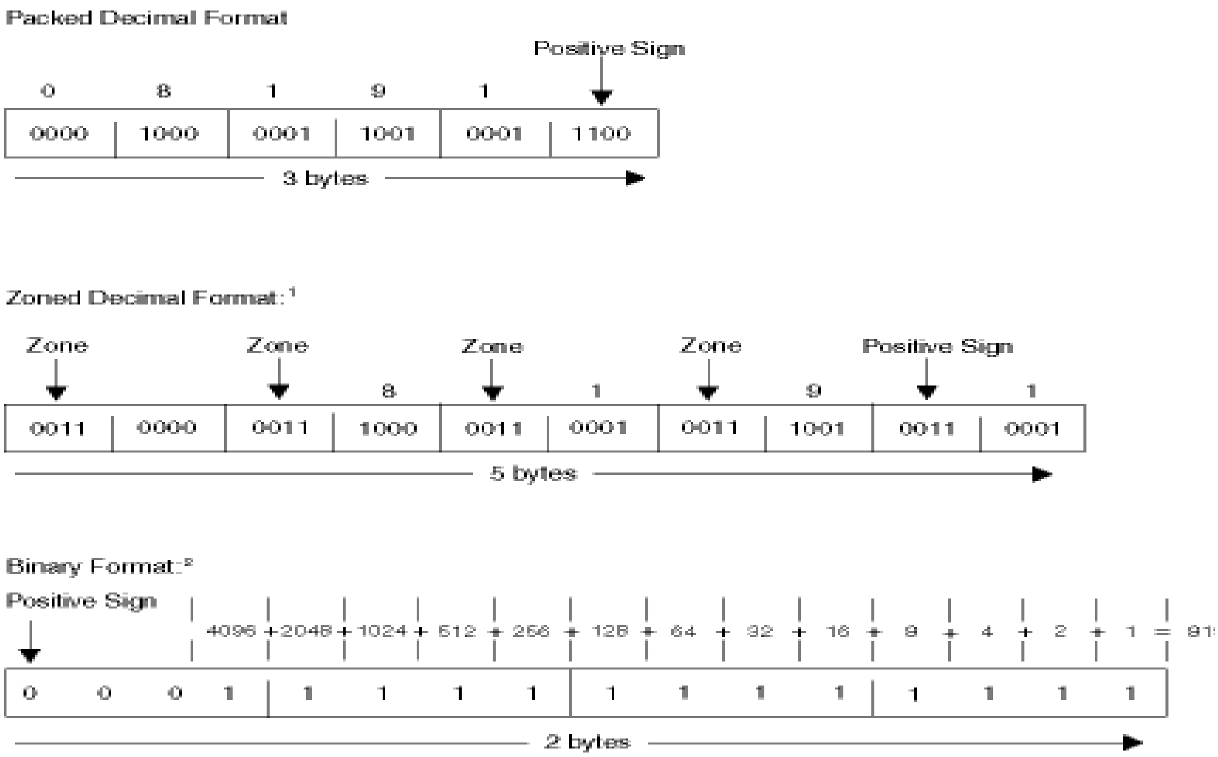


5.1 Binary Format

Binary format means that the sign (positive or negative) is in the leftmost bit of the field and the integer value is in the remaining bits of the field. Positive numbers have a zero in the sign bit; negative numbers have a one in the sign bit and are in twos complement form. In binary format, each field must be either 2 or 4 bytes long.

A binary field can be from one to nine digits in length and can be defined with decimal positions. If the length of the field is from one to four digits, the compiler assumes a binary field length of 2 bytes. If the length of the field is from five to nine digits, the compiler assumes a binary field length of 4 bytes.

Defining Binary Fields



If 8191 is read into storage as a zoned-decimal field, it occupies 4 bytes. If it is converted to packed-decimal format, it occupies 3 bytes. When it is converted back to zoned-decimal format, it occupies 5 bytes.[2]To obtain the numeric value of a positive binary number add the values of the bits that are on (1), do not include the sign bit. To obtain the numeric value of a negative binary number, add the values of the bits that are off (0) plus one (the sign bit is not included).

5.2 Convolutional Encoder/Decoder

Description

The Convolutional Encoder block encodes a sequence of binary input vectors to produce a sequence of binary output vectors. This block can process multiple symbols at a time.

This block can accept inputs that vary in length during simulation. For more information about variable-size signals, see Variable-Size Signal Basics in the Simulink documentation.

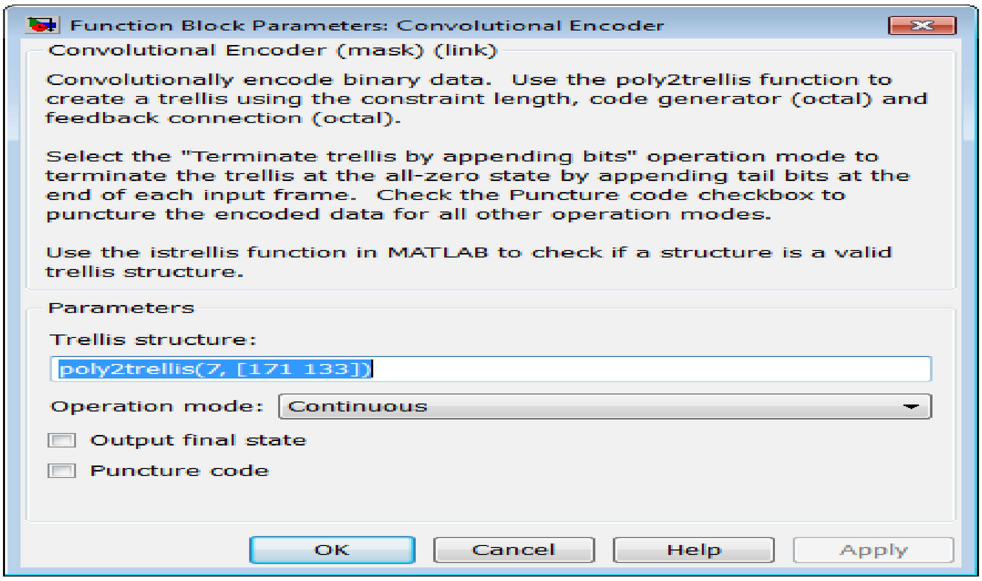
Input and Output Sizes

If the encoder takes k input bit streams (that is, it can receive 2^k possible input symbols), the block input vector length is L*k for some positive integer L. Similarly, if the encoder produces n output bit streams (that is, it can produce 2^n possible output symbols), the block output vector length is L*n.

This block accepts a column vector input signal with any positive integer for L. For variable-size inputs, the L can vary during simulation. The operation of the block is governed by the **Operation mode** parameter.

For both its inputs and outputs for the data ports, the block supports double, single, boolean, int8, uint8, int16, uint16, int32, uint32, and ufix1. The port data types are inherited from the signals that drive the block. The input reset port supports double and boolean typed signals.

Dialog Box



Trellis structure

MATLAB structure that contains the trellis description of the convolutional encoder.

5.3Viterbi Decoder

Decode convolutionally encoded data using Viterbi algorithm

Convolutional sublibrary of Error Detection and Correction

Description

The Viterbi Decoder block decodes input symbols to produce binary output symbols. This block can process several symbols at a time for faster performance.

This block can output sequences that vary in length during simulation. For more information about sequences that vary in length, or variable-size signals, see Variable-Size Signal Basics in the Simulink® documentation.

Input and Output Sizes

If the convolutional code uses an alphabet of 2^n possible symbols, this block's input vector length is L*n for some positive integer L. Similarly, if the decoded data uses an alphabet of 2^k possible output symbols, this block's output vector length is L*k.

This block accepts a column vector input signal with any positive integer value for L. For variable-sized inputs, the L can vary during simulation. The operation of the block is governed by the operation mode parameter."

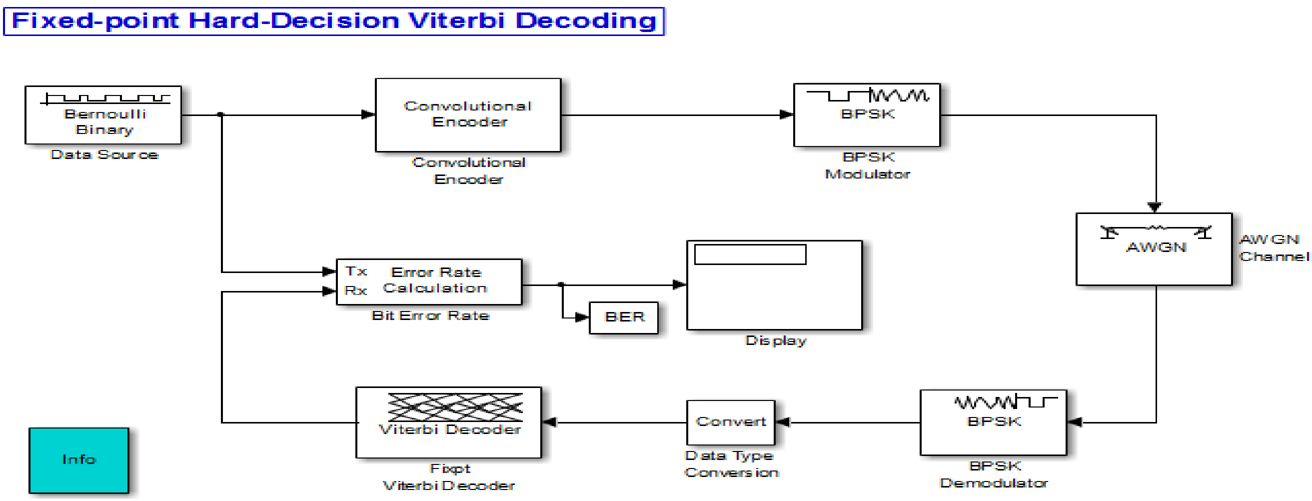
Input Values and Decision Types

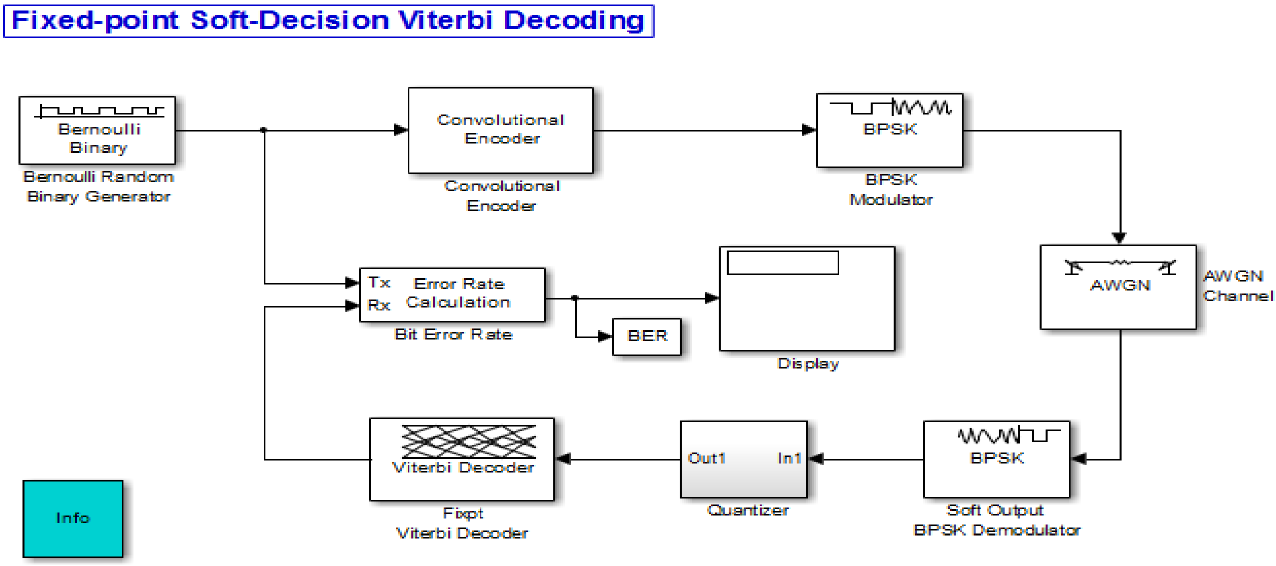
The entries of the input vector are either bipolar, binary, or integer data, depending on the Decision type parameter.

Fixed-Point Viterbi Decoding Examples

The following two example models showcase the fixed-point Viterbi decoder block used for both hard- and soft-decision convolutional decoding.

If you are reading this reference page in the MATLAB® Help Browser, click Fixed-point Hard-Decision Viterbi Decoding and Fixed-point Soft-Decision Viterbi Decoding to open the models. These can also be found as doc_fixpt_vitharddec.mdl and doc_fixpt_vitsoftdec.mdl under help\toolbox\commm\examples .





Comparisons Between Hard and Soft-Decision Decoding

The two models are set up to run from within BERTool to generate a simulation curve that compares the BER performance for hard-decision versus soft-decision decoding.

To generate simulation results for doc_fixpt_vitharddec.mdl, do the following:

1. Type bertool at the MATLAB command prompt.
2. Go to the **Monte Carlo** pane.
3. Set the **Eb/No range** to 2:5.
4. Set the **Simulation model** to doc_fixpt_vitharddec.mdl. Make sure that the model is on path.
5. Set the **BER variable name** to BER.
6. Set the **Number of errors** to 100, and the **Number of bits** to 1e6.
7. Press **Run** and a plot generates.

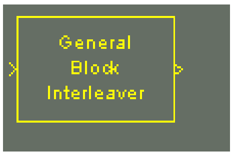
5.4General Block Interleaver

Reorder symbols in input vector

Library

Block sublibrary of Interleaving

Description



The General Block Interleaver block rearranges the elements of its input vector without repeating or omitting any elements. If the input contains *N* elements, then the **Permutation vector** parameter is a column vector of length *N*. The column vector indicates the indices, in order, of the input elements that form the length-*N* output vector; that is,

Output(*k*) = Input(**Permutation vector**(k))

for each integer *k* between 1 and *N*. The contents of **Permutation vector** must be integers between 1 and N, and must have no repetitions.

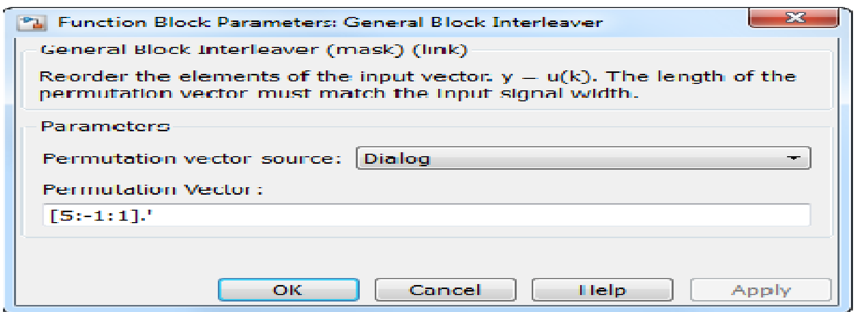
Both the input and the **Permutation vector** parameter must be column vector signals.

This block can output sequences that vary in length during simulation. For more information about sequences that vary in length, or variable-size signals, see Variable-Size Signal Basics in the Simulink® documentation.

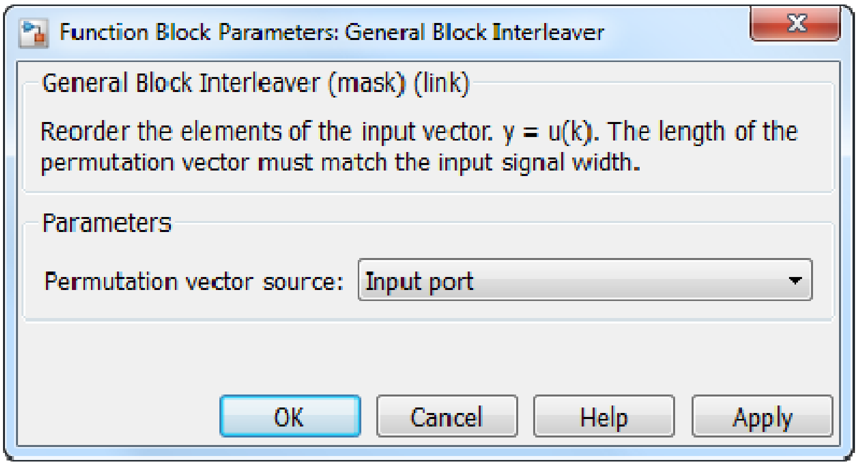
This block accept the following data types: int8, uint8, int16, uint16, int32, uint32, boolean, single, double, and fixed-point. The output signal inherits its data type from the input signal.

Dialog Box

The General Block Interleaver dialog box is shown for the case in which the **Permutation vector source** parameter is set to Dialog.



The General Block Interleaver dialog box is shown for the case in which the **Permutation vector source** parameter is set to Input port.



Permutation vector source

A selection that specifies the source of the permutation vector. The source can be either Dialog or Input port. The default value is Dialog.

Permutation vector

A vector of length N that lists the indices of the output elements that came from the input vector. This parameter is available only when **Permutation vector source** is set toDialog.

Examples

If **Permutation vector** is [4;1;3;2] and the input vector is [40;32;59;1], then the output vector is [1;40;59;32]. Notice that all of these vectors have the same length and that the vector **Permutation vector** is a permutation of the vector [1:4]'.

5.5General Block Deinterleaver

Restore ordering of symbols in input vector

Library

Block sublibrary of Interleaving

Description

The General Block Deinterleaver block rearranges the elements of its input vector without repeating or omitting any elements. If the input contains *N* elements, then the **Permutation vector** parameter is a column vector of length *N*. The column vector indicates the indices, in order, of the output elements that came from the input vector. That is, for each integer *k*between 1 and *N*,

Output(**Permutation vector**(*k*)) = Input(*k*)

The **Permutation vector** parameter must contain unique integers between 1 and *N*.

Both the input and the **Permutation vector** parameter must be column vector signals.

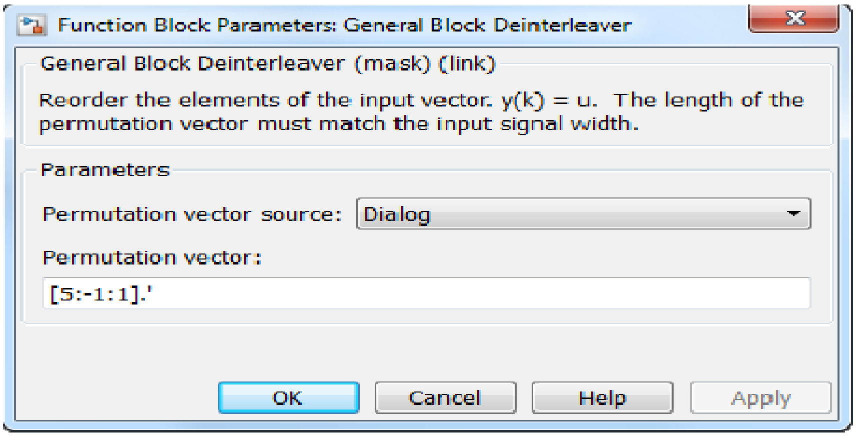
This block can output sequences that vary in length during simulation. For more information about sequences that vary in length, or variable-size signals, see Variable-Size Signal Basics in the Simulink® documentation.

This block accept the following data types: int8, uint8, int16, uint16, int32, uint32, boolean, single, double, and fixed-point. The output signal inherits its data type from the input signal.

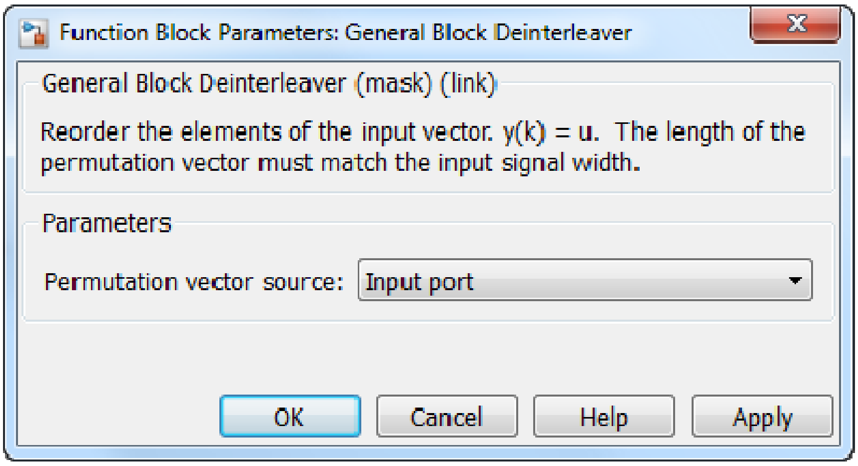
To use this block as an inverse of the General Block Interleaver block, use the same **Permutation vector** parameter in both blocks. In that case, the two blocks are inverses in the sense that applying the General Block Interleaver block followed by the General Block Deinterleaver block leaves data unchanged.

Dialog Box

The General Block Deinterleaver dialog box is shown for the case in which the **Permutation vector source** parameter is set to Dialog.



The General Block Deinterleaver dialog box is shown for the case in which the Permutation vector source parameter is set to Input port.



Permutation

vector source

A selection that specifies the source of the permutation vector. The source can be either Dialog or Input port. The default value is Dialog.

Permutation vector

A vector of length N that lists the indices of the output elements that came from the input vector. This parameter is available only when **Permutation vector source** is set toDialog.

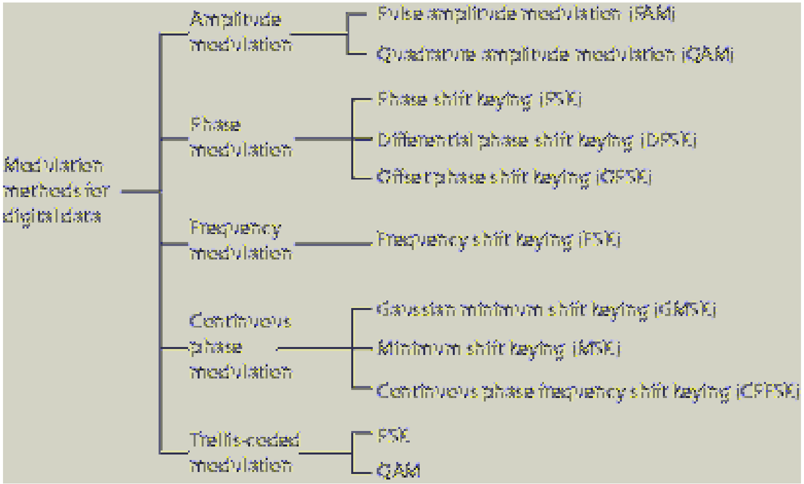
Examples

This example reverses the operation in the example on the General Block Interleaver block reference page. If you set **Permutation vector** to [4,1,3,2]' and you set the General Block Deinterleaver block input to [1;40;59;32], then the output of the General Block Deinterleaver block is [40;32;59;1].

5.6 QAM Modulator & Demodulator

Techniques modulation

The figure below shows the modulation techniques that Communications System Toolbox™ supports for digital data. All the methods at the far right are implemented in library blocks.



Like analog modulation, digital modulation alters a transmittable signal according to the information in a message signal.

SF=16, transport block size 480 bits (including CRC)								
	2 codes		4 codes		6 codes		8 codes	
MCS	# trblks	Bits/TTI	# trblks	Bits/TTI	# trblks	Bits/TTI	# trblks	Bits/TTI
QPSK, rate 1/4	1	480	2	960	3	1440	4	1920
QPSK, rate 1/2	2	960	4	1920	6	2880	8	3840
QPSK, rate 3/4	3	1440	6	2880	9	4320	12	5760
8PSK, rate 1/2	3	1440	6	2880	9	4320	12	5760
8PSK, rate 2/3	4	1920	8	3840	12	5760	16	7680
8PSK, rate 5/6	5	2400	10	4800	15	7200	20	9600
16QAM, rate 1/2	4	1920	8	3840	12	5760	16	7680
16QAM, rate 5/8	5	2400	10	4800	15	7200	20	9600
16QAM, rate 3/4	6	2880	12	5760	18	8640	24	11520
64QAM, rate 3/4	9	4320	18	8640	27	12960	36	17280

The QAM modulator and demodulator are essential building blocks within an overall QAM system

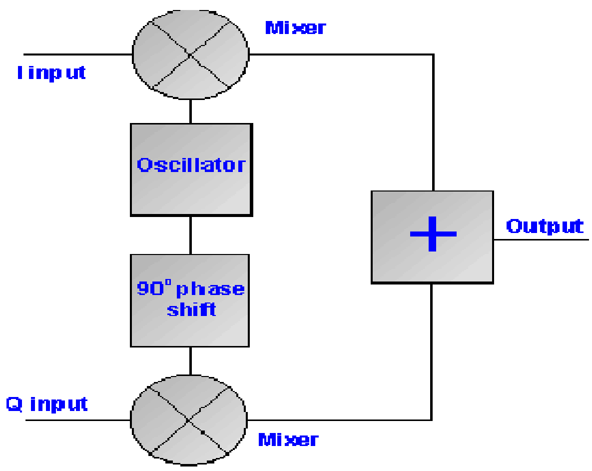
The QAM modulator and QAM demodulator are key elements within any quadrature amplitude modulation system.

The modulator and demodulator are used to encode the signal, often data, onto the radio frequency carrier that is to be transmitted. Then the demodulator is used at the remote end to extract the signal from the RF carrier so that it can used at the remote end.

As quadrature amplitude modulation is a complex signal, specialised QAM modulators and demodulators are required.

QAM modulator basics

The QAM modulator essentially follows the idea that can be seen from the basic QAM theory where there are two carrier signals with a phase shift of 90掳 between them. These are then amplitude modulated with the two data streams known as the I or In-phase and the Q or quadrature data streams. These are generated in the baseband processing area.



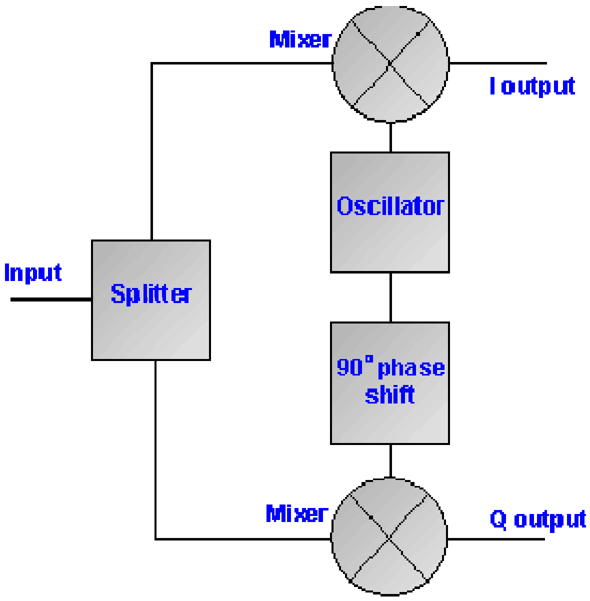
Basic QAM modulator diagram

The two resultant signals are summed and then processed as required in the RF signal chain, typically converting them in frequency to the required final frequency and amplifying them as required.

QAM demodulator basics

The QAM demodulator is very much the reverse of the QAM modulator.

The signals enter the system, they are split and each side is applied to a mixer. One half has the in-phase local oscillator applied and the other half has the quadrature oscillator signal applied.



Basic QAM demodulator diagram

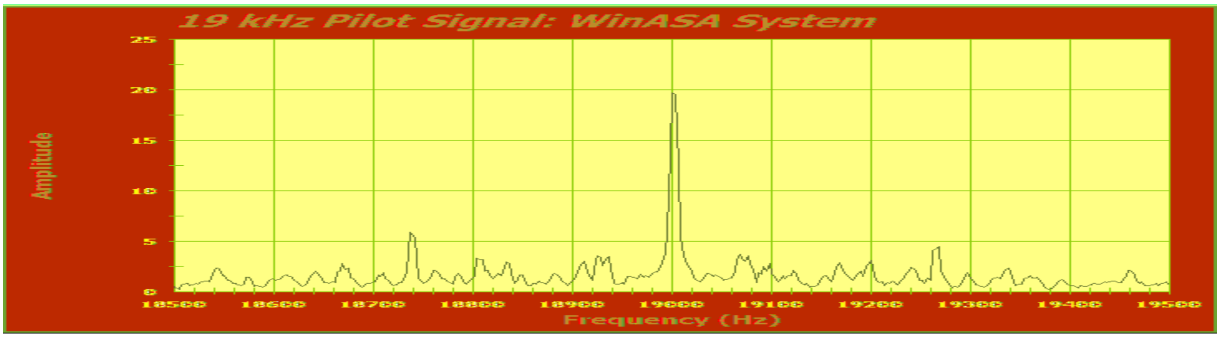
The basic modulator assumes that the two quadrature signals remain exactly in quadrature.

A further requirement is to derive a local oscillator signal for the demodulation that is exactly on the required frequency for the signal. Any frequency offset will be a change in the phase of the local oscillator signal with respect to the two double sideband suppressed carrier constituents of the overall signal.Systems include circuitry for carrier recovery that often utilises a phase locked loop - some even have an inner and outer loop. Recovering the phase of the carrier is important otherwise the bit error rate for the data will be compromised.Modulation is the addition of information to an electronic or optical carrier signal. A carrier signal is one with a steady waveform -- constant height (amplitude) and frequency. Information can be added to the carrier by varying its amplitude, frequency, phase, polarization (for optical signals), and even quantum-level phenomena like spin.

5.7 Pilot signal

"Pilot tone" redirects here. For pilot tones in motion picture sound recording systems, see Pilot tone.

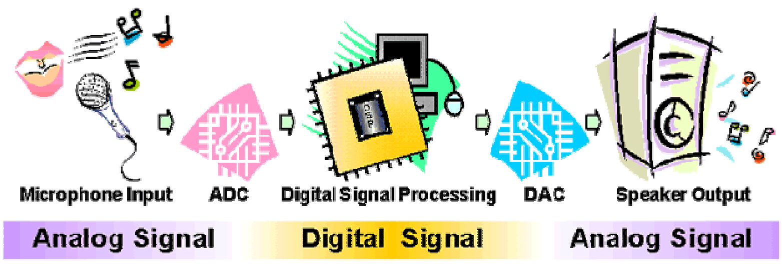
In telecommunications, a **pilot signal** is a signal, usually a single frequency, transmitted over a communications system for supervisory, control, equalization, continuity, synchronization, or reference purposes. Spectrum of an FM broadcast signal. The pilot tone is the orange vertical line on the right of the spectrogram. In FM stereo broadcasting, a **pilot tone** of 19 kHz indicates that there is stereophonic information at 38 kHz (19×2, the second harmonic of the pilot). The receiver doubles the frequency of the pilot tone and uses it as a phase reference to demodulate the stereo information. If no 19 kHz pilot tone is present, then any signals in the 23-53 kHz range are ignored by a stereo receiver. A guard band of ±4 kHz (15-23 kHz) protects the pilot tone from interference from the baseband audio signal (50 Hz-15 kHz) and from the lower sideband of the double sideband stereo information (23-53 kHz). The third harmonic of the pilot (19×3, or 57 kHz) is used for Radio Data System. In AM stereo, the bandwidth is too narrow to accommodate subcarriers, so the modulation itself is changed, and the pilot tone is infrasonic (below the normal hearing range, instead of above it) at a frequency of 25 Hz. In color television, the color burst placed between each pair of video fields is the pilot signal to indicate that there are color subcarriers present. In the NTSC television system, a pilot tone of 15.7342657 kHz is used to indicate the presence of MTS stereo. In some analog video formats (Frequency modulation is the standard method for recording the luminance part of the signal, and is used to record a composite video signal in Direct color systems), e.g. Video 2000 and some Hi-band formats a pilot tone is added to the signal to detect and correct time base errors.Sometimes it is necessary to employ several independent pilot frequencies. Most radio relay systems use radio or continuity pilots of their own but transmit also the pilot frequencies belonging to the carrier frequency multiplex system.

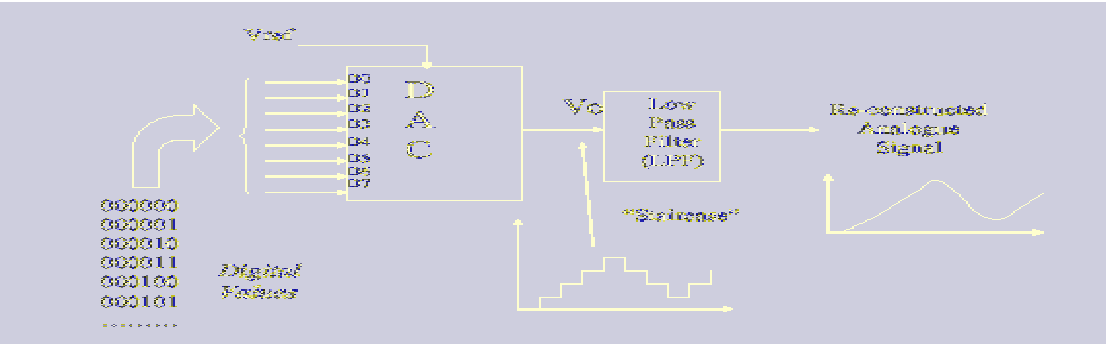


5.8 ADC and DAC

Digital to Analog Converter:-

In modern life, electronic equipment is frequently used in different fields such as communication, transportation, entertainment, etc. Analog to Digital Converter (ADC) and Digital to Analog Converter (DAC) are very important components in electronic equipment. Since most real world signals are analog, these two converting interfaces are necessary to allow digital electronic equipments to process the analog signals. Take the audio signal processing in Figure 1 as an example, ADC converts the analog signal collected by audio input equipment, such as a microphone, into a digital signal that can be processed by computer. The computer may add sound effect such as echo and adjust the tempo and pitch of the music. DAC converts the processed digital signal back into the analog signal that is used by audio output equipment such as a speaker.





5.9 Analog To Digital Converter:-

In electronics, an Analog to Digital Converter (ADC) is a device for converting an analog signal (current, voltage etc.) to a digital code, usually binary. In the real world, most of the signals sensed and processed by humans are analog signals. Analog-to-Digital conversion is the primary means by which analog signal are converted into digital data that can be processed by computers for various purposes, Figure 3.

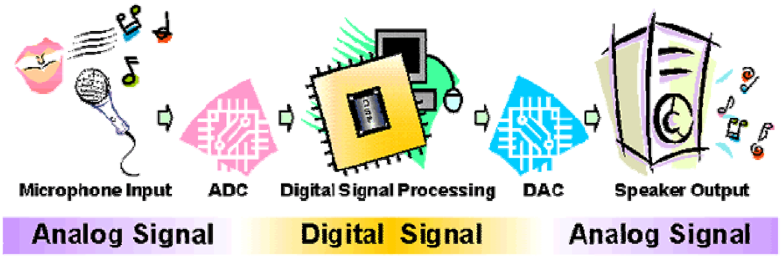
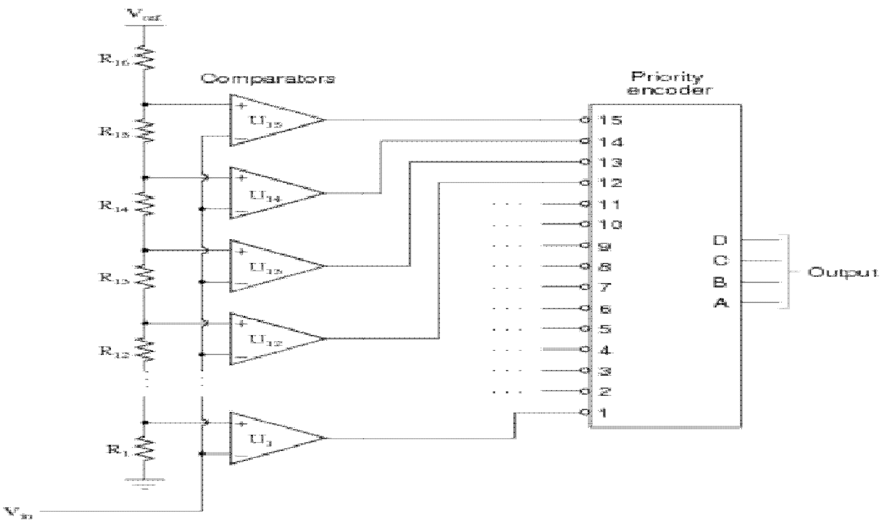


Figure 1: Audio Signal Processing



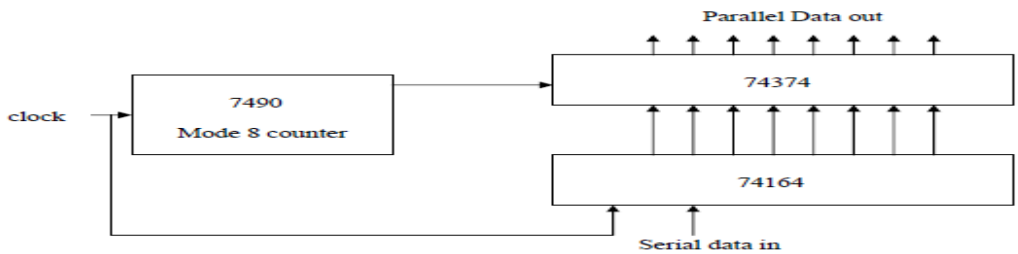
5.10 Serial to parallel converter

This is a simple method to convert a serial data that have been entered to the circuit into parallel one. The reverse parallel to serial can be done in a similar manner. In fact, the first one is useful in computer systems architecture. For example, in ALUs, data can be entered serially and the input, for ALU operations, can be applied in parallel. 2- Equipment:

Type of IC	Description
74164	8 bits serial to parallel Shift registers
74374	Octal D-flip-flop
7490	Decade Counter

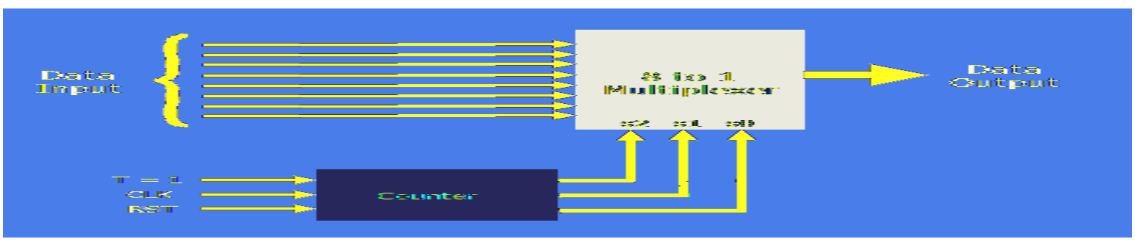
3- ***Procedure:***

- **The function diagram is following :**



5.11 Parallel to serial:-

when data is to be sent over a large distance then the parallel data are send serially over single wire(plus ground). To send parallel data serially from microprocessor or microcontroller or DSP chips, parallel to serial converter is required. Not only is parallel to serial converter required at the microprocessor chip level but also in other parts of communication system. For telephone system for example, parallel to serial and serial to parallel converters are required for implementing modulation/demodulation. QPSK-modulation is one such good example

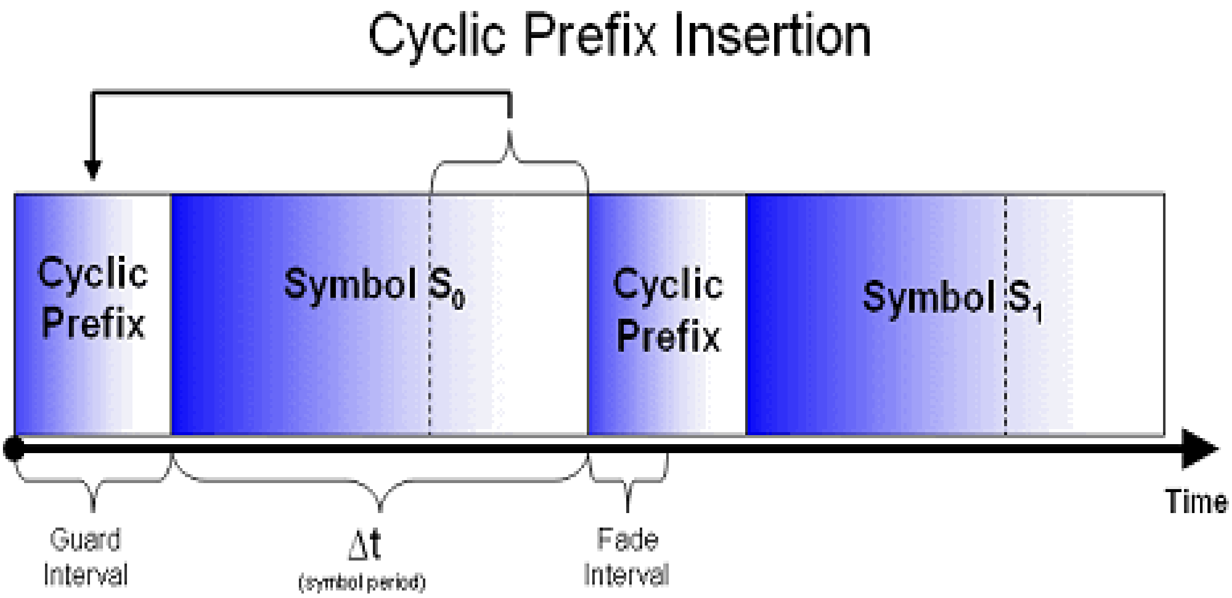


5.12 ADD CYCLIC PREFIX

The term ***cyclic prefix*** refers to the prefixing of a symbol with a repetition of the end. Although the receiver is typically configured to discard the cyclic prefix samples, the cyclic prefix serves two purposes.

- As a guard interval, it eliminates the inter symbol interference from the previous symbol.
- As a repetition of the end of the symbol, it allows the linear convolution of a frequency-selective multipath channel to be modelled as circular convolution, which in turn may be transformed to the frequency domain using a discrete Fourier transform. This approach allows for simple frequency-domain processing, such as channel estimation and equalization.

In order for the cyclic prefix to be effective (i.e. to serve its aforementioned objectives), the length of the cyclic prefix must be at least equal to the length of the multipath channel. Although the concept of cyclic prefix has been traditionally associated with OFDM systems, the cyclic prefix is now also used in single carrier systems to improve the robustness to multipath propagation



5.13 Fft and ifft

The OFDM transmitter and receiver contain Inverse Fast Fourier Transform (IFFT) and Fast Fourier Transform (FFT), respectively[6]. The IFFT/FFT algorithms are chosen due to their execution speed, flexibility and precision [3]. For real
time systems the execution speed is the main concern. The IFFT block providesorthogonality between adjacent subcarriers. The orthogonality makes the signal frame relatively secure to the fading caused by natural multipath
environment. As a result OFDM system has become very popular in modern telecommunication systems. The main objective of this paper is to design IFFT/FFT blocks for OFDM, because these are main blocks for modulation and
demodulation in OFDM transmitter and receiver [2]. The OFDM signal is generated by implementing the Inverse Fast Fourier Transform (IFFT) at the transmitter which is used to convert frequency domain to time domain and Fast FourierTransform (FFT) which is used to convert time domain to frequency domain at the receiver side is implemented.
The basic equation of the FFT is

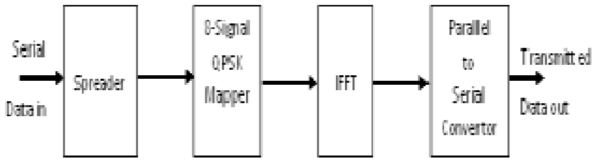
$$X(k)=\sum_{n=1}^{N-1} x(n)e^{-j2\pi kn/N} \quad ,k=0,\ldots,N-1$$

On the other hand, the Inverse FFT equation is

$$x(n)=\frac{1}{N}\sum_{n=1}^{N-1} X(k)e^{-j2\pi kn/N} \quad ,n=0,\ldots,N-1$$

where N is the transform size or the number of sample points in the data frame. X(k) is the frequency output of the FFT at kth point where k=0, 1, …, N-1 and x(n) is the time sample at
nth point with n=0, 1,…, N-1 [4].

OFDM Transmitter



IFFT

The IFFT transform a spectrum (amplitude and phase of each component) into a time domain signal. An IFFT converts a

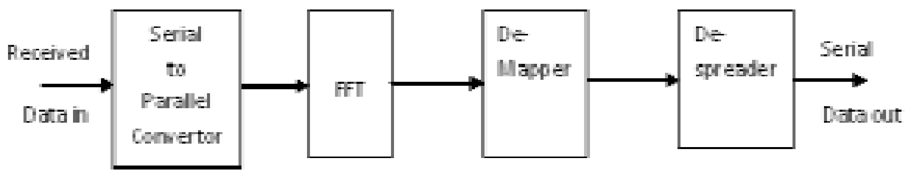
number of complex data points, of length that is power of 2, into the same number of points in time domain. Each data

point in frequency spectrum used for an FFT or IFFT operation is called a bin. The Inverse Fast Fourier Transform

(IFFT) performs N-Point IFFT operation for the received constellation points from the QPSK Mapper.. The output is of

N time domain samples. After N-point computation these values are passed through parallel to serial convertor

OFDM Receiver



FFT

FFT Converters time domain to frequency domain. The parallel symbols which are received from serial to parallel

converter perform N-Point FFT operation and sends to de mapper.

SYMBOL DETECTION

Symbol detection in multi-input multi-output (MIMO)communication systems using different particles warm optimization (PSO) algorithms is presented. This approach is particularly attractive as particles warm intelligence is well suited for real-time applications, where low complexity and fast convergenceis of absolute importance. While an optimal maximum likelihood (ML) detection using an exhaustive search method is prohibitively complex, PSO-assisted MIMO detection algorithms give near-optimal bit error rate (BER) performance with a significant reduction in ML complexity.

SOFTWARE AND HARDWARE

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

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 you 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 noninteractive 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, which together represent 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 application-specific solutions called toolboxes. Very important to most users of MATLAB, toolboxes allow you 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.

The MATLAB System

+

The MATLAB system consists of five main parts:

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 complete large and complex application programs.

The MATLAB working environment.

This is the set of tools and facilities that you work with as the MATLAB user or programmer. It includes facilities for managing the variables in your workspace and importing and exporting data. It also includes tools for developing, managing, debugging, and profiling M-files, MATLAB's applications.

Handle Graphics.

This is the MATLAB graphics system. It includes high-level commands for two-dimensional and three-dimensional data visualization, image processing, animation, and presentation graphics. It also includes low-level commands 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 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 Application Program Interface (API).

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

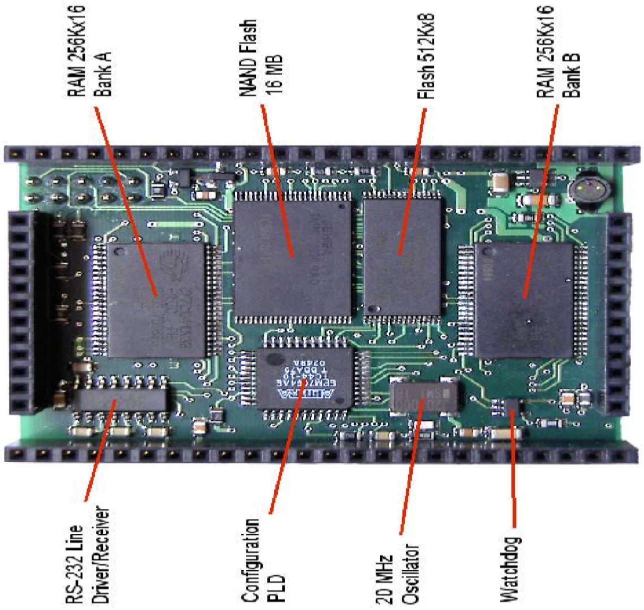
.

APPLICATIONS:

- Communications Systems
- Computational Biology
- Computational Finance
- Control Systems
- Digital Signal Processing
- Embedded Systems
- FPGA Design and Codesign
- Image and Video Processing

FPGA

The Spartan-3 family of Field-Programmable Gate Arrays is specifically designed to meet the needs of high volume, cost-sensitive consumer electronic applications. The eight-member family offers densities ranging from 50,000 to 5,000,000 system gates, as shown in Table 1. The Spartan-3 family builds on the success of the earlier Spartan-IIE family by increasing the amount of logic resources, the capacity of internal RAM, the total number of I/Os, and the overall level of performance as well as by improving clock management functions. Numerous enhancements derive from the Virtex®-II platform technology. These Spartan-3 FPGA enhancements, combined with advanced process technology, deliver more functionality and bandwidth per dollar than was previously possible, setting new standards in the programmable logic industry. Because of their exceptionally low cost, Spartan-3 FPGAs are ideally suited to a wide range of consumer electronics applications, including broadband access, home networking, display/projection and digital television equipment. The Spartan-3 family is a superior alternative to mask programmed ASICs. FPGAs avoid the high initial cost, the lengthy development cycles, and the inherent inflexibility of conventional ASICs. Also, FPGA programmability permits design upgrades in the field with no hardware replacement necessary, an impossibility with ASICs.



Features

• Low-cost, high-performance logic solution for high-volume, consumer-oriented applications

• Densities up to 74,880 logic cells

• Select IO™ interface signaling

• Up to 633 I/O pins

• 622+ Mb/s data transfer rate per I/O

• 18 single-ended signal standards

• 8 differential I/O standards including LVDS, RSDS

• Termination by Digitally Controlled Impedance

• Signal swing ranging from 1.14V to 3.465V

• Double Data Rate (DDR) support

• DDR, DDR2 SDRAM support up to 333 Mb/s

• Logic resources

• Abundant logic cells with shift register capability

• Wide, fast multiplexers

• Fast look-ahead carry logic

• Dedicated 18 x 18 multipliers

• JTAG logic compatible with IEEE 1149.1/1532
• Select RAM™ hierarchical memory
• Up to 1,872 Kbits of total block RAM
• Up to 520 Kbits of total distributed RAM
• Digital Clock Manager (up to four DCMs)
• Clock skew elimination
• Frequency synthesis
• High resolution phase shifting
• Eight global clock lines and abundant routing
• Fully supported by Xilinx ISE® and Web PACK™ software
development systems
• Micro-Blaze™ and Pico-Blaze™ processor, PCI®,
PCI Express® PIPE Endpoint, and other IP cores

Architectural Overview

The Spartan-3 family architecture consists of five fundamental programmable functional elements:

• Configurable Logic Blocks (CLBs) contain RAM-based Look-Up Tables (LUTs) to implement logic and storage
elements that can be used as flip-flops or latches. CLBs can be programmed to perform a wide variety of logical
functions as well as to store data.

• Input /Output Blocks (IOBs) control the flow of data between the I/O pins and the internal logic of the device. Each IOB
supports bidirectional data flow plus 3-state operation. Twenty-six different signal standards, including eight
high-performance differential standards, are available as shown in Double Data-Rate (DDR) registers are
included. The Digitally Controlled Impedance (DCI) feature provides automatic on-chip terminations, simplifying board
designs.

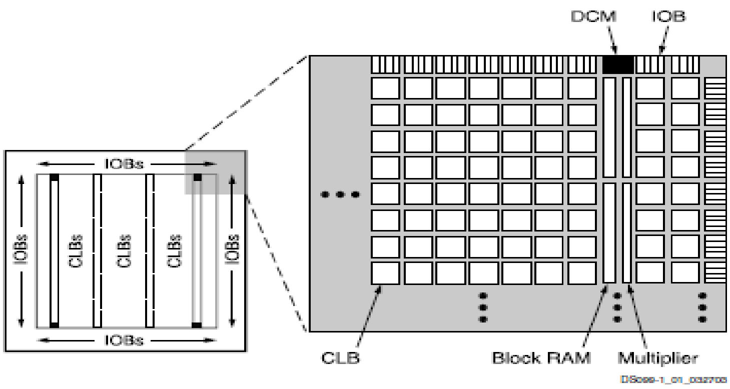
• Block RAM provides data storage in the form of 18-Kbit dual-port blocks.

• Multiplier blocks accept two 18-bit binary numbers as inputs and calculate the product.

• Digital Clock Manager (DCM) blocks provide self-calibrating, fully digital solutions for distributing, delaying, multiplying,
dividing, and phase shifting clock signals.

These elements are organized as shown in Figure 1. A ring of IOBs surrounds a regular array of CLBs. The XC3S50 has a
single column of block RAM embedded in the array. Those devices ranging from the XC3S200 to the XC3S2000 have two
columns of block RAM. The XC3S4000 and XC3S5000 devices have four RAM columns. Each column is made up of several
18-Kbit RAM blocks; each block is associated with a dedicated multiplier. The DCMs are positioned at the ends of the outer
block RAM columns.

The Spartan-3 family features a rich network of traces and switches that interconnect all five functional elements,
transmitting signals among them. Each functional element has an associated switch matrix that permits multiple connections to the routing.



Spartan-3 Family Architecture

Configuration

Spartan-3 FPGAs are programmed by loading configuration data into robust reprogrammable static CMOS configuration
latches (CCLs) that collectively control all functional elements and routing resources. Before powering on the FPGA,
configuration data is stored externally in a PROM or some other nonvolatile medium either on or off the board. After applying
power, the configuration data is written to the FPGA using any of five different modes: Master Parallel, Slave Parallel, Master
Serial, Slave Serial, and Boundary Scan (JTAG). The Master and Slave Parallel modes use an 8-bit-wide SelectMAP port.

The recommended memory for storing the configuration data is the low-cost Xilinx Platform Flash PROM family, which
includes the XCF00S PROMs for serial configuration and the higher density XCF00P PROMs for parallel or serial
configuration.

6:Working

As shown in Figure 1.5, the input serial binary data will be processed by a data scrambler ¯rst and then channel coding is applied to the input data to improve the BER (bit error rate) performance of the system. The encoded data stream is further interleaved to reduce the burst symbol error rate. Dependent on the channel condition like fading, di®eren base modulation modes such as BPSK (binary phase shift keying), QPSK (quadrature phase shift keying) and QAM are adaptively used

to boost the data rate. The modulation mode can be changed even during the transmission of data frames. The resulting complex numbers are grouped into column vectors which have the same number of elements as the FFT size, *N*. For simplicity

of presentation and ease of understanding, we choose to use matrix and vector to describe the

$$\vec{S}(m) = \begin{bmatrix} S(mN) \\ \vdots \\ S(mN+N-1) \end{bmatrix}_{N\times 1},$$

mathematical model. Let *S*(*m*) represent the *m*-th OFDM symbol in the frequency domain i.e. where *m* is the index of OFDM symbols. We assume that the complex-valued elements *fS*(*mN*)*;* *S*(*mN* + 1),……,*S*(*mN* + *N* ¡ 1)*g* of ~*S*(*m*) are zero mean and uncorrelated random variables whose sample space is the signal constellation of the base modulation (BPSK, QPSK and QAM). To achieve the same average power for all mappings, a normalization factor *K*MOD [37] is multiplied to each elements of *S*(*m*) such that the average power of the mappings is normalized to unity. To obtain the time domain samples, as shown by the IDFT block in Figure 1.5, an IFFT (inverse fast Fourier transform) operation is represented by a matrix multiplication.

Let F*N* be the *N*-point DFT matrix whose (*p; q*)-th elements is

$$e^{-j\frac{2\pi}{N}(p-1)(q-1)}.$$

The resulting time domain samples ~$s(m)$ can be described by

$$\vec{s}(m) = \begin{bmatrix} s(mN) \\ \vdots \\ s(mN+N-1) \end{bmatrix}_{N\times 1} = \left(\frac{1}{N}\right)\mathbf{F}_N^H \vec{S}(m).$$

Compared to the costly and complicated modulation and multiplexing of conventional FDM systems, OFDM systems easily implement them by using FFT in baseband processing. To combat the multipath delay spread in wireless channels, the time-domain samples $s(m)$ is cyclically extended by copying the last Ng samples and pasting them to the front, as shown in figure.

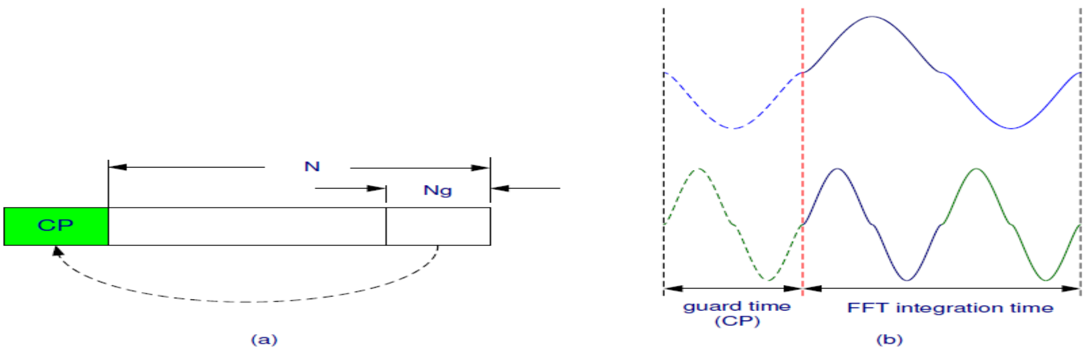


Figure (a) Concept of CP.

(b) OFDM symbol with cyclic extension.

Let $u(m)$ denote the cyclically extended OFDM symbol as

$$\vec{u}(m) = \begin{bmatrix} u(mN_{tot}) \\ \vdots \\ u(mN_{tot}+N_{tot}-1) \end{bmatrix} = \begin{bmatrix} CP \\ \vec{s}(m) \end{bmatrix}_{N_{tot}\times 1}$$

where $N_{tot} = N + N_g$ is the length of *u*(*m*). In the form of matrix, the CP insertion can be readily expressed as a matrix product of *s*(*m*) and an $N_{tot} \times N$ matrix A*CP* . By straight computation, it holds that

$$\vec{u}(m) = \mathbf{A}_{CP}\vec{s}(m).$$

where

$$\mathbf{A}_{CP} = \begin{bmatrix} \mathbf{0} & \mathbf{I}_{N_g} \\ \mathbf{I}_{N-N_g} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{N_g} \end{bmatrix}_{(N+N_g)\times N}$$

One of the challenges from the harsh wireless channels is the multipath delay spread.If the delay spread is relatively large compared to the symbol duration, then a delayed copy of a Sprevious symbol will overlap the current one which implies severe ISI. To eliminate the ISI almost completely, a CP is introduced for each OFDM symbol and the length of CP, *Ng* must be chosen longer than the experienced delay spread, *L*, i.e.,
Ng >*L*. In addition, CP is capable of maintaining the orthogonality among subcarriers which implies zero ICI. It is because the OFDM symbol is cyclically extended and this ensures that the delayed replicas of the OFDM symbol always have an integer number of cycles within the FFT interval, as long as the delay is smaller than the CP.It is clearly illustrated in Figure 1.6(b). No matter where the FFT window starts,provided that it is within the CP, there will be always one or two complete cycles within FFT integration time for the symbol on top and at below respectively. In IEEE 802.11a standard [37], *Ng* is at least 16. The obtained OFDM symbol (including the CP) *u*(*m*), as shown in Figure 1.5, must be converted to the analogue domain by an DAC (digital-to-analog converter) and then up-converted for RF transmission since it is currently not practical to generate the OFDM symbol directly at RF rates. To remain in the discrete-time domain, the OFDM symbol could be up-sampled and added to a discrete carrier frequency. This carrier could be an IF (intermediate frequency) whose sample rate is handled by current technology. It could then be converted to analog and increased to the ¯nal transmit frequency using analog frequency conversion methods. Alternatively, the OFDM modulation could be immediately converted to analog and directly increased to the desired RF transmit frequency.

Either way has its advantages and disadvantages. Cost, power consumption and complexity must be taken into consideration for the selected technique.

The RF signal is transmitted over the air. For the wireless channel, it is assumed in this thesis as a quasi-static frequency-selective Rayleigh fading channel [71]. It indicates that the channel remains constant during the transmission of one OFDM

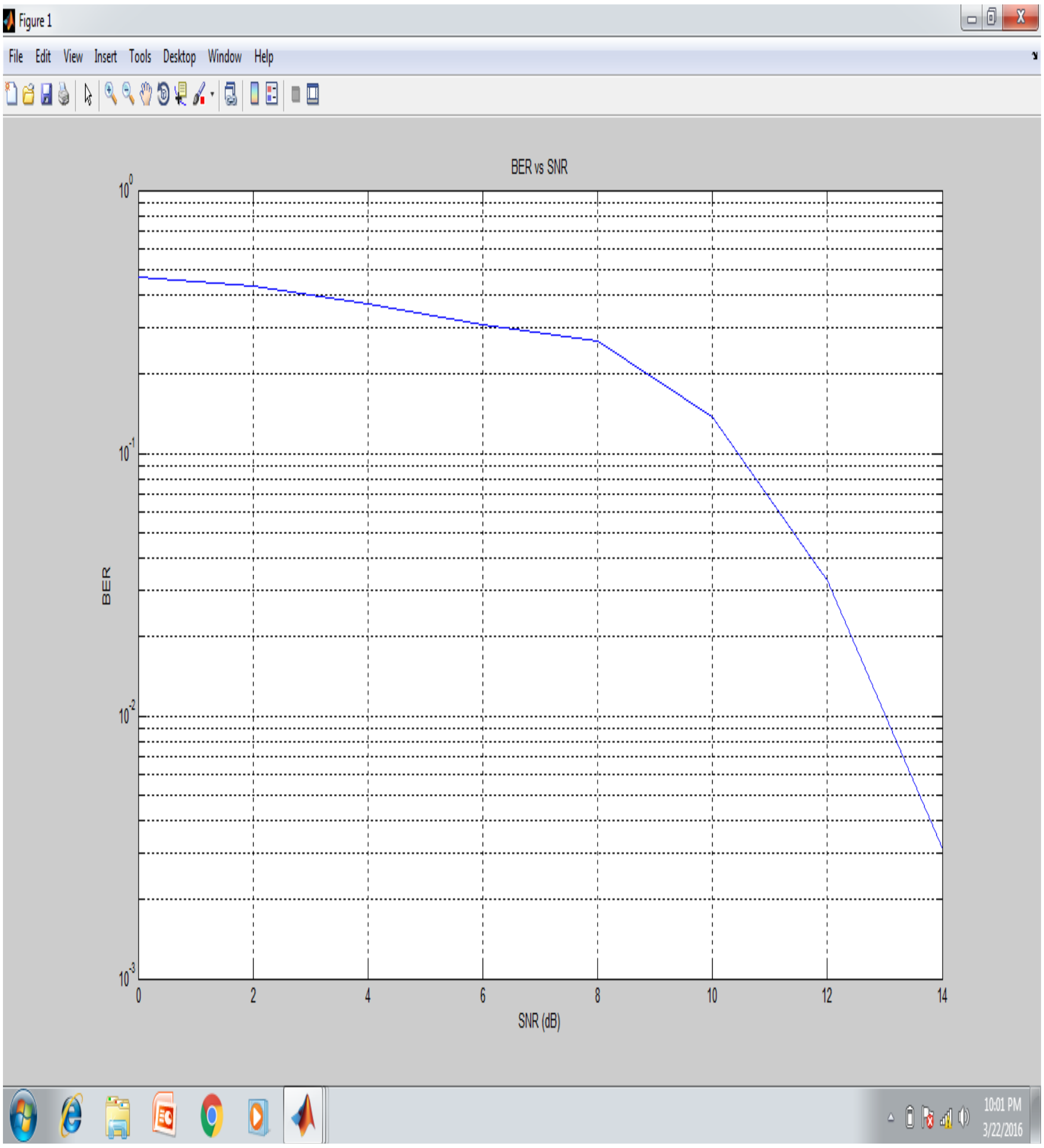
symbol. Suppose that the multipath channel can be modeled by a discrete-time baseband equivalent (L¡1)th-order FIR (Finite impulse response) filter with filter taps $\{h_0, h_1, \ldots, h_l, \ldots, h_{L-1}\}$ It is further assumed that the channel impulse response, i.e. ,the equivalent FIR filter taps, are independent zero mean complex Gaussian random variables with variance of $\frac{1}{2}\mathcal{P}_l$ per dimension. The ensemble of $\{\mathcal{P}_0, \ldots, \mathcal{P}_l, \ldots, \mathcal{P}_{L-1}\}$ is the PDP (power delay profile) of the channel and usually the total power of the PDP is normalized to be 1 as the unit average channel attenuation. Denote the CIR (channel impulse response) vector *hm* as

$$\vec{h}_m = \begin{bmatrix} h_{0,m} \\ \vdots \\ h_{L-1,m} \end{bmatrix}_{L\times 1}$$

where the subscript *m* is kept to imply that the channel may vary from one OFDM symbol to the next one. Then the complex baseband equivalent received signal can be represented by a discrete-time convolution as

$$r(mN_{tot} + n) = \sum_{l=0}^{L-1} h_{l,m} u(mN_{tot} + n - l) + v(mN_{tot} + n)$$

8:result and discussion



10:FUTURE SCOPE

OFDM and MIMO will serve as the physical layer of two key technologies for future mobile communication systems: UMTS LTE and WiMax. OFDM and MIMO will help in the evolving standards for LTE and WiMax. LTE is the 4G evolution of cellular systems, while WiMax is a technology that is expected to deliver last mile wireless broadband access.

- Background to LTE: HSPA Release 7
- LTE Design Goals
- Frame Structure
- Downlink MIMO Modes
- Physical Resource Block
- LTE Oplink: Single Carrier (SC) - FDMA
- WiMax: Overview of MAC layer, MAC protocol data units
- Frame Structure
- Ranging
- Quality of Service Classification
- ARQ
- Scalable OFDMA
- Adaptive Modulation and Coding
- OFDMA Channelization: PUSC, FUSC, AMC, Matlab example
- Multiple Antenna Technology in WiMax

12:CONCLUSION

MIMO-OFDM is a powerful modulation technique used for high data rate, and is able to eliminate ISI. It is computationally efficient due to the use of FFT techniques to implement modulation and demodulation functions. The performance of MIMO-OFDM is tested for two digital modulation techniques namely QPSK and QAM using MATLAB/SIMULINK toolbox. MIMO-OFDM system with QPSK scheme is suitable for low capacity, short distance application. While the OFDM with higher M-ary modulation scheme is used for large capacity, long distance application at the cost of slight increase in Eb/No. The comparison of QPSK and QAM indicates that, BER is large in QPSK as compared to QAM and it generally depends on applications. In this project, we implemented and optimized the 4x4 MIMO-OFDM transmitter and receiver for different embedded platform**.** We reduced the bit error rate of MIMO-OFDM from 10^{0} to 10^{-4} using 4x4 array. We conclude that QAM modulated MIMO - OFDM system achieves better BER results than QPSK and other modulated MIMO - OFDM systems for the same bandwidth efficiency

13:REFERENCE

- Gordan stuber, john barry, steve mclaughlin:”broadband MIMO-OFDM wireless communications”, proceedings of IEEE ,VOL.92, NO.2, feb 2004.

- A.omri, R. bouallegue:“New transmission scheme for MIMO-OFDM system”, International journal of next generation network vol.3,no.1,march 2011.

- Helmut dolcskei, ETH zurich: “MIMO-OFDM wireless system” , IEEE wireless communication aug 2006.

- Johanna kaetonen , markku juntti , joseph cavallaro: “performance complexity comparison of reciever for a LTE MIMO-OFDM system” , IEEE transaction on signal processing , vol.58, no.6, june 2010.

APPENDIX

Program:

```
% OFDM Code

% No. of Carriers: 64
% coding used: Convolutional coding
% Single frame size: 96 bits
% Total no. of Frames: 100
% Modulation: 16-QAM
% No. of Pilots: 4
% Cylic Extension: 25%(16)

close all
clear all
clc



%%
% Generating and coding data
t_data=randint(9600,1)';
x=1;
si=1; %for BER rows
%%
for d=1:100;
data=t_data(x:x+95);
x=x+96;
k=3;
n=6;
s1=size(data,2);  % Size of input matrix
j=s1/k;

%%
% Convolutionally encoding data
constlen=7;
codegen = [171 133];    % Polynomial
trellis = poly2trellis(constlen, codegen);
codedata = convenc(data, trellis);



%%
%Interleaving coded data

s2=size(codedata,2);
j=s2/4;
matrix=reshape(codedata,j,4);

intlvddata = matintrlv(matrix',2,2)'; % Interleave.
intlvddata=intlvddata';
```

```
%%
% Binary to decimal conversion

dec=bi2de(intlvddata','left-msb');


%%
%16-QAM Modulation

M=16;
y = qammod(dec,M);
% scatterplot(y);


%%
% Pilot insertion

lendata=length(y);
pilt=3+3j;
nofpits=4;

k=1;

for i=(1:13:52)

    pilt_data1(i)=pilt;

    for j=(i+1:i+12);
        pilt_data1(j)=y(k);
        k=k+1;
    end
end

pilt_data1=pilt_data1';   % size of pilt_data =52
pilt_data(1:52)=pilt_data1(1:52);    % upsizing to 64
pilt_data(13:64)=pilt_data1(1:52);   % upsizing to 64

for i=1:52

    pilt_data(i+6)=pilt_data1(i);

end


%%
% IFFT

ifft_sig=ifft(pilt_data',64);
```

```
%%
% Adding Cyclic Extension

cext_data=zeros(80,1);
cext_data(1:16)=ifft_sig(49:64);
for i=1:64

    cext_data(i+16)=ifft_sig(i);

end


%%
% Channel

 % SNR

 o=1;
for snr=0:2:50

ofdm_sig=awgn(cext_data,snr,'measured'); % Adding white Gaussian Noise
% figure;
% index=1:80;
% plot(index,cext_data,'b',index,ofdm_sig,'r'); %plot both signals
% legend('Original Signal to be Transmitted','Signal with AWGN');


%%
%                  RECEIVER
%%
%Removing Cyclic Extension

for i=1:64

    rxed_sig(i)=ofdm_sig(i+16);

end


%%
% FFT

ff_sig=fft(rxed_sig,64);

%%
% Pilot Synch%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


for i=1:52
```

```
    synched_sig1(i)=ff_sig(i+6);

end

k=1;

for i=(1:13:52)

    for j=(i+1:i+12);
        synched_sig(k)=synched_sig1(j);
        k=k+1;
    end
end

% scatterplot(synched_sig)


%%
% Demodulation
dem_data= qamdemod(synched_sig,16);


%%
% Decimal to binary conversion

bin=de2bi(dem_data','left-msb');
bin=bin';


%%
% De-Interleaving


deintlvddata = matdeintrlv(bin,2,2); % De-Interleave
deintlvddata=deintlvddata';
deintlvddata=deintlvddata(:)';




%%
%Decoding data
n=6;
k=3;
decodedata =vitdec(deintlvddata,trellis,5,'trunc','hard');  % decoding datausing veterbi decoder
rxed_data=decodedata;

%%
% Calculating BER
rxed_data=rxed_data(:)';
errors=0;
```

```
c=xor(data,rxed_data);
errors=nnz(c);

% for i=1:length(data)
%
%
%     if rxed_data(i)~=data(i);
%         errors=errors+1;
%
%     end
% end



BER(si,o)=errors/length(data);
o=o+1;

 end % SNR loop ends here
 si=si+1;
end % main data loop

%%
% Time averaging for optimum results

for col=1:25;        %%%change if SNR loop Changed
    ber(1,col)=0;
for row=1:100;


        ber(1,col)=ber(1,col)+BER(row,col);
    end
end
ber=ber./100;

%%
figure
i=0:2:48;
semilogy(i,ber);
title('BER vs SNR');
ylabel('BER');
xlabel('SNR (dB)');
grid on
```