# CHANNEL ESTIMATION IN GPRS BASED COMMUNICATION SYSTEM USING BAYESIAN DEMODULATION

Chinni Anil Kumar

Dileep Kumar Appana

Nagappan Palaniappan Nagappan

This thesis is presented as part of Degree of

Master of Science in Electrical Engineering

Blekinge Tekniska Högskola

October 2009

**Blekinge Tekniska Högskola**

**School of Engineering**

**Department of Telecommunication**

**Supervisor: Magnus G Nilsson**

**Examiner: Magnus G Nilsson**

# ABSTRACT

With the increase use of portable devices such as Personal Digital Assistants (PDA), laptops, voice and data integrated cell phones and many more, there is a need of wireless communication method using air as the medium to transmit and receive information between terminals. Radio waves propagate from transmitting antenna and travel through free space undergoing reflections, diffractions and scattering. They are greatly affected by ground terrain, the atmosphere and the objects in their path like buildings, bridges, hills etc. Nowadays, the existence of a direct line of sight path between the transmitter and the receiver is unlikely. These multiple phenomena are responsible for most of the characteristic features like the quality of the received signal.

In the above case propagation is mainly due to reflection and scattering from the buildings and by diffraction. So, in practice the transmitted signal arrives at the receiver via several paths with different time delays creating a multi path situation at the receiver, these multipath waves with randomly distributed amplitudes and phases combine to give a resultant signal that fluctuates in time and space. This phenomenon of random fluctuations in received signal level is termed as fading. The existing demodulation techniques like FM, AM will determine the signal from the received signal based on the mean distance method, which cannot provide the desired level of BER, which fails in proper estimation under high fading and high Doppler-Shift effect.

**SOLUTION:**

This project provides the implementation of an enhancement to the demodulation technique using Bayesian approach for the physical layer simulation of a General Packet Radio System (GPRS) considering variable Rician fading and variable Doppler-Shift effect for an AWGN channel. The system performance is evaluated based on Bit Error Rate (BER) and Signal to Noise Ratio (SNR) for the realized GPRS system. Matlab platform is used for the implementation, analysis of the proposed system with for functional verification in terms of BER and SNR. We have showed the comparative difference between the theoretical calculation of QPSK signal and to the values obtained by our program. The values show difference up to 0.4 db for a 1000 bit random vector. Moreover, we also compared with QAM demodulation technique in MATLAB code to show difference up to 1.4 db for a 1000 bit vector. These results signify better performance of the system as it has saved bandwidth.

# ACKNOWLEDGEMENT

First, we thank our respected Supervisor, **Mr. MAGNUS G NILSSON**, who guided us in proper way to complete this project in time. Thanks to Blekinge Institute of Technology Sweden for providing us such a good platform for completing the course and project in a satisfactory manners. Finally, our gratitude to our all friends and teachers, those who gave help and assistance needed. God bless all of them.

Chinni Anil Kumar

Dileep Kumar Appana

Nagappan Palaniappan Nagappan

October 2009.

# CONTENTS

# LIST OF FIGURES

# LIST OF ABBREVIATIONS

| | |
|---|---|
| AWGN | Additive White Gaussian Noise |
| BER | Bit Error Rate |
| DAC | Digital to Analog Converter |
| DPSK | Differential Phase-Shift Keying |
| FIR | Finite Impulse Response |
| FSK | Frequency-Shift Keying |
| GMRS | General Mobile Radio Service |
| GPRS | General Packet Radio system |
| GPS | Global Positioning System |
| ISI | Inter Symbol Interference |
| LMR | Land Mobile Radio |
| LOS | Line of Sight |
| N-LOS | Non-Line of Sight |
| PDF | Probability Density Function |
| PL | Path Loss |
| PSK | Phase-Shift Keying |
| QPSK | Quadrature Phase-Shift Keying |
| SNR | Signal to Noise Ratio |

# 1 INTRODUCTION TO WIRELESS COMMUNICATIONS

## 1.1: INTRODUCTION:

Since 1800's with the birth of Wireless Communications, its advancements in technology had lead to rapid growth, merging the boundaries for ease of information exchange. Bell and Marconi laid the stones for the present information age and frontier of the future of telecommunications.

These services can be performed via Radio Frequencies for Non Line of Sight (N-LOS) or Microwave Communication for Line of Sight (LOS) [1] in long distance communications where practical implementation of wires is not possible or cost effective, and also through Infrared Communications for short range communications.

The common examples of wireless devices are cell phones, WLAN (Wireless Local Area Networks), professional LMR (Land Mobile Radio) used by business entities, GMRS (General Mobile Radio Service), Amateur Radio Service, GPS (Global Positioning System), satellite televisions etc.

In early stages, communication systems used to be analog, with huge base stations and high power transmitters to overcome large obstacles, weather conditions and other electromagnetic interferences. But the emerging technologies along with increased data transfer rates in limited band width, urge for power saving and decrease in size of the equipment gave new challenges. Today, most of the systems are digitized, comprise binary digits where the bits can be obtained from digitizing the analog signal. Thus, the two fundamental aspects which make the designers a challenge are increase of data rates and improvement of performance while incurring little or no increase in bandwidth or power. The wireless channel is by its nature, random and unpredictable, and in general channel error rates are poorer over a wireless channel than over a wired channel. The fundamental factors that affect the communication in a radio channel are discussed as proceeded.

## 1.2: PROBLEM STATEMENT:

The quality of a radio communication channel can be understood by its error performance [2]. So, network designers who work for deploying of a new channel need to understand and estimate the channel in order to provide an improved communication by modifying the modulation and coding schemes and the network architectures. It is also desirable to have an accurate and thoroughly reproducible error model, which would allow network designers to evaluate a protocol or algorithm and its variations in controlled and repeatable way. However, [3] the problems like physical positioning of antenna (LOS & NLOS), effects on radio propagation, losses, diffractions around corners , scattering caused by the objects and multiple reflections (ISI) lead to errors in a wireless medium, and the task of modelling the error performance of wireless channels is complex. The radio link is highly variable over short distances due to the statistical distribution of Path Loss (PL) and the physical properties of propagation environments, thereby making it difficult to generalize the results of error performance analysis.

This project deals with the implementation of an enhancement to the existing demodulation technique (QPSK) using Bayesian Demodulation for the physical layer simulation of a General Packet Radio system (GPRS) considering variable Rician fading and variable Doppler effect for an Additive White Gaussian Noise (AWGN) channel. Here we are estimating the channel by calculating the Signal to Noise ratio (SNR) and the bit error rate.

## 1.3: OUTLINE OF THESIS:

To increase the readability and understand ability of the document we divide the whole document into 5 chapters. In each chapter again there are subtitles to discuss many more details about the topic.

Chapter 1 contains some introduction about wireless communication. In this we explain about various advancements in wireless technology since 1800. In the Problem Statement quotes the main achievement with our thesis in terms of performance parameters like Bit error rate.

Chapter 2 describes about some available techniques and also those used in the trans-receiving system individually. The QPSK modulator and the Bayesian demodulator are discussed in details as they are main composite of our trans-receiving system.

Chapter 3 contains design analysis of our proposed architecture with neat block diagram and explanation of signal variations at different stages. It gives in-detail description how the bits are varying.

Chapter 4 contains result analysis. This contains the results of various phases with the simulation outputs and the output results of the original data and demodulated data.

Chapter 5 contains conclusion and future scope of our work.

Appendix includes the program coding that is used to produce the simulation output in our thesis.

# 2 WIRELESS COMMUNICATION

## 2.1: INTRODUCTION:

Wireless Telecommunications can be broadly classified into two main categories: Mobile Communications and Fixed Wireless Communications. Both two categories have their own unique requirements based on the technical and customer needs. The present mobile communication trend requires mobility and wireless communications.

The goal of mobility is to provide the service whenever required. Another important characteristic about mobile communication is *roaming* which is the ability to provide service to mobile phone users while they were roaming out of their home network. Unlike this, mobility is not required for a fixed wireless user [4]. Instead, this user needs cost effective telecommunication method from fixed locations e.g., Walkie-Talkie. Since it provides fast and reliable communications, this wireless communication has become the auxiliary mode of providing service to a user, where satellite communication is the only alternative.

## 2.2: FACTORS AFFECTING PROPAGATION OF RADIO TRANSMISSION:

2.2.1: RANGE: The range of a system is defined as the average maximum distance between the two nodes under usual operating conditions (i.e. diameter of the cell-radio of neighbourhoods). When we are measuring the range, transmitted power and sensitivity values are considerable factors of performance [4].

2.2.2: TRANSMITTED POWER: The transmitted power is the strength of the transmissions, which is measured in Watts. System with high transmitting power consumes more power and requires more power supply [4].

2.2.3: SENSITIVITY: The measurement of the weakest signal, which is discovered reliably on the channel, is known as the sensitivity of the signal. The ability to read the bits from the antenna with low error probability illustrates the sensitivity. This indicates the performance of the receiver [4].

2.2.4: ATTENUATION & SIGNAL TO NOISE RATIO (SNR): The attenuation is the reduction in signal strength when transmitted from sender to receiver. In the medium of air, the attenuation is proportional to the square of the distance covered by the signal and if the composition of signal path between these two nodes is known, the attenuation can be calculated [4]. SNR is the “ratio of power in a signal to the power contained in the noise which is present at a particular point in the transmission” [1] [4]. This ratio can be evaluated at the receiver end in order to eliminate the unwanted noise from the signal.

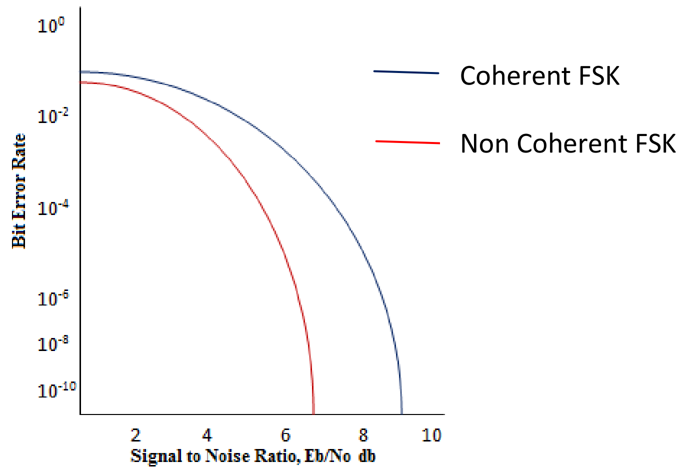


Fig. 2.1: SNR Vs BER.

The above figure explains the different FSK modulation characteristics. From the figure, we can observe that, the maximum level of Bit Error Rate (BER) allowed is attained by some minimum level SNR that should be maintained.

## 2.3: MODULATION:

Modulation is mainly used to transform the digital bits in the form of radio waves, which can be transmitted through a wireless medium. There are many ways in which modulation can be done [5]. Here we are using QPSK modulation, and explanation follows why it is advantageous than other modulations in this project.

2.3.1: AMPLITUDE, FREQUENCY AND PHASE MODULATION: Modulation done by varying amplitude, frequency and phase of the carrier signal in accordance with the input signal respectively. Frequency and phase modulations are more immune to external noise and amplitude modulation is easier to demodulate.

2.3.2: DIGITAL FREQUENCY MODULATION: The digital frequency modulation is varying the carrier frequency in accordance with the magnitude of the modulating input. The output from the modulator is a sine wave, whose frequency depends on the transmitted signal. Demodulation can be done at the receiver's end by passing the signals through different filters and translating the resultant back into logical levels. This form of modulating the frequency is called as Frequency-Shift Keying (FSK).

The main advantages of FSK are robust, and have no ambiguities because one tone is higher than other. FSK is used in all single –channel and radiotelegraph systems. The disadvantage is not bandwidth efficient.

2.3.3: DIGITAL PHASE MODULATION: Digital phase modulation also known as Phase-Shift Keying (PSK), is mostly similar to frequency modulation. It changes the phase of the waveform to digital data. This modulated wave can be generated by the digital data obtained between the two signals of same frequency. If it is further multiplied by a sine wave, a cosine wave along with a frequency independent wave is generated. The amplitude depends on the cosine value of the phase shift. The efficient use of bandwidth and the form of modulation for many telecommunication applications are the main advances of PSK. PSK is especially used for data transmission.

2.3.4: MULTIPLE PHASE-SHIFT KEYING (MPSK): MPSK deals with more than two phases, usually four (0, +90, -90, and 180 degrees) or eight (0, +45, -45, +90, -90, +135, -135, and 180 degrees). If there are four phases (m = 4), it is called Quadrature phase-shift keying or quaternary phase-shift keying (QPSK), and each phase shift represents two bits. If there are eight phases (m = 8), it is called as octal phase-shift keying (OPSK), and each phase shift represents three bits. In MPSK, data can be transmitted at a faster rate, relative to the number of phase changes per unit time, than is the case in BPSK.

2.3.5: QUADRATURE PHASE-SHIFT KEYING (QPSK) MODULATION: In QPSK, carrier undergoes four changes in phase and each symbol is represented by 2 binary bits of data. A special modulation scheme has been engaged to enable the carrier to transmit 2 bits, in order to double the bit rate of the carrier. The process of QPSK modulation and demodulation is carried out as given below, According to Euler's relations;

$$\sin \omega t = \frac{e^{j\omega t} - e^{-j\omega t}}{2j} \qquad \cos \omega t = \frac{e^{j\omega t} + e^{-j\omega t}}{2} \qquad \ldots\ldots\ldots \quad (2.1)$$

Multiplying sine wave to the sine signal gives,

$$\sin^2 \omega t = \frac{e^{j\omega t} - e^{-j\omega t}}{2j} * \frac{e^{j\omega t} - e^{-j\omega t}}{2j} = \frac{e^{2j\omega t} - 2e^{0} + e^{-2j\omega t}}{-4} = \frac{1}{2} - \frac{1}{2}\cos 2\omega t \quad \ldots\ldots\ldots \quad (2.2)$$

From Equation 2.2, The product of two sine waves, results in an output frequency of $\frac{1}{2}\cos 2\omega t$, where one sine wave is the input signal and another one is the local oscillator in the receiver side mixer, $\frac{1}{2}\cos 2\omega t$ is the double that of the input frequency at half the amplitude which is super imposed on a DC offset of the input amplitude. Similarly, the product of sinωt by cos ωt is

$$\sin \omega t * \cos \omega t = \frac{e^{2j\omega t} - e^{-2j\omega t}}{4j} = \sin 2\omega t \qquad \ldots\ldots\ldots \quad (2.3)$$

From this the output frequency of sin2ωt is the double that of the input without DC offset. Now let we consider making the product of sin2ωt by any phase-shifted sine wave (sinωt+ϕ) yields a "demodulated" waveform with an output frequency double that of the input frequency, whose DC offset varies according to the phase shift $\phi$.

To prove this,

$$\sin\omega t * \sin(\omega t+\phi) = \frac{e^{j\omega t}-e^{-j\omega t}}{2j} * \frac{e^{j(\omega t+\phi)}-e^{-j(\omega t+\phi)}}{2j}$$

$$= \frac{e^{j(2\omega t+\phi)}-e^{j(\omega t-\omega t-\phi)}-e^{j(\omega t+\phi-\omega t)}+e^{-j(2\omega t+\phi)}}{-4}$$

$$= \frac{\cos(2\omega t+\phi)}{-2} - \frac{e^{j\phi}+e^{-j\phi}}{-4}$$

$$= \frac{\cos(2\omega t+\phi)}{-2} + \frac{\cos\phi}{2} \quad = \frac{\cos\phi}{2} - \frac{\cos(2\omega t+\phi)}{2} \qquad \text{.........} \quad (2.4)$$

From Equation 2.4, we can understand that the supposition of the phase shift on a carrier can be demodulated into a different output voltage by multiplying the carrier with a sine-wave local oscillator and filtering out the high-frequency term.

But the quadrants of the phase shift are limited for two. The phase shift of π/2 cannot be distinguished from a phase shift of -π/2. Therefore, to get the accurately decode phase shifts in all four quadrants, the input signal should to be multiplied by both sinusoidal and co sinusoidal waveforms, the high frequency is filtered out, and the data is reconstructed. This gives [6].

$$\cos\omega t * \sin(\omega t+\phi) = \frac{e^{j\omega t}+e^{-j\omega t}}{2} * \frac{e^{j(\omega t+\phi)}-e^{-j(\omega t+\phi)}}{2j}$$

$$= \frac{e^{j(2\omega t+\phi)}-e^{j(-\phi)}+e^{j(\phi)}-e^{-j(2\omega t+\phi)}}{4j} \qquad \text{..........} \quad (2.5)$$

$$= \frac{\sin(2\omega t+\phi)}{2} + \frac{e^{j\phi}-e^{-j\phi}}{4j} \quad = \frac{\sin(2\omega t+\phi)}{2} + \frac{\sin\phi}{2}$$

In QPSK, four different phase angles are used, thereby creating four symbols: pi/4, 3pi/4, 5pi/4, and 7pi/4, where the amplitude of the transmitted wave remains constant. The carriers with these four phase angles and constant amplitude are expressed as phase or vectors as shown in Fig 2.2. The symbols-to-bit pattern mapping is such that as the phase angle changes from the neighbouring angle.

For example, the two-bit pattern for ø=45 is 11 and that for ø=135 is 01. Comparing 11 and 01, note that there is one bit difference between the two patterns. This type of symbol-to-bit pattern mapping is referred to as the Gray coding.

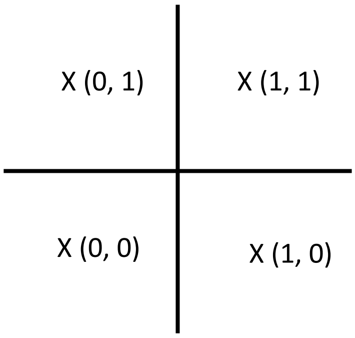


Fig. 2.2: Four Quadrants for the QPSK Modulation Technique.

## 2.4: UP-SAMPLER:

There are many reasons for up sampling the data and the main reason is to increase the sampling rate. The other reasons include to match the rate of output hardware, to make it easier to combine with other higher sampling rate, to vary the playback rate and to follow with some cheaper interpolation for the variable rate The operation of Up-sampling [7] by factor L describes the insertion of L-1 zeros between every sample of the input signal. Up-sampling is carried out for the rate conversion from one rate to another desired rate. This is denoted by "↑L" in block diagrams, as in Fig 2.4.Up sampling can be expressed in the time domain as;

$$y[n] = \begin{cases} x\left[\frac{n}{L}\right] if \frac{n}{L} \in Z \\ 0 otherwise \end{cases} \qquad \text{..........} \quad (2.6)$$

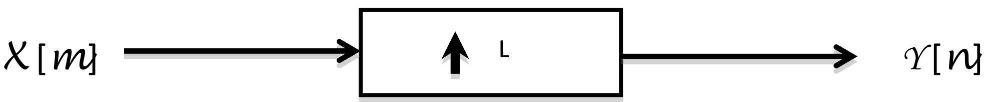


Fig. 2.3: Block Diagram for an Up-Sampler.

Fig. 2.5 shows the effect of sampling the rate on the sampling operation. From the fig, it can be seen that as the rate of sampling rate increases the data rate increases.

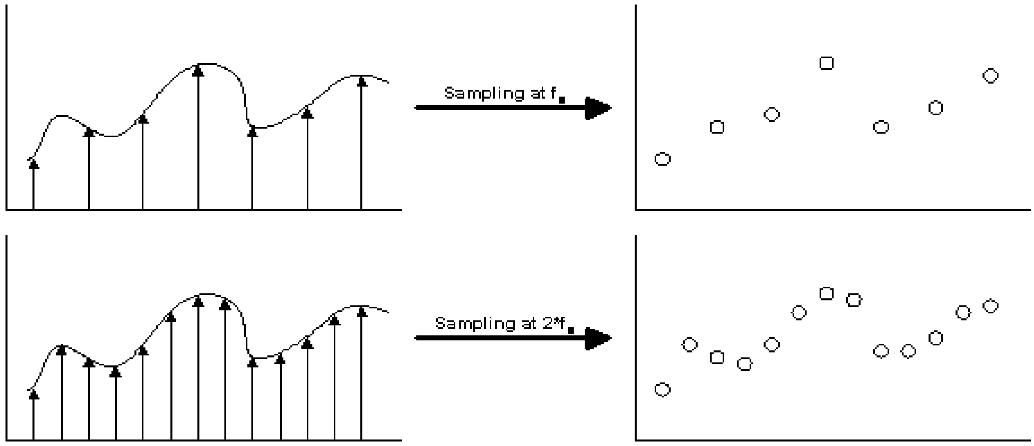


Fig. 2.4: Sampling of data at different sampling rates.

2.4: EFFECTS ON FREQUENCY RESPONSE: Up-sampling the message before transmission increases the bandwidth usage, instead of occupying the entire bandwidth up to fs/2. The up-sampled message occupies relatively less Bandwidth (almost 1/8$^{th}$ of the whole Bandwidth). This uses an analysis filter and a reconstruction filter at the sender and receiver's end respectively. A sharp filter is designed with the same sampling rate to cutoff the frequency exactly at that point. That filter may lose some audible spectrum regardless of the difficulty level and the expense.

## 2.5: PULSE SHAPING FILTERS:

In data transmission system, pulses are transmitted from sender and detected in the receiver side. At receiving end, the signals are sampled at the certain pulse interval, which tends to achieve the maximum probability of an accurate binary decision. Due this process at the optimal sampling point, the pulses won't interfere with each other which make the fundamental shapes of the pulses. There are two characteristics that make non-interference. One is that in the sampling point, where pulse shape exhibits zero crossing in all pulse intervals other than its own pulse. Otherwise in the decision making process, the errors will be introduced due to the residual effect of other pulse, and second is the pulse shape. Hence, the exterior parts of the pulse interval lead to sudden decline of amplitude, which makes the proper pulse shape.

Any real System will have timing jitter i.e. the actual sampling point of the each and every pulse in the receiver side will not be perfect. Though the pulse shape exhibits a zero crossing at the perfect sampling point of different pulse interval, the timing jitter causes the sampling tends to move to miss the zero crossing point. Because of this, the error will be introduce in the decision making process. Hence, the exterior parts of the pulse interval the pulse will suddenly decline. This reduces the possibilities of the timing jitter to introduce the errors while sampling the adjacent pulses. Other than this, limiting the pulse band width gives better performance [7] [8].

2.5.1: RECTANGULAR PULSE: The main characteristic of rectangular pulse is to eliminate the residual effect of those pulses which cause errors into decision making process. But it is not suitable for modern transmission systems because of its spectrum .The pulse spectrum of rectangular pulse is given by sin(p*x*)/p*x*(sinc), where its bandwidth is prolonged to infinity(which means unrestrained frequency response). To overcome this problem, there is a need of pulse shaping filter, though rectangular pulse is not fair for band-limited data transmission. Hence we use raised cosine pulse to fix this problem, where Raised Cosine Pulse is widely used in modern data transmission system.

2.5.2: RAISED COSINE PULSE: Similar to the rectangular pulse, the raised cosine pulse also will take the shape of sine pulse. When the shapes are excellent they can be used for sharing the pulse-shaping load between the transmission and the receiver side.

Fig.2.5 shows the characteristics of ideal raised cosine filter frequency response, which consists of unity gain at low frequencies in the initial stage, a raised cosine function in the middle, and total attenuation at high frequencies. Middle frequencies width can be defined by roll off factor constant Alpha, (0<Alpha<=1) [8] [9].

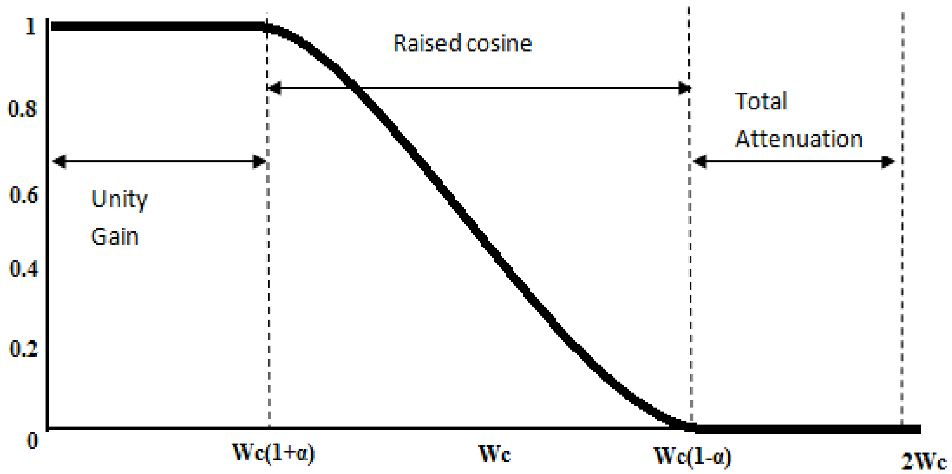


Fig. 2.5: Ideal Raised Cosine Frequency Response [1].

After the pulse shape is defined, we have to decide the category of filter to be used from the two basic category of digital filter.

In General, pulse-shaping filters will engage Finite Impulse Response (FIR) design because the non- zero impulse bares the output samples when the impulse propagates down the delay stage. The fundamental operation in the filters is multiple / adds / delay operators, where the delay stages are clocked due to the operation that occur each time. It is a convolution operation, which is more effective. The convolution operator can be expressed as:

$$y(n) = x(n) * h(n) \qquad \text{..........} \qquad (2.7)$$

Where h(n) is impulse response and x(n) is the input samples.

In the above expression, '*' (asterisk) is the convolution operator. As per Fourier transform (or Z-transform in terms of Digital filters) filtering is identical with convolution operation, which is expressed above.

The main advantage of using a pulse-shaping filter concept is that rectangular pulses can be used as the input to the filter. The convolution of a rectangular pulse with a raised cosine impulse response will results output as a raised cosine pulse as shown in Fig. 2.6.

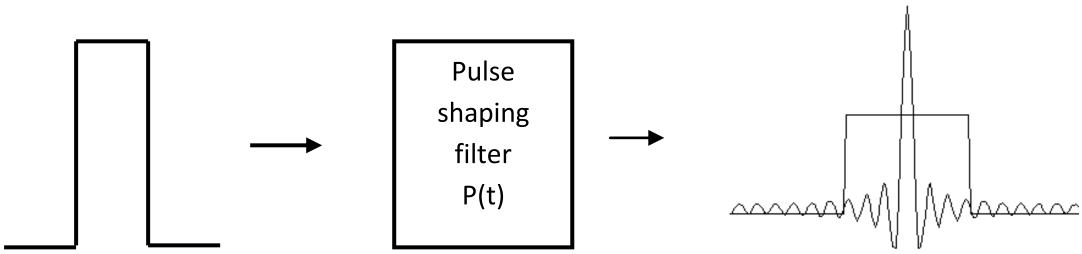


Fig. 2.6: Conversion of an impulse to a raised cosine pulse by filtering.

The input of the filter is 0's and 1's, which climbs to occupy the maximum bit width of the filter's input word size and the output of the raised cosine pulse with all the time and frequency domain advantages. For this process the Digital to Analog Converter (DAC) is required in the output side of the filter, which will convert the digital samples into analog waveform [8] [9] [6].

## 2.6: WIRELESS CHANNEL:

The wireless channel is defined as the medium through which the data gets transmitted from the transmitter to the receiver system. There are various factors affecting these transmitted data. The main parameters that are affected are the wireless noise and the channel fading which results in losing the transmitted data.

2.6.1: NOISE IN COMMUNICATION SYSTEMS: Noise plays a crucial role in communication systems. In theory, it determines the theoretical capacity of the channel. In practice it determines the number of errors occurring in a digital communication. The types of noise are Additive white Gaussian, Random noise. The power spectrum density of sample functions of a wide sense stationary random process at all frequencies is called white noise. However it is unrealizable, as can be seen by the fact that it possesses infinite average power.

In communication system the Additive White Gaussian Noise is the additive of wideband and white noise with a constant spectral density with a Gaussian distribution. Fading, frequency selectivity, interference, dispersion doesn't account for this type of phenomenon. It simply used in analysing the system behaviour and in many satellite and space communication links.

2.6.2: RADIO PROPAGATION MECHANISM: Radio waves are most generally like light, goes through the glass which either reflects or diffracts by the mirrors and stops by most obstacles except the one which are transparent or reflector to radio than to light. In case of a real environment, there are loads of surfaces reflecting the radio waves (walls, ceilings), being semi-transparent (humans, ceilings) or opaque. This may limit the estimation of the range of the system. This states the signal received at a node is coming from various directions with various strengths and it's only the combination of all reflections. This phenomenon is called Multipath.

Multipath is mostly good, since the reflections increase the strength of the signal. The unique feature is that the range is very difficult to evaluate and experience fading. Based upon the reflections and the propagation speed, the signals arriving at the receiver end reaches with an "echo". The following figure depicts the process of multipath and the delays observed during the transmission [10].

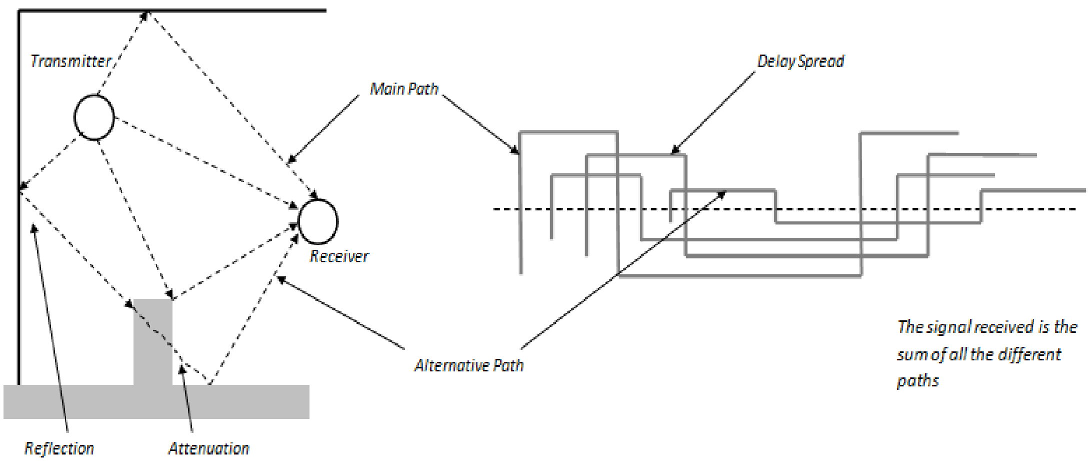


Fig. 2.7: Multipath and Delay Spread [10].

2.6.3: THE CHANNEL FADING: Radio propagation is very complex. The parameters which are affecting the radio propagation in a real-world scenario are reflection, diffraction, and scattering. When electromagnetic radiation reflects off objects or diffracts around objects, it can travel from the transmitter to the receiver over multiple paths, which cause to multipath propagation. This can result in fluctuations in the received signal's amplitude, phase, and angle of arrival, giving rise to multipath fading. The main reason for this phenomenon is due to many reasons like affect of the ground terrain, the atmospheric ducting, ionosphere reflection and the objects in their path like buildings, bridges, hills, trees, etc,.

Once the signal experiences fading, the envelope and the phase fluctuate over time. The fading effects can put down the performance in consistent modulations. The system undergoing analyses assumes the phase effects are adjusted perfectly in consistent demodulations. For inconsistent detection, the information about the phase is not required and also the phase variation due to the fading does not affect performance. Hence the performance depends only on the information obtained [4] [11].

2.6.3.1: RAYLEIGH FADING: In Multipath propagation, the worst case propagation is sometimes when the line-of-sight signal is not reached and reflected, diffracted and refracted waves only reach receiver. Because of these varying path lengths of the received signal, the output signal is random in nature and the instantaneously received power becomes a random variable. In mobile radio channels, this Rayleigh distribution usually describes the statistical time varying nature of the received envelope of a flat fading signal.

2.6.3.2: RICIAN FADING: When there is a dominant stationary signal (such as line-of-sight, ground reflected wave), then the distribution is Rician. In this case there are random Multipath components superimposed on that stationary dominant signal. Thus the main difference between Rayleigh and Rician fading is the presence of the direct line-of-sight signal presence. The Rician degenerates to Rayleigh distribution when line-of-sight signal fade away.

C = speed of light =3x10$^8$, fdoppler = known

If x = 10^(-k/20), where k is Rician fading factor

y = ((1+x)^-1)

the Rayleigh envelope can be written as

$$\sqrt{\tfrac{1}{x+1}}\left(1+\left(2\cos\beta\cos(w_n t)+\sqrt{(2\ \cos(w_d t)\cos\alpha)}\right)+j\left(2\sin\beta\cos(w_n t)+j\sqrt{(2\ \cos(w_d t)\sin\alpha}\ \right)\right)$$

........... (2.8)

Where as $\beta = \frac{n\pi}{N+1}$

Then Rician envelope can be written as y *(1+rayleighenvelop *x) ............ (2.9)

## 2.7: MATCHED FILTER SECTION:

The signal acquired in a communication system requires the convergence of several processing algorithms before reaching at the receiver's end. To receive the useful data from the wireless channel the receiver uses a matched filter [5][12]. A matched filter is a filter whose frequency response is designed to exactly match the frequency spectrum of the input signal. The operation of matched filters is the same as correlating a signal with a copy of itself. These filters are used as signal processors in communication receivers to calculate the correlation between the transmitted signal and the received signal. With the received signal x(t) used as the filter input, the resulting filter output can be defined by the convolution integral.

$$y_j(t)=\int_{-\infty}^{\infty} x(\tau)h_j(t-\tau)d\tau$$ ........... (2.10)

Suppose we now set the impulse response

$$h_j(t) = \phi_j(T - t) \qquad \text{............} \quad (2.11)$$

Then the resulting filter output is in the interval $0 \le t \le T$,

$$y_j(T) = \int_0^T x(T)\phi_j(\tau)d\tau \qquad \text{...........} \quad (2.12)$$

Thus the detector part of the optimum receiver may also be implemented as in figure [5].

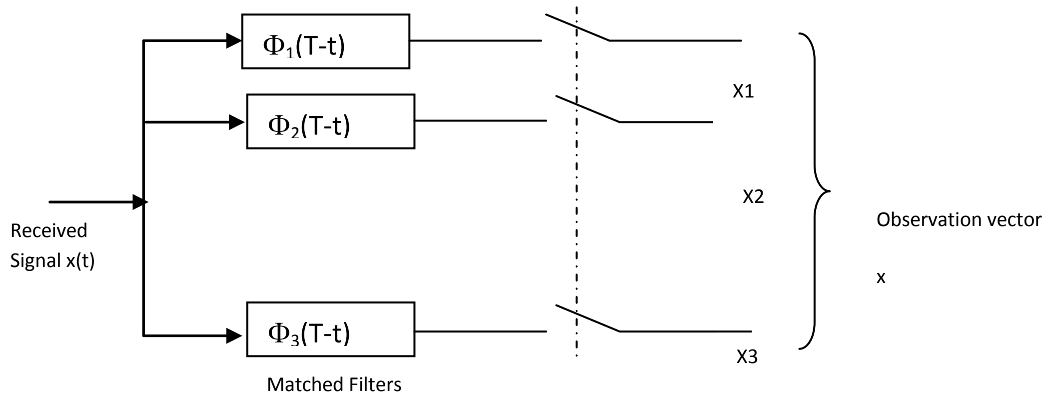


Fig. 2.8: Matched Filter Receiver [2].

A filter whose impulse response is a time reversed and delayed version of some signal is said to be matched to $\Phi_j(t)$.

There are two advantages in these kind of processing, they are

1. Typical Pulse shaping which have low pass response. The received signal will be filtered in the receiver end by using the match filter and also the data signal in the frequency is passed even when the frequencies attenuate. The noise spectrum can be limited by the match filter which will pass into subsequent stage in the receiver end.
2. Correlation of the received signal is obtained with the match filter by transmitting the pulse shape for the period of T. By using convolution operation, the signal x (t) is passed through the filter $\Phi(t)$.

Due to noise the received signal is distorted, though the output of the matched filter is closer to the ideal value when there is no noise. Since AWGN is zero-mean, the integration effectively removes the average sum of noise out.

This makes matched filtering to provide strong signal in the receiver end to work while compared with direct sampling the receiver signal. The processing gain of matched filtering is especially apparent. The output of the matched filter should be sampled at the period of time T, which generates the sample with highest SNR. Sampling the matched filter's output at some time T + Δ, (where Δ represents a receiver timing offset) will significantly reduce the effective SNR [12].

2.7.1: Navigating steps for generating filter coefficients in Matlab:

- Enter ***fdatool*** in command window.
- A window opens up and set values as low pass in response type, FIR in design method, Gaussian in Window. Specify the order that needed.

Go to file → export → select 'export to' as workspace, 'export as' as coefficients, variable names as file name you want and click export.

## 2.8: DOWN SAMPLER:

The down sampling operation is performed by M factor, which describes the process as keeping every $M^{th}$ sample and discarding the rest. This process is mention as ' ↓M' in the block diagram as in Fig 3.6

The general form of down sampling is written as

$$y[n] = x[nM] \qquad \text{..........} \quad (2.13)$$

In the z domain,

$$Y(z) = \sum_n (y[n] z^{-n}) \qquad \text{..........} \quad (2.14)$$

by substitute the value of y[n], we get

$$= \sum_n (x[nM] z^{-n}) = \sum_m \left( x[m] \left( \tfrac{1}{M} \sum_{p=0}^{M-1} \left(e^{i\frac{2\pi}{M}pm}\right)\right) z^{\frac{-m}{M}} \right) \qquad \text{..........} \quad (2.15)$$

$$\chi[m] \longrightarrow \boxed{\downarrow M} \longrightarrow Y[n]$$

Fig. 2.9: Block Diagram for M down sampling.

where

$$\sum_{p=0}^{M-1} \left( e^{i\left(\frac{2\Pi}{M}\right)pm} \right) = \begin{cases} 1 \text{ if m is multiple of M} \\ 0 \text{ otherwise} \end{cases}$$

$$Y(z) = \left(\frac{1}{M}\right) \sum_{p=0}^{M-1} \left( \Sigma \left(x[m] \left( e^{-\left(i\left(\frac{2\Pi}{M}\right)pz\left(\frac{1}{M}\right)\right)} \right)^{-m} \right) \right) \qquad \text{...........} \, . \quad (2.16)$$

$$Y(z) = \left(\frac{1}{M}\right) \sum_{p=0}^{M-1} \left( X\left( e^{-\left(i\left(\frac{2\Pi}{M}\right)pz\left(\frac{1}{M}\right)\right)} \right) \right) \qquad \text{...........} \quad (2.17)$$

This is conversion of Z domain into Frequency domain. The down sampling expands each 2- periodic repetition of X(ei) by a factor of M along the axis, and reduces the gain by a factor of M. If x [m] is not band limited to, aliasing may result from spectral overlap. As shown below in Fig.3.7 [5].

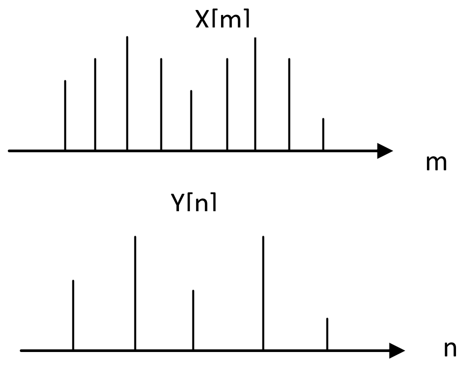


Fig.2.10: Down sampling of message bits in time domain.

## 2.9: DEMODULATOR:

The purpose of communication is to deliver a message signal from an information source in recognizable form to a user destination with the source and the user being physically separated from each other. To do this, the transmitter modifies the message signal into a form suitable for transmission over the channel. This modification is achieved by means of process known as modulation. The receiver re-creates the original message signal from a degraded version of the transmitted signal after propagation through the channel. This re-creation is accomplished by using the process known as demodulation which is the reverse of the modulation process used in the transmitter. However, owing to the unavoidable presence of noise and distortion in the received signal, we find that the receiver cannot re-create the original message signal exactly. The type of modulation influences the resulting degradation in overall system performance is influenced by the type of modulation scheme used. Various modulation techniques are used depending upon the effect of noise and the fading to the signal considered. The demodulation technique considered here follows Bayesian estimation demodulation to retrieve the received data.

2.9.1: BAYESIAN DEMODULATOR: The concepts of Bayesian detection are explored, under the assumption of perfectly known parameters of the communications channel, as the *Maximum Likelihood* Sequence Detector (ML SD) and the *Maximum A Posteriori* Symbol Detector (MAP SD) representing the optimal detection strategies for coded data sequences and un-coded data symbols, respectively. Here in our thesis concepts used are for un-coded data symbols MAP processing where as in it can be used both for joint demodulation and decoding [13][14].

In other words, the Bayesian filter family of algorithms is a recently developed family of de-convolution algorithms for digital communications [23]. The Bayesian filters are approximations to the K-lag optimum symbol-by-symbol estimator with the unique characteristic that the resulting exponential complexity is decoupled from both the filter lag and channel length.

2.9.2: GENERAL CONCEPT: In a fading channel, the optimal MLSD and MAPSD contain the pre-sequence channel estimators [13][14].

$à_{ML}$= max ln p(r|a)=max ln $E_{\Phi}$\{ p(r|a,Φ)\} .......... (2.18)

$à_k^{MAP}$=max $\boldsymbol{\Sigma}$ $E_{\Phi}$\{ p(r|a,Φ)\}, k=1,...., N. .......... (2.19)

Where $a_k$ denotes the symbol sequence, [19][20]Φ is the Gaussian impulse responses. R is number of samples, and E{}is the expectation. Gaussian impulse response simply defined as Gaussian magnitude response + linear phase → Gaussian impulse response

So, for the fading channels the equations can be modified as

$\Phi_{MAP}$=max p(Φ|r)=E{Φ|r } .......... (2.20)

àK=max p($a_k$|r, $\Phi_{MAP}$) k=1,...,N. ......... (2.21)

2.9.3: DEMODULATION OF QPSK SIGNALS: The concept of symbol-by-symbol mapping is process with soft inputs/outputs [15] [16] which use the symbol probabilities to produce an optimal soft decision [17]. Consider a sequence of received samples, the goal of symbol-by-symbol can be defined by determining all possible symbols and all possible times. Therefore symbol soft decisions are made, once these probabilities have been determined [18]. The procedure can be explained by mathematical procedure. We are assuming,

U = vector of transmitted symbols.

N= length of block data

k = hypothesis

r = no of samples

X = vector of transmitted symbol samples

Y = vector of received symbol samples

S,m,Wt are Variables.

The state transition probability is given by

$$\Pr\left\{S_{t-1} = m'; S_t = m Y\right\} = \frac{\sum_{k \in C_i(m^1, m)} \Pr\left\{U' = U(k) Y\right\}}{\sum_k \Pr\left\{U' = U(k) Y\right\}} \qquad \text{.......... (2.22)}$$

$$\Pr\left\{U' = U(k) Y\right\} = \frac{p(YU' = U(k)).\Pr\left\{U' = U(k)\right\}}{p(Y)} \qquad \text{.......... (2.23)}$$

Evaluating the numerator in above equation, assumption of any input symbols, transmitted at distinct times. Therefore, the equation can be reduced as

$$\Pr\{U' = U(k)\} = \bigcap_{t=0}^{N-1} \Pr\{U'_t = U_t(k)\} \qquad \text{..........} \quad (2.24)$$

The 1 to 1 corresponding vector can be written as

$$p(YU' = U(k)) = p(Y X' = X(k)) \qquad \text{..........} \quad (2.25)$$

from equation 2.23 we can write

$$p_{y|x}[Y \mid X(k)] = p_{y|x}[(y_{rN-1,} \ldots\ldots, y_1, y_0) \mid X(k)]$$

$$= \prod_{j=0}^{rN-1} p_{yj} y_{j-1}[y_j \mid Y_{j-1}(k)] \qquad \text{..........} \quad (2.26)$$

As we are considering this a fading channel we should consider the factor of fading also , now the paramters can be written as

$$Wt(m', m) = \prod_{j=rt}^{r(t+1)-1} p_{yj} y_{j-1}[y_j \mid Y_{j-1}(k)] \qquad \text{...........} \quad (2.27)$$

$$p_{y\ x}[Y\ X(k)] = \prod_{t=0}^{N-1} W_t(m', m) \qquad \text{...........} \quad (2.28)$$

Using these equations, 2.26 can be written as

$$p(YU' = U(k)).\Pr\{U' = U(k)\}$$

$$= \prod_{t=0}^{N-1} \Pr\{u'_t = u_t(k)\} gW_t(m', m) \qquad \text{...........} \quad (2.29)$$

$$= \prod_{t=0}^{N-1} \gamma_t(m', m) \qquad \text{...........} \quad (2.30)$$

Where as

$$\gamma_t(m', m) = \Pr\{u'_t = u_t(k)\}.W_t(m', m). \qquad \text{...........} \quad (2.31)$$

$$\Pr\{S_{t-1} = m'; S_t = m Y\} = \frac{\sum_{k \in C_t(m_t' - m_t)} \prod_{i=0}^{N-1} \gamma_i(M_i(k))}{\sum_k \prod_{i=0}^{N-1} \gamma_i(M_i(k))} \qquad \text{...........} \quad (2.32)$$

Now, computing the soft decisions from the state transition probabilities,

$$\Pr\{u_t = q|\mathbf{Y}\} = \frac{\sum_{(m',m)\in A} \sigma_t(m',m)}{\sum_m \alpha_N(m)}$$ .......... (2.33)

Removing the modulation can simply done by linear transformation

$$\text{py'x}[\text{Y'(k)}|X(\text{k})] = [\pi^{\text{N}} R]^{-1} e^{-Y'(k)^H R^{-1} Y'(k)}$$ .......... (2.34)

Now these probabilities can be written as

$$p_{yi}Y_{j-1}[y_j Y_{j-1}(k)] = \frac{r_0}{\pi\sigma^2_{j|j-1}(k)} \exp\left[\frac{y'_j - y'_{jj-1}(k)^2}{\sigma^2_{j|j-1}(k)}\right]$$ .......... (2.35)

where as

$$y'_{j|j-1}(k) = \sum_{i=1}^{j-1} a_i^{j-1} y'_{j-i}(k),\ \sigma^2_{j|j-1}(k) = r_0 Y_{j-1}$$

Using the above equation , 2.27 can be written as

$$W_t(m',m) = \prod_{j=rt}^{r(t+1)-1} \frac{r_0}{\pi\sigma^2_{j|j-1}(k)} \exp\left[-\frac{|y'_j - y'_{j|j-1}(k)|^2}{\sigma^2_{j|j-1}(k)}\right]$$

$$= \prod_{j=rt}^{r(t+1)-1} \frac{1}{\pi V_L} \exp\left[-\frac{\left|\sum_{i=0}^{L} a_i^L x^*_{j-i}(k) y_{j-i}\right|^2}{r_0 V_L}\right]$$

$$= \left(\frac{1}{\pi V_L}\right)^r \exp\left[\frac{\sum_{j=rt}^{r(t+1)-1} B_j(k)}{r_0 V_L}\right]$$ .......... (2.36)

2.9.4: DETECTION PROCESS OVERVIEW: The downloaded bits from a matched filter are collected in to a separate calling function. The stream contains all unique words, payload. The unique words are separated at first followed by Bayesian variables for calculation. The variance of the variables is calculated which can be used in manipulating of common factor which is needed for demodulating the QPSK signals. The MAP probabilities are calculated as in equation 2.36 with values obtained. As QPSK has four phases, considering the possibilities as +1,+j,-1,-j and the corresponding probabilities are calculated and fed in to a matrix form. The maxima are found according to the equation 2.20, and stored as the probability index. Depending on the probability index we will know the four possible combinations of bits as 00,01,10,11. Such a way we can decode the QPSK modulated bits using Bayesian demodulator uses the concept of symbol mapping.

2.9.5: COMPUTATION OF THE PROBABILITY OF ERROR: An advantage of the Bayesian approach for signal detection is that it makes it possible to explicitly evaluate the probability of error. It's the difference between the actual payload sent and the payload that is received.

2.9.6 QPSK DETECTION: A simple coherent type QPSK detection technique can be seen from the diagram below.

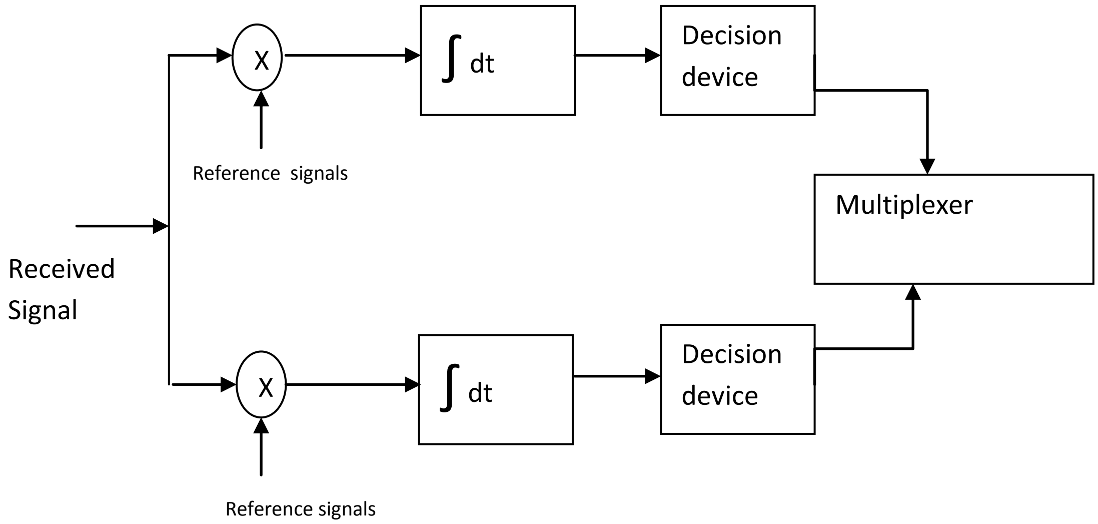


Fig.2.11: A coherent type QPSK demodulator [2].

This is a coherent detection technique. A pair of common correlators with a common input is supplied with a locally generated pair of reference signals. The correlator's output is compared to threshold of zero volts. Then a decision is made in favour of symbol 1 for the upper or inphase channel output, or else 0. Similarly it's done for the lower Quadrature output also. Finally these two binary sequences at the inphase and Quadrature channel outputs are combined in a multiplexer to reproduce the original binary sequence. The reference symbols correspond to equation 2.1.

2.9.7: QAM DETECTION: This is a non coherent detection technique. In QPSK system, inphase and Quadrature components of the modulation signal are interrelated in such a way that the envelope is constrained to remain constant. These constrain manifests itself in circular constellations for the message points. However, if this constraint is removed, and the inphase and quadarture components are there by permitted to be independent, we get the QAM modulation technique. In this modulation scheme, the carrier experiences amplitude as well as phase modulation.

Decoding of each baseband channel is accomplished at the output of the pertinent decision circuit, which is designed to compare the decision threshold. A correlator can generate the multiplying signals required. The two binary sequences so detected are then combined in the parallel to serial converter to reproduce the original binary sequence

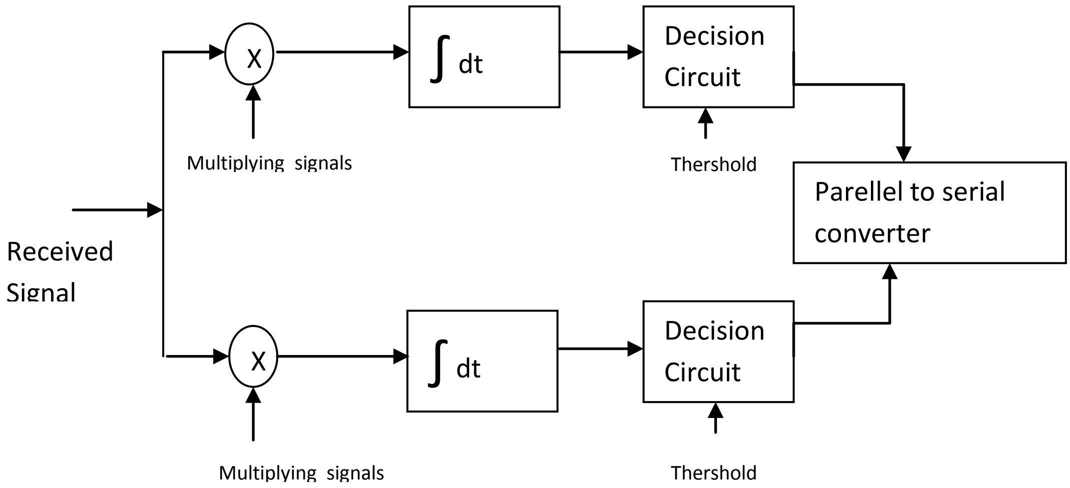


Fig. 2.12: A non-coherent type QAM demodulator.

# 3 DESIGN ANALYSIS

The design can be broadly divided into three modules- transmitter, channel and receiver. In the figure all the components in first row fall under transmitter. The elements in second row fall under channel module. The components in third row fall under receiver section.

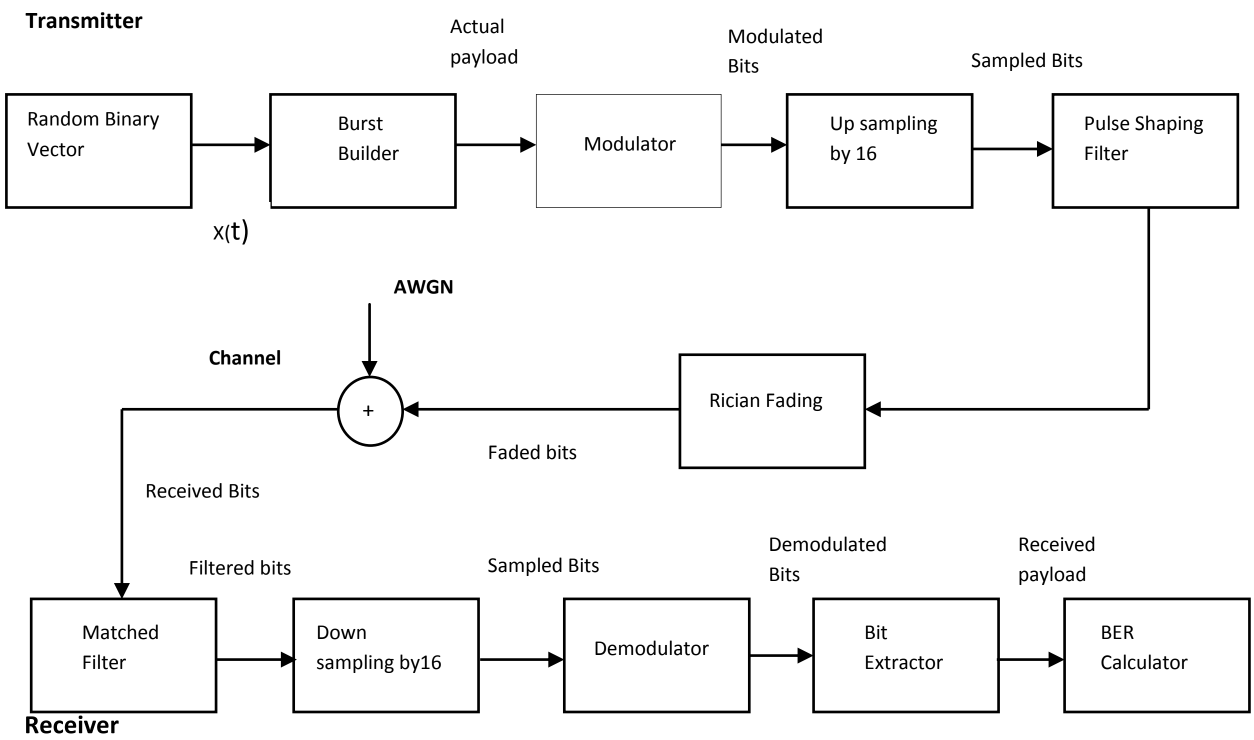


Fig 3.1 Block diagram for process of channel estimation.

Transmitter module has the following components: random bit vector, burst builder, modulator, up sampler, pulse-shaping filter. Now we consider functionality of each component in detail. The first component used is random bit vector. Random bit vector is used to generate random data the payload, guard bits and unique word. Pay load is the actual data, unique words are the information about the payload, and guard bits are the header and tail part of the structure which are used to provide tolerance of error [1] and also for coping with frequency selectivity. Each mobile receiver is assigned a unique word for secure transmission.

Burst builder section forms burst structure with the payload, unique word and guard bits. These burst structures are transmitted through QPSK modulator. With QPSK, the carrier undergoes four changes (1, j, –1, –j) in phase and can thus represent two binary bits of data per one symbol. The bits (00, 01, 10, 11) represent symbol (1, j,-j,-1) respectively. These symbols are up sampled by 16 bit up-sampler.

The up sampling process increases the sampling rate of a signal. This will be achieved by filling 15 zeros between each signal (because we have to fill M-1 zeros between each signal if we want to up sample by a factor M) [3]. These up sampled bits then passed through the pulse-shaping filter for reducing Inter Symbol Interference. The pulse-shaping filter here used is a square root raised cosine filter. Here the author used Rician flat fading channel. (Rician fading occurs when there is line of sight signal in addition to multi path components.). The received waveform is in complex form is given by [4].

$$r(t)=g(t).s(t)+n(t) \qquad \text{..........} \qquad (3.1)$$

where g(t) is the complex channel gain, n(t) is AWGN and s(t) is the transmission signal.

The signal now will pass through matched filter for detection of pulse in presence of AWGN, maximizes the signal power, and minimizes the noise power i.e. it maximizes the SNR value. The matched filter section will match the signal with the help of filter coefficients and the signal passed through the down sampler for down sampling purpose, which will be done by considering the 1st symbol, 17th symbol, and 33rd symbol and so on. Then Bayesian demodulator demodulates these symbols into the bit form by probability estimation method. Burst extractor then separate the payload information from burst format. The last section in the design is BER calculator, which is used to calculate bit error rate by comparing the input bits with receiver. This will tell about how many bit errors occurred for the given data that is passed through our network.

# 4 RESULT ANALYSIS

## 4.1: TEST CONSIDERATION:

Various testing conditions were laid. The design generates a burst from a random vector of length 1000 called payload, an unique word vector length of 120, and guard bits of vector length 6. The burst structure frame comprises of guard symbols at start and end, unique word, and payload. These are generated randomly to make a vector length of 1132 bits. This structure is fed as a input to the QPSK modulator in which two bits are made as one symbol , giving corresponding 566 symbols. The modulator uses bit to symbol mapping as 00 for 1, 01 for j, 10 for –j and 11 for –1. The sampling rate considered is 16 samples per symbol. The up sampled 566*16 bits are pulse shaped using the filter coefficients through 'flt.txt'. The system is convoluted with the filter during the pulse shaping and matched filtering of data. The signal in Rician channel is calculated using the complex convolutions, Doppler shift, Rician factor and the alpha factor. The scaling factor is calculated from the randomly generated noise and multiplied with the signal.

In the receiver section, the obtained bits are matched filtered, down sampled and demodulated using the Bayesian demodulator. The additive white Gaussian noise that is added is filtered and a received payload is extracted. The difference between the actual pay load and the received payload is found for the calculations of the Bit Error Rate. Various plots plotted to show the different modules of the transceiver.

Observations are made at a Maximum Doppler shift of 200Hz, 20Hz, and 0Hz considering the Rician fading factor (K) as 9, 12, 200 dB respectively. The graphs are shown at various levels and the simulation is run and plotted for different SNR values and compared.

4.1: Results Observations made at :

>> Doppler frequency in Hz: 200

>> Rician Factor (Kd) in dB: 9

>> N: 1000

biterrorrate = 0.1567

**Received Pay Load**

Columns 1 through 32

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

Columns 33 through 64

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

Columns 65 through 96

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

Columns 97 through 128

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

Columns 129 through 160

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

Columns 161 through 192

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

Columns 193 through 224

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

Columns 225 through 256

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

Columns 257 through 288

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

Columns 289 through 320

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

Columns 321 through 352

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

Columns 353 through 384

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

Columns 385 through 416

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

Columns 417 through 464

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

Columns 465 through 496

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

Columns 497 through 512

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

Columns 513 through 544

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

Columns 545 through 576

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

Columns 577 through 608

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

Columns 609 through 640

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

Columns 641 through 672

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

Columns 673 through 704

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

Columns 705 through 736

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

Columns 737 through 768

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

Columns 769 through 800

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

Columns 801 through 832

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

Columns 835 through 864

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

Columns 865 through 896

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

Columns 897 through 928

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

Columns 929 through 960

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

Columns 961 through 992

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

Columns 609 through 640

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

Columns 641 through 672

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

Columns 673 through 704

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

Columns 705 through 736

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

Columns 993 through 1000

1 1 0 0 1 1 1 1

**Modulated Bits Real**

Columns 1 through 32

-1 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1

Columns 33 through 64

0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 -1

Columns 65 through 96

-1 0 0 0 1 0 0 -1 0 0 0 -1 -1 0 -1 -1 0 0 -1 1 -1 0 -1 -1 1 0 -1 0 -1 0 1 1

Columns 97 through 128

-1 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1

Columns 129 through 160

-1 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1

Columns 161 through 192

0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 -1

Columns 193 through 224

-1 0 0 0 1 0 0 -1 0 0 0 -1 -1 0 -1 -1 0 0 -1 1 -1 0 -1 -1 1 0 -1 0 -1 0 1 1

Columns 225 through 256

-1 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1

Columns 257 through 288

-1 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1

Columns 289 through 320

0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 -1

Columns 321 through 352

-1 0 0 0 1 0 0 -1 0 0 0 -1 -1 0 -1 -1 0 0 -1 1 -1 0 -1 -1 1 0 -1 0 -1 0 1 1

Columns 353 through 384

-1 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1

Columns 385 through 416

-1 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1

Columns 417 through 464

0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 -1

Columns 465 through 496

-1 0 0 0 1 0 0 -1 0 0 0 -1 -1 0 -1 -1 0 0 -1 1 -1 0 -1 -1 1 0 -1 0 -1 0 1 1

Columns 497 through 512

-1 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1

Columns 513 through 544

-1 0 0 0 1 0 0 -1 0 0 0 -1 -1 0 -1 -1 0 0 -1 1 -1 0 -1 -1 1 0 -1 0 -1 0 1 1

Columns 545 through 566

-1 1 -1 -1 0 1 1 1 0 0 -1 -1 -1 1 0 -1 1 1 -1 -1 -1 0

**Modulated Real Bits Imaginary**

Columns 1 through 32

-1 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1

Columns 33 through 64

0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 -1

Columns 65 through 96

-1 0 0 0 1 0 0 -1 0 0 0 -1 -1 0 -1 -1 0 0 -1 1 -1 0 -1 -1 1 0 -1 0 -1 0 1 1

Columns 97 through 128

-1 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1

Columns 129 through 160

-1 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1

Columns 161 through 192

0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 -1

Columns 193 through 224

-1 0 0 0 1 0 0 -1 0 0 0 -1 -1 0 -1 -1 0 0 -1 1 -1 0 -1 -1 1 0 -1 0 -1 0 1 1

Columns 225 through 256

-1 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1

Columns 257 through 288

-1 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1

Columns 289 through 320

0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 -1

Columns 321 through 352

-1 0 0 0 1 0 0 -1 0 0 0 -1 -1 0 -1 -1 0 0 -1 1 -1 0 -1 -1 1 0 -1 0 -1 0 1 1

Columns 353 through 384

-1 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1

Columns 385 through 416

-1 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1

Columns 417 through 464

0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 -1

Columns 465 through 496

-1 0 0 0 1 0 0 -1 0 0 0 -1 -1 0 -1 -1 0 0 -1 1 -1 0 -1 -1 1 0 -1 0 -1 0 1 1

Columns 497 through 512

-1 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1 0 -1 0 1

Columns 513 through 544

-1 0 0 0 1 0 0 -1 0 0 0 -1 -1 0 -1 -1 0 0 -1 1 -1 0 -1 -1 1 0 -1 0 -1 0 1 1

Columns 545 through 566

-1 1 -1 -1 0 1 1 1 0 0 -1 -1 -1 1 0 -1 1 1 -1 -1 -1 0

Graphs:

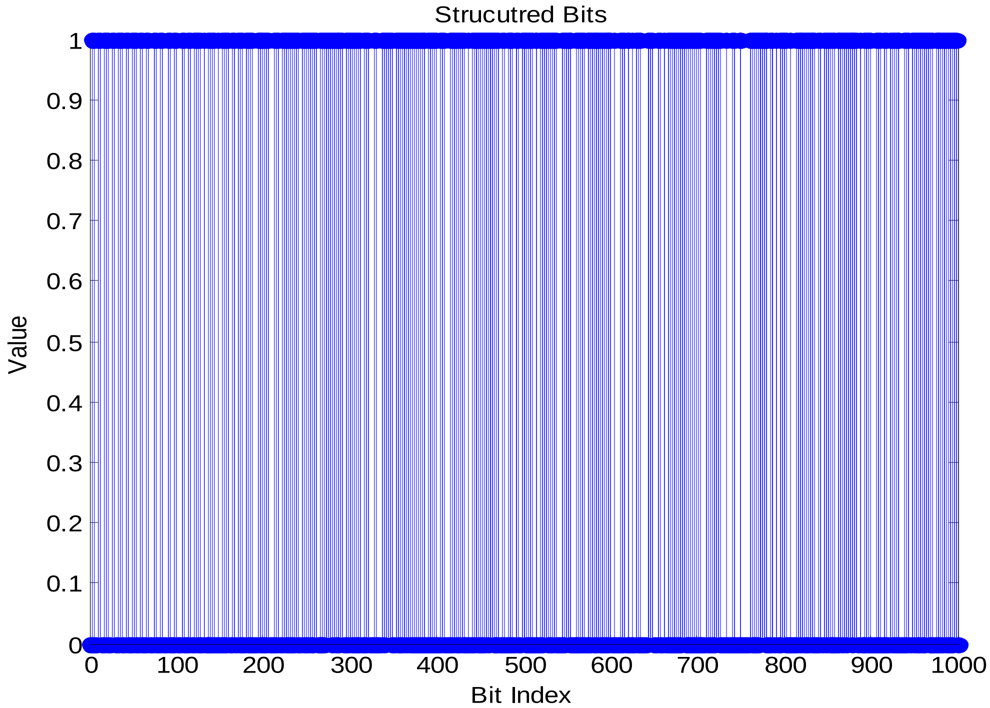


Fig. 4.1: The burst packet bits.

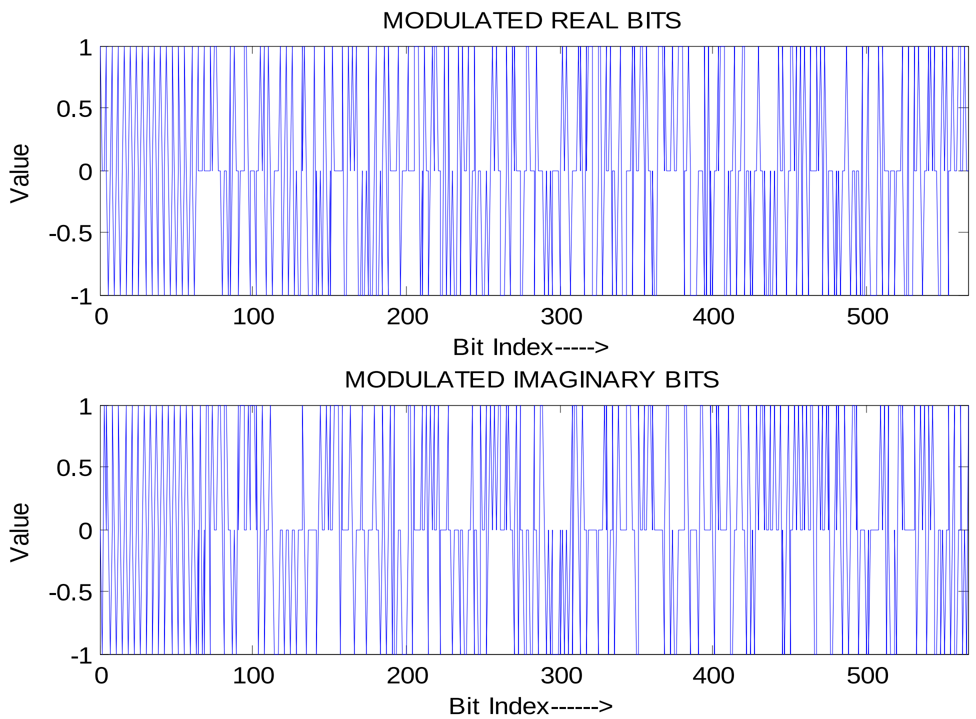


Fig. 4.2: The QPSK modulated bits.

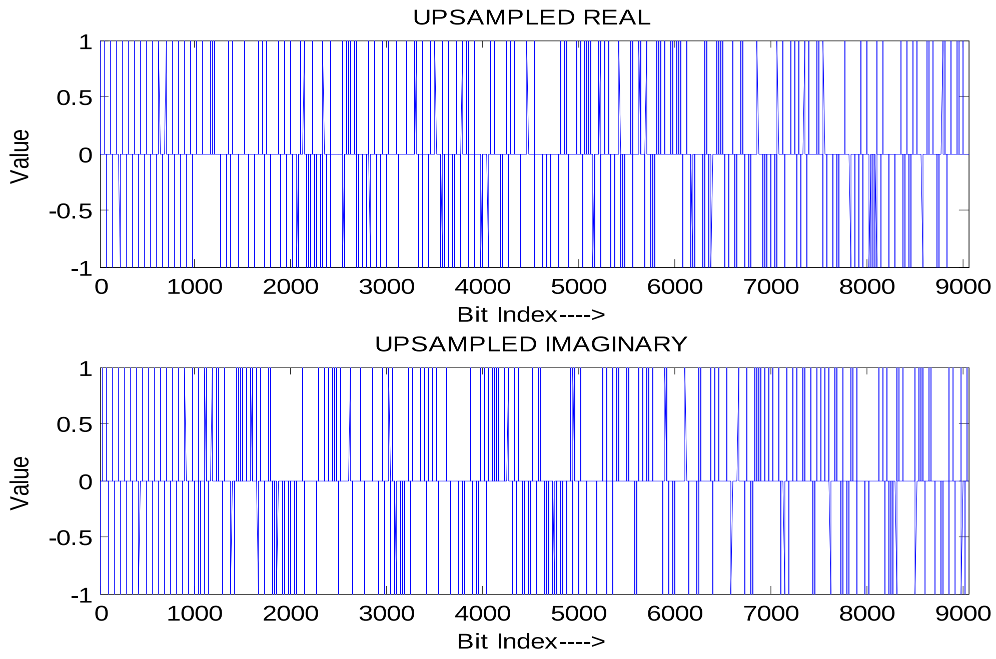


Fig. 4.3: The Up-sampled bits by 16.

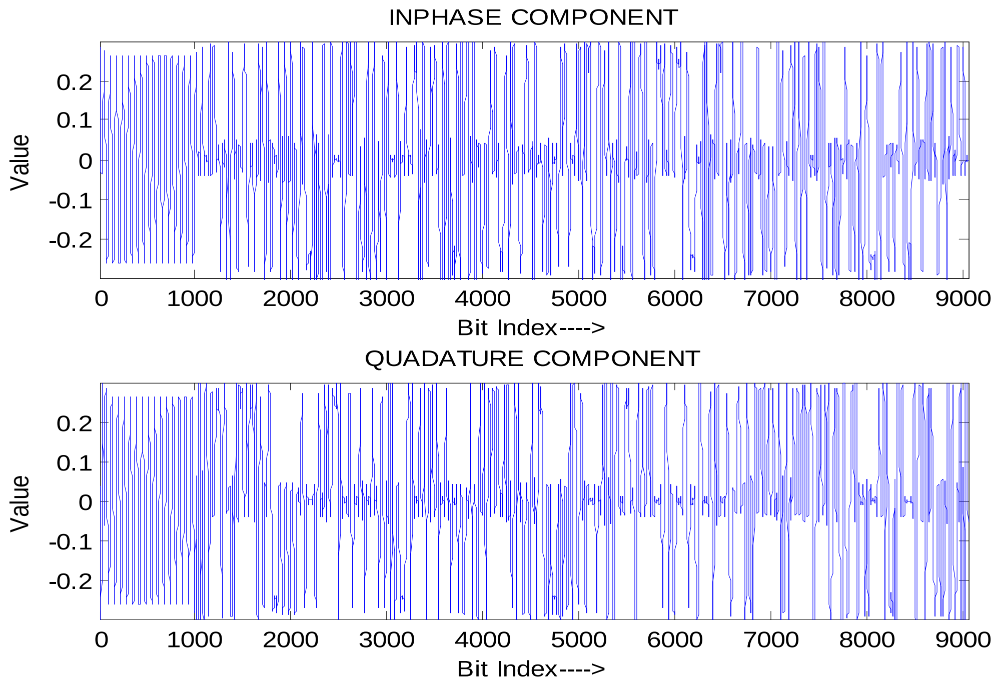


Fig. 4.4: Pulse shaped bits.

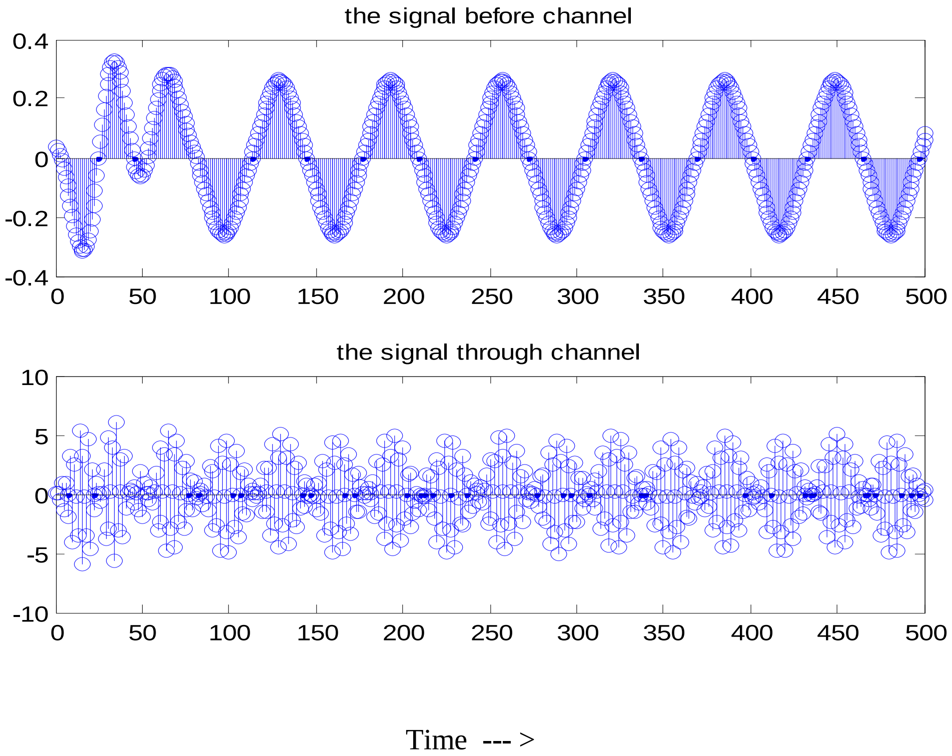


Fig. 4.5: Signal before and after transmission.

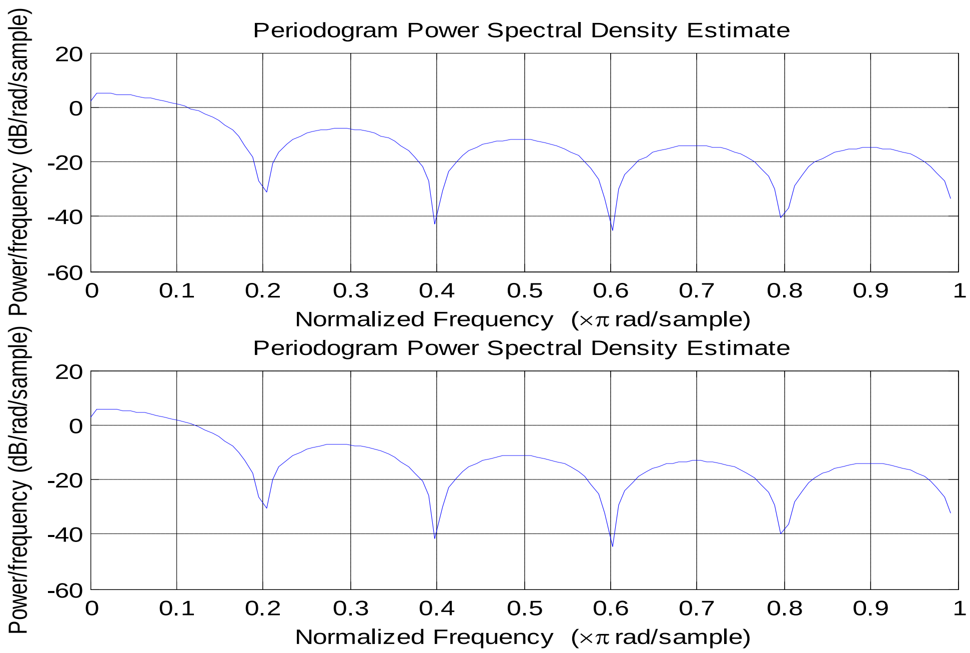


Fig. 4.6: The spectral density of the actual and the received signal.

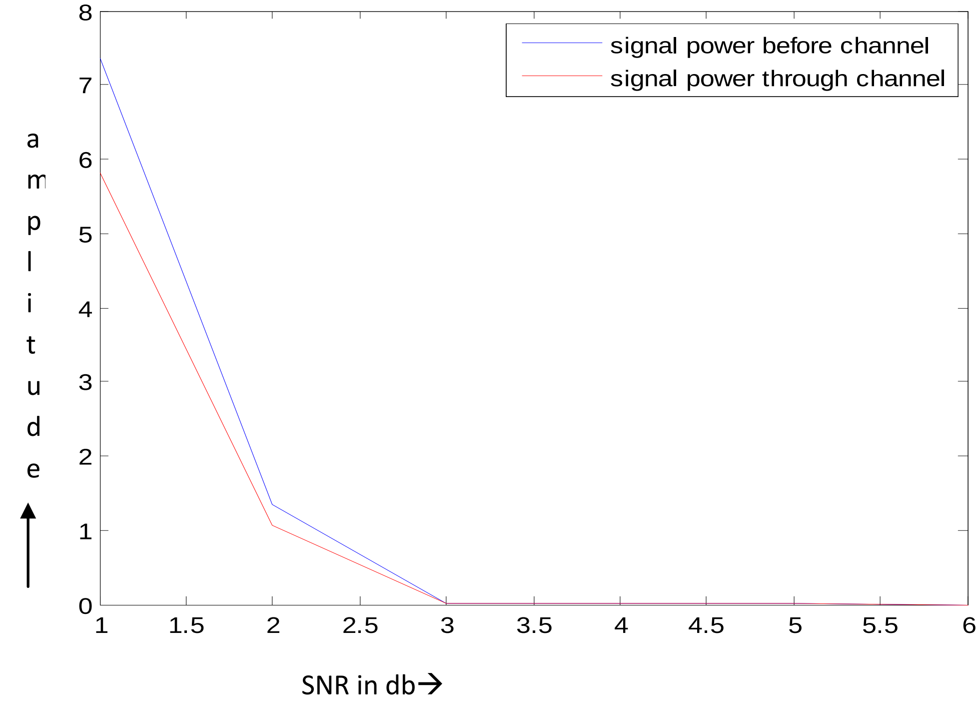


Fig. 4.7: Signal power comparison.

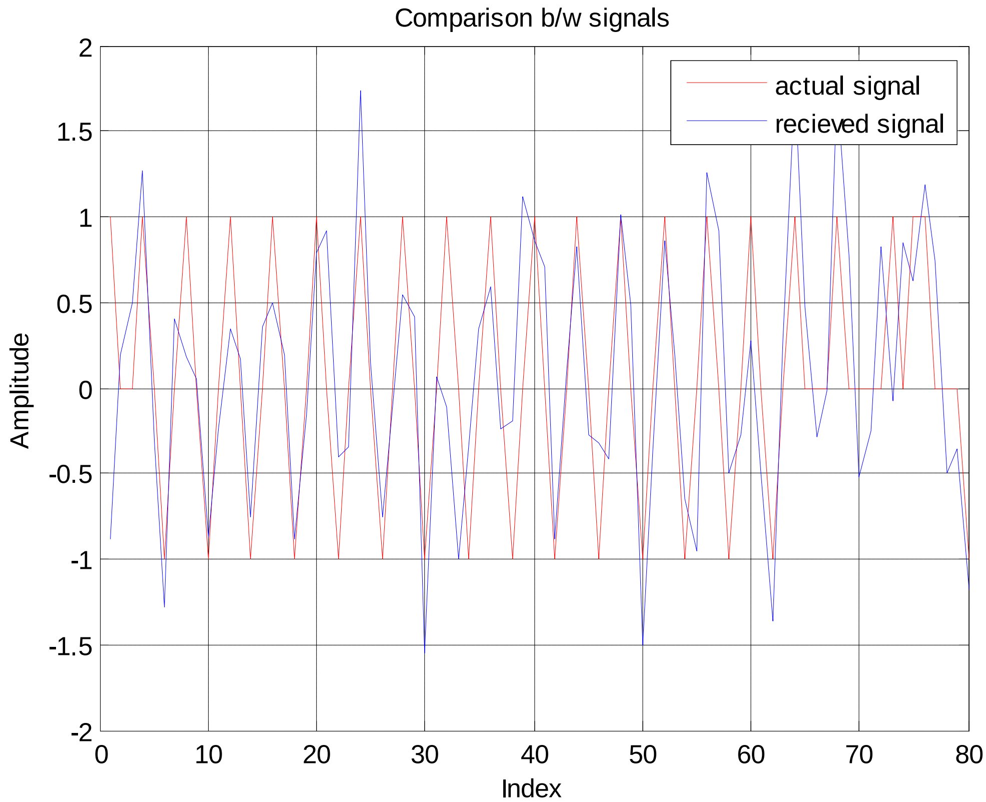


Fig. 4.8: Comparison of original and demodulated signals.

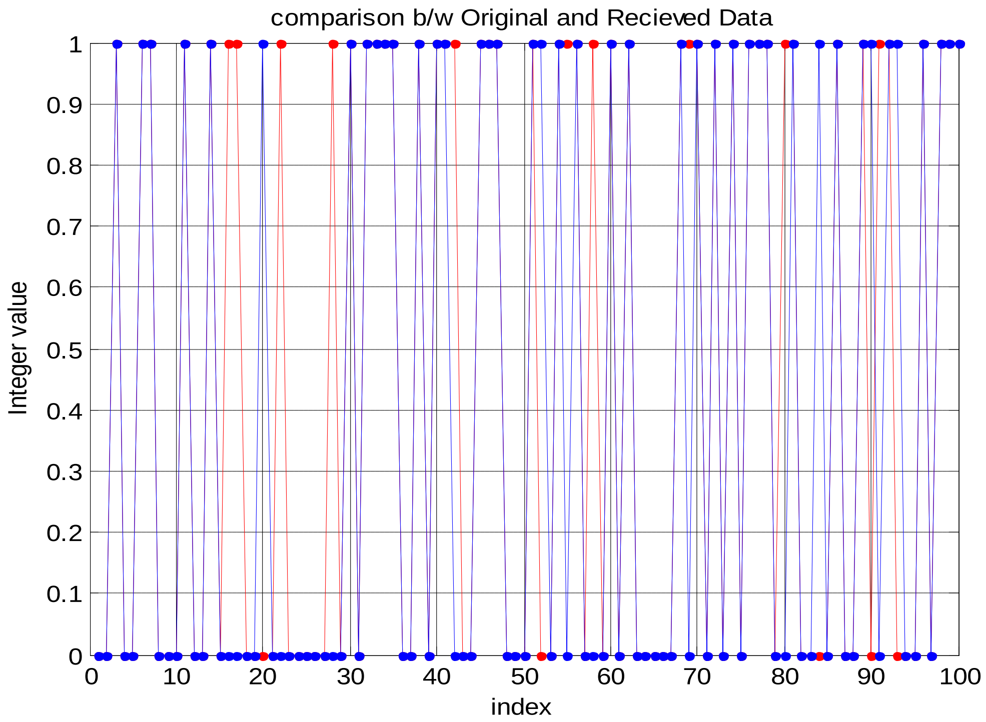


Fig 4.9: Comparison between signal data of original and demodulated.

The following graphs show how the Bit Error Rate varies with different SNR.

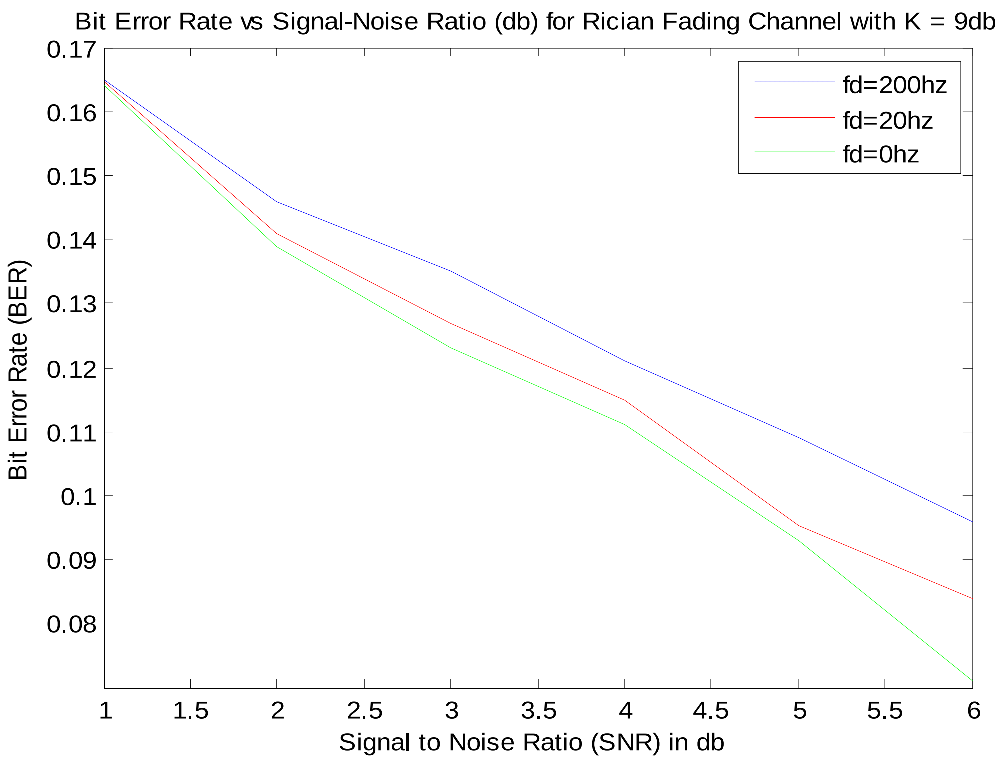


Fig. 4.10: BER vs SNR for constant Rician fading.

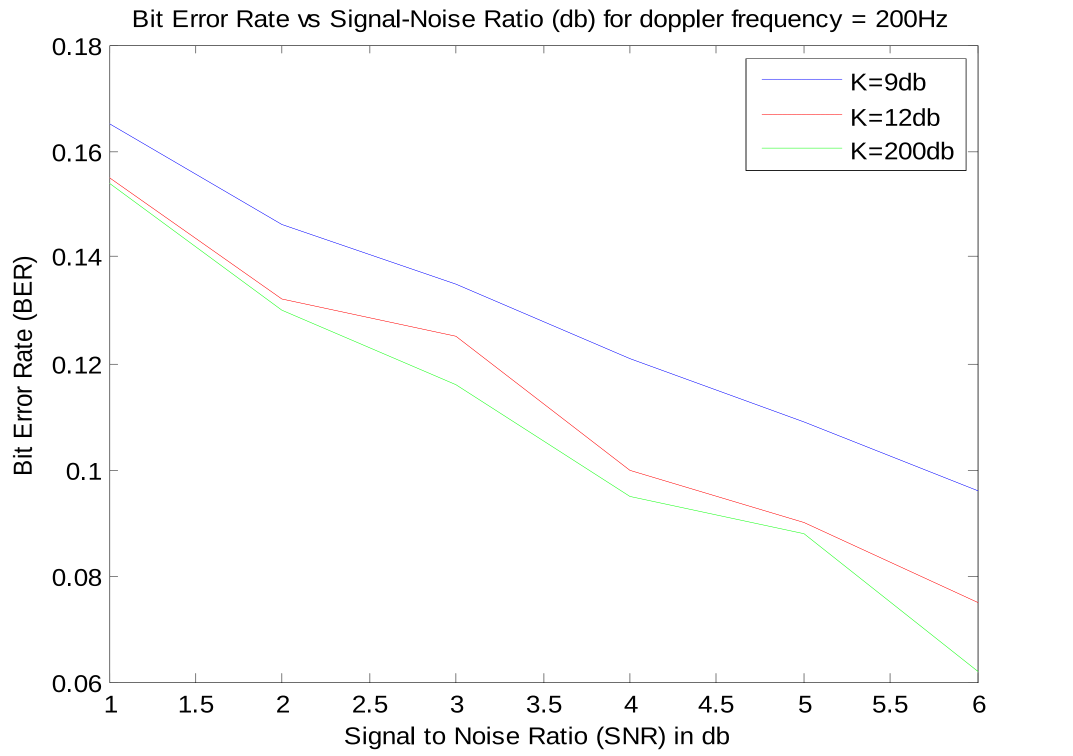


Fig. 4.11: BER vs SNR for constant Doppler frequency.

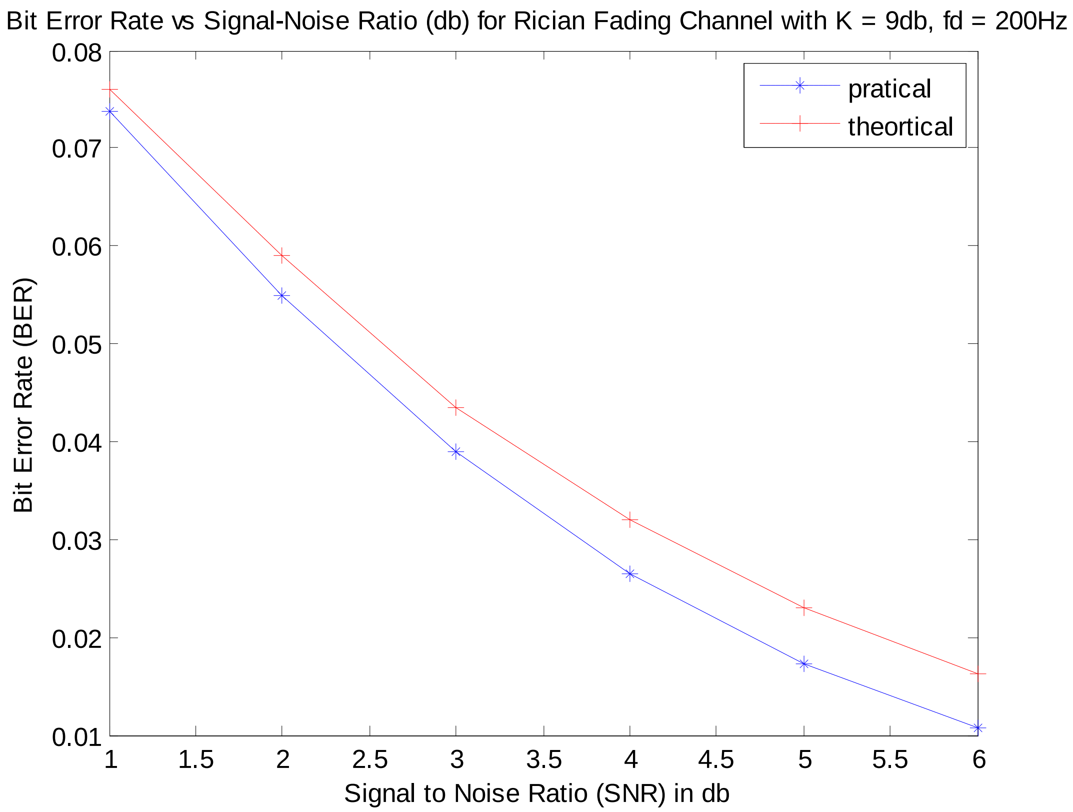


Fig. 4.12: SNR vs BER graph for K=9db, fd=200hz.

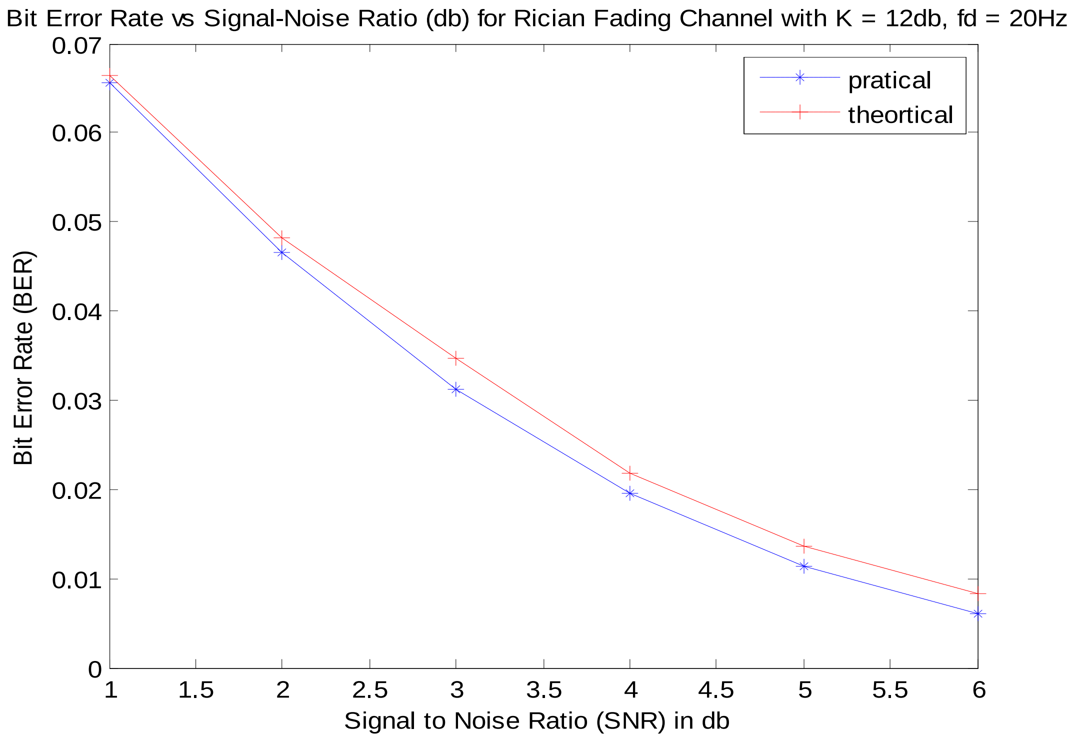


Fig. 4.13: SNR vs BER graph for K=12db, fd=20hz.

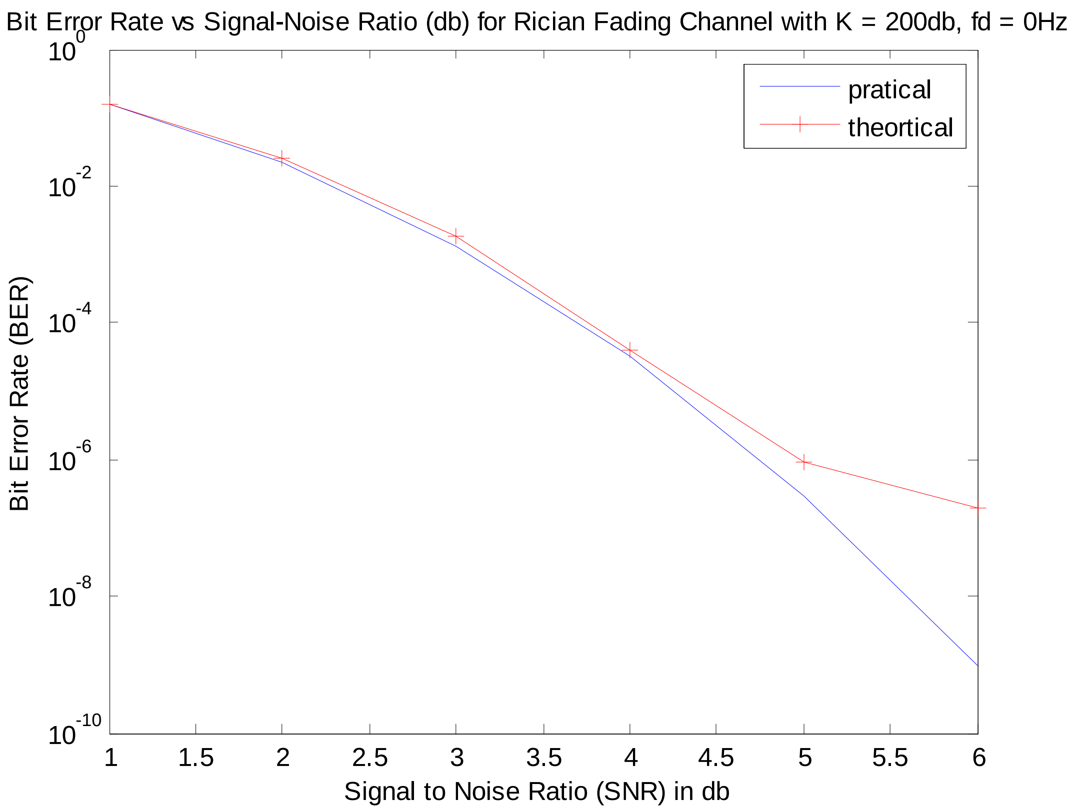


Fig. 4.14: SNR vs BER graph for K = 200db, fd = 0Hz.

## 4.2 OBSERVATIONS:

The simulation of the above design was performed using Matlab 7.0.

To determine the performance of Bayesian Demodulator we are comparing the obtained values with the theoretical values of a QPSK demodulator.

When you look at the comparing graphs the BER is decreased around 0.1 db to 0.5 db is varied according to different K and fd values.

Here we are generating only 1000 random bits and coming to real world scenario there will be huge amount of data and the variation between of BER will varying and be considerable.

For more comparison of performance of our demodulator a QAM demodulator is simulated considering with zero Doppler frequency and zero Rician fading. A program 'test_qam' is run and the corresponding noise power 'No', signal to noise ratio 'SNR' for a given N value are entered as input to our program to calculate the BER. Now these results are driven for comparing. When observed we find less BER with in Bayesian demodulator. Hence can conclude a better system performance.

Now, tests were run for various scenarios with the different values of maximum Doppler shift, Rician fading factor, number of transmitted bursts and SNR. In each case the BER values are averaged over the number of bursts run and these values are plotted against the values of Eb/No. (dB).

From the observations made it is seen that

1) The value of BER decreases with the increase in the value of SNR as obtained from the rain fall curve shown above.
2) For a given value of Doppler shift, the BER reduces with the increase in the value of Rician fading factor.
3) For a given value of Rician fading factor, the BER reduces with the decrease in the maximum Doppler shift.

# 5 CONCLUSION

Today’s technology usage demands for very high performance over the communication channel. During the transmission channel overcomes of various interferences and various other parameters that affect the performance hence resulting in higher bit error rate and resulting in wrong interpretation of message bits. Hence the communication system developed must be such developed that it should efficiently detect these message bits.

In most of the mobile or cellular systems, message gets disrupted due to multipath, reflections, interference and interferences resulting in fading of the signal. From the analysis on such short wireless channel for data communication it can be observed that the Bit error rate (BER) drops down with the increase in signal to noise ratio (SNR). The value of BER falls down with the increase in Rician fading factor under constant Doppler shift. If the Rician fading factor is constant for a channel the bit error rate (BER) reduces with the decrease in Doppler shift. From the observation made it can be concluded that for a wireless communication system with narrow channel undergoing fading the bit error rate fall down with the increase in signal to noise ratio, decrease in maximum Doppler shift and increase in Rician fading factor.

By Bayesian approach, a new deconvolution method, we can decrease the BER to an extent where there is considerable increase in system performance. Hence, reduces the cost effect.

Comparisons are made with both the coherent QPSK transreciever and non coherent type of QPSK modulation and QAM Detection. Though the QPSK modulator and QAM demodulator is not in extent now it is used here to show that a different modulator and a demodulator can be used which very familiar.

## FUTURE SCOPE:

The system designed considers 1132 bits of data as burst and frames a packet work can be extended for higher packet length resulting in more data transmission over each burst for achieving higher rate of bit rate. The work can be targeted to implement for other modulation technique.

# APPENDIX

----------------------------------------------------------------------------------------------------------

Main

----------------------------------------------------------------------------------------------------------

```
clear all; close all; clc;
n = input('Iterations to run: ');                    %No of iterations
SNR_EBNO = input('SNR in dB: ');                     %Signal to noise ratio
fd = input('Doppler freq in Hz: ');                  %doppler frquency
k = input('Rician Factor (K) in dB: ');
alfa = input('Alfa: ');                              %Reflection Factor
N = input('N: ');
filter=load('flt.txt');                              %matched filter coefficients
filt = filter.';
 for iteration = 1:n
   iteration
[struc payload] = burst(0);
figure(1);                                           % plotting digital bit stream
stem(struc(1:1000),'b-','filled');
title('Strucutred Bits'); xlabel('Bit Index'); ylabel('Value');
 [modreal modimag] = qpsk_mod(struc);                % QPSK Modulator
figure(2)
subplot(2,1,1)
plot(modreal);xlim([0 length(modreal)]);
xlabel('Bit Index----->'); ylabel('Value'); title('MODULATED REAL BITS')
subplot(2,1,2)
plot(modimag);xlim([0 length(modreal)])
xlabel('Bit Index------>'); ylabel('Value');
title('MODULATED IMAGINARY BITS')
 [upreal upimag] = upsampler(modreal,modimag);       % Up-Sampling by 16
figure(3)
subplot(2,1,1)
```

```
plot(upreal(1:9056));
xlim([0 length(upreal)]); xlabel('Bit Index---->'); ylabel('Value');
title('UPSAMPLED REAL');
subplot(2,1,2)
plot(upimag(1:9056));
xlim([0 length(upimag)]); xlabel('Bit Index---->'); ylabel('Value');
title('UPSAMPLED IMAGINARY');
[convreal convimag] = rect_filter(upreal,upimag,filt);             % Raised cosine Pulse Shaping
figure(4)
subplot(2,1,1)
plot(convreal(1:9056));
xlim([0 length(convreal)]); ylim([-0.3 0.3]); xlabel('Bit Index---->'); ylabel('Value');
title('INPHASE COMPONENT');
subplot(2,1,2)
plot(convimag(1:9056)); xlim([0 length(convimag)]); ylim([-0.3 0.3]);
xlabel('Bit Index---->'); ylabel('Value');
title('QUADATURE COMPONENT');
signal = complex(convreal,convimag);                               % Channel Section
signalthruchannel = channel(signal,fd,N,alfa,k);
figure(5)
subplot(2,1,1)
stem(signal); xlim([0 500]); title('the signal before channel')
subplot(2,1,2)
stem(signalthruchannel);xlim([0 500]);  title('the signal through channel')
 signalreal1= real(signal); signalimag1= imag(signal);
signalreal= real(signalthruchannel); signalimag= imag(signalthruchannel);
for ctr = 1:10                                                     % Signal Power calculation( without noise )
[matchsignalreal1 matchsignalimag1] = matchedfilter(signalreal1,signalimag1,filt);
[downsamplesignalreal1 downsamplesignalimag1] = downsampler(matchsignalreal1,matchsignalimag1);
   signalpower1(ctr) = var(downsamplesignalreal1) + var(downsamplesignalimag1);
end
meansignalpower1 = mean(signalpower1);
```

```
 for ctr = 1:10                                         % Noise Power calculation
noisereal = randn(1,9056); noiseimag = randn(1,9056);
 [matchnoisereal matchnoiseimag] = matchedfilter(noisereal,noiseimag,filt);
[downsamplenoisereal downsamplenoiseimag] = downsampler(matchnoisereal,matchnoiseimag);
    noisepower(ctr) = var(downsamplenoisereal) + var(downsamplenoiseimag);
end
 meannoisepower = mean(noisepower);
SNR = SNR_EBNO + 3.01;                                  % Calculation of SNR for (Es / N0)
A = sqrt((10^( -SNR / 10)) * (meansignalpower1 / meannoisepower));
                                                        % Multiplying of Scaling factor to Signal and Noise
[signalreal2 signalimag2] = awgn1(signalreal,signalimag,noisereal,noiseimag,A);
for ctr2=1:10                                           % Matched Filter
[matchsignalreal matchsignalimag] = matchedfilter(signalreal2,signalimag2,filt);
                                                        % DownSampling by 16
[downsamplereal downsampleimag] = downsampler(matchsignalreal,matchsignalimag);
                                                        % Calculation of signal power
    signalpower(ctr2) = var(downsamplereal) + var(downsampleimag);
end
meansignalpower = mean(signalpower);                    % Calculation of Mean Signal Power
h1=psd(signalpower1);h=psd(signalpower);
 s = spectrum.periodogram;
figure(6)
subplot(2,1,1)
psd(s,signalpower1)
subplot(2,1,2)
psd(s,signalpower)
figure(7)
plot(h1);
xlim([1 6])
hold on
plot(h,'r');xlim([1 6])
legend('signal power before channel','signal power through channel')
```

```
recdpayload = bayesian_demod(downsamplereal,downsampleimag);     % Bayesian Demodulator
err = abs(recdpayload - payload);                                 % Error calculation
errornum(iteration) = sum(err);
ber(iteration) = errornum(iteration) / 1000;
 end
figure(8);
plot(modreal(1:80),'r-');grid on                                  % B4 modulation
hold;
plot(downsamplereal(1:80),'b-');grid on;                          % After Demodulation
title('Comparison b/w signals');
xlabel('Index'); ylabel('Amplitude');
figure(9);
plot(payload(1:100),'.r-');hold;                                  % Original data
plot(recdpayload(1:100),'.b-');grid on;                           % Recieved data
title('comparison b/w Original and Recieved Data');
xlabel('index'); ylabel('Integer value');
errornumber = sum(errornum) / n                                   % Results Display
biterrorrate = errornumber / 1000
-----------------------------------------------------------------------------------------------------------
%Burst Builder
-----------------------------------------------------------------------------------------------------------
function [struc payload] = burst(tempvariable)
z = rand(1000,1);                                                 % Payload of 1000 vector length generation
for i = 1:1:1000
   if z(i) > 0.5
      payload(i) = 0;
   else      payload(i) = 1;
   end    end
x = rand(6,1);                                                    % Guard Symbols generation for the Burst
for i = 1:1:6
   if z(i) > 0.5
      guard(i) = 0;
```

```
else      guard(i) = 1;
end    end
temp = [];                                          % Unique Word for Burst generation
 for ij = 1:15
    temp2 = [0 0 0 1 1 1 1 0];    uniword = [temp temp2];
    temp = uniword;
    temp2 = [];
 end
struc = [guard uniword payload guard];              % Burst Structure formation
```

```
-----------------------------------------------------------------------------------------------------------
% QPSK Modulator
-----------------------------------------------------------------------------------------------------------
function [modburstreal,modburstimag] = qpsk_mod(structure)
temp = [];
lcount = 1;
for k = 1:2:1132
        temp(1) = structure(k);
          temp(2) = structure((k+1));
                if temp == [0 0]
                     modburstreal(lcount) = 1;                    modburstimag(lcount) = 0;
                elseif temp == [0 1]
                     modburstreal(lcount) = 0;                    modburstimag(lcount) = 1;
                elseif temp == [1 1]
                     modburstreal(lcount) = -1;                    modburstimag(lcount) = 0;
                elseif temp == [1 0]
                     modburstreal(lcount) = 0;                    modburstimag(lcount) = -1;
                end
        temp = [];
        lcount = lcount+1;
end
```

-----------------------------------------------------------------------------------------------------

% Up Sampling

-----------------------------------------------------------------------------------------------------

```
 function [upreal,upimag] = upsampler(modreal,modimag)
 for i = 1:1:9056
    upreal(i) = 0;  upimag(i) = 0; end
 upreal(1) = modreal(1);
upimag(1) = modimag(1);
 for i = 1:1:565
    tmp = ((i * 16)+1);
    upreal(tmp) = modreal(i+1);
    upimag(tmp) = modimag(i+1);
end
 upreal(9040) = modreal(566);
upimag(9040) = modimag(566);
```

-----------------------------------------------------------------------------------------------------

FILTER COFFICIENTS

-----------------------------------------------------------------------------------------------------

| | | | |
|---|---|---|---|
| 0.000758 | -0.000574 | -0.001911 | -0.003132 |
| -0.004113 | -0.004743 | -0.004928 | -0.004605 |
| -0.003751 | -0.002386 | -0.000577 | 0.001561 |
| 0.003867 | 0.006148 | 0.008184 | 0.009746 |
| 0.010610 | 0.010579 | 0.009497 | 0.007267 |
| 0.003867 | -0.000638 | -0.006092 | -0.012245 |
| -0.018757 | -0.025208 | -0.031119 | -0.035966 |
| -0.039211 | -0.040330 | -0.038843 | -0.034343 |
| -0.026526 | -0.015214 | -0.000376 | 0.017862 |
| 0.039211 | 0.063222 | 0.089299 | 0.116719 |
| 0.144658 | 0.172226 | 0.198505 | 0.222589 |
| 0.243624 | 0.260849 | 0.273633 | 0.281500 |
| 0.284155 | 0.281500 | 0.273633 | 0.260849 |
| 0.243624 | 0.222589 | 0.198505 | 0.172226 |

| | | | |
|---|---|---|---|
| 0.144658 | 0.116719 | 0.089299 | 0.063222 |
| 0.039211 | 0.017862 | -0.000376 | -0.015214 |
| -0.026526 | -0.034343 | -0.038843 | -0.040330 |
| -0.039211 | -0.035966 | -0.031119 | -0.025208 |
| -0.018757 | -0.012245 | -0.006092 | -0.000638 |
| 0.003867 | 0.007267 | 0.009497 | 0.010579 |
| 0.010610 | 0.009746 | 0.008184 | 0.006148 |
| 0.003867 | 0.001561 | -0.000577 | -0.002386 |
| -0.003751 | -0.004605 | -0.004928 | -0.004743 |
| -0.004113 | -0.003132 | -0.001911 | -0.000574 |

```
--------------------------------------------------------------------------------------------------------------------
% Pulse Shaping
--------------------------------------------------------------------------------------------------------------------
function [convreal,convimag] = rect_filter(realpart,imagpart,filt)
convtmpreal = conv(realpart,filt);
convtmpimag = conv(imagpart,filt);
for i = 49:1:9104
    convreal(i-48) = convtmpreal(i);
    convimag(i-48) = convtmpimag(i);
end


--------------------------------------------------------------------------------------------------------------------
%Matched Filter
--------------------------------------------------------------------------------------------------------------------
function [convreal,convimag] = matchedfilter(realpart,imagpart,filt)
convtmpreal = conv(realpart,filt);
convtmpimag = conv(imagpart,filt);
for i = 49:1:9104
    convreal(i-48) = convtmpreal(i);
    convimag(i-48) = convtmpimag(i);
end
```

-----------------------------------------------------------------------------------------------------------------

% Rician Channel

-----------------------------------------------------------------------------------------------------------------

```
function [signalthruchannel] = channel(signal,fd,N,alfa,k)
    c = 3 * (10)^8;    wd = 2 * pi * fd;    t = rand(1);     M = 0.5 * ( N / 2 - 1);
   y = 10^(-k / 20);    x = sqrt((1 + y)^(-1));
for tcounter = 1:9056                                            % Rayleigh channel implementation
       temp1 = 0;        temp2 = 0;        t = t + 0.002;
       for n = 1:N
           beta = n*pi/(N+1);
           wn = wd*cos(2*pi*n/N);
           gi(tcounter)=(2*cos(beta)*cos(wn*t))+(sqrt(2)*cos(wd*t)*cos(alfa))+temp1;
gq(tcounter)=(2*sin(beta)*cos(wn*t))+(sqrt(2)*cos(wd*t)*sin(alfa))+temp2;
                     temp1 = gi(tcounter);         temp2 = gq(tcounter);
              end
            gi(tcounter) = ((1 / (2 * (M + 1)))^0.5) * gi(tcounter);
           gq(tcounter) = ((1 / (2 * M))^0.5) * gq(tcounter);
           rayleighenvelope(tcounter) = (gi(tcounter) + j * gq(tcounter));
  end
         ricianenvelope = x*(1 + rayleighenvelope*y);
         signalthruchannel = ricianenvelope .* signal;
```

-----------------------------------------------------------------------------------------------------------------

% Bayesian Demodulator

-----------------------------------------------------------------------------------------------------------------

```
 function payloadinbits = bayesian_demod(downreal,downimag)
  recunireal = downreal(4:63);    recuniimag = downimag(4:63); % uniqueword extraction
  recuni = complex(recunireal,recuniimag);
                                                        % Individual unique words
  R_UW_p1real = recunireal(1:4:60);
  R_UW_pjreal = recunireal(2:4:60);
  R_UW_m1real = recunireal(3:4:60);
```

```
R_UW_mjreal = recunireal(4:4:60);
R_UW_p1imag = recuniimag(1:4:60);
R_UW_pjimag = recuniimag(2:4:60);
R_UW_m1imag = recuniimag(3:4:60);
R_UW_mjimag = recuniimag(4:4:60);
                              % Mean calculation
meanR_UW_p1real = mean(R_UW_p1real);
meanR_UW_pjreal = mean(R_UW_pjreal);
meanR_UW_m1real = mean(R_UW_m1real);
meanR_UW_mjreal = mean(R_UW_mjreal);
meanR_UW_p1imag = mean(R_UW_p1imag);
meanR_UW_pjimag = mean(R_UW_pjimag);
meanR_UW_m1imag = mean(R_UW_m1imag);
meanR_UW_mjimag = mean(R_UW_mjimag);
                    % Bayesian detection's variable calculation
ruwp1mean = [meanR_UW_p1real meanR_UW_p1imag]';
ruwpjmean = [meanR_UW_pjreal meanR_UW_pjimag]';
ruwm1mean = [meanR_UW_m1real meanR_UW_m1imag]';
ruwmjmean = [meanR_UW_mjreal meanR_UW_mjimag]';
R_UW_p1 = recuni(1:4:60);
R_UW_pj = recuni(2:4:60);
R_UW_m1 = recuni(3:4:60);
R_UW_mj = recuni(4:4:60);
                              %variance calculation
varR_UW_p1 = var(R_UW_p1);
varR_UW_pj = var(R_UW_pj);
varR_UW_m1 = var(R_UW_m1);
varR_UW_mj = var(R_UW_mj);
vartotal = [varR_UW_p1 varR_UW_pj varR_UW_m1 varR_UW_mj];
varavg = mean(vartotal);
varmatrix = [varavg 0;0 varavg];
varfinal = sqrt(det(varmatrix));
```

```
                                   % Payload extraction
recdpayloadreal = downreal(64:563);
recdpayloadimag = downimag(64:563);
commonfactor = (1 / (2 * pi * varfinal));
temp = [];
 for i = 1:500
   r = [recdpayloadreal(i) recdpayloadimag(i)]';
                                   % Probability calucaltion
   pbp1 = commonfactor * exp((-1 / 2) * (r - ruwp1mean)' * (inv(varfinal)) * (r - ruwp1mean));
   pbpj = commonfactor * exp((-1 / 2) * (r - ruwpjmean)' * (inv(varfinal)) * (r - ruwpjmean));
   pbm1 = commonfactor * exp((-1 / 2) * (r - ruwm1mean)' * (inv(varfinal)) * (r - ruwm1mean));
   pbmj = commonfactor * exp((-1 / 2) * (r - ruwmjmean)' * (inv(varfinal)) * (r - ruwmjmean));
    pbmatrix = [pbp1 pbpj pbm1 pbmj];
   [pbsort pbindex] = max(pbmatrix);
                                   % symbols to bits conversion
      if pbindex == 1
         temp2 = [0 0];
       elseif pbindex == 2
         temp2 = [0 1];
       elseif pbindex == 3
         temp2 = [1 1];
       else
         temp2 = [1 0];
       end
   payloadinbits = [temp temp2];
   temp = payloadinbits;
   temp2 = [];


end
```

```
--------------------------------------------------------------------------------------------------
% program for comparing graphs
--------------------------------------------------------------------------------------------------
snr = [1 2 3 4 5 6];
 ber200 =[0.165 0.146 0.135 0.121 0.109 0.096];
ber20 = [0.1645 0.141 0.127 0.115 0.0954 0.0840];
ber0 = [0.1640 0.1390  0.1230 0.1110 0.0930 0.0710];
 ber12=[0.1550 0.132 0.125 0.1 0.09 0.075];
 ber=[0.154 0.13 0.116 0.095 0.088 0.062]
 figure(1)
plot(snr,ber200,'b:')
hold on
plot(snr,ber20,'r:')
hold on
plot(snr,ber0,'g')
title('Bit Error Rate vs Signal-Noise Ratio (db) for Rician Fading Channel with K = 9db');
xlabel('Signal to Noise Ratio (SNR) in db ');
ylabel('Bit Error Rate (BER)');
legend('fd=200hz','fd=20hz','fd=0hz');

figure(2)
plot(snr,ber200,'b:')
hold on
plot(snr,ber12,'r:')
hold on
plot(snr,ber,'g')
title('Bit Error Rate vs Signal-Noise Ratio (db) for doppler frequency = 200Hz');
xlabel('Signal to Noise Ratio (SNR) in db ');
ylabel('Bit Error Rate (BER)');
legend('K=9db','K=12db','K=200db');
```

# REFRENCES

[1] Andrea Goldsmith, Wireless Communications, Cambridge University Press, 2005, pp. 2-27.

[2] J. G. Proakis, *Digital Communications.* New York: McGraw-Hill,1983.

[3] W. Y. C. Lee, *Mobile Communications Engineering.* New York: McGraw-Hill, 1982.

[4] Wireless Communications & Networks, 2/E William Stallings ISBN-10: 0131918354 ISBN-13: 9780131918351 Publisher: Prentice Hall Copyright: 2005.

[5] Simmon Hakins ‘‘Digital Communications’’ ISBN 0-471-62947-2, 1988 version.pages.

[6] F. M. Gardner, “A BPSK/QPSK Timing Error Detector for Sampled Receivers,” IEEE Transactions on Communications, vol. COM-34, pp. 423-429, May 1986.

[7] John G Proakis, Dimitris G Manolakis: Digital Signal Processing, 4rd edition ISBN: 0-13-187374-1

[8]http://mobiledevdesign.com/images/archive/0402Gentile50.pdf , http://mobiledevdesign.com/images/archive/0402Gentile50.pdf

[9]http://www.filter-solutions.com/FIR.html#book1 [http://www.filter-solutions.com/FIR.html#book1

[10]http://www.disi.unige.it/person/GianuzziV/MobComp/lucidi/Physical.html [http://www.disi.unige.it/person/GianuzziV/MobComp/lucidi/Physical.html]

[11] Marvin K.Simon, Mohamed-slim Alouini “Digital communication over fading channels”, John Wiley & Sons, INC, 2002.

[12]Matched filtering and timing recovery in digital receivers, Louis Litwin. http://mobiledevdesign.com/tutorials/radio_matched_filtering_timing/

[13] M. Gertsman, “Symbol-by-symbol MAP demodulation of CPM signals on Rayleigh flat fading channels,” M. Eng. thesis, Dep. Syst. Comput. Eng., Carleton Univ., Ottawa, Canada, 1996.

[14] X. Yu and S. Pasupathy, “Innovations-based MLSE for Rayleigh fading channels,” *IEEE Trans. Commun.*, vol. 43, pp. 1534–1544, Feb./Mar./Apr. 1995.

[15] A. Anastasopoulos and K. M. Chugg, “Adaptive soft-input soft-output algorithms for iterative detection with parametric uncertainty,” *IEEE Trans.Commun.*, vol. 48, pp. 1638–1649, Oct. 2000.

[16] J. Hagenauer, “Soft-in/soft-out—The benefits of using soft values in all stages of digital receivers,” in *3rd Int. Workshop Digital Signal Processing Techniques Applied Space Commun.*, 1992.

[17] J. Lodge and M. Moher, “Maximum likelihood sequence estimation of CPM signals transmitted over Rayleigh flat-fading channels,” *IEEE Trans. Commun.*, vol. 38, pp. 787–794, June 1990.

[18] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, “Optimal decoding of linear codes for minimizing symbol error rate,” *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 284–287, Mar. 1974.

[19] A. Logothetis and V. Krishnamurthy, “Expectation maximization algorithms for MAP estimation of jump Markov linear systems,” *IEEE Trans.Signal Proc.*, vol. 47, pp. 2139–2156, Aug. 1999.

[20] C. Douillard, M. J´ez´equel, C. Berrou, A. Picard, P. Didier, and A. Glavieux, “Iterative correction of intersymbol interference: Turboequalization,” *Europ. Trans. Telecommun. (ETT)*, vol. 6, pp. 507–511,Sept.–Oct. 1995.

[21] Q. Dai and E. Shwedyk, “Detection of bandlimited signals over frequency selective Rayleigh fading channels,” *IEEE Trans. Commun.*, vol. 42, pp. 941–950, Feb./Mar./Apr. 1994.

[22] G.K. Lee, S.B. Gelfand, and M.P. Fitz. "Bayesian Techniques for Deconvolution," Technical Report, School of Electrical Engineering, F'urdue University, West Lafayette, IN, 1993.

[23] Lee William C, Mobile Cellular Telecommunications Systems, Mc Graw-Hill, 1989 449 s, ISBN 0-07-100790-3.

[24] Alberto Leon-Garcia, Probability and Random Processes for Electrical Engineering, 2nd edition, Addison Wesely, 1994, 677 pages, ISBN 0-201-50037-X.

[25] Stephen G. Wilson, ‘Digital Modulation and Coding’, Prentice Hall, 1996, 677 pages, ISBN 0-13-210071-1.

[26] Lars Ahlin, Jens Zander, Principles of Wireless Communications, Studentlitteratur, 1998, 418 pages,ISBN 91-44-00762-0.