DATA TRANSMISSION OVER GSM VOICE CHANNEL

By

NC Hira Imtiaz

NC Ifrah Waheed

PC Badar Mehmood

Submitted to the Faculty of Department of Electrical Engineering, Military College of Signals, National University of Sciences and Technology, Islamabad in partial fulfillment for the requirements of a B.E Degree in Telecom Engineering

AUGUST 2015

ABSTRACT

DATA TRANSMISSION OVER GSM VOICE CHANNEL

While the cellular revolution has made voice connectivity ubiquitous in the developing world, data services are largely absent or are prohibitively expensive. GSM voice channel is designed for low-bit rate digitized speech which makes it unsuitable for data communication mainly because of the voice codecs used in GSM. These codecs greatly distort any signal that doesn't have speech-like properties.

This project is about the development of modem for GSM voice channel using speech-like waveform that can go through GSM voice channel without experiencing significant distortion. The objective of this project is to carefully evaluate the recent research developments in the area, design and implement a suitable waveform for real-time data communication over GSM voice channel.

GSM voice channel when used for data communication will make wide spread GSM channel attractive for variety of applications. This data transmission will have greater throughput when compared with SMS data service and that too at the lower cost per bit.

CERTIFICATE OF CORRECTNESS AND APPROVAL

It is certified that the work contained in this thesis title “Data Transmission over GSM voice Channel”, carried out by Hira Imtiaz, Ifrah Waheed and Badar Mehmood under the supervision of Brig.Ashraf Masood in partial fulfillment of Degree of Bachelor of Telecommunication Engineering, is correct and approved.

Approved By

Brig.Ashraf Masood (R)

PROJECT SUPERVISOR

Dean Military College of Signals ,NUST.

DECLARATION

No portion of the work presented in this dissertation has been submitted in support of another award or qualification either at this institution or elsewhere.

DEDICATION

To Almighty Allah, for whose greatness we do not have enough words,

To our parents and friends, without whose unflinching support and unstinting cooperation, our work of this magnitude would not have been possible.

ACKNOWLEDGEMENTS

All praises to the Almighty Allah, who enlightened us with the requisite knowledge to accomplish the project goals that we set for ourselves prior to the start of the project, and on a broader level in the completion of our degree.

We wish to express our gratitude to our supervisor Brig.Ashraf Masood, Dean Military of Signals , National University of Sciences and Technology, for his continuous support and supervision during the course of the project. His persistent guidance is certainly commendable.

We would specially like to thank Co-supervisor Azam. It would not have been possible to complete the project on time without his technical and moral support.

We would also like to acknowledge the struggle of our parents for our well being and education. They have done endless efforts to make us standout. We would designate our success to them.

TABLE OF CONTENTS

LIST OF TABLES

TABLE OF FIGURES

LIST OF ABBREVIATIONS

ACELP: Algebraic Code Excitation Linear Prediction

AWGN: Additive white Gaussian noise

BFSK: Binary Frequency-Shift keying

CRC: Cyclic Redundancy Check

DSK: DSP Starter Kit

Δ: delta

FDMA: Frequency Division Multiple Access

FEC: Forward Error Correction

FED: Forward Error Detection

FSK: Frequency-Shift keying

$\boldsymbol{\Delta f}$**:** Peak frequency-deviation

$\boldsymbol{f_m}$**:** Highest Frequency

f_{base}: Base Frequency

GMSK: Gaussian Minimum Shift Keying

GSM: Global System for Mobile communications

Hz: Hertz

ISI: Inter-Symbol Interference

LPC: Linear Predictive Coding

LTP: Long Term Prediction

MATLAB: Matrix Laboratory

P2P: Peer to Peer

PSK: Phase-Shift keying

SMS: Short Message Service

TDMA: Time Division Multiple Access

VAD: Voice Activity Detection

Chapter 1 Introduction

1.1 Motivation and Problem Statement

The motivation of this project lead us to the problem statement and ultimately to the selection of this project. There are two distinct motivations which are explained below:

a) For applications that require simultaneous transmission of data and voice, we need to have this modem. This modem would allow transmission of data over voice channel simultaneously and bidirectional.
b) Data communication over communication channels have traditionally been performed using a modem. Cellular networks have data and voice channels over which data may be transmitted. Data communication may be utilized for a variety of purpose. One of the examples being vehicle telematics system which utilizes bi-directional data transmission between vehicles and call centers. Vehicle telematics generally utilize a cellular communication. Since safety and security is the large part of these system data communications between the vehicle and call centers should be delivered quickly and reliably. But as the data channel of cellular networks work on IP protocol, the transmission could take several seconds to several hours. On the other hand voice channels for all cellular communication system provide a low delay audio path between users. It is therefore natural to use this path for data communication in systems such as vehicle telematics or in order words for real time data communication.

Thus problem statement of our project is:

“GSM voice channel is designed for low-bit rate digitized speech which is unsuitable for data communication mainly because of the voice codec used in GSM.”

1.2 Project description and Silent features

While the cellular revolution has made voice connectivity ubiquitous in the developing world, data communication services are largely absent or are prohibitively expensive. GSM voice channel is designed for low-bit rate digitized speech which makes unsuitable for data communication mainly because of the voice codecs (AMR) used in GSM. These codecs greatly distort any point-to-point digital data signal that doesn't have speech-like properties. This project is about the development of data modem for GSM voice channel using speech-like waveform that can go through GSM voice channel without experiencing significant distortion. The objective of this project is to carefully evaluate the recent research developments in the area, design and implement a suitable waveform for real-time data communication over GSM voice channel.

Data communication along with voice will make wide spread GSM network attractive for variety of applications. Although SMS service can be used for low data rate transfer, the data modem will offer increased throughout as lower cost-per bit than SMS. It will enable new mobile services in rural developing regions where no data connectivity solution exists today.

End goal of the project is to demonstrate real-time data transfer (text file or image) during a voice call session between two GSM nodes.

1.3 Scope of work

Scope of the work includes:

1) Literature review to evaluate the recent developments in this area

2) Complete temporal and frequency characterization of GSM voice channel

3) Design of a suitable waveform that can carry data through GSM channel without undergoing significant distortion

4) Simulation in MATLAB and performance characterization of the waveform

5) Real-time implementation of the waveform on Taxes Instrument's DSP SDK.

1.4 Objective

Non-linear voice codec operation makes GSM voice channel one the most difficult channels to communicate data reliably. We will try to understand how to characterize a communication channel and its distortion and how to design suitable waveform for it. It will enable us to design communication systems for most practical wireless/wired channels in future.

End objective of the project is to demonstrate real-time data transfer (text file or image) during a voice call session between two GSM nodes. This project is a part of our final year BS degree project.

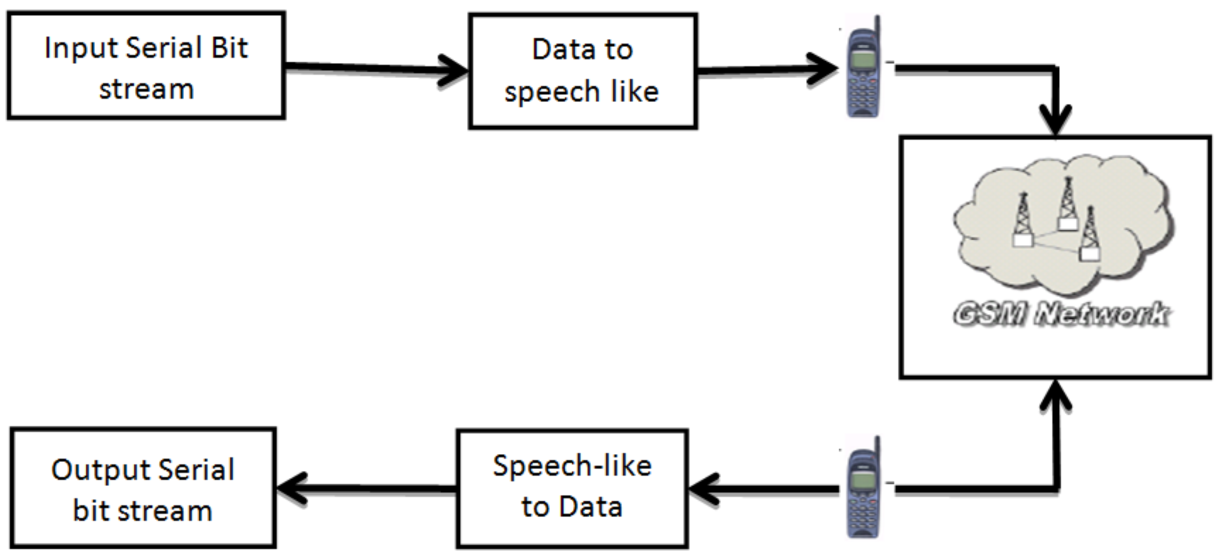

Figure 1.1: Methodology [2]

1.5 Deliverables

Our project basically has main part understanding GSM channel impairments and designing a communication system that can mitigate them. We will do coding in MATLAB to start with and later we are going to implement it on hardware.

At first we did research regarding our project, read few papers, understood working of different GSM codec, and learned about the different modulation techniques. Then we started MATLAB coding for designing transmitter and receiver and are currently working on it.

.

Chapter 2 Literature Review

GSM stands for Global System for Mobiles. This is a world-wide standard for digital cellular telephony. GSM was created by the Europeans, and originally meant "Groupe Special Mobile". GSM is a published standard by ETSI, and it has widespread implementation in Europe, Asia, and increasingly America.

GSM has used a variety of voice codec to squeeze 3.1 kHz audio into between 6.5 and 13 kb/s. Originally, two codec, named after the types of data channel they were allocated, were used, called Half Rate (6.5 kb/s) and Full Rate (13 k/s). These used a system based on linear predictive coding (LPC). The GSM uses Gaussian Minimum Shift Keying (GMSK) modulation method. GSM chose a combination of TDMA/FDMA as its method. The uplink frequency range specified for GSM is 933 - 960 MHz (basic 900 MHz band only). The downlink frequency band 890 - 915 MHz (basic 900 MHz band only). GSM uses linear predictive coding (LPC). The purpose of LPC is to reduce the bit rate.[1]

Dr. N. Katugampal and Dr. K. Al-Naimi have mentioned in their paper, “Voice-Data Tunneling over GSM Voice Channel”,that the GSM speech service is secure up to the point where speech enters the core network. However over the core network it has no security. Moreover the security over the air interface is optional, and some operators do not implement the encryption algorithms. In order to have an end-to-end security, speech must be encrypted before it enters the GSM network. However the GSM speech transcoding process will severely distort an encrypted signal that does not possess characteristics of speech. As a result it is not possible to use standard modem techniques over the GSM speech channel.[2]

Although the GSM data channel can be used for encrypted speech transmission, this approach suffers from a number of disadvantages. The GSM data channel typically requires 28-31 seconds to establish a connection, of which approximately 18 seconds are taken up by the GSM modem handshaking time. Also the round-trip time of the GSM data channel is between 1 and 2 seconds for the 95th percentile. This is the solution adopted by current secure GSM systems, which therefore suffer from the problems associated with the GSM data channel: long delays and call set-up times, and interoperability problems. Moreover they require a dedicated handset and subscription to the GSM data services.[2]

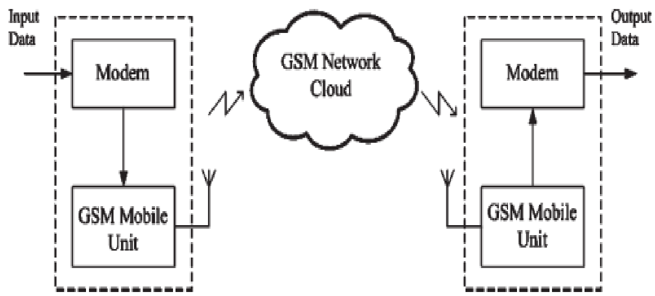

Figure 2.1: Data communication over GSM cloud[3]

LaDue, C., Sapozhnykov, V., and Fienberg, K have statedin their research paper, “A data modem for GSM voice channel”,that the data communicationover the GSM voice channel has unique set of problems; therefore we design a modem capable of communicating data through a compressed speech medium. In figure 2.1, it is shown that it will convert input data to modulated signal, which is fed into GSM mobile unit exactly like a speech signal. The GSM mobile unit encodes and modulates this signal and then sends it in air. The receiver GSM mobile unitdemodulates and decodes the received signal, which is, in turn, fed into the modem. The modem outputs estimates of sent data. As far as the GSM network is concerned, it is a normal voice call.[3]

2.1 Codec

2.1.Full-Rate

The full-rate codec is a regular pulse excitation, long-term prediction (RPE-LTP) linear predictive coder that operates on a 20-ms frame composed of one hundred sixty 13-bit samples.

2.1.2EFR

The EFR codec is an algebraic code excitation linear prediction (ACELP) codec. The pre-processing stage for EFR consists of an 80 Hz high-pass filter, and some downscaling to reduce implementation complexity.

2.1.3GSM AMR Codec

ACELP codec is used in it. In fact, the 12.2 Kbit/s AMR codec is computationally the same as the EFR codec. Discontinuous transmission is employed so that when there is no speech activity the transmission is cut.Additionally Voice Activity Detection (VAD) is usedto indicate when there is onlybackground noise and no speech.

MODE	BIT RATE (KBPS)	FULL RATE (FR) / HALF RATE (HR)
AMR 12.2	12.2	FR
AMR 10.2	10.2	FR
AMR 7.95	7.95	FR / HR
AMR 7.40	7.40	FR / HR
AMR 6.70	6.70	FR / HR
AMR 5.90	5.90	FR / HR
AMR 5.15	5.15	FR / HR
AMR 4.75	4.75	FR / HR

Table 2.1: AMR CODEC data rates [4]

As clearly shown in figure 2.2 that the AMR codec has a total of eight rates, eight are available at full rate (FR), while six are available at half rate (HR). [4]

2.2 Inter-symbol Interference

Then we have gone through inter symbol interference and pulse shaping filter as well. Inter-symbol interference (ISI) is an unavoidable consequence of both wired and wireless communication systems.

The main problem is that energy, which we wish to confine to one symbol, leaks into others.

So one of the simplest things we can do to reduce ISI is to just slow down the signal. Pulse shaping filters are used to counter ISI.CharanLangton has proposed different techniques in his paper,“Inter symbol interference and raised cosine filtering”, for reducing ISI. Those techniques are discussed below:

- Using a square pulse shape.[5]
- Using sinc function [5]
- Using raised cosine filter

2.3 Modulation Techniques

Then afterwards, we have gone through different modulation techniques.

2.3.1 Frequency shift keying

This is a modulation technique where data is transmitted through discrete frequency changes of a carrier wave.

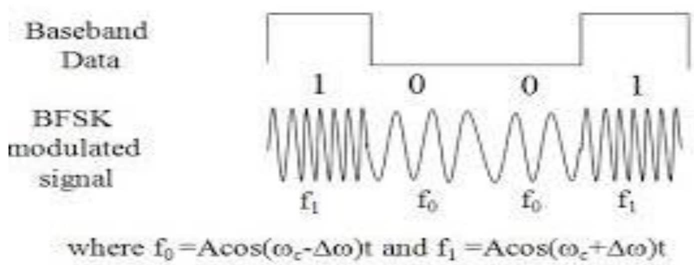

Figure 2.2: Frequency shift keying Technique [6]

Figure 2.7 shows the simplest method of performing FSK. In BFSK, we transmit one sine wave per bit. A 0 or 1 is transmitted as a sinusoid of frequency f0 or f1 respectively.
Intuitively, we want f0 and f1 to be far-apart, in order to be able to distinguish the symbols. However, this leads to abrupt and large changes in frequency over short time scales, rendering the audio stream to become non voice-like; the voice codec therefore significantly distort to the audio stream. Therefore, we also need to keep the symbols close enough (in the frequency domain) to please the voice codec. [6]

2.3.2Phase shift keying

This is a modulation technique in which digital information is transmitted through discrete phase changes of a carrier wave.

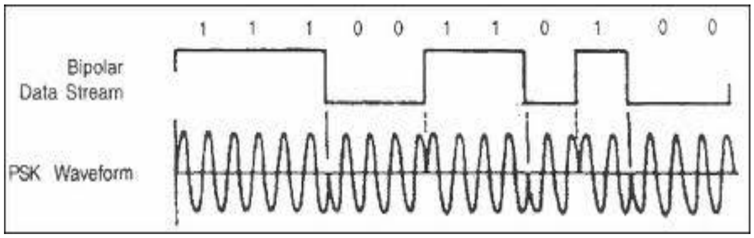

Figure 2.3: Phase shift keying Technique [6]

Figure 2.8 shows that the phase of the waveform changes for every single sinusoid that is transmitted, this creates a large number of discontinuities (abrupt jumps in amplitude) in the time domain signal. Even very small changes in phase are sufficient to cause discontinuities in the input signal. Since such audio signals are not voice-like, they get severely distorted by the codec.[6]

2.4 Cyclic Redundancy Check (CRC)

The cyclic redundancy check, or CRC, is a technique for detecting errors in digital data, but not for making corrections when errors are detected. It is used primarily in data transmission. In the CRC method, a certain number of check bits, often called a checksum, are appended to the message being transmitted. The receiver can determine whether or not the check bits agree with the data, to ascertain with a certain degree of probability whether or not an error occurred in transmission. If an error occurred, the receiver sends a “negative acknowledgement” (NAK) back to the sender, requesting that the message be retransmitted.

The basic ideabehind CRCs is to treat the message string as a single binary word M, and divide it by a key word k that is known to both the transmitter and the receiver. The remainder r left after dividing M by k constitutes the "check word" for the given message.The transmitter sends both the message string M and the check word r, and the receiver can then check the data by repeating the calculation, dividing M by the key word k, and verifying thatthe remainder is r.

The ITU-TS (CCITT) has a standard for a 16-bit polynomial to be used to obtain the cyclic redundancy code (CRC) that is appended. IBM's Synchronous Data Link Control and other protocols use CRC-16, another 16-bit polynomial. A 16-bit cyclic redundancy code detects all single and double-bit errors and ensures detection of 99.998% of all possible errors. This level of detection assurance is considered sufficient for data transmission blocks of 4 kilobytes or less. For larger transmissions, a 32-bit CRC is used. The Ethernet and token ring local area network protocols both used a 32-bit CRC

2.5 Transcoding

Transcoding is the direct analog-to-analog or digital-to-digital conversion of one encoding to another. This is usually done in cases where a target device (or workflow) does not support the format or has limited storage capacity that mandates a reduced file size, or to convert incompatible or obsolete data to a better-supported or modern format.Transcoding is the ability to adapt digital files so that content can be viewed on different playback devices.

Transcoding servers and services reformat material that would otherwise have to be developed separately for different platforms. They are commonly used for adapting content for mobile devices or serving video. There are a number of different ways that transcoding can take place but the overall process remains the same. The source format is translated into a raw intermediate format and then re-translated into a format the end user's device recognizes

Although transcoding can be found in many areas of content adaptation, it is commonly used in the area of mobile phone content adaptation. In this case, transcoding is a must, due to the diversity of mobile devices and their capabilities. This diversity requires an intermediate state of content adaptation in order to make sure that the source content will adequately function on the target device to which it is sent

Chapter 3 Design and Development

For any system to work in an appropriate way, first of all an ideal system is designed and later on it is improved to deal with practical issues. Real time implementation of the system is carried out once the design has been completed. In a design process, the first step is to accomplish it in an ideal scenario. Hence to study ideal systems and to design them is one of the major phases of every project to get it done.

Development is the extension of the design and it follows the same sequence of events that are being suggested in design layout. The design layout of our project is shown below in block diagram 3.1

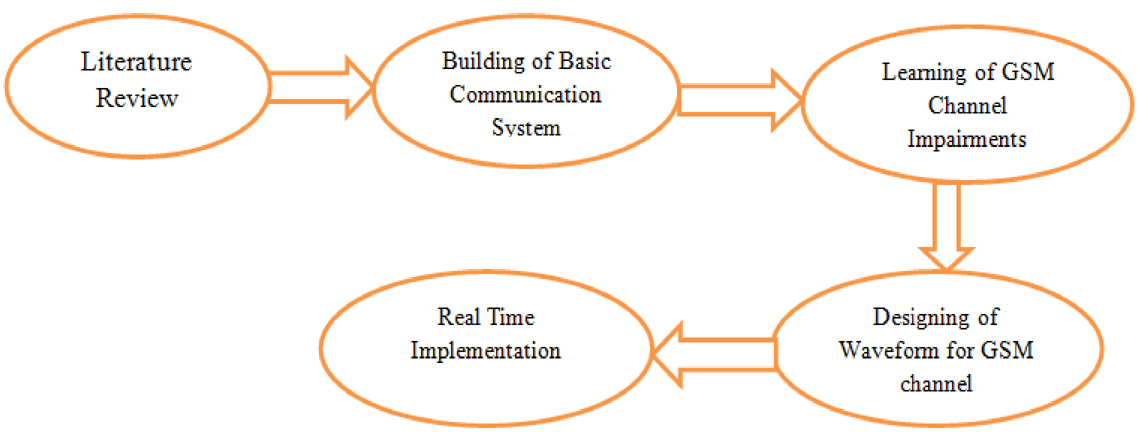

Figure 3.1: Design Layout

3.1 Building of basic Communication System over AWGN Channel

The two basic communication systems that have been designed are phase-shift keying and frequency-shift keying. Understanding their basic behavior in the presence of white Gaussian noise will help us in analysis of the problem in a better way.

3.1.1 Phase Shift Keying

It is a digital modulation scheme that conveys data by changing or modulating the phase of a reference (carrier) signal.

The block diagram of PSK transmitter-receiver is shown below in figure 3.2

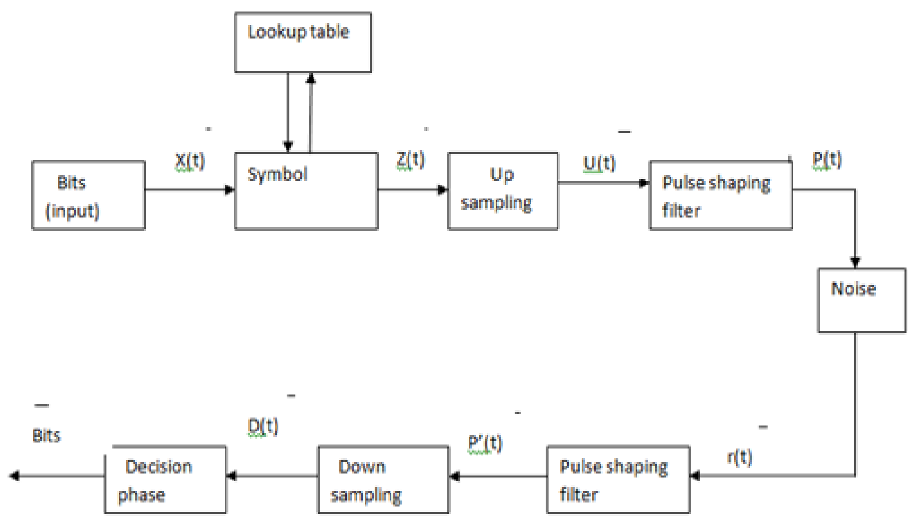

Figure 3.2: PSK transmitter-receiver Block Diagram

3.1.1.1Transmitter

On the transmitter side, a bit of stream is transmitted which are then converted into decimal and encoded with the help of look up table. A look up table is an array that replaces run time computation with a simple array indexing operation. An up sampling with an appropriate or required number of samples is done with the incoming data. Up sampling is interpolation and it produces an approximation of the sequence that would have been obtained by sampling the signal at a higher rate. Pulse shaping is performed for the reason to avoid inter symbol interference at the receiver end. The spreading and smearing of symbols such that the energy from one symbol effects the next ones in such a way that the received signal has a higher probability of being interpreted incorrectly is called Inter Symbol Interference or ISI.

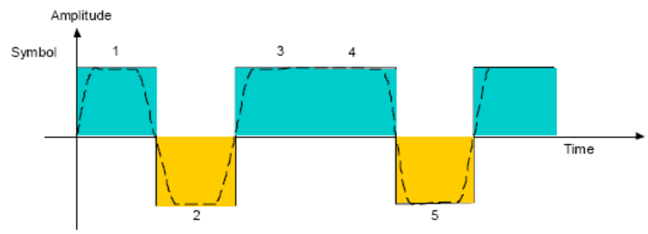

Figure 3.3a: Condition of Transmitted bits before ISI [4]

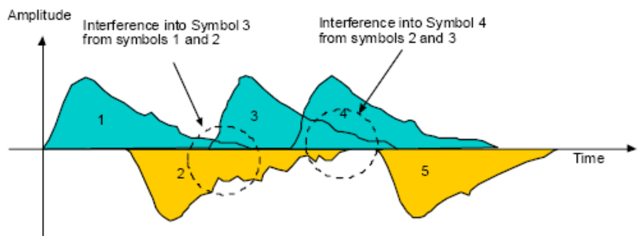

Figure 3.3b: Condition of transmitted bits after ISI [4]

Figure 3.3a shows transmitted data bits which were not encountered by ISI whereas figure 3.3b shows spreading of symbols which clearly depict that these data bits are affected and encountered by ISI.

To counter ISI, a pulse shaping filter called raised cosine is used. Its name stems from the fact that the non-zero portion of the frequency spectrum of its simplest form (α=1) is a cosine function, 'raised' up to sit above the *f* (horizontal) axis. [4]

3.1.1.2 Channel

A channel is a pathway through which data is transmitted from one end to the other end or from transmitter to the receiver. In our project, GSM is the channel and a stream of data bits is to be transmitted over this channel.

A noisy channel distorts the signal and makes it inappropriate for receiver. Thus there should be some mechanisms which reduce the noise or saves the signal from distortion caused by noise.

Additive white Gaussian noise (AWGN) is a basic noise model used in Information theory to mimic the effect of many random processes that occur in nature. The modifiers denote specific characteristics:

- 'Additive' because it is added to any noise that might be intrinsic to the information system.
- 'White' refers to idea that it has uniform power across the frequency band for the information system. It is an analogy to the color white which has uniform emissions at all frequencies in the visible spectrum.
- 'Gaussian' because it has a normal distribution in the time domain with an average time domain value of zero.

AWGN is often used as a channel model in which the only impairment to communication is a linear addition of wideband or white noise with a constant spectral density and a Gaussian distribution of amplitude. The model does not account for fading, frequency selectivity, interference, nonlinearity or dispersion. However, it produces simple and tractable mathematical models which are useful for gaining insight into the underlying behavior of a system before these other phenomena are considered. [6]

3.1.1.3 Receiver

In order to receive the signal, a filter called matched filter is added in order to filter out the noise from the incoming noisy signal. Decimation is the process of reducing the sampling rate of a signal. It is a process which is used to avoid aliasing and hence after passing from filter, a signal is down sampled in order to eliminate the effect of aliasing and to reduce the size of a signal. Decision phase comes at the end where a signal is detected and a decision is taken that which of the bits were received by the receiver.

3.1.2 Frequency Shift Keying

Frequency-shift keying (FSK) is a frequency modulation scheme in which digital information is transmitted through discrete frequency changes of a carrier wave. The simplest FSK is binary FSK (BFSK). BFSK uses a pair of discrete frequencies to transmit binary (0s and 1s) information. With this scheme, the "1" is called the mark frequency and the "0" is called the space frequency.

The block diagram of FSK transmitter is shown below in figure 3.3

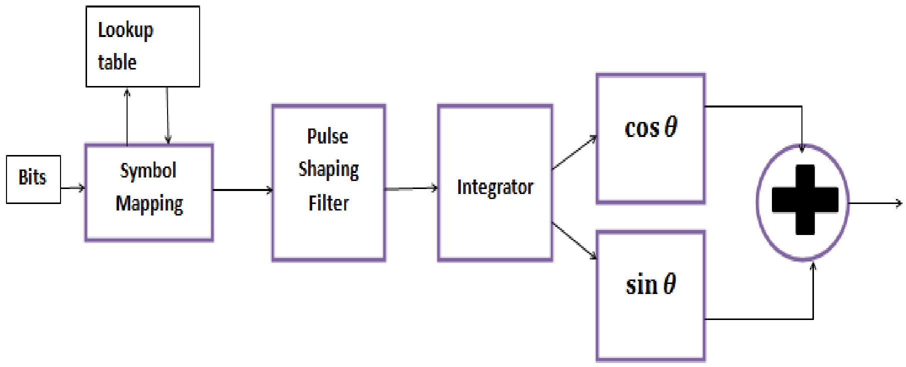

Figure 3.4: FSK transmitter Block Diagram

The block diagram of FSK receiver is shown below in figure 3.4

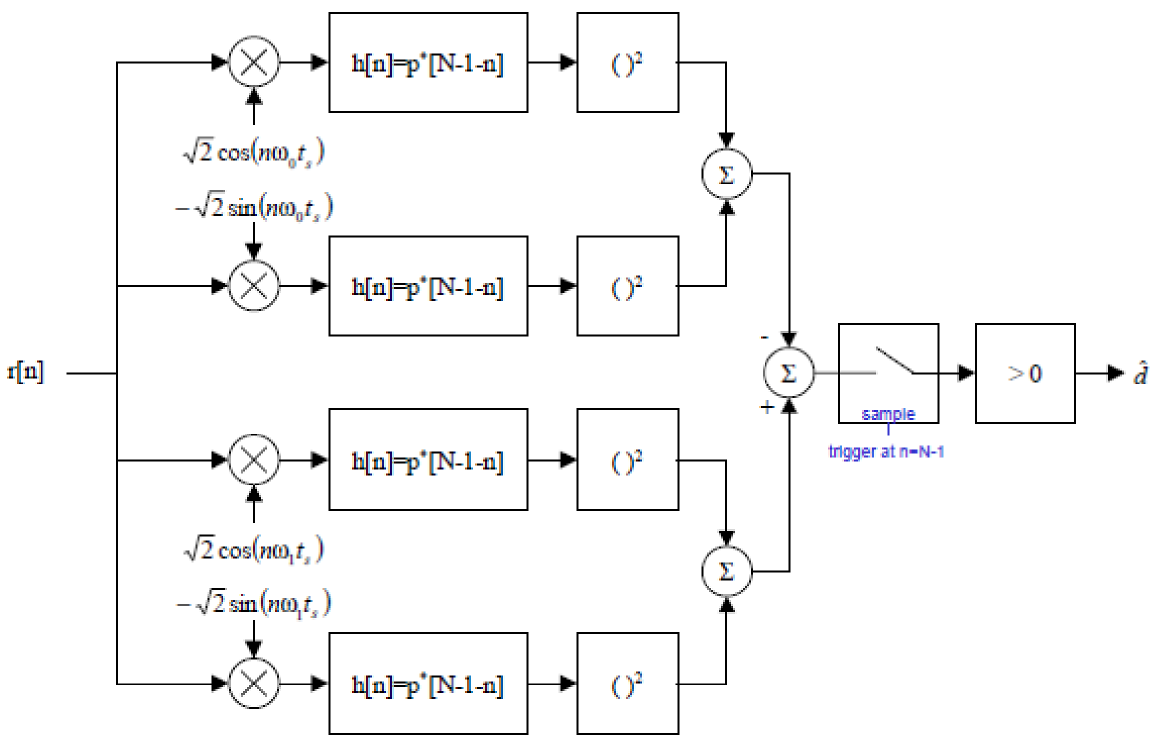

Figure 3.5: FSK receiver Block Diagram [5]

The working of transmitter and receiver is similar as that of PSK except that it detects on the basis of varying frequencies in the receiving signals. A modulation index is another factor that is being used in this type of modulation. The value of the modulation index indicates by how much the modulated variable varies around its un-modulated level [7]. It relates to variations in the carrier frequency:

$$\boldsymbol{h = \frac{\Delta f}{f_m}}\textbf{[7]}$$

Where f_m is the highest frequency component present in the modulating signal $x_m(t)$, and Δf is the peak frequency-deviation—i.e. the maximum deviation of the instantaneous frequency from the carrier frequency.

3.1.2.1 Differential frequency shift keying

It is a technique of transmitting two frequencies derived from a single base frequency f_{base}. A suitable value of delta δ is decided after simulation which is then subtracted from f_{base} to encode a sinusoidal waveform of 1 and added to f_{base} in order to have a sinusoid of 0. Referring to Hermes, results are good when value of δ is taken to be (10-25) percent of the base frequency.

We have applied the technique of differential frequency shift keying in our project. There is a major flaw in using FSK as two absolute frequencies (f0 and f1) are used: f0 for transmitting 0 and the other sinusoid of frequency f1 for transmitting 1. In order to distinguish between the symbols, f0 and f1 are taken far apart. It leads to a non-voice like waveform due to abrupt and large changes in frequency over short time scale. Consequently voice codec will distort and block the audio waveform. In order to please the codec we need to keep the symbols close enough in the frequency domain.

3.2 Learning of GSM Channel Impairments:

In a wireless environment, the impairment of communication channels can affect significantly the performance of broadband wireless system. Therefore the learning of a GSM channel is important in order to transmit data over a voice channel. [1]

GSM voice channel is effectively a band-limited nonlinear channel with memory, which is designed for voice-like signals. To allow greater channel capacity, the GSM voice codec extracts the parameters characterizing speech according to the corresponding speech model, and only these parameters are sent over the air. But here the problem is that data is sent over a voice channel and hence the characteristics of data that is being transmitted must be like the characteristics of voice so that the channel would not block or stop it. The parameters characterizing data must be adjusted and set so that it would be received at the receiver without being distorted. However in achieving it, the error rate may not remain same as it was achieved in case of voice transmission. [3]

GSM voice channel is specifically designed for voice communication. Transmitting data over that channel would be a difficult and a tough task. Understanding the characteristics of GSM voice channel would help to a great deal.

Direct transmission of data over the GSM voice channel would not be beneficial as GSM voice codecs block and distort the data completely and to an extend that would un able the user at the receiver end to take any decision out of it.

The solution of this problem is to make a waveform whose characteristics would resemble speech but whose information would depict data encoded in it. The encoding scheme used is transcoding scheme which is to convert to a different format of similar or like quality to gain compatibility with another program or application.

The basic parameter which characterizes voice is fundamental frequency. Estimating the frequency shift at the receiver end will help us to verify which frequency is best for transmitting data encoded in a voice like waveform.

3.3 Designing of Waveform for GSM channel

This phase of design will include designing of a waveform that can communicate over GSM channel. An algorithm is used to build the desired signal as a set of waveforms (symbol dictionary). The modem takes these pre-generated symbols and maps the input data onto them. The symbols are concatenated and sent over the air via the GSM unit. On the receiver side, the symbol outputs of the GSM unit are converted back to data. The data estimate is the index of the symbol from the codebook that maximizes the inner product with the received symbol. The actual over the air transmission is handled by the existing GSM system with its own modulation, forward error correction (FEC), forward error detection (FED), and equalization. Thus, if their correct functioning is assumed, the major source of errors is the voice codec itself. [3]

GSM channel has a problem of shifting the frequency thus causing frequency deviation at the receiver end. Sometimes this deviation approaches to such an extent that it becomes difficult for the receiver to take a suitable decision and hence bit error rate increases as a result. In order to avoid this situation, waveforms of different frequencies will be transmitted one after the other and decision will be taken based on whether the received frequency is greater than the previous frequency or lesser than it. In short a suitable range of delta will be decided through testing and optimization of waveforms will be carried out.

3.3.1 Conversion of character input into Binary and vice versa

As this project allows the user at one end to send file of any size at the other end over the GSM voice channel so there is a need to convert characters into their corresponding binary numbers. Each character has its own ASCII value which makes it possible to have binary representation of every character. For the generation of a waveform against each bit that has to be transmitted over the channel we first took an input from the user who wants to transmit. That input is taken in character form which may include alphabets, numbers and symbols.

Just take an example of letter **"a"** which has an ASCII value of **"97"**(decimal value). Binary conversion of 97 would be **"01100001"**. In this way for every word we have a stream of zeros and ones. Keeping in mind that ASCII values are case sensitive so one should be careful while writing a text for transmission if he wants to check binary representation of each letter he typed.

At the receiver end reverse technique is applied because the output should be in the form of character so that it could be readable for the user at the receiving end. Therefore binary to ASCII conversion is applied at the other end.

3.3.2 CRC check

A cyclic redundancy check (CRC) is an error-detecting code commonly used in digital networks and storage devices to detect accidental changes to raw data. Blocks of data entering these systems get a short check value attached, based on the remainder of a polynomial division of their contents. On retrieval the calculation is repeated, and corrective action can be taken against presumed data corruption if the check values do not match. It is usually 16 or 32 bits in length.

For error detection and correction, CRC generator will generate a 32 bit code in our project which will be added before data packet thus making a packet of length 240 in which 208 bits are of information bits. At the transmitter end CRC generator code is written and at the receiver end CRC detector code is written in order to detect and correct errors in the transmitting file.

3.3.3 Making a packet of finite length

A data packet should be of finite length so that errors in a transmitting message could be easily detected. Packet consists of two parts: first part comprises of information bits and CRC (correction and redundancy code) bits make the second part of a packet.

During our simulations we have fixed 208 bits for information and 32 CRC bits are added to make a total of 240 bits. But one can have a packet of any finite length by changing the length of information bits as well as CRC bits.

One thing must be kept in mind that all packets should be of same length otherwise correct detection at the receiver end would not be possible. Padding is done in order to make all packets length constant.

3.3.4 Transcoding

Transcoding is the direct analog-to-analog and digital-to-digital conversion of one encoding into another.

In this project, we have transcoded the inputstream into another representation before passing it to the modulation layer to be converted into sounds. The goals ofthe transcoding layer are summarized as follows:

• The stream fed by the transcoder to the modulator should be such that the modulator is able to guarantee a fixed fundamental frequency.

• The modulation layer should use a minimum number of unique frequencies over the air; this minimizes the distortions. Also, the modulator should keep the value of f within acceptable limits.

• The receiver should be able to recover the original bit stream with a very high probability, even in the face of bit insertions, deletions and flips in the post-transcoded stream of data that is modulated and sent over the air.

Transcoding of the data is performed using a very simple algorithm. It is a 1/2 code, meaning that it takes in one input bit and produces two output bits, as shown below:

0 →01

1 → 10

3.3.5 Transmission of Analog Signal

Final form of a signal which is going to be transmitted over the GSM channel is now in the form of an analog signal. It has been transcoded and modulated after passing through FSK modulator. It has the characteristics of speech thus allowing it to pass through the GSM channel which blocks and distorts the waveforms other than speech badly. In our simulations, this audio signal has sampling frequency equal to 8000 Hz.

3.4 Real Time Implementation

The last phase of this project design is the real time implementation of it. A data file is transmitted from a mobile or any transmitting device and then it is received at the receiving mobile or device. This data will pass through transmitter and receiver circuit as well as through GSM channel.

We have created a GSM environment in our laptop and simulated our codes such that user at transmitter end will write a message on notepad and received characters will be seen on the MATLAB after passing through GSM environment. We have seen some flipping of bits but because of redundancy one can easily understand what was being transmitted.

Chapter 4 Analysis and Evaluation

4.1 Frequency Selection using Sound Scope

We began to use software which is named as "Sound Scope" in which we have made our simulations and interpreted our results based on those simulations. A simple snap shot of this software is shown below:

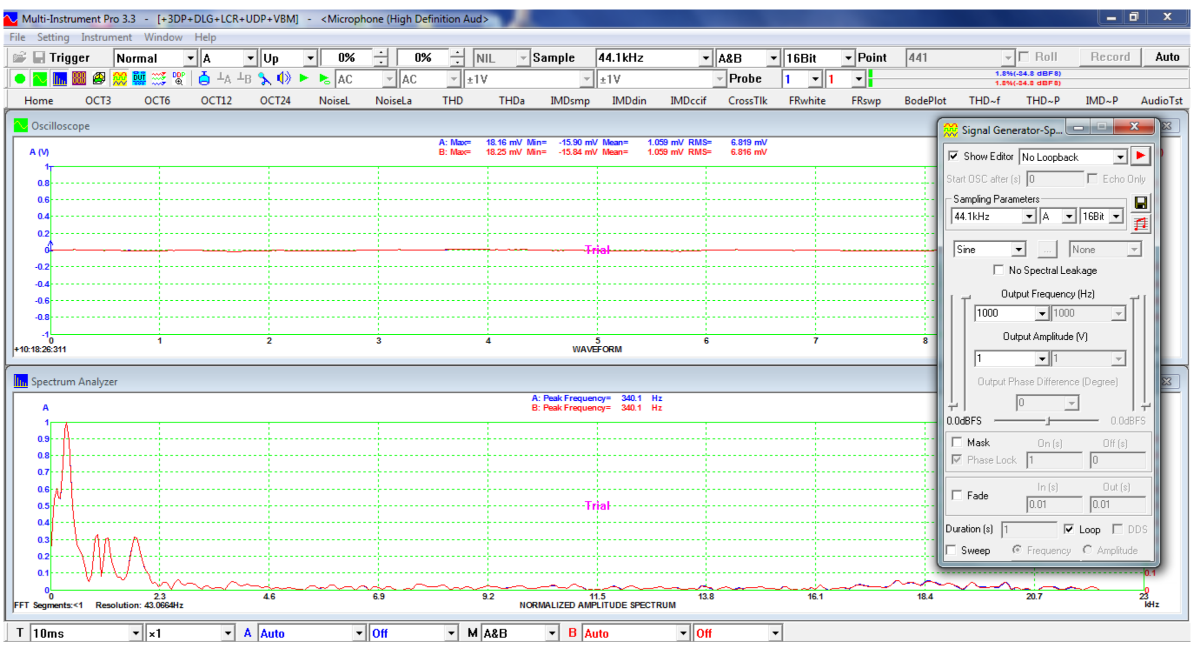

Figure 4.1: Sound Scope

First of all, simulations were done between two laptops and results were compared. From one laptop, we transmitted a sine wave and by connecting the two laptops with a stereo cable, we received that sine wave at another laptop along with some distortions because of transmission line losses. The validity of change in gain was checked by simulating at different volumes of the laptop.

After successful testing between two laptops, simulations were done over the GSM voice channel. A live call was made and mobile phones were connected to laptops via stereo cable. Testing is performed at different frequencies. Starting from 500 Hz and ending at 3500 Hz, different gains and attenuations were seen and out of these frequencies, we select a frequency with least attenuations and highest gain. Therefore, we select 2240 Hz as an optimum frequency for GSM voice channel. Following table shows the frequencies at which testing are being performed:

Sr.no	Central Frequency (Hz)	Gain at Reciever (dB)	Attenuation
1	500	95	more
2	1000	95	less
3	1500	97	more
4	2000	98	very less
5	2500	98	more
6	3000	80	less
7	3500	78	more

Table 4.1: Table of frequencies

At 2000 Hz and 2500 Hz, maximum gain is achieved but with respect to attenuations, the response is best at 2000 Hz.

At Transmitter:

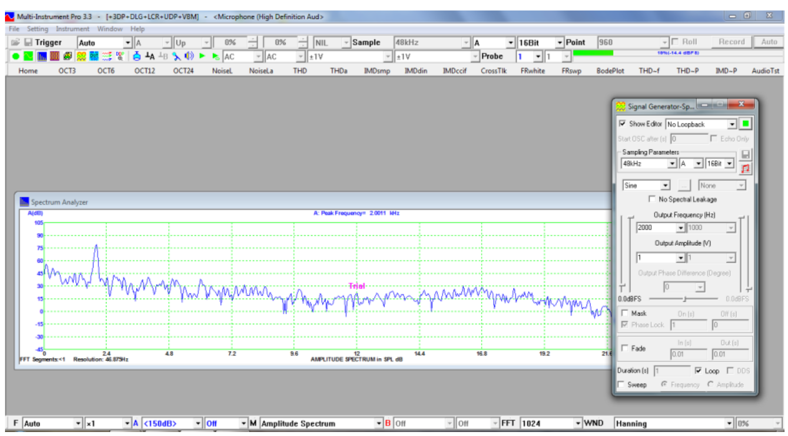

Figure 4.2a: Transmission at 2000 Hz

At Receiver:

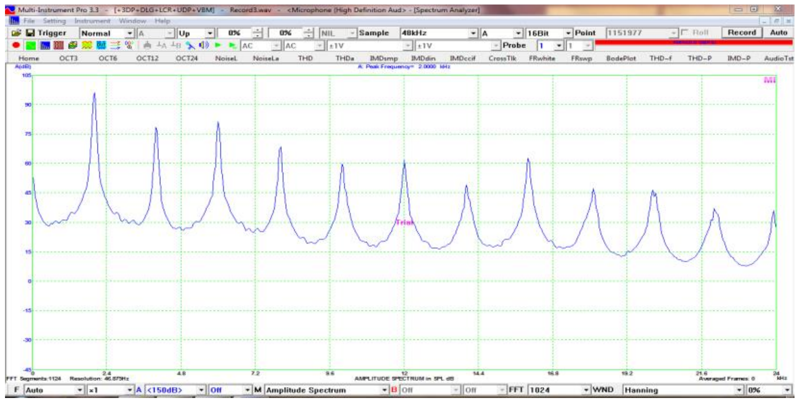

Figure 4.2b: Reception at 2000 Hz

The algorithms used for transcoding, modulation, and demodulation and decoding are all quite simple, since we would like them to run directly on cellular phones, which are not necessarily high-end. However, since we are still prototyping and testing the protocols, these algorithms are currently implemented on regular desktop computers that interface with and use the cellular phones as the underlying physical communication channel. We are in the process of implementing the algorithms on the Android[6] platform.

4.2 Simulations

The simulations of Hermes approach were also carried out in MATLAB. At the transmitter end a data file is transformed into speech like waveform and the resultant audio signal from the modulator was sent to the other end through the voice channel using two mobiles. At the receiver end speech like data waveforms were demodulated and output bit streams were received. The simulations mainly focused on the transmitted and received waveform as well as the transmitted and received frequencies.

4.2.1 At the Transmitter end

- **Take an input or file written in characters having ASCII values**.

  For data to be transmitted, write your text in notepad .Each character has anASCII value associated with it. There is no limitation in the number of characters in the file. The file should have '.txt' format.

  As an example we have sent a file from the transmitter side. Figure is shown below:

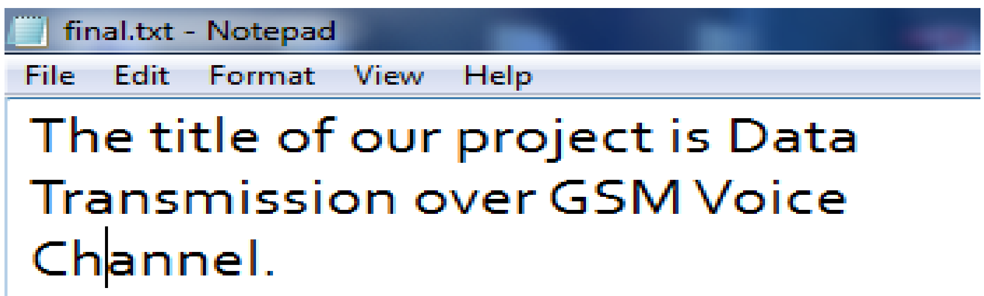

Figure 4.3: Text file from transmitter end

- **Making an information bit of length 208.**

Padding is done for the purpose of making a packet length constant.The length is made constant to 208 bits by padding 1's at the end of information bits.

Following figure shows the result of transmitted text file:

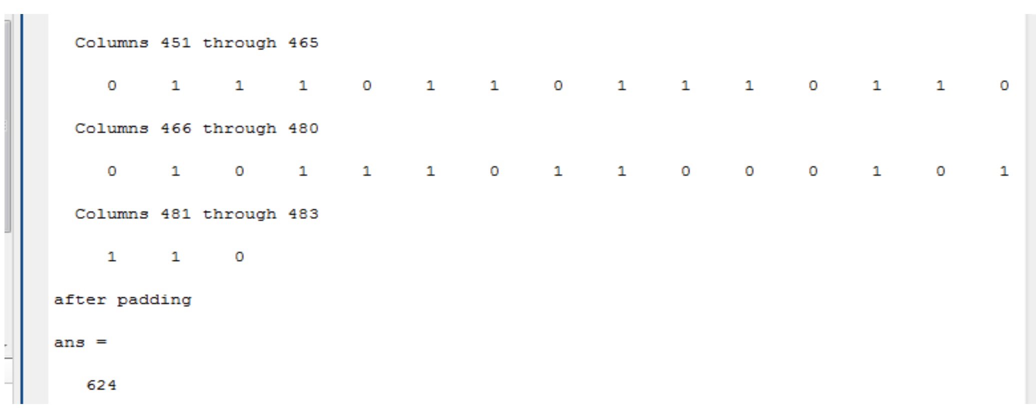

Figure 4.4: Transmitted text after padding

The figure showed that there are 3 information bit packets for the transmitted text file. As 624/208 = 3 which means that each packet contains 208 information bits after padding.

❖ CRC generator

For error detection and correction, CRC generator will generate a 32 bit code in our project which will be added before data packet thus making a packet of length 240 in which 208 bits are of information bits. At the transmitter end CRC generator code is written which generates 32 bits of CRC for the transmitting file.

Following figure shows the data packet after applying CRC generator:

```
codeword length

ans =

   720
```

Figure 4.5: Transmitted data packet with 32 bit code

In total 96 bits were added in all three information bit packets. Each data packet has 32 bit code before information bits generated by CRC generator.

❖ Transcoding

We have transcoded the binary bitstreams into another representation before passing it to the modulation layer to be converted into speech like waveforms. Transcoding of the data is performed using a very simple algorithm. It is a 1/2 code, meaning that it takes in one input bit and produces two output bits.

The following figure shows the algorithm for transcoding:

$$\boxed{\begin{array}{c} 0 \rightarrow 01 \\ 1 \rightarrow 10 \end{array}}$$

Figure 4.6: Transcoding Algorithm

Following figure shows the code of ½ transcoding at transmitter end.

```
for i=2:length(codeword)
    if codeword(i)==0
       transcodedMatrixAfterCRC=[transcodedMatrixAfterCRC 0 1];
    else
        transcodedMatrixAfterCRC=[transcodedMatrixAfterCRC 1 0];
    end
end
disp('trascoded');
```

Figure 4.7: Transcoding Code at Transmitter side

- **Wave file is generated by applying differential FSK modulator on binary packet of length 240 bits.**

After transcoding, differential FSK modulation technique is applied to transmitting packet. Following figure shows result after applying this technique:

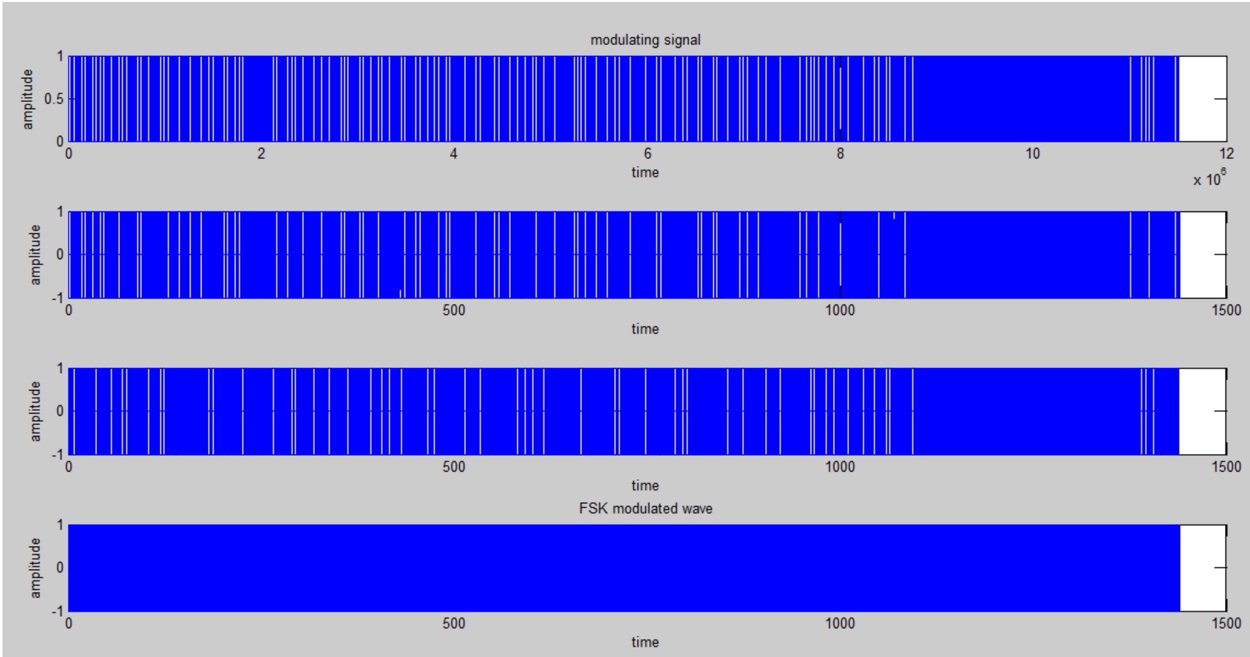

Figure 4.8: FSK modulated wave

Second graph shows modulation of only zeros whereas third graph represents modulation pattern of ones. We can see modulation of both zeros and ones in the third graph but because frequency is high enough so we cannot differentiate ones and zeros by looking at graph only. Last graph shows the actual modulating FSK signal which is going to be transmitted over the GSM voice channel.

Pseudo code for Differential FSK applied in our project is as follows:

Given: base frequency f_{base}, delta frequency δ.
$f = f_{base}$
for each bit b in the input string do
  if $b = 0$ then
    $f = f - \delta$
  else
    $f = f + \delta$
  end if
  Generate a sinusoid of frequency f
end for

Figure 4.9: Pseudo code for Differential FSK modulator

A base frequency is selected through simulations on Sound Scope and delta is generally 10-25 percent of the base frequency. For bit 0, transmitting frequency according to pseudo code would be calculated by subtraction delta from base frequency. Similarly for bit 1, delta is added to base frequency to obtain transmitting frequency of bit 0.

4.2.2 At the Receiver end

- **FSK Differential demodulator**

At the receiver end, first of all FSK differential demodulation technique is applied. Now the wave file is given to demodulatorand binary data packet is generated.

- **Anti-transcoding**

Now reverse transcoding is done at demodulator output.The receiver should be able to recover the original bit stream with a very high probability. Following technique is applied for reverse transcoding:

$$01 \longrightarrow 0$$

$$10 \longrightarrow 1$$

Following figures shows the algorithm for reverse-transcoding:

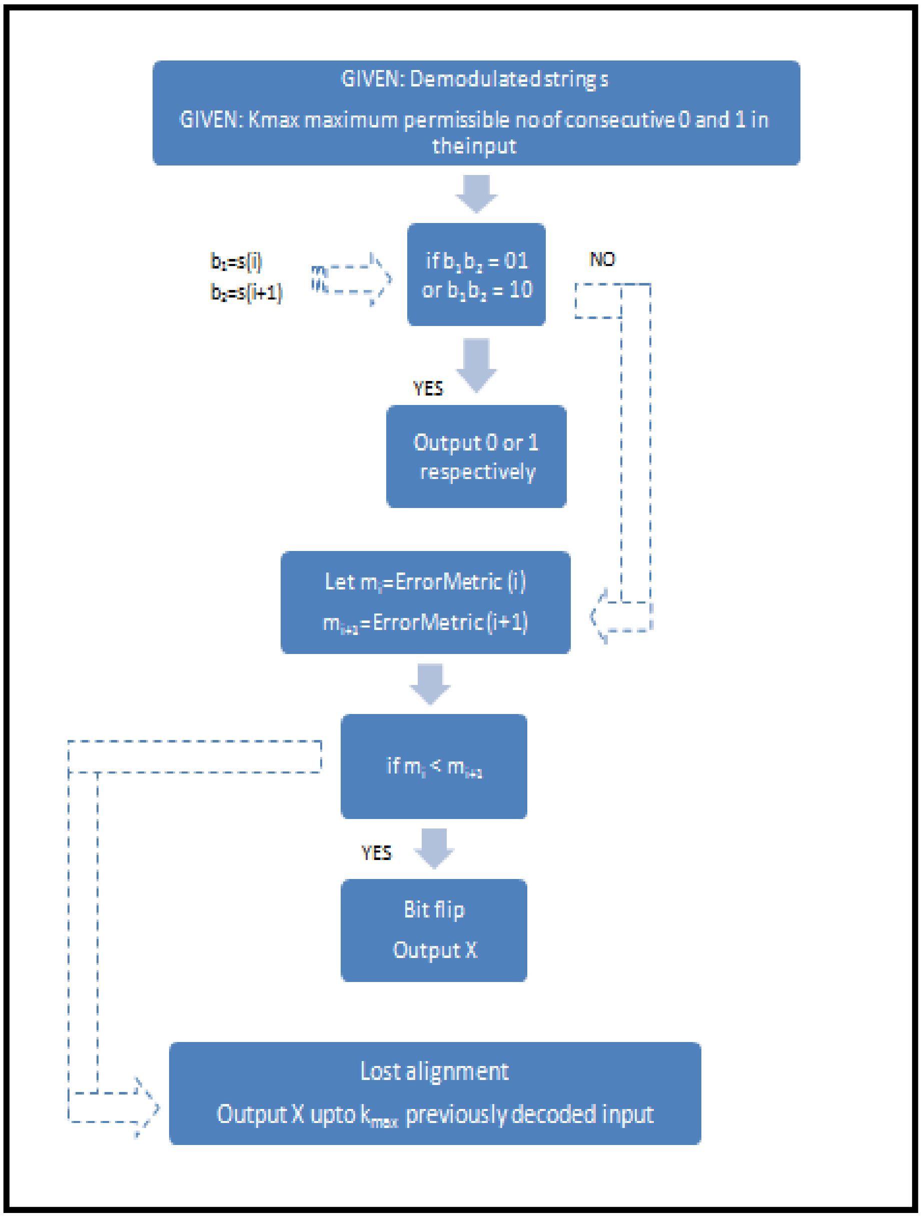

Figure 4.10 a: Reverse Transcoding Algorithm

Given: Demodulated string s, consisting of post-transcoded bits.
Given: k_{max_rep}, the maximum permissible number of consecutive 0s or 1s in the input.
for $i = 1; i \le length(s); i = i + 2$ **do**
  $b_1 = s[i]$
  $b_2 = s[i+1]$
  if $b_1 b_2 = 01$ or $b_1 b_2 = 10$ **then**
    Output 0 or 1, respectively.
  else
    There is either a bit flip or an insertion/deletion error.
    $m_i = ErrorMetric(i)$
    $m_{i+1} = ErrorMetric(i+1)$;
    if $m_i \le m_{i+1}$ **then**
      The alignment is correct, but there is a bit flip.
      Output X
    else
      We have lost alignment by one bit
      $i = i + 1$
      Output X on up to k_{max_rep} previously decoded output bits.
    end if
  end if
end for

Figure 4.10 b: Pseudo code for Reverse Transcoding

A string of post transcoded bits are received at the receiver end. Anti-transcoding is performed in order to reach to the final result. Two bits in series are checked for error detection. If there is no error, the transcoded bits are decoded back into 0 and 1. However, an error metric variable is introduced in case of an error which will check whether we have lost an alignment or there is a flip in bit.

- **CRC detector**

CRC detector will detect errors in a packet.At the receiver end CRC detector code is written in order to detect and correct errors in the transmitting file.

Following figure shows the bits which are detected at demodulator after applying CRC detector.

```
>> hermesdemod('hermes1.wav')

ans =

     11520099


ans =

     11520099


ans =

   720
```

Figure 4.11: CRC detector Output

Therefore the three transmitted data packets (720 bits) will be received here.

- **Conversion of ASCII values into characters.**

The ASCII values generated in the above step are converted into characters. Here we have received the original text that was transmitted from the transmitter side.

Following figure shows the received text file with some bits flipped:

```
ans =

The title og our project is Data Transmission over GSm Voice ChaoneL>
```

Figure 4.12: Received text file

Comparison of original text file and received text shows that the transmitted bits have undergone some bit flips which is clearly evident from the words like “og” which was originally “of” and “chaonel>” which was originally “channel”.

Chapter 5 Recommendations and Conclusion

5.1 Recommendations for future work

5.1.1Extensions

The usage of data modems have substantially increased during the last 5 years. The need for high priority data has increased so has the technology which has resulted in catering this need quite well as well.

A lot of research papers have already been written on it still there is a need of implementing a MATAB code on DSP-SDK. It requires conversion of MATLAB code into C language. Moreover understanding of DSK-KIT is important for this extension.

5.1.2 Improvements

CRC generator and detector however removes error to a greater extend but still we see flipping of a bit which changes our end result.The solution of this problem is to use Viterbi decoder whose decoding algorithm uses two metrics: the branch metric (BM) and the path metric (PM). The branch metric is a measure of the distance between what was transmitted and what was received, and is defined for each arc in the trellis. In hard decision decoding, where we are given a sequence of digitized parity bits, the branch metric is the Hamming distance between the expected parity bits and the received ones.

5.2 Conclusion

5.2.1 Overview

The problem of data transmission over unknown voice channels has several important implications on enabling new mobile services in rural developing regions where no data connectivity solution exists today. A real time data modem with the aim of data transmission over GSM voice channel has been implemented. This modem modulates and demodulates data on GSM AMR voice channel which is nonlinear channel with memory.

5.2.2 Objectives Achieved

Following objectives have been met:

a) Characterization of a GSM voice channel and its distortions have been understood and handled well.

b) Designing of a speech like waveform which would not be distorted or blocked by the channel. Encoding of data in a waveform whose characteristics and parameters are speech-like have been done successfully.

c) Real time data transfer during a voice call session between two GSM nodes has been accomplished. One can send text of any length over a GSM voice channel to the receiver at the other end during a voice call session.

5.2.3 Applications

a) While SMS is available as a possible data channel, it is extremely low-bandwidth, where every SMS message is limited to 140 bytes and furthermore, the cost per bit is quite high. In most areas, a single SMS costs $0.05 to $0.25 which is comparable to per-minute voice call rates. Hence, if there exists an efficient data connectivity service over the cellular voice channel, one can enable a new class of mobile data services and also significantly reduce the cost per bit for data connectivity.

b) In situations where real-time, high priority, and low bit rate data channel is needed, this data modem could be useful.

c) It may be used for encrypted data and voice transmission, online text message, online multimedia message, real-time monitoring system, etc.

d) This modem can be used to develop secure voice communication and data transmission to achieve end-to-end security, because in GSM network, security is provided by network operator, not by end user.

e) It is useful in remote and rural areas where no data connectivity exists. Using an existing GSM voice channel one can avail this opportunity of sending data in addition to voice.

5.2.4 Limitations

No doubt this data modem is not a replacement of GPRS (general packet radio service), in situations where high bit data rate channel is required it is not a viable solution. For such requirements GPRS works well as compared to data transmission over GSM voice channel.New technologies like 3G and 4G offers enhanced bit rates as compared to GSM voice channel. It offers faster data uploads and downloads.

6. Bibliography

Following are the papers regarding our project which we went through:

6.1 Simon, M.K. and Divsilar, D., “Maximum-likelihood Block Detection of Non coherent
Continuous Phase Modulation,” IEEE Transactions on Communications, Jan. 1993.

6.2 University of Hull, “Digital Modulation and GMSK”.

6.3 Jan Cernock´y, ValentinaHubeika, “Speech coding II”.

6.4 JuttaDegener, “Digital Speech Compression”, copyright 1994, December 1.

6.5 Eric S. Otto and Phillip L. De León, “Digital CPFSK transmitter and non-coherent receiver/demodulator implementation”, New Mexico State University, Center for Space Telemetry and Telecommunications.

6.6 Hazem Ismail Abdel Aziz Ali, Mohammad NazrulIshlamPatoary, “Design and Implementation of anAudio Codec (AMR-WB) using Data Flow Programming Language CAL in the Open DF Environment”, Halmstad University.

6.7 David Brandon,JeffKeip,(January 2012), Efficient FSK/PSK Modulator Uses Multichannel DDS to Switch at Zero Crossings[Online].Available: http://www.analog.com/library/analogdialogue/archives/46-01/zero_crossing.html

6.8 Richard Metson. (2003, November 7). Sorting through GSM codecs [Online]. Available: http://www.eetimes.com/document.

6.9 LaDue, C., Sapozhnykov, V., and Fienberg, K. A data modem for GSM voice channel. Vehicular Technology, IEEE Transactions on 57, 4 (july 2008), 2205 –2218.

6.10 Bernard Sklar, “Band pass modulation and demodulation detection” in Digital Communications, 2nd edition, California, pp. 175

6.11 Jean Paul M.G. Linnartz (1993). Additive White Gaussian Noise [Online]. Available: http://www.wirelesscommunication.nl/reference/chaptr05/digimod/awgn.htm.

- Similar project done at MCS:

Data transmission over GSM channel by NC Zahra Arshad, NC Hassnain Ali Chohan, PC Omar Luqman, TCC-22, Military College of Signals, NUST.

- Similar project done outside MCS:

  i. Dhananjay, A. Sharma, M. Paik, J. Chen, T. Karthik, Kuppusamy, J, Li, L. Subramanian “Hermes: Data Transmission over Unknown Voice Channels”, Courant Institute of Mathematical Sciences, New York University, (2010).

  ii. LaDue, C., Sapozhnykov, V., and Fienberg, K., “A data modem for GSM voice channel”, Vehicular Technology, IEEE Transactions on 57, (2008, july 4), 2205 – 2218.

iii. Katugampala, N., Al-Naimi, K., Villette, S., and Kondoz, A., " Real time data transmission over GSM voice channel for secure voice and data applications", In 2nd IEE Secure Mobile Communications Forum (2004).

7. References

[1]; Dhananjay, A. Sharma, M. Paik, J. Chen, T. Karthik, Kuppusamy, J, Li, L. Subramanian. Hermes: Data Transmission over Unknown Voice Channels. Courant Institute of Mathematical Sciences, New York University, (2010).

[2]; Katugampala, N., Al-Naimi, K., Villette, S., and Kondoz, A.Real time data transmission over GSM voice channel for secure voice and data application. In 2nd IEE Secure Mobile Communications Forum ,(2004).

[3]; LaDue, C., Sapozhnykov, V., and Fienberg, K. A data modem for GSM voice channel. Vehicular Technology, IEEE Transactions on 57, (2008, july 4), 2205 –2218.

[4];Charan Langton, "Inter symbol interference and raised cosine filtering", copyright 2002.

[5];DavidBrandon,JeffKeip,(January 2012),Efficient FSK/PSK Modulator Uses Multichannel DDS to Switch at Zero Crossings[Online].Available:http://www.analog.com/library/analogdialogue/archives/46-01/zero_crossing.html

[6];Jean Paul M.G. Linnartz. (1993). Additive White Gaussian Noise [Online]. Available: http://www.wirelesscommunication.nl/reference/chaptr05/digimod/awgn.htm

[7];BernardSklar, "Band pass modulation and demodulation detection" in Digital Communications, 2nd edition, California, pp. 175

Appendix A

MODULATOR END (AT TRANSMITTER)

```
function [ audio ] = hermes(filename)
%UNTITLED2 Summary of this function goes here
%   Detailed explanation goes here
% 01111110 escape sequence
% 26*8 = 208 bits in one frame + 32 bits of CRC = 240 bits = 30 bytes
+8
% bit preamble =00000000
fileID = fopen(filename,'r');
Intro = textscan(fileID,'%s',10,'Delimiter','\n');
matrixWithData=[];
a=dec2bin(Intro{1}{1});
sizeOfa = size(a);
a= a';
a =reshape(a,1,sizeOfa(1)*sizeOfa(2));
matrixWithData=a;
%Converting the data into binary from ascii from the test1.txt file
for i=2:length(Intro{1})
    a = dec2bin(Intro{1}{i});
    a = reshape( a.', 1, numel(a) );
   matrixWithData=[matrixWithData a];

end
matrixWithData = logical(matrixWithData(:)-'0')';
disp('matrix with data length');
matrixWithData
matrixWithData= [matrixWithData ones(1,208-
rem(length(matrixWithData),208))];
disp('after padding');
length(matrixWithData')
%Making them into packets with 32 bit-CRC
hGen = comm.CRCGenerator([32 26 23 22 16 12 11 10 8 7 5 4 2 1 0],
'ChecksumsPerFrame',length(matrixWithData)/208);
codeword=step(hGen,matrixWithData');
disp('codeword length');
length(codeword)

%transcoding 1/2 code transcoding starting
transcodedMatrixAfterCRC=[];
if(codeword(1)==0)
    transcodedMatrixAfterCRC=[0 1];
else
    transcodedMatrixAfterCRC=[1 0];
end

for i=2:length(codeword)
    if codeword(i)==0
       transcodedMatrixAfterCRC=[transcodedMatrixAfterCRC 0 1];
    else
        transcodedMatrixAfterCRC=[transcodedMatrixAfterCRC 1 0];
```

```
    end
end
disp('trascoded');

 [audio] = binary2fsk2(transcodedMatrixAfterCRC);
 audiowrite('hermes1.wav',audio,8000);
end



function [ binout ] = fsk2binary2( audio_input )
%UNTITLED Summary of this function goes here
%   Detailed explanation goes here
l1=length(audio_input);
basefrequency=2250;
delta=340;
% Enter the two frequencies
% Frequency component for 0 bit
f1 = basefrequency+delta;

% Frequency component for 1 bit
f2 = basefrequency-delta;
t1=0:0.01:.99;
r1=sin(2*pi*f1*t1);
r1=fliplr(r1);
l2=length(r1);
l3=l1+l2-1;
u=fft(audio_input',l3);
length(u)

v=fft(r1,l3);
length(v)
k1=u.*v;
k11=ifft(k1,l3);
r2=sin(2*pi*f2*t1);
r2=fliplr(r2);
w=fft(r2,l3);
k2=u.*w;
k22=ifft(k2,l3);
k=k11-k22;
z=1;
while z*8000<=length(k)
t(z)=k(z*8000);
if t(z)>0.25
s(z)=1;
else
s(z)=0;
end
z=z+1;
end
ii=1;
for i=1:2:length(s)
    if s(i)==0 && s(i+1)==1
        ss(ii)=0;
    end
    if s(i)==1 && s(i+1)==0
```

```
            ss(ii)=1;
        end
        if s(i)==1 && s(i+1)==1
            ss(ii)=1;
        end
         if s(i)==0 && s(i+1)==0
            ss(ii)=0;
        end
 ii=ii+1;
end
length(ss)
 hDetect = comm.CRCDetector([32 26 23 22 16 12 11 10 8 7 5 4 2 1 0],
'ChecksumsPerFrame',floor(length(ss)/208));
    [tx, err] = step(hDetect, ss');

str_x = num2str(tx);
str_x(isspace(str_x)) = '';
binout=convertor(str_x');
end
```

DEMODULATOR END (AT RECEIVER)

```
function [ output_args ] = hermesdemod( input_args )
%UNTITLED Summary of this function goes here
%   Detailed explanation goes here
[y,Fs] = audioread(input_args);
output_args=fsk2binary2(y);

end


function [ FSK_signal ] = binary2fsk2( bit_stream )
%UNTITLED Summary of this function goes here
%   Detailed explanation goes here

x=bit_stream;
l=length(x);
for i=1:1:l
m(((i-1)*8000)+1:i*8000)=x(i);
end
figure;
subplot(4,1,1);
plot(m);
xlabel('time');
ylabel('amplitude');
title('modulating signal');
f=8000;
basefrequency=2250;
delta=340;
% Enter the two frequencies
% Frequency component for 0 bit
f1 = basefrequency+delta;

% Frequency component for 1 bit
f2 = basefrequency-delta;
```

```
t=0:(1/f):(l-(1/f));

c1=sin(2*pi*f1*t);
y1=m.*c1;
subplot(4,1,2);
plot(t,y1);
xlabel('time');
ylabel('amplitude');
for j=1:l
if x(j)==1
x(j)=0;
else x(j)=1;
end
m1((j-1)*8000+1:j*8000)=x(j);
end
c2=sin(2*pi*f2*t);
y2=m1.*c2;
subplot(4,1,3);
plot(t,y2);
xlabel('time');
ylabel('amplitude');
y=y1+y2;
subplot(4,1,4);
plot(t,y);
xlabel('time');
ylabel('amplitude');
title('FSK modulated wave');
r=randn(size(y));
FSK_signal=y;

figure;
subplot(3,1,1);
plot(FSK_signal);
xlabel('time');
ylabel('amplitude');
title('noise added FSK signal');
```