# TRANSMISSION OF SPEECH THROUGH LAN USING IP

**By**


**ADITYA KUMAR SINGH-051056**

**AAKASH SHARMA-051055**

**NIKHILESH CHAUHAN-051044**




**MAY-2009**

**Submitted in partial fulfillment of the Degree of Bachelor of Technology**


**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**

**JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY-WAKNAGHAT**

# CERTIFICATE

This is to certify that the work entitled, "Transmission of speech through LAN using IP" submitted by Aditya Kumar Singh, Aakash Sharma, and Nikhilesh Chauhan in partial fulfillment for the award of degree of Bachelor of Technology in Electronics and Communication of Jaypee University of Information Technology has been carried out under my supervision. This work has not been submitted partially or wholly to any other University or Institute for the award of this or any other degree or diploma.

Dr. Vinay Kumar

Dr. Sunil V. Bhooshan

# ACKNOWLEDGEMENT

There were many steps to take to complete this work. Each step forward would have been impossible without the help of many people. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

First of all, we are deeply grateful to our advisor, **Dr. Vinay Kumar.** We really appreciate the patience he had towards our project. We acknowledge his advice and guidance throughout the year.

We take this opportunity to express our sincere gratitude to the Head of department (Electronics and Communication Engineering) **Dr. Sunil V. Bhooshan** for all his support and valuable inputs.

We would also like to express our profound gratitude to our panelist **Dr. D.C Kulshreshtha, Ms. Neetu Singh, Ms. Shruti Jain** for their constant and valuable suggestions during the development of this project.

The successful compilation of final year project depends on the knowledge and attitude inculcated in the total length of course. So we want to express our sincere gratitude to all the faculties who taught us during the four years of B.Tech.

Aakash Sharma
**(Aakash Sharma)**

Nikhilesh Chauhan
**(Nikhilesh Chauhan)**

Aditya Singh
**(Aditya Kumar Singh)**

# TABLE OF CONTENTS

## LIST OF FIGURES

## LIST OF ABBREVIATIONS

1. A/D ..........................................................................Analog to Digital
2. LAN ..........................................................................Local Area Network
3. IP ..........................................................................Internet Protocol
4. GUI ..........................................................................Graphical User Interface
5. PCM ..........................................................................Pulse Code Modulation
6. JPEG ..........................................................................Joint Photographic Experts Group

# ABSTRACT

A speech transmission system through LAN using IP is meant to transfer a given wave file containing speech to any other computer over LAN. Transmission will take place in coded and compressed form. We are using Huffman coding in our project for the transmission of speech. Wave file will be converted into two MATLAB variables that will be send as a single file over LAN using a GUI. All the coding and compression will be done by a separate GUI in MATLAB. Speech will be transmitted in coded form and also as a much compressed signal. The software is being tested on different wave samples and has shown satisfactory results. It has been able to get a compression of over 80%. It can even increase in some of the cases. Software has been made very user friendly since for GUI has been made for simplifying the procedure to run code.

Chapter - 1

# INTRODUCTION

## 1.1 Speech Coding:

Speech coding is the application of data compression of digital audio signals containing speech. Speech coding uses speech-specific parameter estimation using audio signal processing techniques to model the speech signal, combined with generic data compression algorithms to represent the resulting modeled parameters in a compact bit stream.

The techniques used in speech coding are similar to that in audio data compression and audio coding where knowledge in psychoacoustics is used to transmit only data that is relevant to the human auditory system. For example, in narrowband speech coding, only information in the frequency band 400 Hz to 3500 Hz is transmitted but the reconstructed signal is still adequate for intelligibility.

Speech coding differs from other forms of audio coding in that speech is a much simpler signal than most other audio signals, and that there is a lot more statistical information available about the properties of speech. As a result, some auditory information which is relevant in audio coding can be unnecessary in the speech coding context. In speech coding, the most important criterion is preservation of intelligibility and "pleasantness" of speech, with a constrained amount of transmitted data.

It should be emphasized that the intelligibility of speech includes, besides the actual literal content, also speaker identity, emotions, intonation, timbre etc. that are all important for perfect intelligibility.

Hearing is not a purely mechanical phenomenon of wave propagation, but is also a sensory and perceptual event. When a person hears something, that something arrives at the ear as a mechanical sound wave traveling through the air, but within the ear it is transformed into neural action potentials. These nerve pulses then travel to the brain where they are perceived. Hence, in many problems in acoustics, such as for audio processing, it is advantageous to take into account not just the mechanics of the environment, but also the fact that both the ear and the brain are involved in a person's listening experience.

The inner ear, for example, does significant signal processing in converting sound waveforms into neural stimulus, so certain differences between waveforms may be imperceptible. MP3 and other audio compression techniques make use of this fact. In addition, the ear has a nonlinear response to sounds of different loudness levels. Also human ears have nonlinear frequency response, which is the measure of spectrum response across different frequencies.

## 1.2 DATA COMPRESSION:

Now comes the second part of the project only speech coding is not the aim of the project the other aim is bring a certain amount of compression in the file.

In computer science and information theory, data compression or source coding is the process of encoding information using fewer bits (or other information-bearing units) than an un-encoded representation would use through use of specific encoding schemes.

As with any communication, compressed data communication only works when both the sender and receiver of the information understand the encoding scheme. For example, this text makes sense only if the receiver understands that it is intended to be interpreted as characters representing the English language. Similarly, compressed data can only be understood if the decoding method is known by the receiver.

Compression is useful because it helps reduce the consumption of expensive resources, such as hard disk space or transmission bandwidth. On the downside, compressed data must be decompressed to be used, and this extra processing may be detrimental to some applications. For instance, a compression scheme for video may require expensive hardware for the video to be decompressed fast enough to be viewed as it's being decompressed (the option of decompressing the video in full before watching it may be inconvenient, and requires storage space for the decompressed video). The design of data compression schemes therefore involves trade-offs among various factors, including the degree of compression, the amount of distortion introduced (if using a lossy compression scheme), and the computational resources required to compress and uncompress the data.

## 1.3 COMPRESSION ALGORITHMS:

There are basically two types of compression algorithms based on their principles of compression. The two different algorithms are explained below.

### 1.3.1 Lossy Data Compression:

In this algorithm there occurs a loss in the information or data that is needed to be compressed. These algorithms basically remove all those data that is not used often or the type of data whose absence will not make any difference.

### 1.3.2 Lossless Data Compression:

It is a type of compression scheme in which no data is lost while sending. The information is fully preserved and safe.
Coding that we have applied in our project i.e. Huffman Coding is a lossless data compression algorithm. Hence no data is being lost in coding and compression.

### 1.4 Data Transmission:

Data transmission, digital transmission or digital communications is the physical transfer of data (a digital bit stream) over a point-to-point or point-to-multipoint communication channel. Examples of such channels are copper wires, optical fibers, wireless communication channels, and storage media. The data is often represented as an electro-magnetic signal, such an electrical voltage signal, a radio-wave or microwave signal or an infra-red signal.

Data transmitted may be the digital messages originating from a data source, for example a computer or a keyboard. It may also be an analog signal such as a phone call or a video signal, digitized into a bit-stream for example using pulse-code modulation (PCM) or more advanced source coding (data compression) schemes. This source coding and decoding is carried out by codec equipment.

In our project we have done DATA Transmission through LAN using JAVA language. We have also used internet protocol in our transmission.

CHAPTER – 2

# SAMPLING

## 2.1 Theory

In signal processing, sampling is the reduction of a continuous signal to a discrete signal. A common example is the conversion of a sound wave (a continuous-time signal) to a sequence of samples (a discrete-time signal).

A sample refers to a value or set of values at a point in time and/or space.

A sampler is a subsystem or operation that extracts samples from a continuous signal.

A theoretical ideal sampler produces samples equivalent to the instantaneous value of the continuous signal at the desired points.

Let x(t) be a continuous signal which is to be sampled, and that sampling is performed by measuring the value of the continuous signal every T seconds, which is called the sampling interval. Thus, the sampled signal x[n] given by:

$x[n] = x(nT)$, with $n = 0, 1, 2, 3, \ldots$

## 2.2 Speech Sampling

Speech sampling is much easier to apply since human ears are not so sensitive to small frequency changes. Hence we have also applied sampling for our speech signal.

We have done re-sampling in our project it is because any data that is being stored in a computer is already in sampled form.

Function that is being used in our project is resample. It is used to again sample the signal sored in the wave file.

# QUANTIZATION:

We have done quantization in our signal to approximate values and to reduce number of samples in our signal. Numbers of samples were needed to be less so that it would become easier for us to perform the Huffman coding and to increase the Compression.

## 3.1 Theory:

In digital signal processing, quantization is the process of approximating ("mapping") a continuous range of values (or a very large set of possible discrete values) by a relatively small ("finite") set of ("values which can still take on continuous range") discrete symbols or integer values. For example, rounding a real number in the interval [0,100] to an integer

In signal processing, quantization refers to approximating the output by one of a discrete and finite set of values, while replacing the input by a discrete set is called discretization, and is done by sampling: the resulting sampled signal is called a discrete signal (discrete time), and need not be quantized (it can have continuous values). To produce a digital signal (discrete time and discrete values), one both samples (discrete time) and quantizes the resulting sample values (discrete values).

## 3.2 Quantization and data compression

Quantization plays a major part in lossy data compression. In order to determine how many bits are necessary to effect a given precision, algorithms are used. Quantization was done in our project in MATLAB for reducing the number of samples from the original signal. Quantization plays a major part in lossy data compression. In many cases, quantization can be viewed as the fundamental element that distinguishes lossy data compression from lossless data compression, and the use of quantization is nearly always motivated by the need to reduce the amount of data needed to represent a signal.

### 3.3 Use of Quantization in Project:

In Speech Coding quantization that is being done is very useful since it can create only certain number of samples that are helpful for applying Huffman Coding. The lower number of samples is the easier it becomes to apply Huffman Coding.

CHAPTER 4

# HUFFMAN CODING

## 4.1 Theory:

Huffman Coding is applied in the signal after quantization. Huffman Coding produces a small length code for the samples that are being repetitive. We have done Huffman Coding in the signal to reduce the size and include coding in the signal.

Huffman Coding can easily bring a compression of about 80-95% in a wave file. Now the data that is needed to transfer is only coded signal and a dictionary file.

One of the main advantage of the Huffman coding is that it is easy to apply, quick to perform and easy to decode.

In computer science and information theory, Huffman coding is an entropy encoding algorithm used for lossless data compression. The term refers to the use of a variable-length code table for encoding a source symbol (such as a character in a file) where the variable-length code table has been derived in a particular way based on the estimated probability of occurrence for each possible value of the source symbol. It was developed by David A. Huffman .

Huffman coding results in a code that contains the coded form of original signal that is been made by comparing it with the dictionary that contains compressed code for each sample.

Hence we now have two compressed files one is code and other is dictionary. Now we only need to transmit code and dictionary instead of the whole signal. It reduces the size of the file that is to be transmitted.

## 4.2 Encoding Huffman Code.

Without relation of how Huffman Codes are calculated, there is a common scheme of encoding the Static Huffman Code.

(i) Get the samples that are present in the original file.

(ii) Find out their probability of occurrence in the signal.

(iii) Give each sample a unique code and small length code to the signal that have high probability of occurrence.

## 4.3 Decoding Huffman Code.

Decoding is symmetric to encoding, however, here it is:

(i) Compare the incoming code with the dictionary and decode accordingly.

(ii) Replace each coded value by their respective value as per in dictionary.

# CHAPTER 5

## GUI (Graphic User Interface)

Main element in any software is it's GUI. A simpler GUI can easily make a good software into a perfect one.

Our main aim was to provide users with a good GUI so that it becomes easier for them to use our software.

Our project is having two GUI's one for all the processing in the MATLAB and other for all the processing or transmission that are done in JAVA.

### 5.1 MATALAB GUI:

Our GUI in MATLAB performs two main things:

1. It performs coding and plays the coded signal and decoded signal.

2. It prepares code and dictionary two files that are needed for transmission.

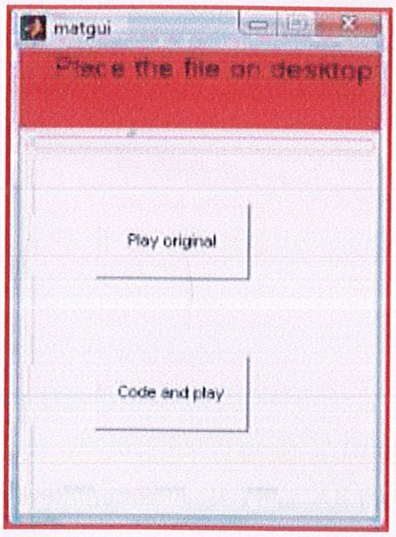


Figure 5.1: Speech Transmitter
(MATLAB GUI)


Figure 5.2: Speech Receiver
(MATLAB GUI)

This was all the processing that is being done in MATLAB and the transmission is done through JAVA GUI.

**5.2 JAVA GUI:**

A JAVA GUI (Graphical User Interface) is easy to create after you have a few basic elements from these groups:

1. Top-level Containers (eg, JFrame) :

A JFrame is the total frame or window that appears in front of us. It contains all panels and menubar and everything that a GUI has. All the other components are being added in the JFrame.

2. Intermediate Containers (eg, JPanel, JMenuBar) :

Jpanel and JMenuBar are the other items that are added in the JFrame. JPanel contains the buttons text field that are used in a GUI.

3. Components (eg, JLabel, JButton, JTextField) :

These are the components having specific purpose. Each perform a pre assigned task thak is given to them if they are pressed.

4. Listeners (eg, ActionListener)

These are for assigning buttons and text field their tasks. They are used to let the program know what task is to be performed.

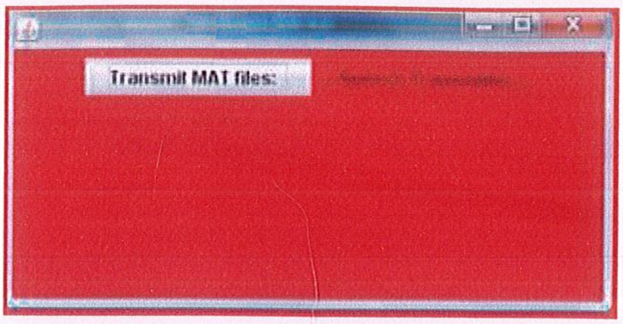


Figure 5.3: MAT File transmitter

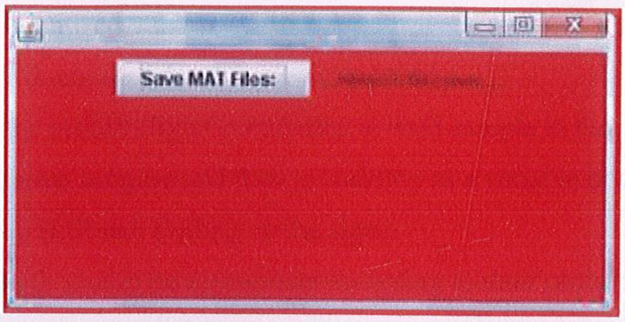


Figure 5.4 MAT File receiver

# Chapter 6

## ALGORITHM:

### 6.1 Algorithm for the transmitter GUI in MATLAB:

1. Get the wave file and place it desirable location.
2. Now use the wave file as a single variable in MATLAB.
3. Perform the resampling on the wave file.
4. Store the resampled signal.
5. Now use Quantization in the resampled signal.
6. Save the quantized values.
7. Get the different samples that are to be coded.
8. Find out the probability of occurrence of each sample in the signal.
9. Store different samples and their probability in a variable or array.
10. Now apply Huffman Coding on the signal.
11. Codes length should be appropriate according to their probability of occurrence in the original signal.
12. Code the original signal with respect to the dictionary that contains Huffman code for each sample.
13. Now get the coded signal and play it for the user to show how compressed wave file sounds.
14. Load the code and dictionary in a separate .MAT file that is being used for transmission.
15. Now place all the functions that are to be performed to their respective buttons in the GUI.

## 6.2 Algorithm for the receiver GUI in MATLAB:

1. Load the .MAT file that has the code and dictionary saved in it.
2. Now decode the coded signal by matching it's codes with that of dictionary.
3. Place the original sample in place of each code .
4. Store the decoded signal.
5. Now play the wave file that is being made after decoding.
6. Now place all the functions that are to be performed to their respective buttons in the GUI.

## 6.3 Algorithm for the transmitter GUI in JAVA:

1. Get the .Mat file that is needed for transmission.
2. Get the port number that is being used.
3. Now load the given file to a file variable in JAVA.
4. Read the data from the file .
5. Place the read data in a buffer.
6. Buffer is used for the transmission purpose.
7. Buffer will keep transmitting certain values of file every time till the file is fully transmitted.
8. Now assign all the functions to different buttons in the GUI using actionListeners.

### 6.4 Algorithm for the receiver GUI in JAVA:

1. Get the server address and port number
2. Get connected to the server.
3. Start receiving files from the server.
4. Now use the buffer to store values that are being transmitted.
5. Use the write function to write each value in .MAT file .
6. Perform the writing part until full file is being saved using buffer.
7. Save the .MAT file to a pre-desired place.

## CONCLUSION:

Speech is the main mode of Communication that is being used nowadays. Hence, speech communication is a much popular and much used way of communication. So there will be different ways of transferring speech signal. In each and every way there is emphasis given on the speed.

Speed can increase easily if the speech signal is being send in the compressed form. Much of the emphasis in communication system is given on how much compression can be brought in the transmitting signal. It is because low size samples or signals are much easy and fast to send over LAN or Internet.

Speech coding or speech compression is the application of data compression of digital audio signals containing speech. Speech coding uses speech-specific parameter estimation using audio signal processing techniques to model the speech signal, combined with generic data compression algorithms to represent the resulting modeled parameters in a compact bit stream. The objective of the speech coding is to represent speech signal with minimum number of bits yet maintain the perceptual quality.

The advantages with coded speech signals are lower sensitivity to channel noise, easier to error-protect, encrypt, multiplex and packetize and lastly it is efficient transmission over bandwidth constrained channels due to lower bit rate.

**Commercialization Potential:**

Our software can be used in Hostels and Offices having LAN as a simple way to communicate with speech signal which can be heard whenever wanted.

## Chapter 8

### RESULTS:

After applying the coding on a sample wave file these were the outputs at different levels.

Example 1:

### 1. Original Signal

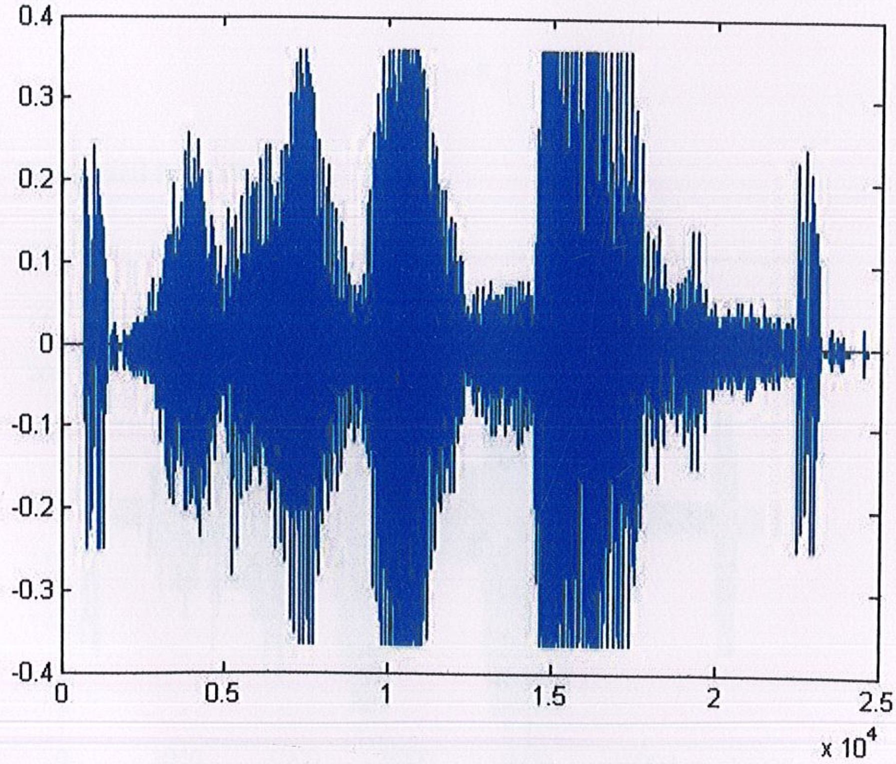


Figure 8.1

2. **Sampled Signal**

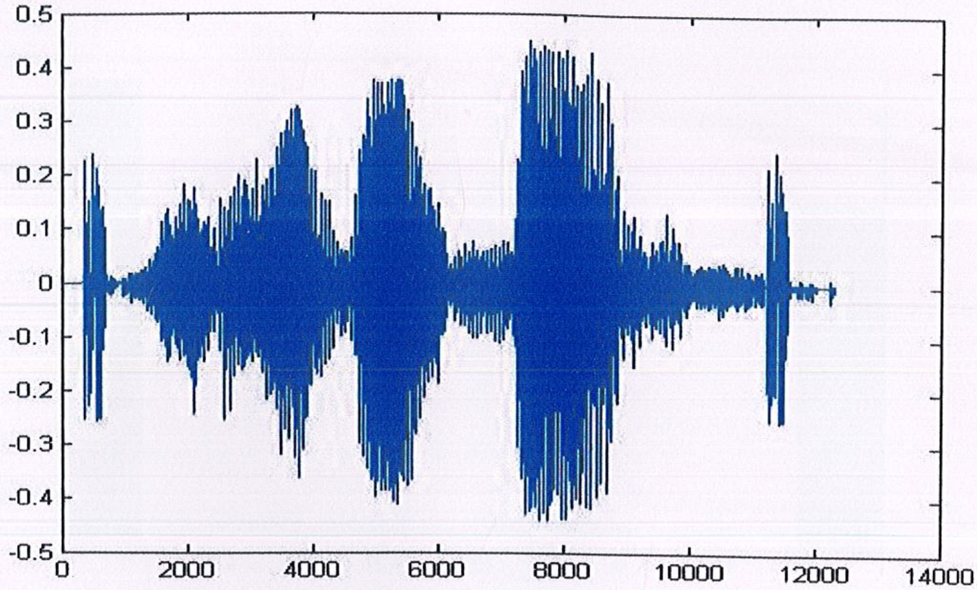


**Figure 8.2**

3. **Quantized Signal**

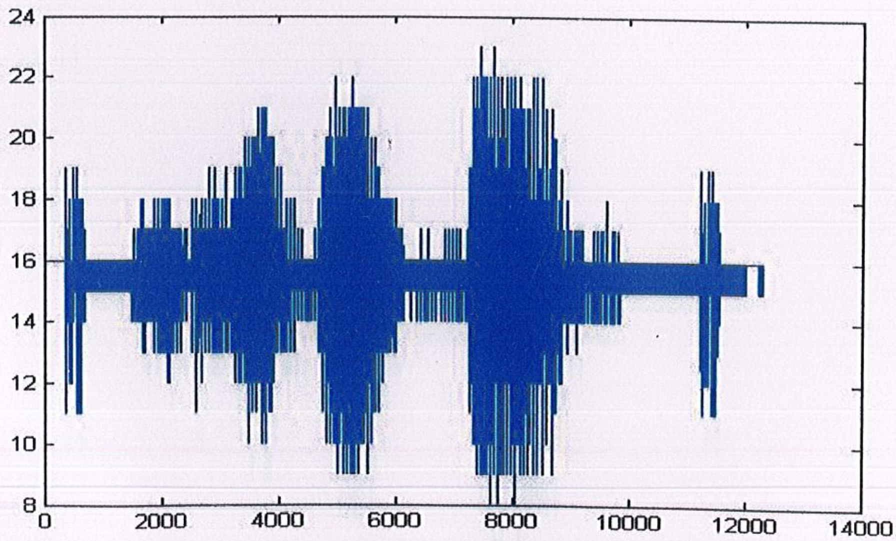


**Figure 8.3**

4. **Number of zeros and ones in the coded signal.**

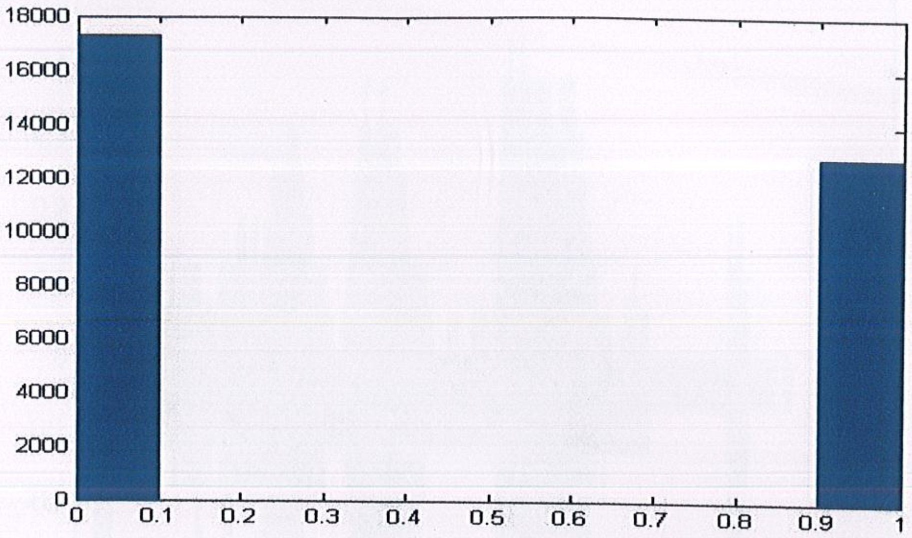


**Figure 8.4**

5. **Digital signal that we get after Huffman Decoding.**

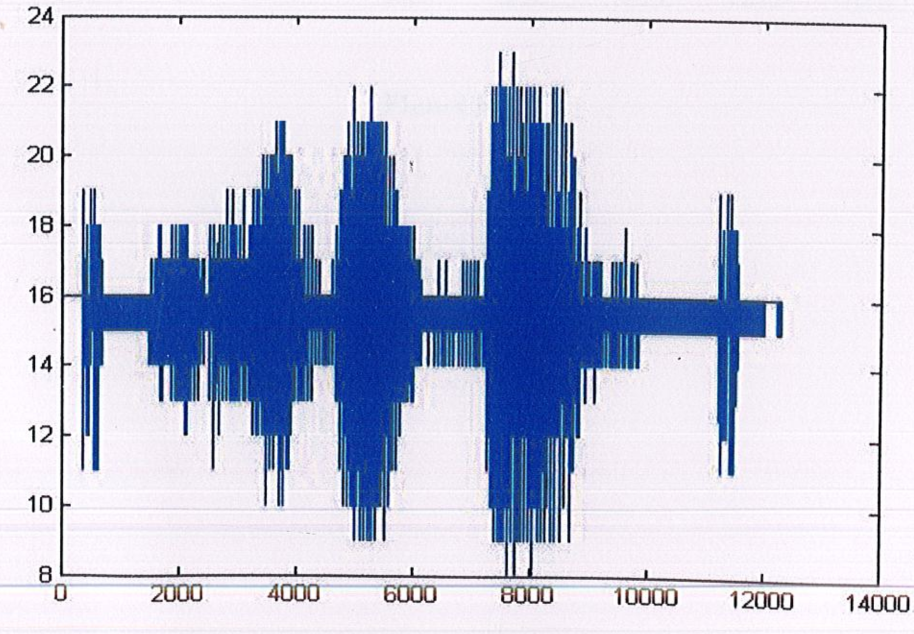


**Figure 8.5**

6. **Reconstructed Signal.**

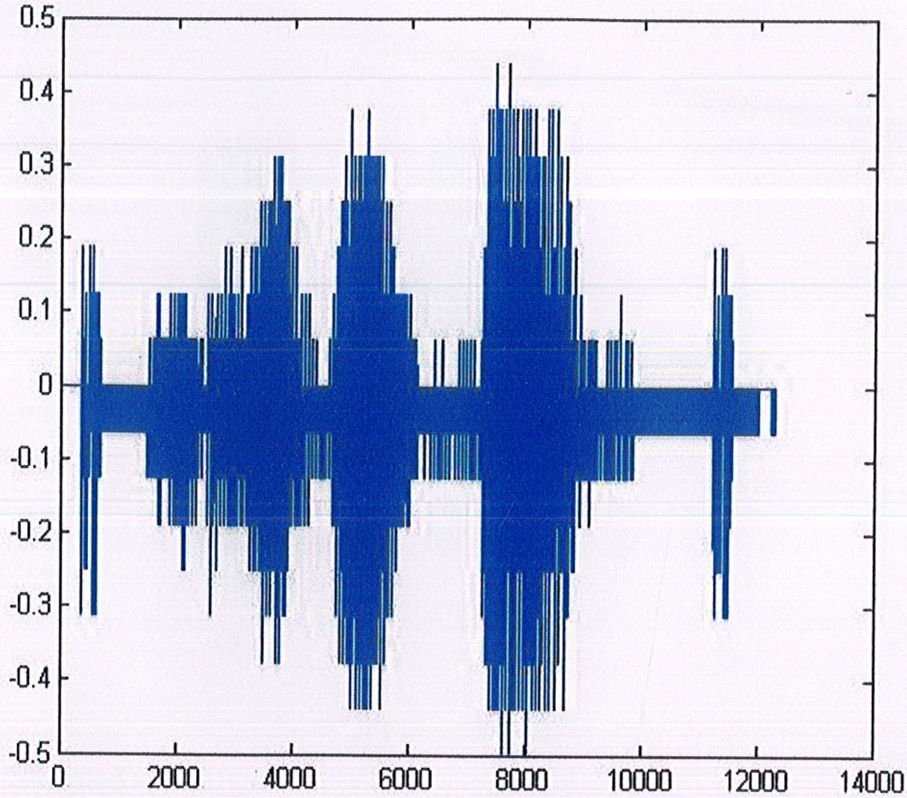


**Figure 8.6**

**Example 2: Results when other wave is processed by our software.**

1. **Original Signal**

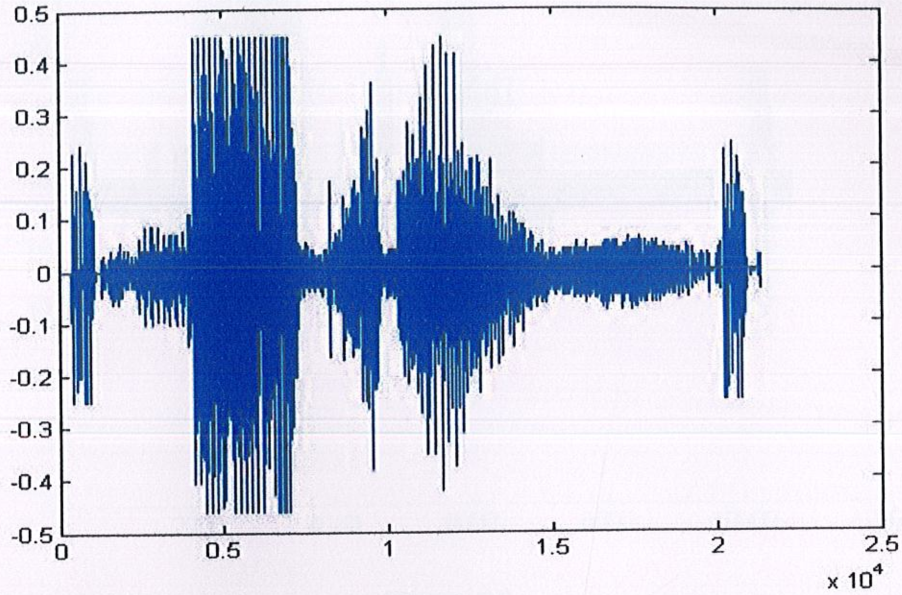


**Figure 8.7**

2. **Sampled Signal**

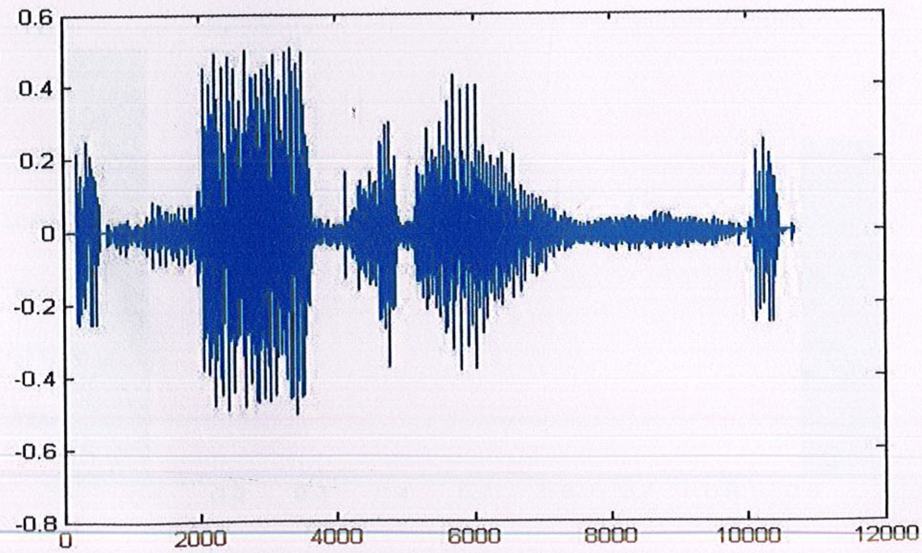


**Figure 8.8**

3. **Quantized Signal**

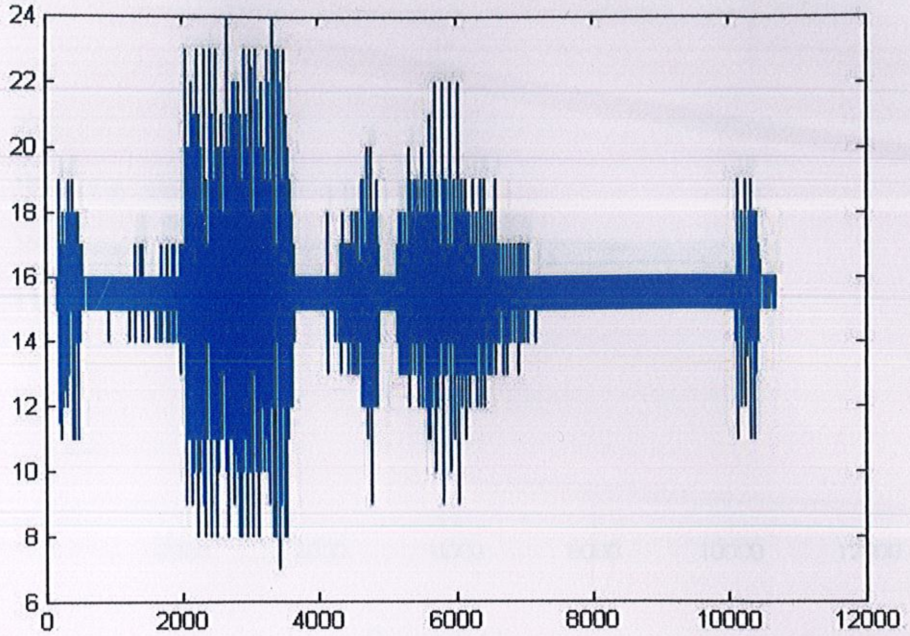


**Figure 8.9**

4. **Number of zeros and ones in the coded signal.**

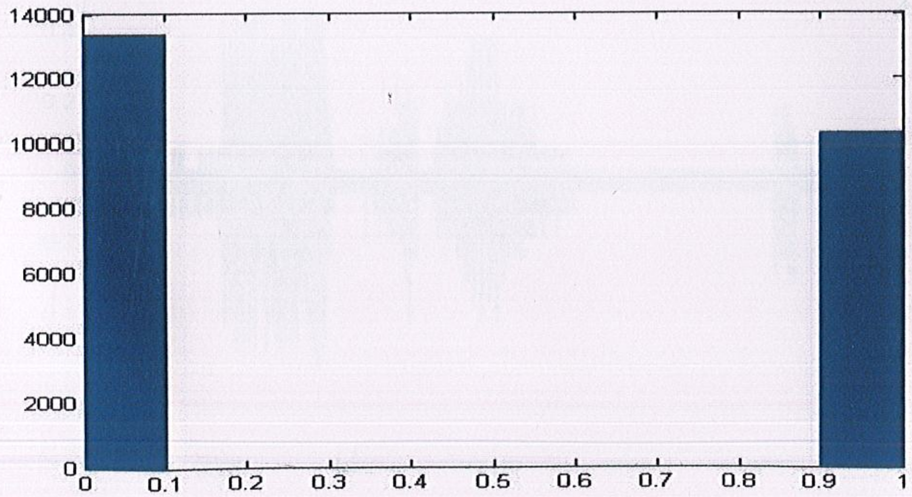


**Figure 8.10**

5. **Digital signal that we get after Huffman Decoding.**

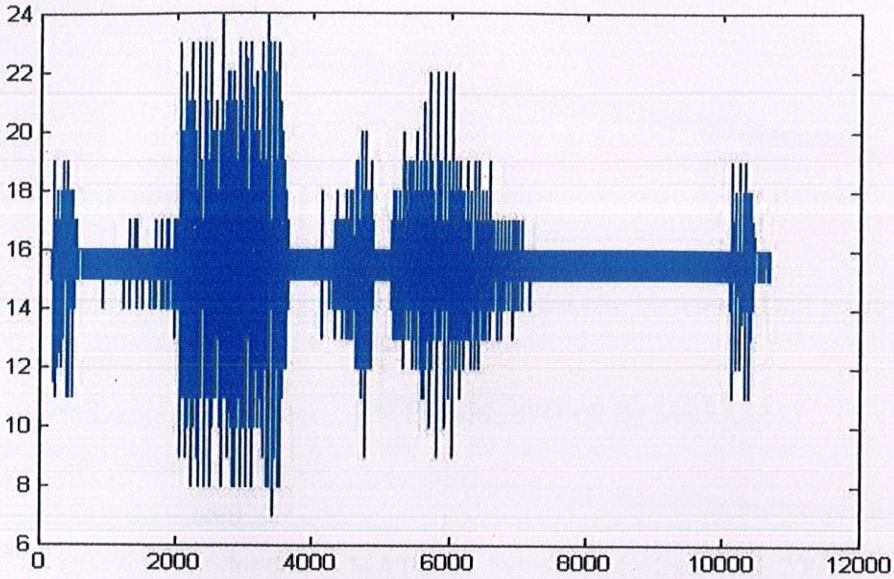


**Figure 8.11**

6. **Reconstructed Signal.**

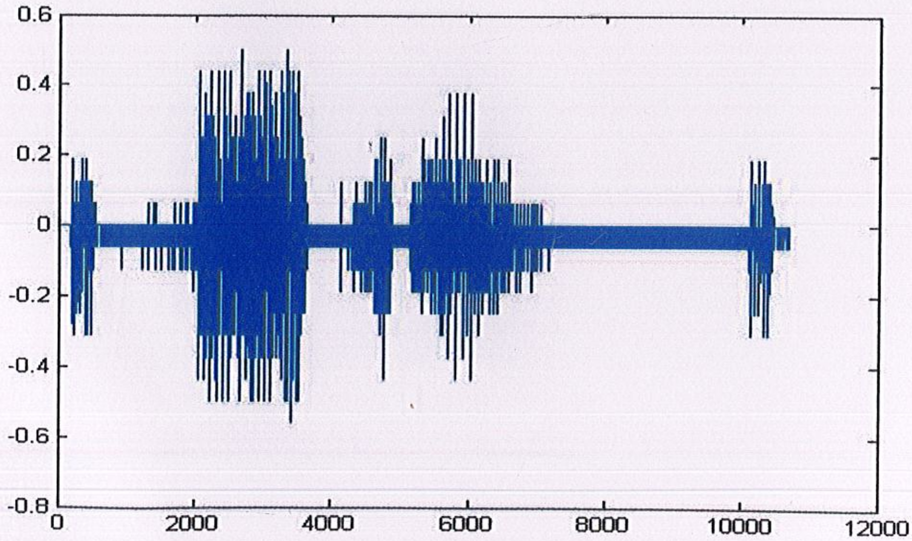


**Figure 8.12**

Chapter 9

## BIBLIOGRAPHY


1. Java Programming Language Handbook By Anthony Potts David H., Jr. Friedel
2. Java Language Reference, 2nd Edition, 1997-08 By Mark Grand
3. www.wikipedia.com
4. www.google.com
5. A Concise Introduction to MATLAB Palm III McGraw-Hill, 2008