# DESIGN AND DEVELOPMENT OF A VOICE BASED MACHINE CONTROL AT REMOTE LOCATION

**A Thesis**

***Submitted in partial fulfillment of the requirements for the award of degree of***

**Master of Engineering**

in

**Electronics Instrumentation and Control Engineering**

Submitted by


**JASLEEN KAUR**

Regn. No- **8044212**


Under the Guidance of


| | |
|---|---|
| **Mr. Sunil Singla** | **Mr. Nitin Narang** |
| Sr.Lecturer | Lecturer |
| Department of Electrical & Instrumentation Engineering | Department of Electrical & Instrumentation Engineering |
| T.I.E.T. (Patiala) | T.I.E.T., (Patiala) |

**Department of Electrical & Instrumentation Engineering**

**THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY**

**(DEEMED UNIVERSITY)**

**PATIALA (PUNJAB)-147004, INDIA**


**JUNE-2006**

# ***CERTIFICATE***

This is to certify that the thesis titled, "**Design and Development of a voice based machine control at remote location**." being submitted by **Jasleen Kaur**, Regn. No.: 8044212, in partial fulfillment for the requirements of the award of the degree of Master of Engineering in Electronics Instrumentation and Control Engineering to Thapar Institute of Engineering And Technology (Deemed University), Patiala, is a record of student's own work carried out under my supervision and guidance and no part of this thesis has been submitted to any other university or institute for award of any degree.

**Supervisor:**
**Mr. Sunil Singla**
**Sr. Lecturer**

**Co-Supervisor:**
**Mr. Nitin Narang**
**Lecturer**

**Countersigned By**

**Ms.Manbir Kaur**
**Assistant Professor and Head**

**Dr.T.P.Singh**
**Prof. and Dean of Academics**

**Department of Electrical and Instrumentation Engineering**

**THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY**

**(DEEMED UNIVERITY)**

**PATIALA (PUNJAB) -147004**

# ACKNOWLEDGMENT

*At this time of my substantial enhancement, before we get into the thick of the things, I would like to add a few heartfelt words for the people who gave their unending support with their unfair humor and warm wishes. I am vastly indebted to many people who have helped and inspired me, in various ways, to start, to continue, and complete this work.*

*First a very special thanks to my supervisor, Mr. Sunil Singla who gave me the confidence and support to begin my Master's program and helped me to set my benchmark even higher and to look for solutions to problems rather than focus on the problem. I wish to express my sincere gratitude to Mr.Nitin Narang, my co-supervisor who guided me in all times of need and helped me in all possible ways.*

*I am grateful to our esteemed Director, Deans and Head of the Department for there valuable support. I would also like to gratefully acknowledge the support of my friend Ashutosh Chauhan. He helped me immensely by giving me encouragement and mirrored back my ideas so I heard them aloud, an important process for me to shape my thesis paper and future work.*

*Finally, I wish to express my profound regards to my parents and thanks to my brothers without whose constant encouragement and support, I would not have made it this echelon.*

*Let some words of thanks to those people, whom I can't mention…………*

*Place: TIET, Patiala* *Jasleen Kaur*

*Date: Friday, June 30, 2006*

***DEDICATED***

***TO***

***MY PARENTS***

**CERTIFICATE** I
**ACKNOWLEDGEMENT** II
**DEDICATIONS** III
**TABLE OF CONTENT** IV
**TABLE OF FIGURES** V
**ABSTRACT** 1
**LITERATURE SURVEY** 2
**CHAPTER 1**……………………………………………………………………………………….6
**REMOTE CONTROL TECHNIQUES AND SPEECH SIGNAL**……………………………………. 6

1.1 INTRODUCTION ....................................................................................................................6
1.2 REMOTE CONTROL USING TELEPHONE SIGNAL .................................................................... 6
1.3 SPEECH SIGNAL ................................................................................................................. 7
*1.3.1 Vocal Tract.............................................................................................................. 8*
*1.3.2 Characteristics of Speech signal ............................................................................ 19*
*1.3.3 OSCILOGRAM (WAVEFORM) ................................................................................ 19*
*1.3.4 Different Voice characteristics ................................................................................ 19*
*1.3.5 FUNDAMENTAL FREQUENCY (PITCH, F0) ........................................................... 20*
*1.3.6 SPECTRUM ............................................................................................................ 21*
*1.3.7 SPECTROGRAM .................................................................................................... 21*
1.4 PROBLEM FORMULATION.................................................................................................. 22
**CHAPTER 2 .................................................................................................................... 23**
**DTMF SIGNAL ................................................................................................................ 23**
2.1 INTRODUCTION……………………………………………………………………………..15
2.2 DTMF SIGNAL GENERATION.............................................................................................. 25
2.3 DTMF DECODER ............................................................................................................... 17
**CHAPTER 3 .................................................................................................................... 28**
**LABVIEW AND SOUND ...................................................................................................... 28**
3.1 INTRODUCTION…………………………………………………………………………......20
3.2 LABVIEW BLOCK DIAGRAM AND FRONT PANEL ................................................................ 29
*3.2.1 Terminals ................................................................................................................ 31*
*3.2.2 Nodes ..................................................................................................................... 31*
*3.2.3 Structures ............................................................................................................... 31*
*3.2.4 Icon and Connector Pane ......................................................................................... 32*
3.3 LABVIEW ENVIRONMENT................................................................................................... 32
*3.3.1 Controls on the front panel ...................................................................................... 32*
*3.3.2 Functions Palette ..................................................................................................... 33*
*3.3.3 Tools Palette ........................................................................................................... 33*
3.4 MENUS AND THE TOOLBAR ................................................................................................ 34
*3.4.1 Menus ...................................................................................................................... 34*
*3.4.2 Shortcut Menus ........................................................................................................ 34*
*3.4.3 Shortcut Menus in Run Mode .................................................................................. 34*
3.5 BUILDING THE FRONT PANEL ............................................................................................ 35
*3.5.1 Configuring Front Panel Objects.............................................................................. 35*
*3.5.2 Showing and Hiding Optional Elements .................................................................. 35*
*3.5.3 Changing Controls to Indicators and Indicators to Controls .................................... 36*
*3.5.4 Replacing Front Panel Objects ................................................................................. 27*

*3.5.5 Front Panel Controls and Indicators* ........ *27*
*3.5.6 3-D and Classic Controls and Indicators* ........ *27*
*3.5.7 Slides, Knobs, Dials, Digital Displays, and Time Stamps* ........ *28*
*3.5.8 Slide Controls and Indicators* ........ *28*
*3.5.9 Rotary Controls and Indicators* ........ *28*
*3.5.10 Time Stamp Control and Indicator* ........ *29*
*3.5.11 Buttons, Switches, and Lights* ........ *29*
*3.5.12 Text Entry Boxes, Labels, and Path Displays* ........ *29*
*3.5.13 String Controls and Indicators* ........ *29*
*3.5.14 Array and Cluster Controls and Indicators* ........ *39*
3.6 BUILDING THE BLOCK DIAGRAM ........ 39
*3.6.1Relationship between Front Panel Objects and Block Diagram Terminals* ........ *40*
*3.6.2 Block Diagram Objects* ........ *40*
*3.6.3 Block Diagram Terminals* ........ *40*
*3.6.4 Control and Indicator Data Types* ........ *41*
*3.6.5 Functions Overview* ........ *41*
*3.6.6 Numeric Functions* ........ *41*
*3.6.7 Boolean Functions* ........ *41*
*3.6.8 String Functions* ........ *42*
*3.6.9Array Functions* ........ *42*
*3.6.10 Cluster Functions* ........ *42*
*3.6.11 Comparison Functions* ........ *42*
*3.6.12 Time and Dialog Functions* ........ *42*
3.7 SOUND VIS ........ 43
*3.7.1 Beep (Not in Base Package)* ........ *43*
*3.7.2GET SOUND INFO* ........ *33*
*3.73 SND READ WAVEFORM* ........ *33*
*3.7.4 Snd Write Waveform* ........ *44*
3.8 SOUND FILE VIS ........ 44
*3.8.1 Snd Read Wave File* ........ *44*
*3.8.2 Snd Write Wave File* ........ *44*
3.9 SOUND INPUT VIS ........ 45
*3.9.1 SI Clear* ........ *45*
*3.9.2SI Config* ........ *45*
*3.9.3SI Read* ........ *45*
*3.9.4 SI Start* ........ *46*
*3.9.5 SI Stop* ........ *46*

**CHAPTER 4** ........ **37**

**HARDWARE AND SOFTWARE IMPLEMENTATION** ........ **37**

4.1 INTRODUCTION ........ 37
4.2 SYSTEM DESCRIPTION ........ 37
4.3VOICE INPUT ........ 38
4.4 READ SOUND CARD (SOUNDCARDREAD.VI) ........ 38
4.5 WORD RECOGNITION (MACHINERUN.VI) ........ 50
*4.5.1 Decimate* ........ *53*
*4.5.2 Cross Correlation* ........ *54*
*4.5.3 Mean* ........ *55*
*4.5.4 Maximum of three* ........ *46*
*4.5.5 Output at parallel port* ........ *57*
*4.5.5.1 Build Array* ........ *47*
*4.5.5.2 Type Cast* ........ *57*
*4.5.5.3 VISA Write* ........ *58*
*4.6 PARALLEL PORT TO DTMF CONVERTER* ........ *48*

*4.7 Transmitter Circuit (DTMF)* ........ *58*
*4.7.1 Keypad* ........ *58*
*4.72 DTMF Encoder* ........ *59*
4.8 RECEIVER CIRCUIT ........ *49*
4.9 SEVEN SEGMENT DISPLAY ........ 62
4.10 4 TO 16 LINE DECODER ........ 63
4.11 DEVICE SWITCHING UNIT (RELAYS) ........ 63
4.12CIRCUIT DIAGRAM OF THE SYSTEM ........ 54

**CHAPTER 5 ........ 66**

**RESULTS, DISCUSSIONS AND CONCLUSIONS ........ 55**

5.1RESULTS AND DISCUSSIONS ........ 55
5.2 CONCLUSIONS ........ 57

**FUTURE SCOPE ........ 58**

**REFERENCES ........ 71**

# TABLE OF FIGURES


FIGURE 1-1:THE ACTUAL SPEECH PRODUCTION SYSTEM .......... 8
FIGURE 1-2:TYPICAL SEGMENT OF VOICED SPEECH .......... 9
FIGURE 1-3:POWER SPECTRAL DENSITY FOR A SEGMENT OF VOICED SPEECH.......... 9
FIGURE 1-4:TYPICAL SEGMENT OF UNVOICED SPEECH .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 1-5: POWER SPECTRAL DENSITY FOR A SEGMENT OF VOICED SPEECH.......... 10
FIGURE 1-6: WAVEFORM OF THE SPOKEN WORD PHONETICIAN .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 1-7: FEMALE AND MALE VOICE CHARACTERISTICS .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 1-8: FUNDAMENTAL FREQUENCY AGAINST TIME .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 1-9: SPECTRUM.......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 1-10: SPECTROGRAM .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 2-1: TONE FREQUENCY FOR 1 KEY.......... 15
FIGURE 2-2: DTMF KEYPAD.......... 16
FIGURE 2-3: DTMF ENCODER.......... 17
FIGURE 2-4: DTMF DECODER.......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 3-1: FRONT PANEL, TOOL BOX AND CONTROLS BOX .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 3-2: BLOCK DIAGRAM AND FUNCTION BOX .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 3-3: BEEP.......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 3-4: GET SOUND INFO.......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 3-5: SND READ WAVEFORM .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 3-6: SND WRITE WAVEFORM .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 3-7: SND READ WAVE FILE .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 3-8: SOUND INPUT CLEAR .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 3-9: SOUND INPUT CONFIGURATION.......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 3-10: SOUND INPUT READ .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 3-11: SOUND INPUT START.......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 3-12: SOUND INPUT STOP .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 4-1: BLOCK DIAGRAM OF THE SYSTEM.......... 38
FIGURE 4-2: SOUND READ CARD .......... 39
FIGURE 4-3: SOUND READ VI.......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 4-4(A): WORD RECOGNITION VI .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 4-4(B): WORD RECOGNITION VI.......... 42
FIGURE 4-5: PARALLEL PORT VI.......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 4-6: DECIMATE .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 4-7: CROSS CORRELATION.......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 4-8: MEAN .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 4-9: MAXIMUM OF THREE VI.......... 46
FIGURE 4-10: BUILD ARRAY .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 4-11: TYPE CAST .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 4-12: DTMF KEYPAD.......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 4-13: DTMF ENCODER.......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 4-14: DTMF DECODER.......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 4-15: 4 TO 16 LINE DECODER .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 4-16: DEVICE SWITCHING UNIT .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 4-17: CIRCUIT DIAGRAM OF THE SYSTEM.......... 54
FIGURE 5-1: MACHINE RUN VI .......... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE 5-2: WAVEFORM OF THE SPOKEN WORD 'ONE' AND 'THREE'... **ERROR! BOOKMARK NOT DEFINED.**
FIGURE5-3: PROGRAM FOR THE RUNNING OF THE MACHINE 3 .......... **ERROR! BOOKMARK NOT DEFINED.**

# ABSTRACT

*Recent advancements in computer technology and availability have allowed the computer industry to develop hardware and software applications that address the needs of the physically disabled .The concept of "voice controlled home automation" or "voice controlled smart home" is not new. In fact, integrated home systems have been offered since the 1960s. There furthermore, most of the industries invest in developing communication networks, but not much on the applications or appliances themselves. This thesis proposes device control technique using speech recognition technique. The key technology to realize the system is* ***(1)*** *distant-talking speech recognition and* ***(2)*** *the control of devices/machines at remote location.*

*This thesis first does the speech recognition using the functions available in LabVIEW (Virtual Instrumentation). Next, the possibility of remote control using LabVIEW to control such devices is proposed. DTMF (Dual Tone Multiple Frequency) signal is used to remote control of devices. The objective of the Virtual Laboratory for the Disabled is to create a realistic, real-time, electrical engineering virtual laboratory. This project targets individuals who do not have adequate mobility of their upper bodies to perform laboratory experiments. To provide a more realistic and enhanced learning experience, the users of the virtual laboratory are allowed the freedom to control the devices and machines using just by speaking the machine number.*

# LITERATURE SURVEY

**Rabiner, L.R.** in *"Applications of voice processing to telecommunications"* has described the ways in which people communicate are changing rapidly. The options are many and diverse, ranging from voice calls over wireless networks, to video calls over the conventional wired network, ISDN video, FAX, e-mail, voice mail, beeper services, data services, audio teleconferencing, video teleconferencing, and so-called scribble phone service (transmission of arbitrary handwritten input). This revolution in communications is being fueled by several sources, including the availability of low-cost, low-power, computation in both DSP and RISC chips, larger and cheaper memory chips, improved algorithms for communications (e.g., modems, signaling) and signal processing, and finally the creation of world-wide standards for transmission, signal compression, and communication protocols. The broad goal of the communications revolution is to provide seamless and high-quality communications between people (or groups of people), anywhere, anytime, and at a reasonable price. Although there are many technologies that form the bases for the communications environment of the twenty-first century, one of the key technologies for making the vision a reality is voice processing. In this paper we attempt to show, by example, how voice processing has been applied to specific problems in telecommunications, and how it will grow to become an even more essential component of the communications systems of the twenty-first century.[8]

**R Nakatsu** and **Y Suzuki** in their paper *"What does voice-Processing Technology Support today?"* describes the state of the art in applications of voice-processing technologies. In the first part, technologies concerning the implementation of speech recognition and synthesis algorithms are described. Hardware technologies such as microprocessors and DSPs (digital signal processors) are discussed. Software development environment, which is a key technology in developing applications software, ranging from DSP software to support software also is described. In the second part, the state of the art of algorithms from the standpoint of applications is discussed. Several issues concerning evaluation of speech recognition/synthesis algorithms are covered, as well as issues concerning the robustness of algorithms in adverse conditions.[11]

**Rabiner, L.R** in *" The role of voice processing in telecommunications"* During the decade of the 1990s, the fields of communications, computing, and networking are coming together in the form of personal information/communication terminals, and in the associated services (so-called personal communications services, PCS). Several technologies will play major roles in this communications revolution, but one of the key ones will be voice processing. The authors review several voice processing technologies, discuss current capabilities and the associated applications, and try to forecast where they see progress being achieved in the next decade and what applications will become commonplace as a result of the increased capabilities. They show how progress in voice processing is accompanied and stimulated by progress in microelectronics (memory and processing power of single chip architectures), and how, by the 21st century, telecommunications will have made major advances as a result of the use of voice processing.[9]

**C Seelbach** in *"A Perspective on Early Commercial Applications of Voice-Processing Technology for Telecommunications and Aids for the Handicapped"* The Colloquium on

Human-Machine Communication by voice highlighted the global technical community's focus on the problems and promise of voice-processing technology, particularly, speech recognition and speech synthesis. Clearly, there are many areas in both the research and development of these technologies that can be advanced significantly. However, it is also true that there are many applications of these technologies that are capable of commercialization now. Early successful commercialization of new technology is vital to ensure continuing interest in its development. This paper addresses efforts to commercialize speech technologies in two markets: telecommunications and aids for the handicapped.[4]

**Wong, E.M.C** in *"A phone-based remote controller for home and office automation"* describes that in modern society, home and office automation become increasingly important and interesting. They not only provide better ways to transfer information within homes/offices and between homes/offices, they provide better time management too. These also improve the quality of our lives by automating some of the electrical home appliances such as light source, A/V equipment, computer, security device, etc. The paper describes a hardware-based remote controller for power point control. Users can input the control commands and their own pass codes by using local or external telephones. The paper also discusses the operational sequence of the remote controller.[15]

**Coskun, I. Ardam**, H. *"A remote controller for home and office appliances by telephone"* This paper describes the design and development of a phone-based remote controller for home and office automation. The circuit is designed based on the Turkish telephone standards and connected to the telephone network just like any normal telephone sets. Any tone dialing dual tone multiple frequency (DTMF) telephone set or hand-held tone dialer may be used to send commands to the control unit, and remotely control, a wide range of mains appliances in homes and offices. The designed circuit can also detect the user identification number for preventing non-authorized use of the control unit. The feedback signal informs the user about the results of the commands.[3]

**Michael Duartkand Brian P. Bud** *"The Virtual Laboratory for the Disabled"* stated that the objective of the Virtual Laboratory forth Disabled are to create a realistic, real-time, electrical engineering virtual laboratory. This project targets individuals who do not have adequate mobility of their upper bodies' toper form laboratory experiments. To provide a more realistic and enhanced learning experience, the users of the virtual laboratory are allowed the freedom to build and test a wide variety of realistic electrical circuits, and be able to perform curriculum-based experiments. The main goal is to create an environment similar to a real electrical engineering laboratory, and to offer the user a way to learn the different aspects of instrumentation and circuitry.[7]

**Schneider, S.Swanson and J. Peng-Yung Woo** *"Remote telephone control system"* Today's technology is increasing at such an incredible rate that in the future the entire house we live in will be centrally controlled by a computer. The device detailed in this paper is the starting point for such a house. The goal of this device is to give a houseowner the ability to remotely check or control his household devices through the use of a Touch-Tone telephone. This means that any house can be controlled from any location where there is a Touch-Tone telephone available. [13]

# CHAPTER 1

## Remote Control Techniques and Speech Signal

### *1.1 Introduction*

Earlier, we were looking into the face of future when we talked about automated devices, which could do anything on instigation of a controller, but today it has become a reality. An automated device can replace good amount of human working force, moreover humans are more prone to errors and in intensive conditions the probability of error increases, whereas an automated device can work with diligence, versatility and with almost zero error. Home/office automation is the control of any or all electrical devices in our home or office, whether we are there or away. Home/office automation is one of the most exciting developments in technology for the home that has come along in decades. There are hundreds of products available today that allow us control over the devices automatically, either by remote control; or even by voice command!

Virtually anything in the home/office that is powered by electricity can be automated and/or controlled. We can control our electrical devices with our cordless phone from our easy chair. We can turn our porch lights on automatically at dark or when someone approaches and can see who is at the front door from any nearby television, and talk to them or unlock the door from any nearby telephone. Have the security system turn off lights, close drapes and setback the temperature when we leave and turn on the alarm system. The possibilities are only limited by our imagination!

### *1.2 Remote control using Telephone Signal*

Controlling devices using switches are common. From a few decades controlling using remote control switches like infrared remote control switch, wireless remote control switches, light activated switches are becoming popular. But these technologies have their own limitations. Laser beams are harmful to mankind. Some technologies like IR remote control are used for short distance applications. Such limitations can be overcome by using a system in which telephone is used as a media, which serves the main part of the system. This system does not require any radiations, is not harmful, and has no limitation of range i.e. can be used from any distance from meters to thousand kilometers. By using home phone as a local phone and another phone - either landline or mobile phone as a remote phone we can control devices. It has following features:

- This system can control up to 10 devices. It may be any electric or electronic appliances or devices with simple to heavy appliances. Each device is given a unique code.
- It makes accurate switching, any false switching of device are not done.
- There is no risk for false switching.
- The local phone (i.e., home phone or office phone) can be used for normal use by using a DPDT switch. So the system does not require a separate telephone line for this device controlling.
- To perform any operations through remote phone line, the user needs to dial to the local telephone (to which the interfacing circuit is connected) then the respective code of the device is dialed.
- The system circuit does not require any complex or programmable IC's.
- This system can detect the ringing signal from your exchange with the help of ring detector and automatically switches ON.
- Before changing the state of the device we can confirm the present status of the device. This circuit gives an acknowledgement tone after switching ON the devices to confirm the status of the device.
- We can control devices from local telephone. It can also be controlled by PCO.

## *1.3 Speech Signal*

Speech is produced when air is forced from the lungs through the vocal cords and along the vocal tract. The vocal tract extends from the opening in the vocal cords (called the glottis) to the mouth, and in an average man is about 17 cm long. It introduces short-term correlations (of the order of 1 ms) into the speech signal, and can be thought of as a filter with broad resonances called formants. The frequencies of these formants are controlled by varying the shape of the tract, for example by moving the position of the tongue. An important part of many speech codecs is the modelling of the vocal tract as a short term filter.

### *1.3.1 Vocal Tract*

The vocal tract is shown in cross-section. Air pressure produced by the lungs forces air through the vocal cords that, when under tension, produce puffs of air that excite resonances in the vocal and nasal cavities. The brain and the musculature not showed control the entire speech production process.

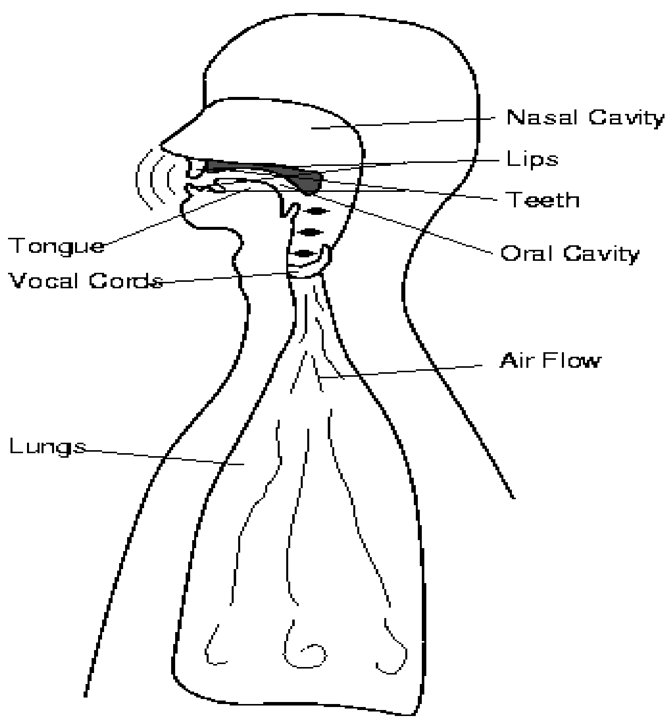


**Figure 6-1: The actual speech production system**

Speech sounds can be broken into three classes depending on their mode of excitation.

- Voiced sounds are produced when the vocal cords vibrate open and closed, thus interrupting the flow of air from the lungs to the vocal tract and producing quasi-periodic pulses of air as the excitation. The rate of the opening and closing gives the pitch of the sound. This can be adjusted by varying the shape of, and the tension in, the vocal cords, and the pressure of the air behind them. Voiced sounds show a high degree of periodicity at the pitch period, which is typically between 2 and 20 ms. This long-term periodicity can be seen in Figure 1-2 which shows a segment of voiced speech sampled at 8 kHz. Here the pitch period is about 8 ms or 64 samples. The power spectral density for this segment is shown in Figure 1-3.

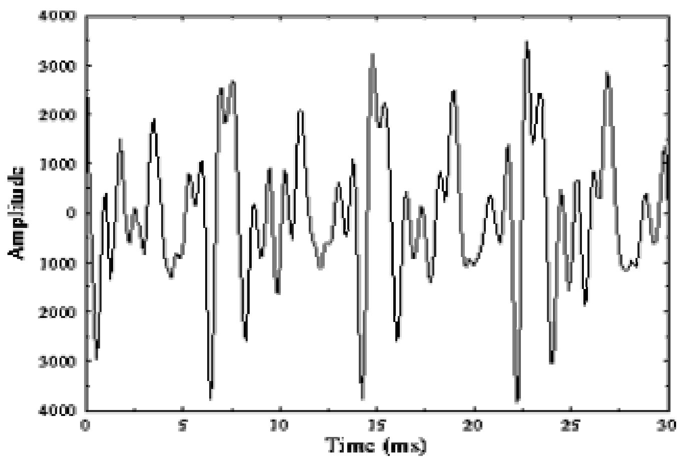


**Figure 1-7**: **Typical Segment of Voiced Speech**

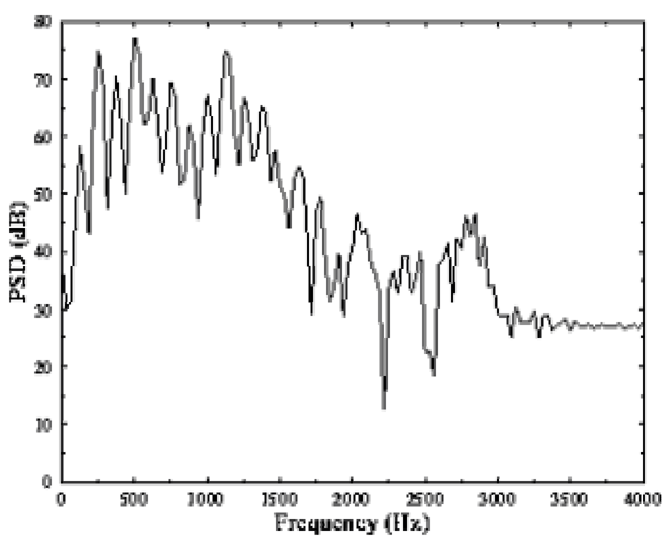


**Figure 1-8**: **Power Spectral Density for a Segment of Voiced Speech**

- Unvoiced sounds result when the excitation is a noise-like turbulence produced by forcing air at high velocities through a constriction in the vocal tract while the glottis is held open. Such sounds show little long-term periodicity as can be seen from Figures 1-4 and 1-5, although short-term correlations due to the vocal tract are still present.

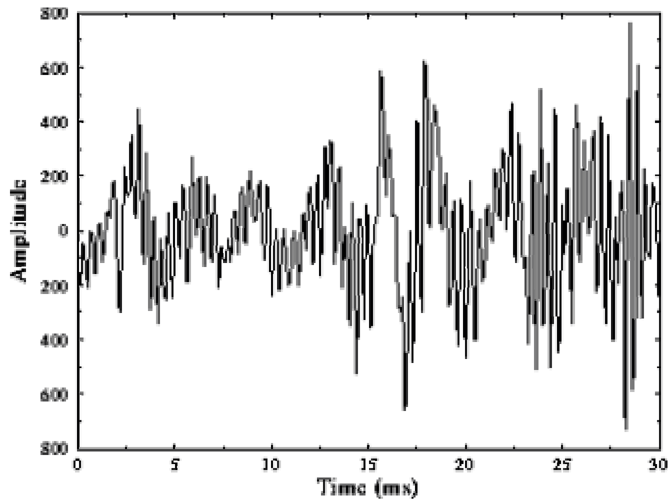


**Figure 1-9**: **Typical Segment of Unvoiced Speech**

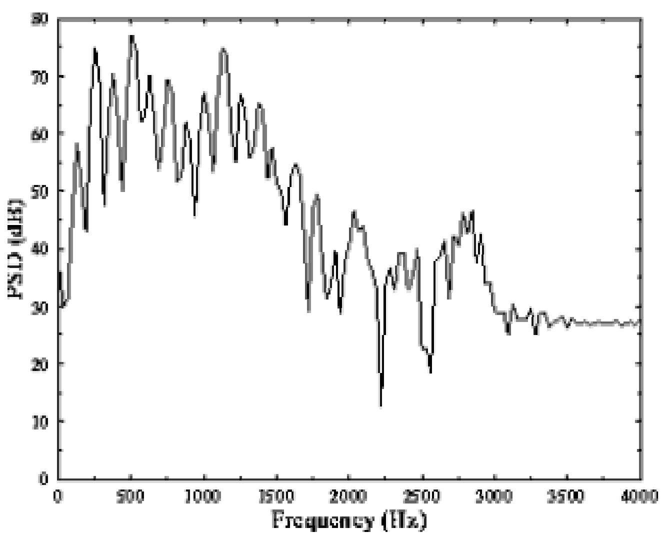


**Figure 1-10: Power Spectral Density for a Segment of Unvoiced Speech**

- Plosive sounds result when a complete closure is made in the vocal tract, and air pressure is built up behind this closure and released suddenly.

Some sounds cannot be considered to fall into any *one* of the three classes above, but are a mixture. For example voiced fricatives result when both vocal cord vibration and a constriction in the vocal tract are present.

### 1.3.2 Characteristics of Speech signal

- Speech signal is non uniform probability distribution of speech amplitude.
- Non-zero autocorrelation between successive speech samples.
- A speech spectrum has Non-flat nature.
- Existence of voiced and unvoiced segments in speech.
- Qasi periodicity of voiced speech signals.
- Speech signal is band limited.

### 1.3.3 OSCILOGRAM (WAVEFORM)

Physically the speech signal (actually all sound) is a series of pressure changes in the medium between the sound source and the listener. The most common representation of the speech signal is the oscillogram, often called the waveform. In this the time axis is the horizontal axis from left to right and the curve shows how the pressure increases and decreases in the signal. The utterance we have used for demonstration is **"phonetician"**, spoken by a male adult.

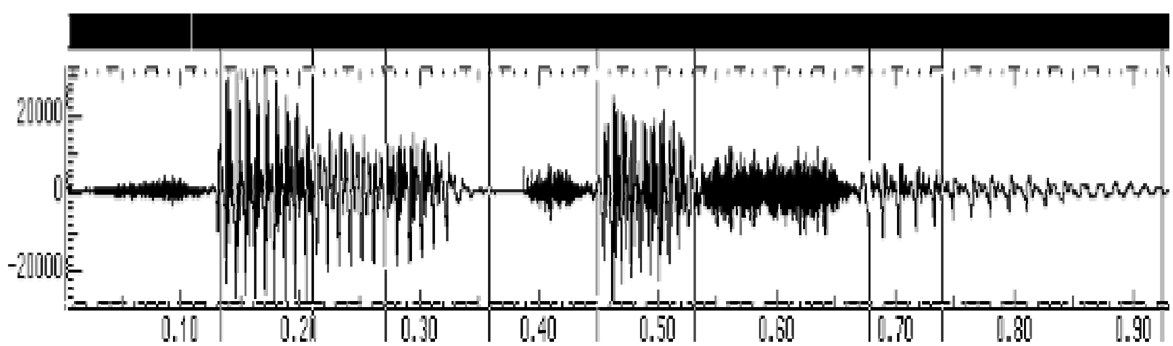


**Figure 1-11: WAVEFORM of the spoken word Phonetician**

### 1.3.4 Different Voice characteristics

The primary difference between adult male and female/prepubescent speech is pitch. Before puberty, pitch frequency for normal speech ranges between 150-400 Hz for both males and females. After puberty, the vocal cords of males undergo a physical change, which has the effect of lowering their pitch frequency to the range 80-160 Hz.

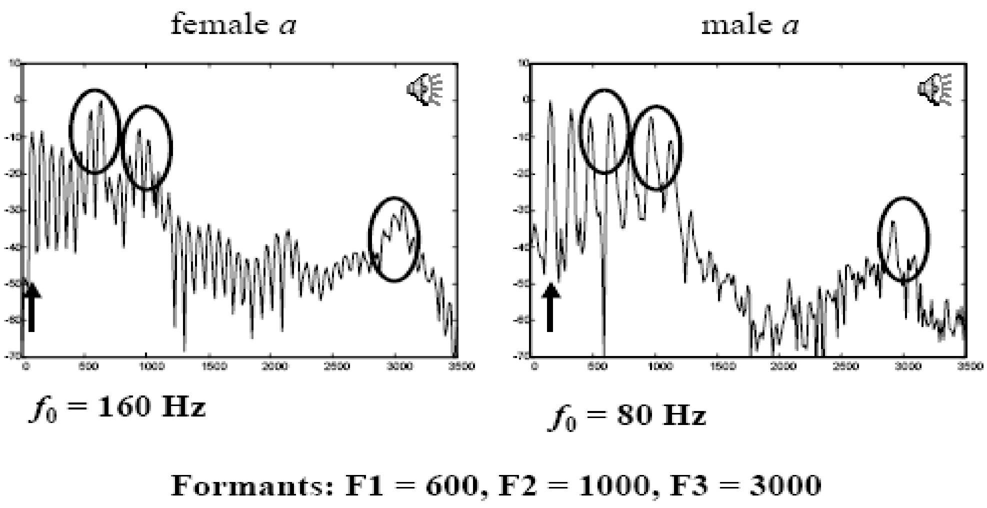


**Figure 1-12: Female and Male voice characteristics**

### *1.3.5 FUNDAMENTAL FREQUENCY (PITCH, F0)*

Another representation of the speech signal is the one produced by a pitch analysis. Speech is normally looked upon as a physical process consisting of two parts: a product of a sound source (the vocal chords) and filtering (by the tongue, lips, teeth etc). The pitch analysis tries to capture the fundamental frequency of the sound source by analyzing the final speech utterance. The fundamental frequency is the dominating frequency of the sound produced by the vocal chords. This analysis is quite difficult to perform. There are several problems in trying to decide which parts of the speech signal are voiced and which are not. It is also difficult to decipher the speech signal and try to find which oscillations originate from the sound source, and which are introduced by the filtering in the mouth. Several algorithms have been developed, but no algorithm has been found which is efficient and correct for all situations. The fundamental frequency is the strongest correlate to how the listener perceives the speakers' intonation and stress.

In the figure 1-8 fundamental frequency (often called F0 to be coherent with the terms for the formants, F1, F2 etc) is plotted against time. The F0 curve is visible only at points where the speech is voiced, i.e. where the vocal chords vibrate. The values for F0 lie between 100 and 150 Hz. This is common for a male speaker. The typical F0 range for a male is 80-200 Hz, and for females 150-350 Hz. Naturally, there is great variation in these figures.

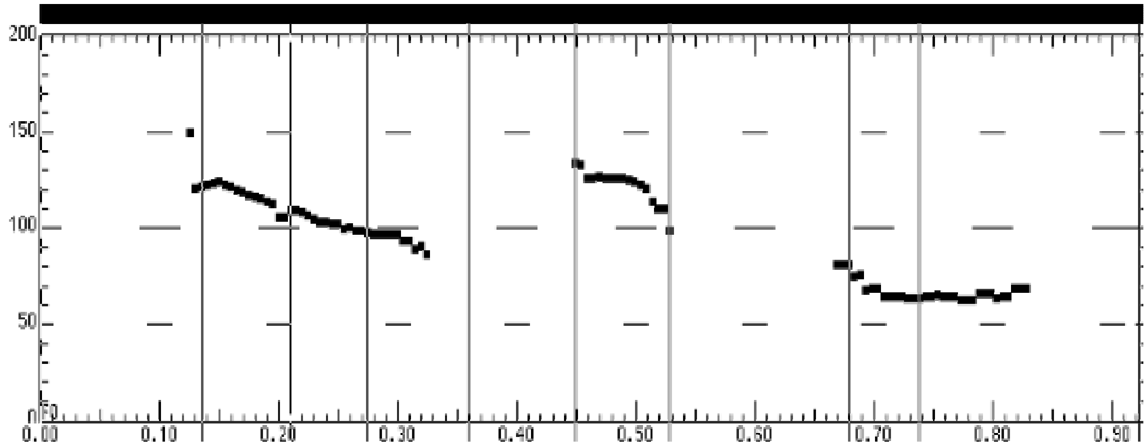


**Figure 1-13: Fundamental frequency against time**

### 1.3.6 SPECTRUM

According to general theories each periodical waveform may be described as the sum of a number of simple sine waves, each with a particular amplitude, frequency and phase. The spectrum gives a picture of the distribution of frequency and amplitude at a moment in time. The horizontal axis represents frequency, and the vertical axis amplitude. If we want to plot the spectrum as a function of time we need a way of representing a three-dimensional diagram, one such representation is the spectrogram. The figure shows the spectrum 0.15 seconds into the utterance, in the beginning of the "o" vowel.

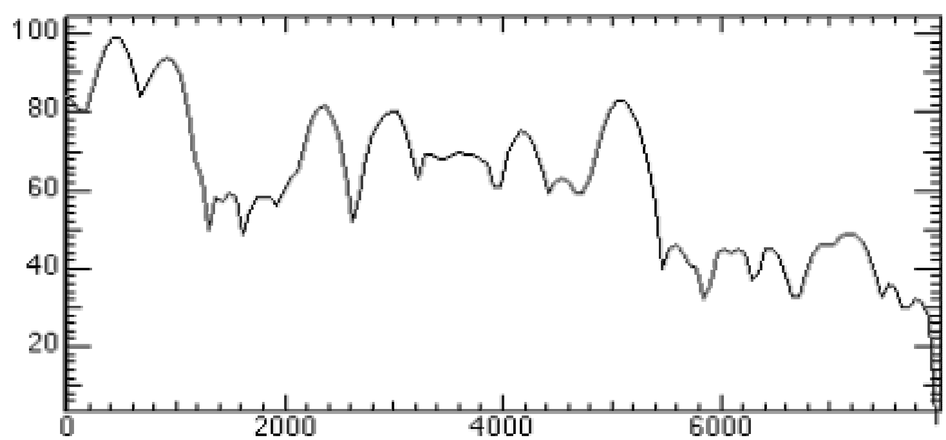


**Figure 1-14: Spectrum**

### 1.3.7 SPECTROGRAM

In the spectrogram the time axis is the horizontal axis, and frequency is the vertical axis. The third dimension, amplitude, is represented by shades of darkness. Consider the spectrogram to be a number of spectrums in a row, looked upon "from above", and where the highs in the spectra are represented with dark spots in the spectrogram.
The figure 1-10 shows how different the speech sounds are from a spectral point of view. In the unvoiced fricative sounds, the energy is concentrated high up in the frequency band, and quite disorganized (noise-like) in its appearance. In other unvoiced sounds, e.g. the plosives, much of the speech sound actually consists of silence until strong energy appears at many frequency bands, as an "explosion". The voiced sounds appear more organized. The spectrum highs (dark spots) actually form horizontal bands across the spectrogram. These bands represent frequencies where the shape of the mouth gives resonance to sounds. The bands are called formants, and are numbered from the bottom up as F1, F2, F3 etc. The positions of the formants are different for different sounds and they can often be predicted for each phoneme.

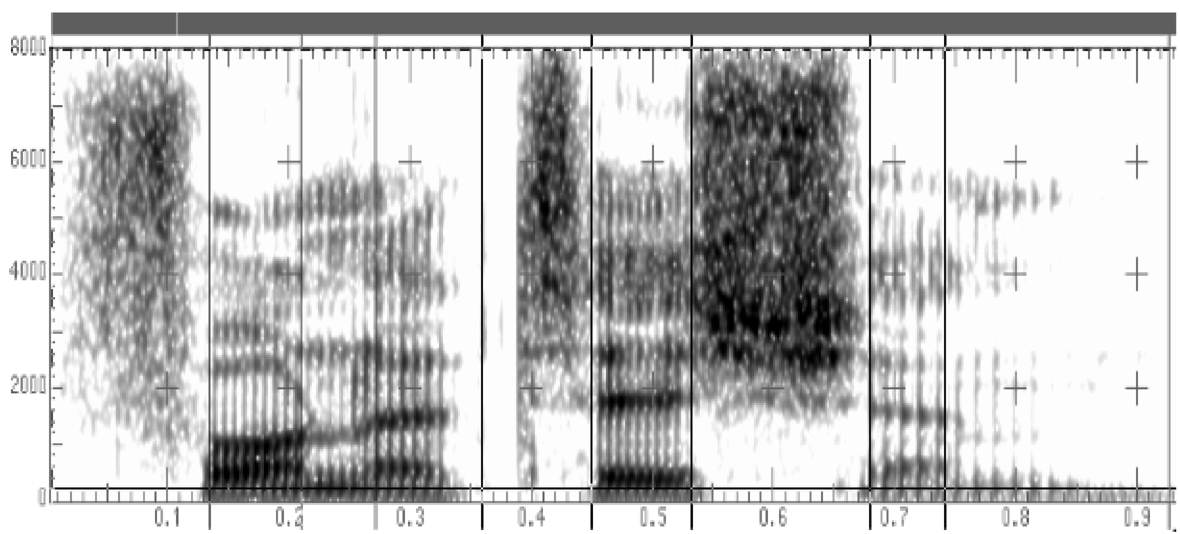


**Figure 1-15: Spectrogram**

## *1.4 Problem Formulation*

The present work focuses on voice controlled remote devices. Voice recognition is the main part of the problem and has been carried out with the help of LabVIEW.As LabVIEW has tremendous potential to access the hardware and strong inbuilt Library. The thesis work also provides easiness to control the devices which are located at far place. DTMF signal is used to send the information from the control section to the remote location. As existing telephone lines helps us to realize the system and hence cost of the project is very less.

# CHAPTER 2

# DTMF SIGNAL

## *2.1 Introduction*

When you press a button in the telephone set keypad, a connection is made that generates a resultant signal of two tones at the same time. These two tones are taken from a row frequency and a column frequency. The resultant frequency signal is called "*Dual Tone Multiple Frequency*". These tones are identical and unique.

A DTMF signal is the algebraic sum of two different audio frequencies, and can be expressed as follows:

$f(t) = A_0\sin(2*\Pi*f_a*t) + B_0\sin(2*\Pi*f_b*t) + ..........$ -------> (1)

Where $f_a$ and $f_b$ are two different audio frequencies with A and B as their peak amplitudes and f as the resultant DTMF signal. $f_a$ belongs to the low frequency group and $f_b$ belongs to the high frequency group.

Each of the low and high frequency groups comprise four frequencies from the various keys present on the telephone keypad; two different frequencies, one from the high frequency group and another from the low frequency group are used to produce a DTMF signal to represent the pressed key.

The amplitudes of the two sine waves should be such that

$(0.7 < (A/B) < 0.9)$ V --------> (2)

The frequencies are chosen such that they are not the harmonics of each other. The frequencies associated with various keys on the keypad are shown in figure 2-2. A Typical frequency is shown in the figure below:

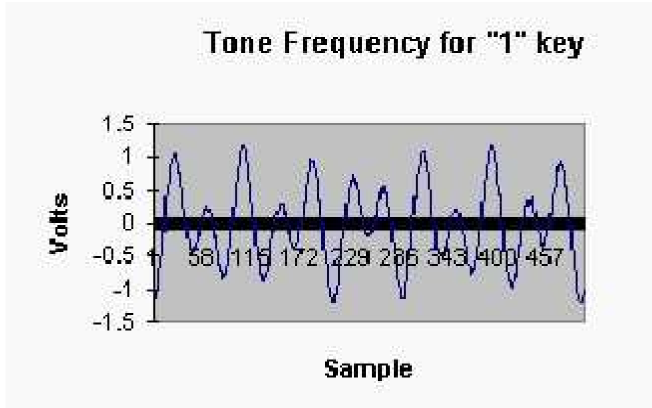


**Figure 2-1: Tone frequency for “1” key**

When you send these DTMF signals to the telephone exchange through cables, the servers in the telephone exchange identifies these signals and makes the connection to the person you are calling. When you press the digit 5 in the keypad it generates a resultant tone signal, which is made up of frequencies 770Hz, and 1336Hz. Pressing digit 8 will produce the tone taken from tones 852Hz and 1336Hz. In both the cases, the column frequency 1336 Hz is the same. These signals are digital signals, which are symmetrical with the sinusoidal wave. The row and column frequencies are given below:

| LOW FREQUENCY GROUP | HIGH FREQUENCY GROUP | | | |
|---|---|---|---|---|
| | 1209Hz | 1336Hz | 1447Hz | 1663Hz |
| 697Hz | 1 | 2 | 3 | A |
| 770Hz | 4 | 5 | 6 | B |
| 852Hz | 7 | 8 | 9 | C |
| 941Hz | * | 0 | # | D |

**Figure 2-2: DTMF Keypad**

Along with this DTMF generator in our telephone set provides a set of special purpose groups of tones, which is normally not used in our keypad. These tones are identified as 'A', 'B', 'C', 'D'. These frequencies have the same column frequency but uses row frequencies given in the table in figure 2.2. These tones are used for communication signaling. Due to its accuracy and uniqueness, these DTMF are used in controlling systems using telephones. By using some DTMF generating IC's (UM-91214, UM-91214, etc) we can generate DTMF tones without depending on the telephone set.

## *2.2 DTMF Signal Generation*

***DTMF*** signal is generated using the IC UM-91214B. This IC produces DTMF. It contains four row frequencies & three column frequencies. The pins of IC UM-91214B from 12 to 14 produces high frequency column group and pins from 15 to 18 produces the low frequency row group. By pressing any key in the keyboard corresponding DTMF signal is available in its output pin at pin no.7. For producing the appropriate signals it is necessary that a crystal oscillator of 3.58MHz is connected across its pins 3 & 4 so that it makes a part of its internal oscillator.

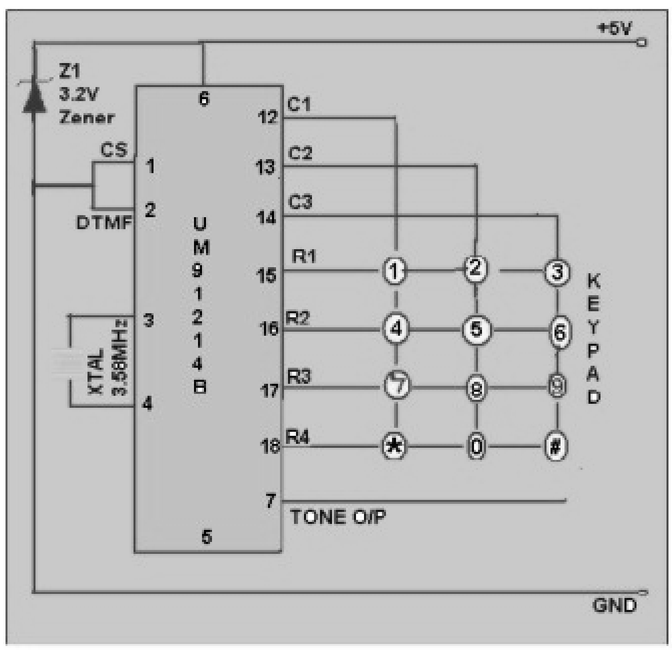


**Figure 2-3: DTMF Encoder**

This encoder IC requires a voltage of 3V. For that IC is wired around 4.5V battery. And 3V backup Vcc for this IC is supplied by using 3.2V zener diode.

The row and column frequency of this IC is as on the figure 2-3. By pressing the number 5 in the key pad the output tone is produced which is the resultant of addition of two frequencies, at pin no. 13 & pin no.16 of the IC and respective tone which represents number '5' in key pad is produced at pin no.7 of the IC . This signal is sent to the local control system through telephone line via exchange.

## *2.3 DTMF Decoder*

The receiver circuit decodes the DTMF signal and IC MT-8870/KT-3170 serves as DTMF decoder. This IC takes DTMF signal coming via telephone line and converts that signal into respective BCD number. It uses same oscillator frequency used in the remote section so same crystal oscillator with frequency of 3.85M Hz is used in this IC. The MT-8870 is a full DTMF receiver that integrates both band split filter and decoder functions into a single 18-pin DIP. Its filter section uses switched capacitor technology for both the high and low group filters and for dial tone rejection. Its decoder uses digital counting techniques to detect and decode all 16 DTMF tone pairs into a 4-bit code. External component count is minimized by provision of an on-chip differential input amplifier, clock generator, and latched tri-state interface bus. Minimal external components required include a low-cost 3.579545 MHz crystal, a timing resistor, and a timing capacitor. The MT-8870 can also inhibit the decoding of fourth column digits. MT-8870 operating functions include a band split filter that separates the high and low tones of the received pair, and a digital decoder that verifies both the frequency and duration of the received tones before passing the resulting 4-bit code to the output bus.

The low and high group tones are separated by applying the dual-tone signal to the inputs of two $6^{th}$ order switched capacitor band pass filters with bandwidths that correspond to the bands enclosing the low and high group tones. The filter also incorporates notches at 350 and 440 Hz, providing excellent dial tone rejection. Each filter output is followed by a single-order switched capacitor section that smoothes the signals prior to limiting. Signal limiting is performed by high gain comparators provided with hysteresis to prevent detection of unwanted low-level signals and noise. The MT-8870 decoder uses a digital counting technique to determine the frequencies of the limited tones and to verify that they correspond to standard DTMF frequencies. When the detector recognizes the simultaneous presence of two valid tones (known as signal condition), it raises the Early Steering flag (ESt). Any subsequent loss of signal condition will cause ESt to fall. Before a decoded tone pair is registered, the receiver checks for valid signal duration (referred to as character- recognition-condition). This check is performed by an external RC time constant driven by ESt. A short delay to allow the output latch to settle, the delayed steering output flag (StD) goes high, signaling that a received tone pair has been registered. The contents of the output latch are made available on the 4-bit output bus by raising the three state control input (OE) to logic high. Inhibit mode is enabled by a logic high input to pin 5 (INH). It inhibits the detection of 1633 Hz.

The output code will remain the same as the previous detected code. On the MT-8870 models, this pin is tied to ground (logic low).The input arrangement of the MT-8870 provides a differential input operational amplifier as well as a bias source (VREF) to bias the inputs at mid-rail. Provision is made for connection of a feedback resistor to the op-amp output (GS) for gain adjustment. The internal clock circuit is completed with the addition of a standard 3.579545 MHz crystal. The input arrangement of the

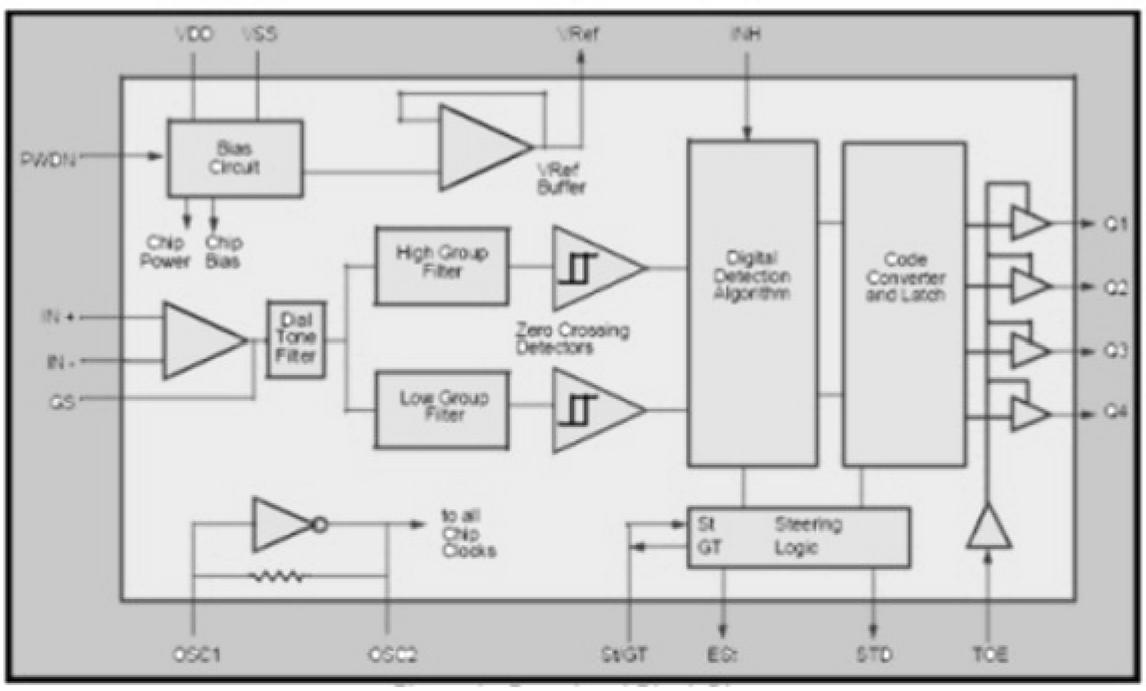


**Figure 2-4: DTMF Decoder**

MT-8870 provides a differential input operational amplifier as well as a bias source (VREF) to bias the inputs at mid-rail. Provision is made for connection of a feedback resistor to the op-amp output (GS) for gain adjustment. The internal clock circuit is completed with the addition of a standard 3.579545 MHz crystal.

# CHAPTER 3

# LABVIEW AND SOUND

## *3.1 Introduction*

LabVIEW is a graphical programming language used as a powerful and flexible instrumentation and analysis software system in industry and academia. LabVIEW uses a graphical programming language, G, to create programs called virtual instruments (VIs) in a pictorial form, eliminating much of the syntactical details of other text-based programming languages, such as C and MATLAB. LabVIEW includes many tools for data acquisition, analysis, and display of results. LabVIEW is available for all the major platforms and is easily portable across platforms. LabVIEW has the ability to create stand-alone executable applications, which run at compiled execution speeds. LabVIEW uses icons instead of lines of text to create applications. In contrast to text-based programming languages, where instructions determine program execution, LabVIEW uses dataflow programming, where the flow of data determines execution.

In LabVIEW, you build a user interface by using a set of tools and objects. The user interface is known as the front panel. You then add code using graphical representations of functions to control the front panel objects. The block diagram contains this code. In some ways, the block diagram resembles a flowchart.

### *3.2 LabVIEW Block Diagram and Front panel*

LabVIEW programs are called virtual instruments, or VIs, because their appearance and operation imitate physical instruments, such as oscilloscopes and millimeters. LabVIEW contains a comprehensive set of tools for acquiring, analyzing, displaying, and storing data, as well as tools to help you troubleshoot your code. In LabVIEW, you build a user interface, or front panel, with controls and indicators. Controls are knobs, push buttons, dials, and other input devices. Indicators are graphs, LEDs, and other displays.

After you build the user interface, you add code using VI and structures to control the front panel objects. The block diagram contains this code. Use LabVIEW to communicate with hardware such as data acquisition, vision, and motion control devices and GPIB, PXI, VXI, RS-232, and RS-485 instruments.

Each VI contains three parts.

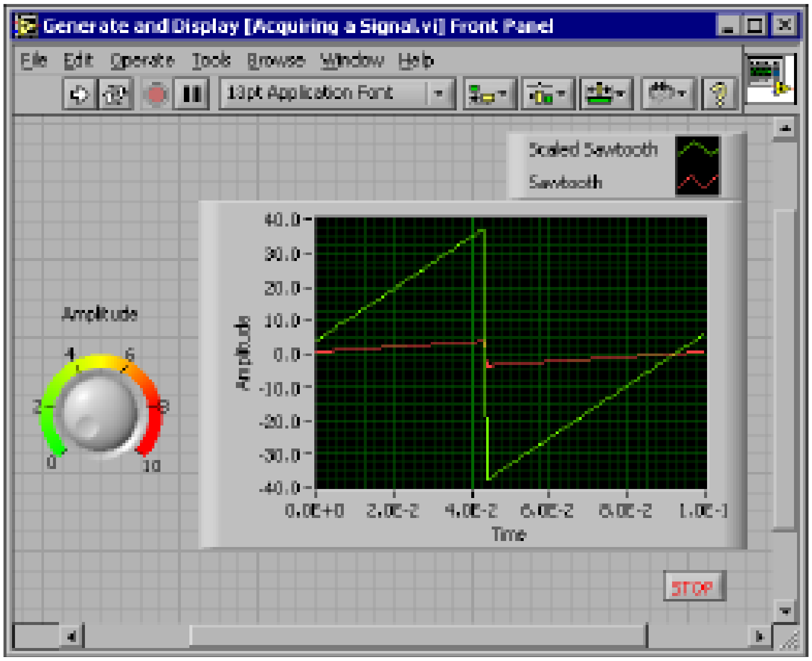


**Figure 3-16: Front Panel, Tool Box And Control Box**

1. The front panel contains the user interface control inputs, such as knobs, sliders, and push buttons, and output indicators to produce such items as charts and graphs. Inputs can be fed into the system using the mouse or keyboard. A typical front panel, used to provide intuitive GUIs to vary the parameters of the algorithm.

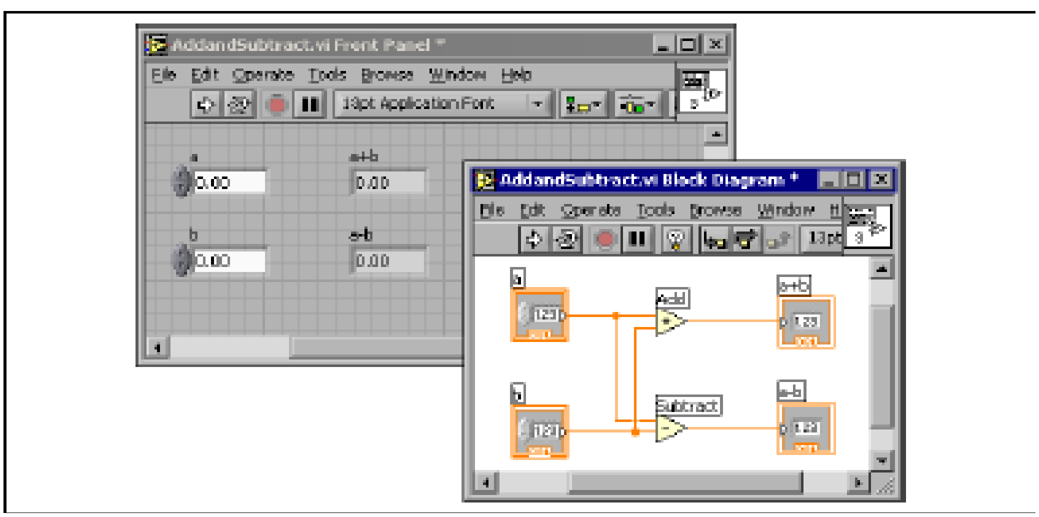


**Figure 3-2: Block Diagram And Function Box**

2 .The block diagram is the equivalent of a “source code” for the VI. The blocks are interconnected, using wires to indicate the data flow. Front-panel indicators pass data from the user to their corresponding terminals on the block diagram. The results of the operation are then passed back to the front-panel indicators.

3. Sub-VIs are analogous to subroutines in conventional programming languages.

### 3.2.1 Terminals

The terminals represent the data type of the control or indicator. We can configure front panel controls or indicators to appear as icon or data type terminals on the block diagram. By default, front panel objects appear as icon terminals. For example, a knob icon terminal shown at left represents a knob on the front panel. The DBL at the bottom of the terminal represents a data type of double-precision, floating-point numeric. A DBL terminal, shown at left, represents a double-precision, floating-point numeric control or indicator. Terminals are entry and exit ports that exchange information between the front panel and block diagram. Data you enter into the front panel controls enter the block diagram through the control terminals.

### 3.2.2 Nodes

Nodes are objects on the block diagram that have inputs and/or outputs and perform operations when a VI runs. They are analogous to statements, operators, functions, and subroutines in text-based programming languages. Wires you transfer data among block diagram objects through wires. Wires are different colors, styles, and thicknesses, depending on their data types. A broken wire appears as a dashed black line with a red X in the middle.

### 3.2.3 Structures

Structures are graphical representations of the loops and case statements of text-based programming languages. Use structures on the block diagram to repeat blocks of code and to execute code conditionally or in a specific order.

### 3.2.4 Icon and Connector Pane

After you build a VI front panel and block diagram, build the icon and the connector pane so you can use the VI as a subVI. Every VI displays an icon, such as the one shown at left, in the upper right corner of the front panel and block diagram windows. An icon is a graphical representation of a VI. It can contain text, images, or a combination of both. If you use a VI as a subVI, the icon identifies the subVI on the block diagram of the VI. You can double-click the icon to customize or edit it. You also need to build a connector pane, shown at left, to use the VI as a subVI. The connector pane is a set of terminals that correspond to the controls and indicators of that VI, similar to the parameter list of a function call in text-based programming languages. The connector pane defines the inputs and outputs you can wire to the VI so you can use it as a subVI. A connector pane receives data at its input terminals and passes the data to the block diagram code through the front panel controls and receives the results at its output terminals from the front panel indicators. When you view the connector pane for the first time, you see a connector pattern. You can select a different pattern if you want to. The connector pane generally has one terminal for each control or indicator on the front panel. You can assign up to 28 terminals to a connector pane. If you anticipate changes to the VI that would require a new input or output, leave extra terminals unassigned.

## 3.3 LabVIEW Environment

LabVIEW (Virtual Instrumentation) palettes, tools, and menus to build the front panels and block diagrams of VIs (Virtual Instrumentation). You can customize the Controls and Functions palettes, and you can set several works Palette. The Controls palette is available only environment options.

### 3.3.1 Controls on the front panel

The Controls palette contains the controls and indicators you use to create the front panel. The controls and indicators are located on sub palettes based on the types of controls and indicators. The controls and indicators located on the Controls palette depend on the palette view currently selected. Select Window»Show Controls Palette or right-click the front panel workspace to display the Controls palette. You can place the

Controls palette anywhere on the screen. LabVIEW retains the Controls palette position and size so when you restart LabVIEW, the palette appears in the same position and has the same size. You can change the way the Controls palette appears.

### 3.3.2 Functions Palette

The Functions palette is available only on the block diagram. The Functions palette contains the VIs and functions you use to build the block diagram. The VIs and functions are located on sub palettes based on the types of VIs and functions. The VIs and functions located on the Functions palette depend on the palette view currently selected. Select Window»Show Functions Palette or right-click the block diagram workspace to display the Functions palette. You can place the Functions palette anywhere on the screen. LabVIEW retains the Functions palette position and size so when you restart LabVIEW, the palette appears in the same position and has the same size. You can change the way the Functions palette appears. When you click a sub palette icon, the entire palette changes to the sub palette you selected. Click an object on the palette to place the object on the cursor so you can place it on the front panel or block diagram. You also can right-click a VI icon on the palette and select Open VI from the shortcut menu to open the VI.

### 3.3.3 Tools Palette

The Tools palette is available on the front panel and the block diagram. A tool is a special operating mode of the mouse cursor. The cursor corresponds to the icon of the tool selected in the palette. Use the tools to operate and modify front panel and block diagram objects. Select Window»Show Tools Palette to display the Tools palette. You can place the Tools palette anywhere on the screen. LabVIEW retains the Tools palette position so when you restart LabVIEW, the palette appears in the same position. Tip Press the <Shift> key and right-click to display a temporary version of the Tools palette at the location of the cursor. If automatic tool selection is enabled and you move the cursor over objects on the front panel or block diagram, LabVIEW automatically selects the corresponding tool from the Tools palette. You can disable automatic tool selection by clicking the Automatic Tool Selection button on the Tools palette, shown at left. Press the

<Shift-Tab> keys or click the Automatic Tool Selection button to enable automatic tool selection again. You also can disable automatic tool selection by manually selecting a tool on the Tools palette. Press the <Tab> key or click the Automatic Tool Selection button on the Tools palette to enable automatic tool selection again.

## 3.4 Menus and the Toolbar

The menu and toolbar items to operate and modify front panel and block diagram objects. Use the toolbar buttons to run VIs.

### 3.4.1 Menus

The menus at the top of a VI window contain items common to other applications, such as Open, Save, Copy, and Paste, and other items specific to LabVIEW. Some menu items also list shortcut key combinations. (Mac OS) The menus appear at the top of the screen. (Windows and UNIX) The menus display only the most recently used items by default. Click the arrows at the bottom of a menu to display all items. You can display all menu items by default by selecting Tools»Options and selecting Miscellaneous from the top pulls-down menu.

### 3.4.2 Shortcut Menus

The most often-used menu is the object shortcut menu. All LabVIEW objects and empty space on the front panel and block diagram have associated shortcut menus. Use the shortcut menu items to change the look or behavior of front panel and block diagram objects. To access the shortcut menu, right-click the object, front panel, or block diagram.

### 3.4.3 Shortcut Menus in Run Mode

When a VI is running, or is in run mode, all front panel objects have an abridged set of shortcut menu items. Use the abridged shortcut menu items to cut, copy, or paste the contents of the object, to set the object to its default value, or to read the description of the object. Some of the more complex controls have additional options. Toolbar Use the toolbar buttons to run and edit a VI. When you run a VI, buttons appear on the toolbar that you can use to debug the VI.

## 3.5 *Building the Front Panel*

The front panel is the user interface of a VI. Generally, you design the front panel first, then design the block diagram to perform tasks on the inputs and outputs you create on the front panel You build the front panel with controls and indicators, which are the interactive input and output terminals of the VI, respectively. Controls are knobs, push buttons, dials, and other input devices. Indicators are graphs, LEDs, and other displays. Controls simulate instrument input devices and supply data to the block diagram of the VI. Indicators simulate instrument output devices and display data the block diagram acquires or generates. Select Window»Show Controls Palette to display the Controls palette, then select controls and indicators from the Controls palette and place them on the front panel.

### *3.5.1 Configuring Front Panel Objects*

Property dialog boxes or shortcut menus to configure how controls and indicators appear or behave on the front panel. Use property dialog boxes when you want to configure a front panel control or indicator through a dialog box that includes context help or when you want to set several properties at once for an object. Use shortcut menus to quickly configure common control and indicator properties. Options in the property dialog boxes and shortcut menus differ depending on the front panel object. Any option you set using a shortcut menu overrides the option you set with a property dialog box. You cannot access property dialog boxes for a control or indicator while a VI runs.

### *3.5.2 Showing and Hiding Optional Elements*

Front panel controls and indicators have optional elements you can show or hide. Set the visible elements for the control or indicator on the Appearance tab of the property dialog box for the front panel object. You also can set the visible elements by right-clicking an object and selecting Visible Items from the shortcut menu. Most objects are a label and a caption. Refer to the Labeling section of this chapter for more information about labels and captions.

### *3.5.3 Changing Controls to Indicators and Indicators to Controls*

LabVIEW initially configures objects in the Controls palette as controls or indicators based on their typical use. For example, if you select a toggle switch it appears on the front panel as a control because a toggle switch is usually an input device. If you select an LED, it appears on the front panel as an indicator because an LED is usually an output device. Some palettes contain a control and an indicator for the same type or class of object. For example, the Numeric palette contains a digital control and a digital indicator. You can change a control to an indicator by right-clicking the object and selecting Change to Indicator from the shortcut menu, and you can change an indicator to a control by right-clicking the object and selecting Change to Control from the shortcut menu.

### *3.5.4 Replacing Front Panel Objects*

We can replace a front panel object with a different control or indicator. When you right-click an object and select Replace from the shortcut menu, a temporary Controls palette appears, even if the Controls palette is already open. Select a control or indicator from the temporary Controls palette to replace the current object on the front panel. Selecting Replace from the shortcut menu preserves as much information as possible about the original object, such as its name, description, default data, dataflow direction (control or indicator), color, size, and so on. However, the new object retains its own data type. Wires from the terminal of the object or local variables remain on the block diagram, but they might.

### *3.5.5 Front Panel Controls and Indicators*

Use the front panel controls and indicators located on the Controls palette to build the front panel. Controls are knobs, push buttons, dials, and other input devices. Indicators are graphs, LEDs, and other displays. Controls simulate instrument input devices and supply data to the block diagram of the VI. Indicators simulate instrument output devices and display data the block diagram acquires or generates.

### 3.5.6 3-D and Classic Controls and Indicators

Many front panel objects have a high-color, three-dimensional appearance. Set your monitor to display at least 16-bit color for optimal appearance of the objects. The 3D front panel objects also have corresponding low-color, two-dimensional objects. Use the 2D controls and indicators located on the Classic Controls palette to create VIs for 256-color and 16-color monitor settings. Select File»VI Properties and select Editor Options from the Category pull-down menu to change the style of control or indicator LabVIEW creates when you right-click a terminal and select Create»Control or Create»Indicator from the shortcut menu. Select Tools»Options and select Front Panel from the top pull-down menu to change the style of control or indicator LabVIEW creates in new VIs when you right-click a terminal and select Create»Control or Create» Indicator from the shortcut menu.

### 3.5.7 Slides, Knobs, Dials, Digital Displays, and Time Stamps

Numeric controls and indicators located on the Numeric and Classic Numeric palettes to simulate slides, knobs, dials, and digital displays. The palette also includes color boxes and color ramps for setting color values and a time stamp for setting the time and date for the data. Use numeric controls and indicators to enter and display numeric data.

### 3.5.8 Slide Controls and Indicators

The slide controls and indicators include vertical and horizontal slides, a tank, and a thermometer. Change the value of a slide control or indicator by using the Operating tool to drag the slider to a new position, by clicking a point of the slide object, or by using the optional digital display. If you drag the slider to a new position and the VI is running during the change, the control passes intermediate values to the VI, depending on how often the VI reads the control. Slide controls or indicators can display more than one value. Right-click the object and select Add Slider from the shortcut menu to add more sliders. The data type of a control with multiple sliders is a cluster that contains each of the numeric values.

### *3.5.9 Rotary Controls and Indicators*

The rotary controls and indicators include knobs, dials, gauges, and meters. The rotary objects operate similarly to the slide controls and indicators. Change the value of a rotary control or indicator by moving the needles, by clicking a point of the rotary object, or by using the optional digital display. Rotary controls or indicators can display more than one value. Right-click the object and select Add Needle from the shortcut menu to add new needles. The data type of a control with multiple needles is a cluster that contains each of the numeric values. Digital controls and indicators are the simplest way to enter and display numeric data. You can resize these front panel objects horizontally to accommodate more digits. You can change the value of a digital control or indicator by using the following methods:

- Operating or Labeling tool to click inside the digital display window and enter numbers from the keyboard.
- Operating tool to click the increment or decrement arrow buttons of a digital control.
- Operating or labeling tool to place the cursor to the right of the digit you want to change and press the up or down arrow key on the keyboard.

### *3.5.10 Time Stamp Control and Indicator*

Time stamp control and indicator to send and retrieve a time and date value to or from the block diagram. You can change the value of the time stamp control by using the following methods:

- Right-click the constant and select Format & Precision from the shortcut menu.
- Click the Time/Date Browse button, shown at left, to display the Set Time and Date dialog box.
- Select Data Operations»Set Time and Date from the shortcut menu to display the Set Time and Date dialog box. You also can right-click the time stamp control and select Data Operations»Set Time to Now from the shortcut menu.

### *3.5.11 Buttons, Switches, and Lights*

Boolean controls and indicators to simulate buttons, switches, and lights. Use Boolean controls and indicators to enter and display Boolean (TRUE/FALSE) values. For example, if you are monitoring the temperature of an experiment, you can place a Boolean warning light on the front panel to indicate when the temperature goes above a certain level. Use the shortcut menu to customize the appearance of Boolean objects and how they behave when you click them.

### *3.5.12 Text Entry Boxes, Labels, and Path Displays*

String and path controls and indicators to simulate text entry boxes and labels and to enter or return the location of a file or directory.

### *3.5.13 String Controls and Indicators*

Enter or edit text in a string control on the front panel by using the Operating tool or the Labeling tool. By default, new or changed text does not pass to the block diagram until you terminate the edit session. You terminate the edit session by clicking elsewhere on the panel, changing to a different window, clicking the Enter button on the toolbar, or pressing the <Enter> key on the numeric keypad. Pressing the <Enter> key on the keyboard enters a carriage return.

### *3.5.14 Array and Cluster Controls and Indicators*

Use the array and cluster controls and indicators located on the Array & Cluster and Classic Array & Cluster palettes to create arrays and clusters of other controls and indicators. The Array & Cluster palettes also contain standard error cluster controls and indicators and the variant control.

## *3.6 Building the Block Diagram*

After you build the front panel, you add code using graphical representations of functions to control the front panel objects. The block diagram contains this graphical source code.

### 3.6.1 Relationship between Front Panel Objects and Block Diagram Terminals

Front panel objects appear as terminals on the block diagram. Double-click a block diagram terminal to highlight the corresponding control or indicator on the front panel. Terminals are entry and exit ports that exchange information between the front panel and block diagram. Data you enter into the front panel controls enter the block diagram through the control terminals. During execution, the output data flow to the indicator terminals, where they exit the block diagram, reenter the front panel, and appear in front panel indicators.

### 3.6.2 Block Diagram Objects

Objects on the block diagram include terminals, nodes, and functions. You build block diagrams by connecting the objects with wires.

### 3.6.3 Block Diagram Terminals

We can configure front panel controls or indicators to appear as icon or data type terminals on the block diagram. By default, front panel objects appear as icon terminals. For example, a knob icon terminal, shown at left, represents a knob on the front panel. The DBL at the bottom of the terminal represents a data type of double-precision, floating-point numeric. A DBL terminal, shown at left, represents a double-precision, floating-point numeric control or indicator. Right-click a terminal and select Display Icon from the shortcut menu to remove the checkmark and to display the data type for the terminal. Use icon terminals to display the types of front panel objects on the block diagram, in addition to the data types of the front panel objects. Use data type terminals to conserve space on the block diagram. Note Icon terminals are larger than data type terminals, so you might unintentionally obscure other block diagram objects when you convert a data type terminal to an icon terminal. A terminal is any point to which you can attach a wire, other than to another wire. LabVIEW has control and indicator terminals, node terminals, constants, and specialized terminals on structures, such as the input and output terminals on the Formula Node. You use wires to connect terminals and pass data to other terminals. Right-click a block diagram object and select Visible Items»Terminals from the shortcut menu to view the terminals. Right-click the object and select Visible Items»Terminals again to hide the terminals. This shortcut menu item is not available for expandable VIs and functions.

### *3.6.4 Control and Indicator Data Types*

The color and symbol of each terminal indicate the data type of the control or indicator. Control terminals have a thicker border than indicator terminals. Also, arrows appear on front panel terminals to indicate whether the terminal is a control or an indicator. An arrow appears on the right if the terminal is a control, and an arrow appears on the left if the terminal is an indicator.

### *3.6.5 Functions Overview*

Functions are the essential operating elements of LabVIEW. Function icons on the Functions palette have pale yellow backgrounds and black foregrounds. Functions do not have front panels or block diagrams but do have connector panes. You cannot open nor edit a function. The Functions palette also includes the VIs that ship with LabVIEW. Use these VIs as subVIs when you build data acquisition, instrument control, communication, and other VIs.

### *3.6.6 Numeric Functions*

Numeric functions to create and perform arithmetic, trigonometric, logarithmic, and complex mathematical operations on numbers and to convert numbers from one data type to another.

### *3.6.7 Boolean Functions*

Boolean functions to perform logical operations on single Boolean values or arrays of Boolean values, such as the following tasks:

- Change a TRUE value to a FALSE value and vice versa.
- Determine which Boolean value to return if you receive two or more Boolean values.
- Convert a Boolean value to a number (either 1 or 0).
- Perform compound arithmetic on two or more Boolean values.

### *3.6.8 String Functions*

String functions to perform the following tasks:

• Concatenate two or more strings.

• Extract a subset of strings from a string.

• Search for and replace characters or subsets of strings in a string.

• Convert numeric data into strings.

• Format a string for use in a word processing or spreadsheet application.

### *3.6.9 Array Functions*

Array functions to create and manipulate arrays, such as the following tasks:

• Extract individual data elements from an array.

• Add individual data elements to an array.

• Split arrays.

### *3.6.10 Cluster Functions*

Cluster functions to create and manipulate clusters, such as the following tasks:

• Extract individual data elements from a cluster.

• Add individual data elements to a cluster.

• Break a cluster out into its individual data elements.

### *3.6.11 Comparison Functions*

Comparison functions to compare Boolean values, strings, numerics, arrays, and clusters.

### *3.6.12 Time and Dialog Functions*

Use the Time and Dialog functions to perform the following tasks:

• Manipulate the speed at which an operation executes.

• Retrieve time and date information from your computer clock.

• Create dialog boxes to prompt users with instructions.

## 3.7 Sound VIs

### 3.7.1 Beep (Not in Base Package)

Causes the system to issue an audible tone. LabVIEW ignores the frequency, duration, and intensity, and error parameters.

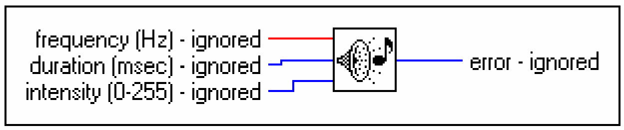


**Figure 3-3: Beep**

### 3.7.2 Get Sound Info

Returns the information associated with a sound operation associated with the task ID.

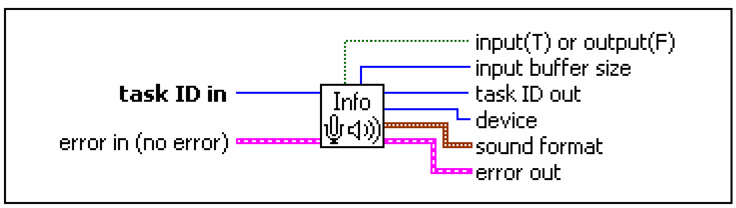


**Figure 3-4: Get Sound Info**

### 3.7.3 Snd Read Waveform

Reads data from a sound input device. You specify the size of the data through buffer size**.**

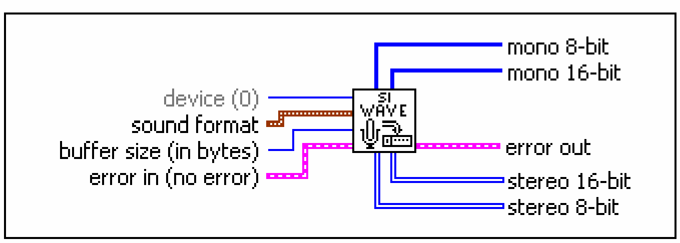


**Figure 3-5: Snd Read Waveform**

### 3.7.4 Snd Write Waveform

Writes a piece of data to the sound output device and waits until the device finishes playing the data.

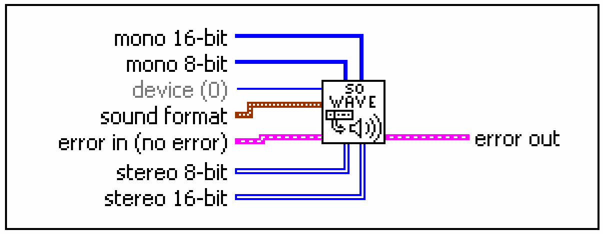


**Figure 3-6: Snd Write Waveform**

## 3.8 Sound File VIs

### 3.8.1 Snd Read Wave File

Retrieves a PC wave file (.wav) specified in wave file path**.** The information returned includes both waveform data and sound format data, which is necessary for configuring a sound device to play the waveform.

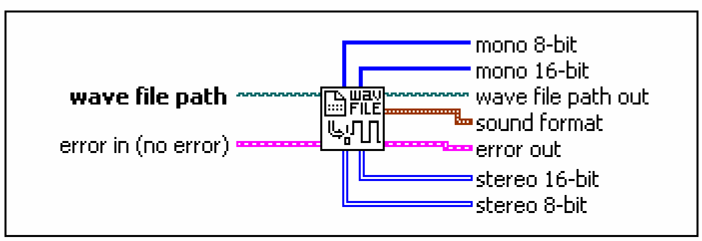


**Figure 3-7: Snd Read Wave File**

### 3.8.2 Snd Write Wave File

Stores waveform data and its sound format information into a file. The file is in a PC wave file (.wav) format and can be read by the Snd Read Wave File VI or other applications.

## 3.9 Sound Input VIs

### 3.9.1 SI Clear

Closes the sound input device associated with the task ID in and releases all the resources the device uses to the computer system.

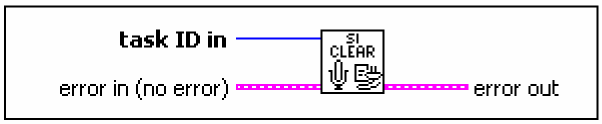


**Figure 3-8: Sound Input Clear**

### 3.9.2 SI Config

Configures a sound device for sound input operation. After configuration, you can use the SI Start VI and the SI Read VI to transfer data from device**.**

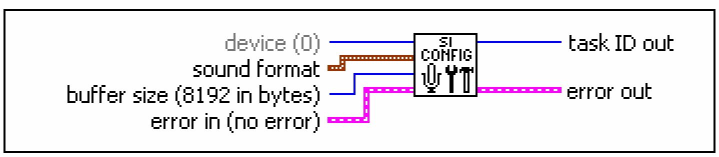


**Figure 3-9: Sound Input Configuration**

### 3.9.3 SI Read

Reads data from the sound input device. If data has arrived in the device buffer, it returns that data after buffering, otherwise it waits until data arrives. If for some reason the unbuffered data is overwritten, no data is returned, and instead, an overwrite error is reported

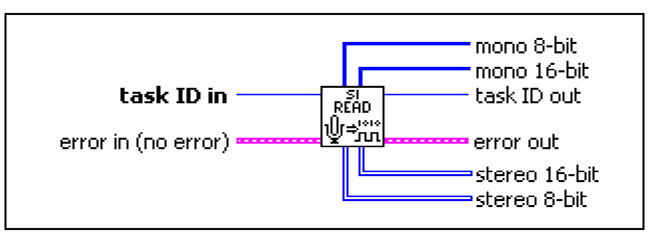


**Figure 3-10: Sound Input Read**

### 3.9.4 SI Start

Prompts the sound input device to begin accumulating incoming data. If the device is running already, calling this VI has no effect.

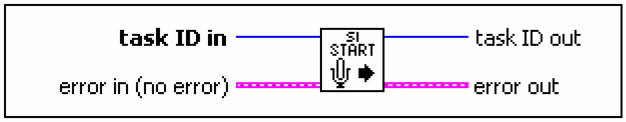


**Figure 3-11: Sound Input Start**

### 3.9.5 SI Stop

Prompts the sound input device to cease accumulating incoming data. The collected data remains in the system buffer, and it can be collected with the SI Read VI. If the device no longer is running, calling this VI has no effect.

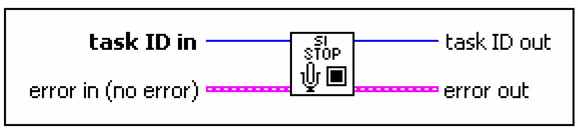


**Figure 3-12: Sound Input Stop**

# CHAPTER 4

# HARDWARE AND SOFTWARE IMPLEMENTATION

## *4.1 Introduction*

A voice based remote controller for home and office automation describes that in modern society, home and office automation become increasingly important and interesting. They not only provide better ways to transfer information within homes/offices and between homes/offices, they provide better time management too. These also improve the quality of our lives by automating some of the electrical home appliances such as light source, A/V equipment, computer, security device, etc. In our design we have envisaged a voice based remote control of devices using LabVIEW. User can input the control command through the head phone and by speaking one, two or three and accordingly the device one, two or three.

LabVIEW is used to process and recognize the voice signal then DTMF signals is used to send commands to the control unit, and remotely control a wide range of mains appliances in homes and offices. The use of DTMF makes the designed circuit cheap and enhances its capabilities of controlling the devices at long distances.

## *4.2 System Description*

The block diagram of the system has been shown in the figure 4-1. Main parts of the system are as follows:

- Voice Input (Microphone)
- Read Sound Card (Readsoundcard.vi)
- Word Recognition (Machinerun.vi)
- Data O/P at parallel port (Machinerun.vi)
- Parallel port to DTMF converter
- Transmitter Circuit (DTMF)
- Receiver Circuit (DTMF)

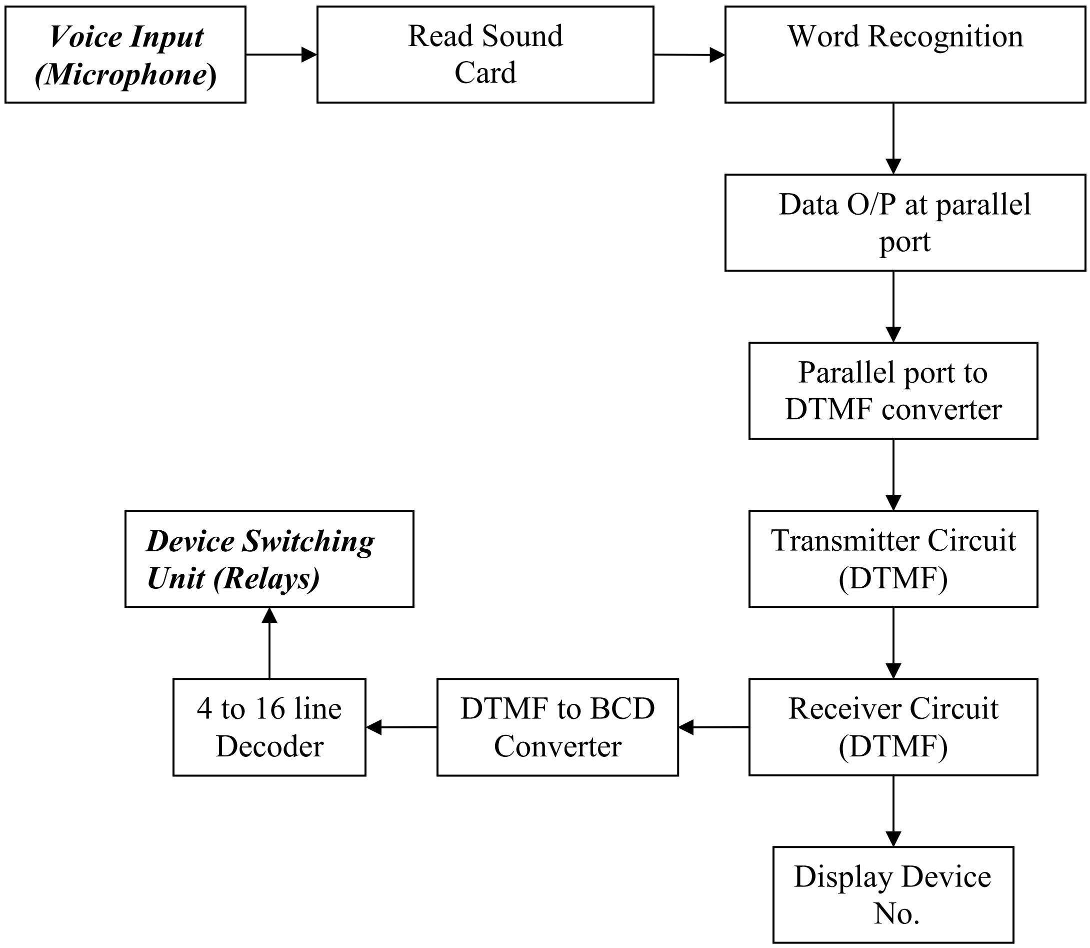


**Figure 4-1: Block Diagram of the System**

- Seven Segment Display
- DTMF to BCD Converter
- 4 to 16 line Decoder
- Device Switching Unit (Relays)

## *4.3 Voice Input (Microphone)*

The person is allowed to speak through the microphone which is attached to the computer.

## *4.4 Read Sound Card (soundcardread.vi)*

After the person has spoken the word next stage is to read the sound card for the LabVIEW (Virtual Instrumentation) program has been prepared. The data from the sound card is accessed the subVI (Snd read waveform.vi).

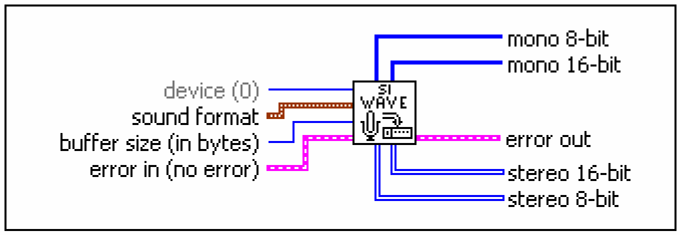


**Figure 4-2: Sound Read Card**

Snd read waveform.vi reads data from a sound input device. Different parameters to be adjusted are: **1.)** Buffer Size **2.)** Sound Format **3.)** Rate. Buffer size is the size of the internal buffer that LabVIEW uses to transfer data from a device The default value is 8192 bytes and we have selected the value 55000 bytes. Sound format specifies how the sound operation is set up (Mono or Stereo), sets its playing rate (speed), and sets up the sound as 8 or 16-bit sound. As in our case sound format is mono so we selected mono. Rate sets the sample rate for the sound input operation or the update rate for the output operation. We have selected 11025. Device is the input device you access for a sound operation on Windows XP. In our program we have given it the value 0. The sub VI is shown in figure 4-3. The VI that reads the word from sound card and converts it into the16-bit mono data is shown in the figure 4-4(A) and 4-4(B).

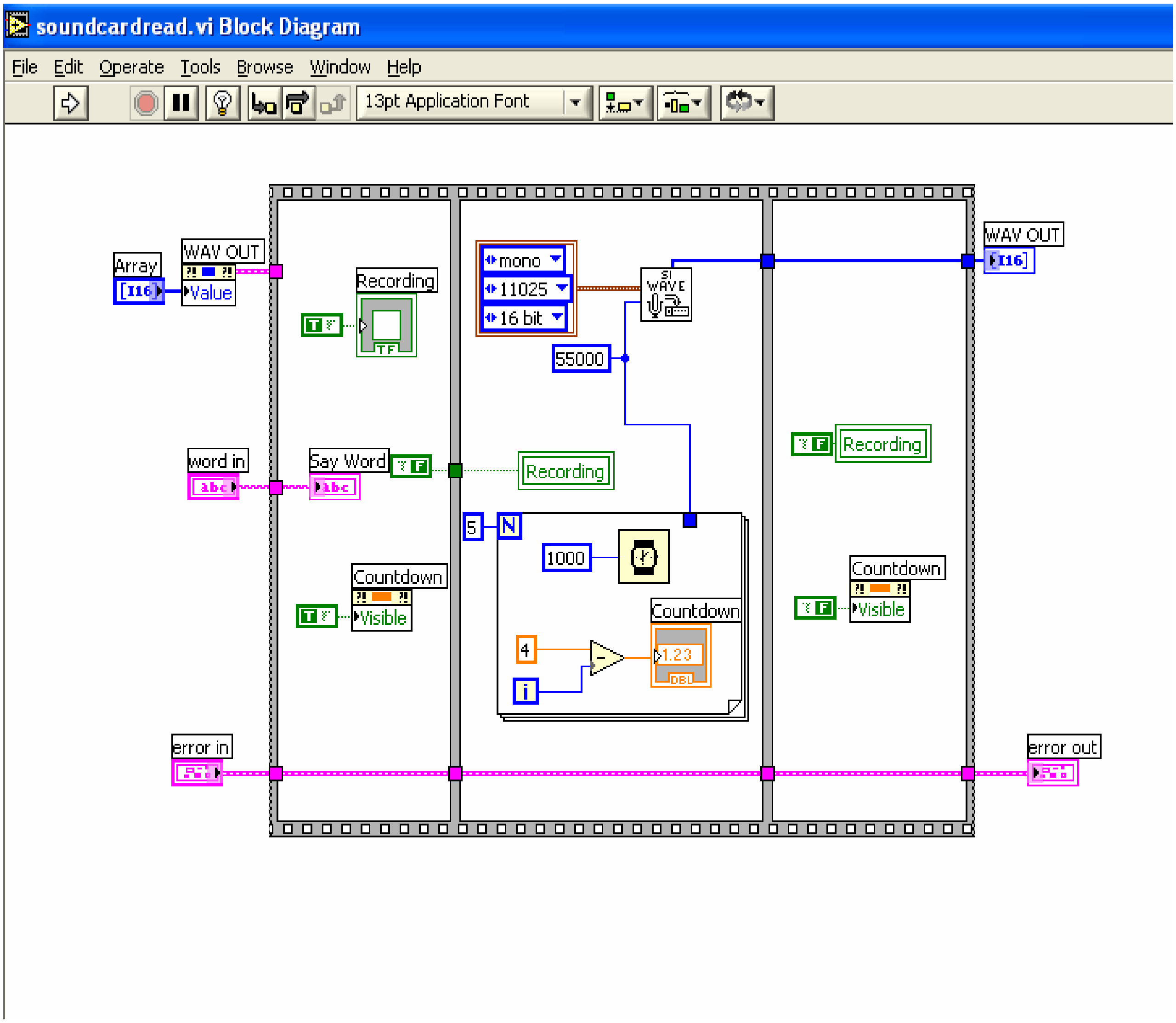


**Figure 4-3: Sound Read VI**

A countdown of 5 sec is also used so that the user can speak in the time span. The test word is also displayed in the waveform using waveform chart. The Front panel of the VI (soundcard read) has been shown in the figure 5-2.

## *4.5 Word Recognition (Machinerun.vi)*

In the machinerun.vi we have used soundcardread.vi as a subVI. First we have used the sequence structure and we have used five sequences. In the first sequence we have entered the first word; in the second we have entered second word then third word that are one, two and three respectively. In the fourth sequence we compare the test word with the stored words (i.e one, two, three).In the last sequence the data is written on the parallel port.

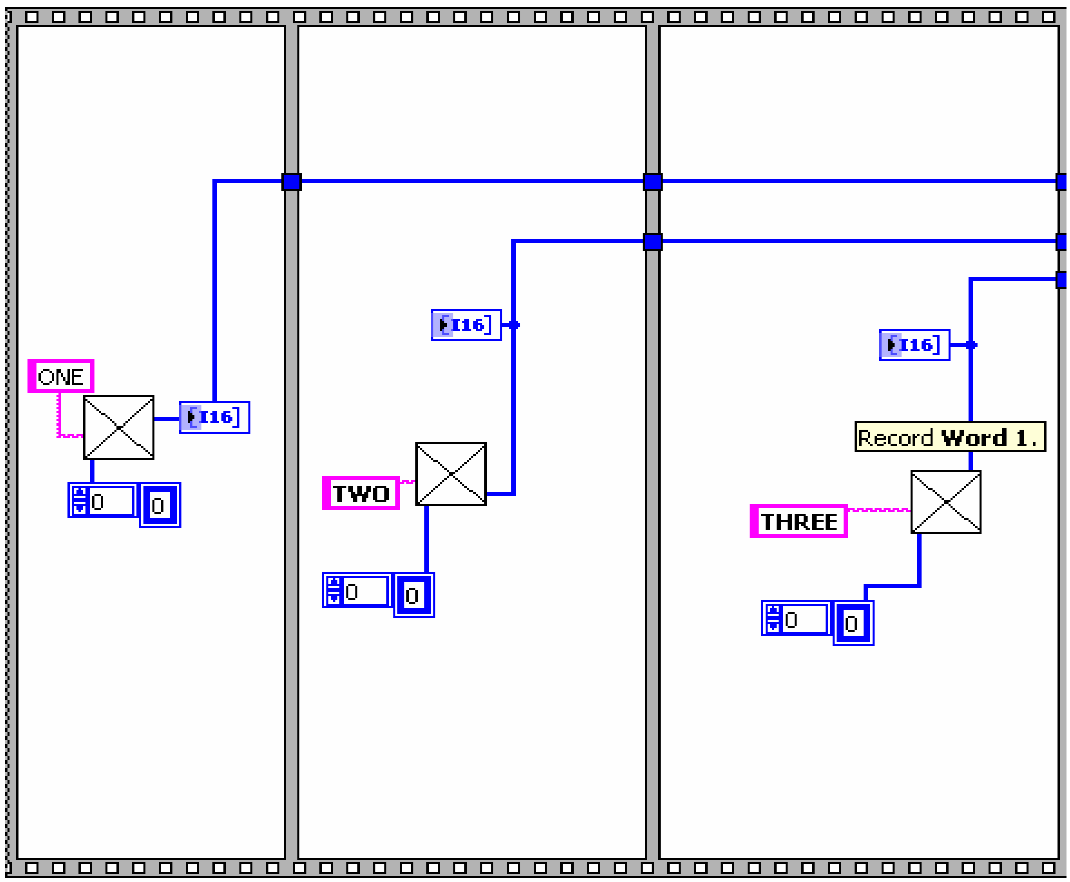


**Figure 4-4 (A): Word Recognition VI**

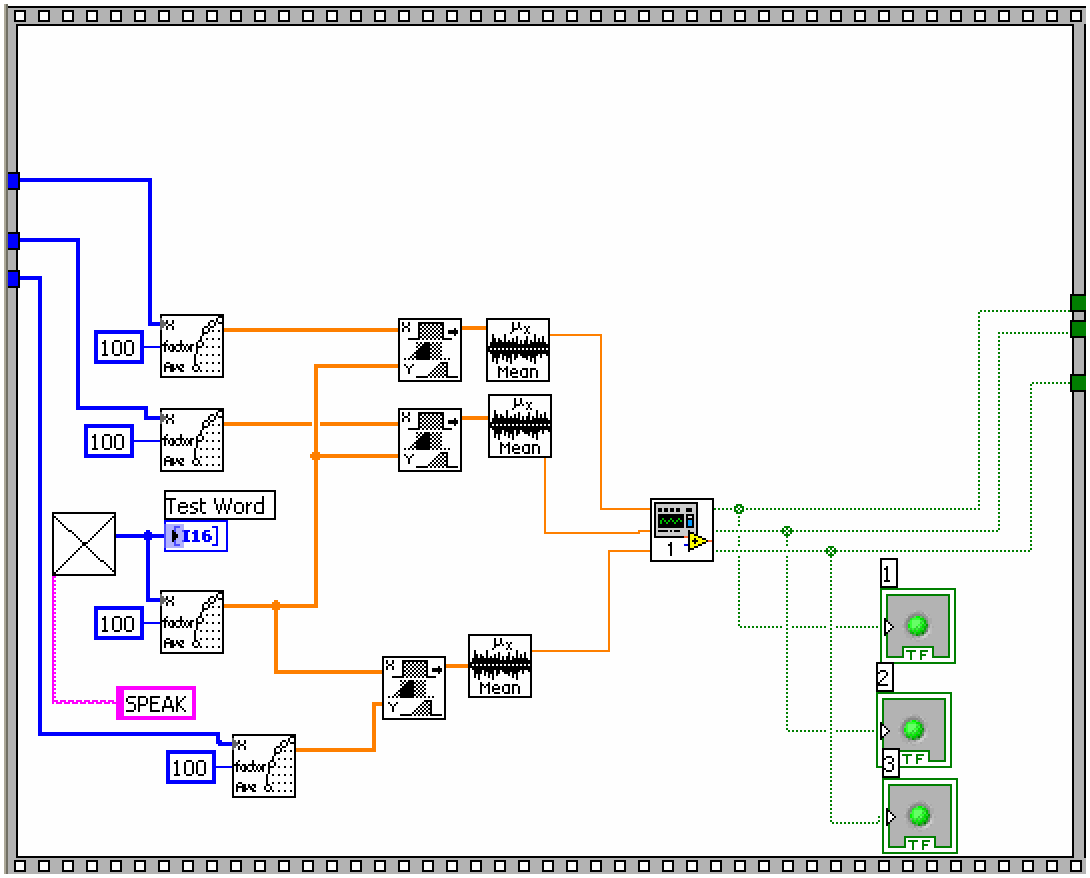


**Figure 4-4(B): Word Recognition VI**

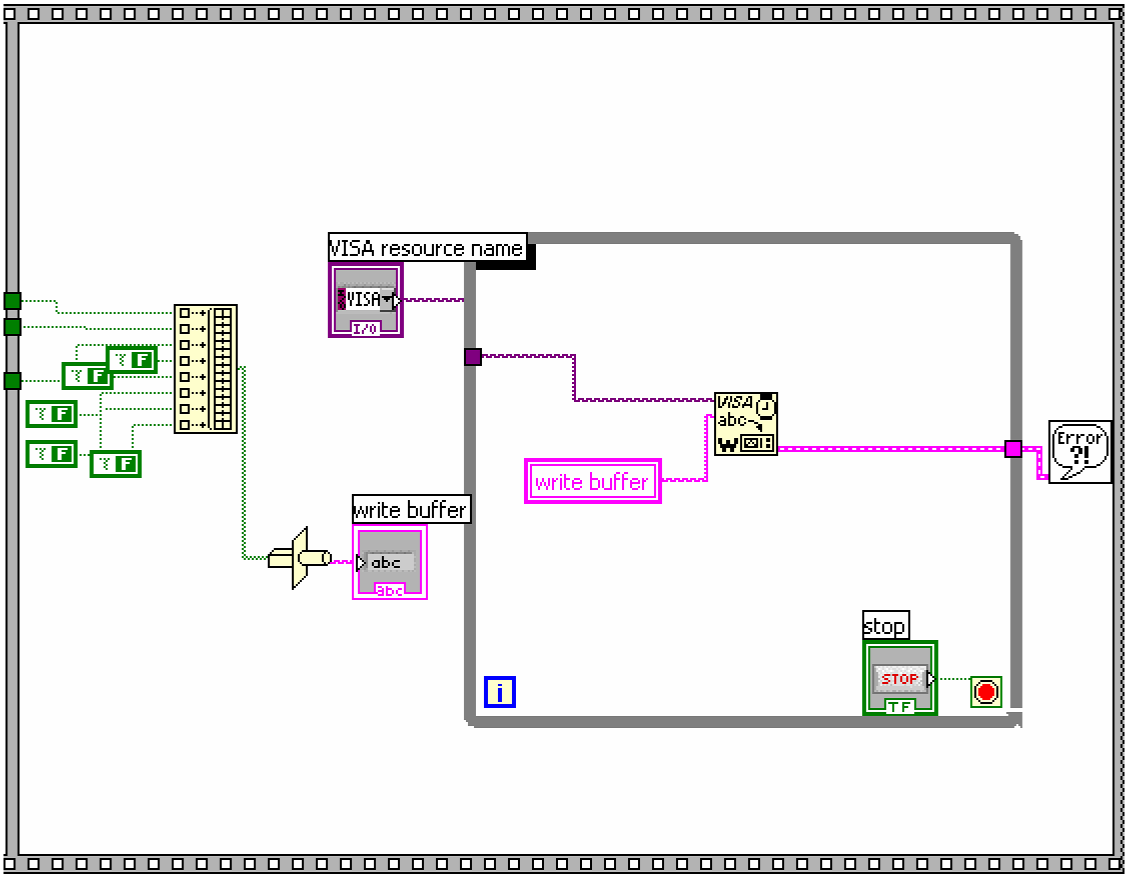


**Figure 4. 5: Parallel Port VI**

### *4.5.1 Decimate*

Decimates the input sequence X by the decimating factor and the averaging binary control. X is the input sequence. The number of elements in **X** must be greater than or equal to the decimating factor**.** If *Y* represents the output sequence Decimated Array, the Decimate VI obtains the elements of the sequence *Y* using

$$y_i = \begin{cases} x_{im} & \text{if ave is FALSE} \\ \frac{1}{m}\sum_{k=0}^{m-1} x_{((i*m)+k)} & \text{if ave is TRUE} \end{cases}$$

$$\text{for } i = 0, 1, 2, \ldots, \mathit{size} - 1$$

Here we decimate by a factor of 100. Averaging specifies how the Decimate VI handles the data points in X**.** The default is FALSE**.** When averaging is set to FALSE, every decimating factor point is kept from **X**.

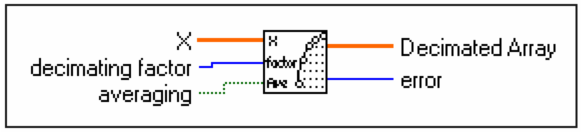


**Figure 4-6: Decimate**

When averaging is set to TRUE, each output point in decimated array is the mean of the decimating factor input points, where *n* is the number of elements in **X**, *m* is the decimating factor, *avg* is the averaging option, and *size* is the number of elements in the output sequence Decimated Array.

### *4.5.2 Cross Correlation*

Cross correlation is the main operation in our function. Computes the cross correlation of the input signals X and Y. Algorithm specifies the method used for computing the cross correlation. When algorithm is set to direct, the VI computes the correlation using the direct form, Rxy[i] = Sum(x[k]y[i+k]). When algorithm is set to frequency domain (default), the VI computes the correlation using a FFT-based technique. The direct method is typically faster for small input sizes, whereas the frequency-domain method

is faster for large input sizes. Additionally, slight numerical differences can exist between the two methods.

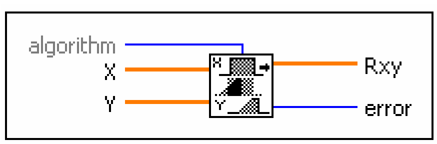


**Figure 4-7: Cross Correlation**

Here x and y are the input function while Rxy is the crosscorelation of the signal x and y. The cross correlation Rxy*(t)* of the signals *x*(*t*) and *y*(t) is defined as

$$R_{xy}(t) = x(t) \otimes y(t) = \int_{-\infty}^{\infty} x(\tau)y(t+\tau)\,d\tau,$$

Where the symbol $\otimes$ denotes correlation. The discrete implementation of the CrossCorrelation VI is as follows. Let h represent a sequence whose indexing can be negative, let *n* be the number of elements in the input sequence X, let m be the number of elements in the sequence Y, and assume that the indexed elements of X and Y that lie outside their range are equal to zero,

$$X_j = 0, j < 0 \text{ or } j => n$$

And

$$Y_j = 0, j < 0 \text{ or } j => m.$$

Then the Cross Correlation VI obtains the elements of h using the formula,

$$h_j = \sum_{k=0}^{n-1} x_k y_{j+k}$$

For $j = -(n-1), -(n-2), \ldots, -2, -1, 0, 1, 2, \ldots, m-1$

The elements of the output sequence Rxy are related to the elements in the sequence *h* by

$Rxy_i = h_{i-(n-1)}$ For $i = 0, 1, 2, \ldots, size - 1$, $size = n + m - 1$,

where *size* is the number of elements in the output sequence Rxy.

### *4.5.3 Mean*

Computes the mean of the values in the input sequence X. X is the input sequence. If the input sequence X is empty, mean is NaN. Mean is the average of the values in the input sequence X.

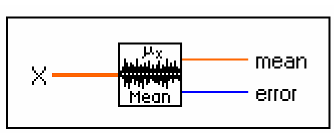

**Figure 4-8: Mean**

Error returns any error or warning from the VI. We can wire error to the error Cluster from error Code VI to convert the error code or warning into an error cluster. The VI calculates mean using the following equation.

$$\mu = \frac{1}{n}\sum_{i=0}^{n-1} x_i ,$$

Where μ is mean and n is the number of elements in X.

### *4.5.4 Maximum of three*

This function is implemented to choose the maximum value from the three values whose cross-correlation has been calculated. The maximum value of cross correlations determines that which one word matches with the test word. The VI used to calculate the maximum of three values is as shown in figure 4.9:

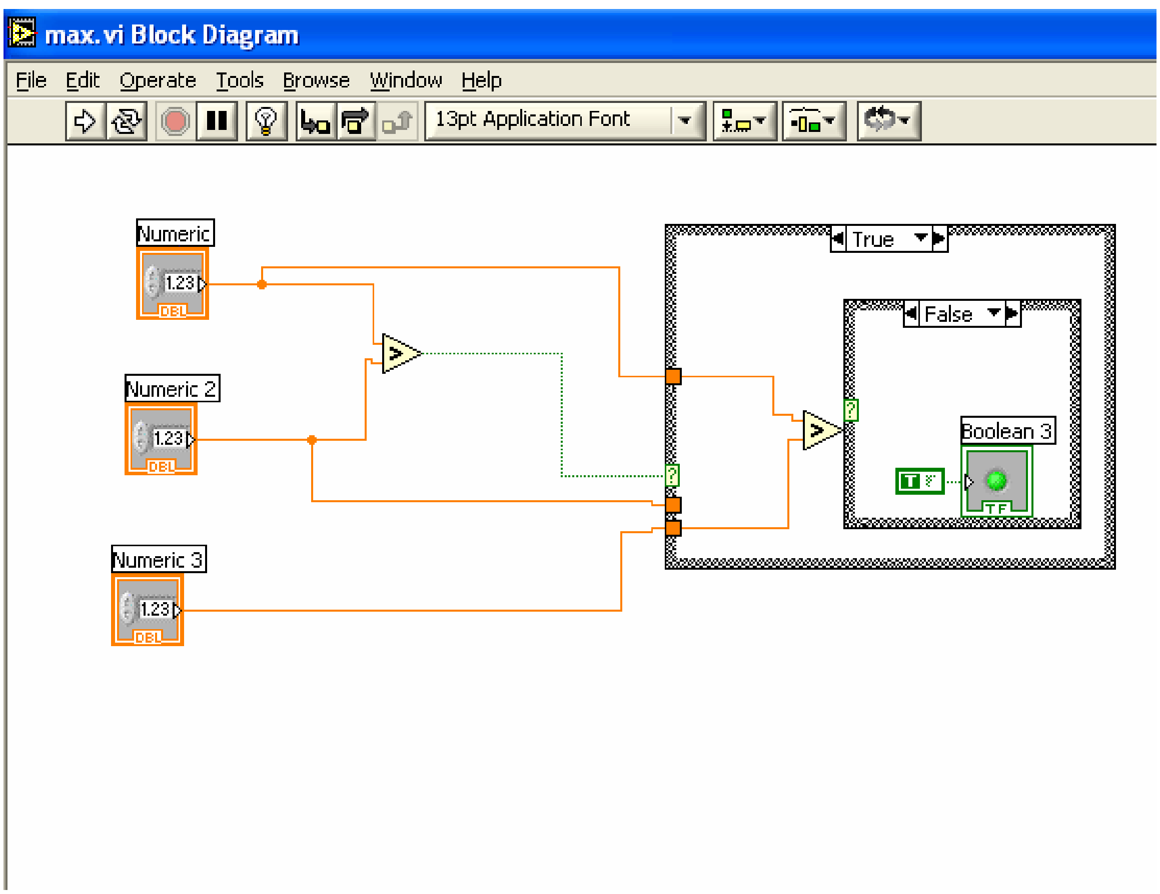

**Figure 4-9: Maximum of three VI**

## *4.5.5 Output at parallel port*

To connect the VI to external environment we have used the parallel port of the computer. Sequence of the parallel port has been shown in the figure 4-5. It contains following VI functions:

### *4.5.5.1 Build Array*

Build array concatenates multiple arrays or appends elements to an n-dimensional array. We also can use the Replace Array Subset function to modify an existing array. The connector pane displays the default data types for this polymorphic function.

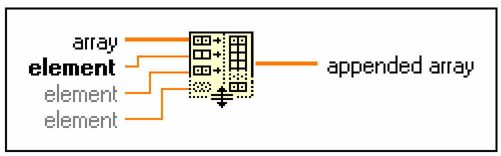


**Figure 4-10: Build Array**

### *4.5.5.2 Type Cast*

Casts **x** to the data type, **type**, by flattening it and unflattening it using the new data type. The connector pane displays the default data types for this polymorphic function.

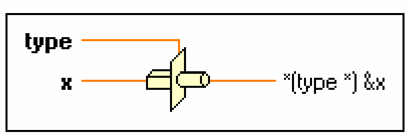


**Figure 4-11: Type Cast**

### *4.5.5.3 VISA Write*

Writes the data from write buffer to the device or interface specified by VISA resource name**.** Whether the data is transferred synchronously or asynchronously is platform-dependent. Right-click the node and select Do I/O Synchronously from the shortcut menu to write data synchronously. The operation returns only when the transfer terminates.

## *4.6 Parallel Port to DTMF Converter*

Parallel port (LPT1) of the computer is used to send the data. This converter converts the data at the parallel port into the DTMF signal which will control the device. The complete circuit diagram of the system is shown in the figure 4-17. In the circuit an opto-coupler IC-433T/817B used to isolate the computer from the external circuitry. To derive the circuit, voltage is stepped-down to 40V by using step-down transformer. Then the signal is given to voltage regulator L7805CV which is used to generate the 12V for the transistors. Transistors are used to derive the relay(JQC-3FC). The two output wires of the relay have been connected to the selected row and column of the IC generating the DTMF signal.

## *4.7 Transmitter Circuit (DTMF)*

Transmitter circuit receiver the signal from the parallel port of the computer and it activates one of the switches of the key pad. Transmitter circuit consists of following parts.

### *4.7.1 Keypad*

This is a 3x3array of switches (basically push buttons) when a button is pressed it connects the row1 with column1 and so on. The figure 4-12 shows the different frequency generated by pressing the buttons from 1 to 9.

| | | HIGH FREQUENCY GROUP | | |
|---|---|---|---|---|
| | | 1209Hz | 1336Hz | 1447Hz |
| LOW | 697Hz | 1 | 2 | 3 |
| FREQUENCY | 770Hz | 4 | 5 | 6 |
| GROUP | 852Hz | 7 | 8 | 9 |
| | 941Hz | * | 0 | # |

**Figure 4-12: DTMF (Dual Tone Multiple Frequency) Keypad**

### *4.7.2 DTMF Encoder*

To generate the DTMF signal IC UM-91214B is used. It generates the DTMF signal depending upon the key pressed on the keypad. It uses DTMF encoder integrated circuit, Chip UM-91214B. This IC produces DTMF signals. It contains four row frequencies & three column frequencies. The pins of IC UM-91214B from 12 to 14 produces high frequency column group and pins from 15 to 18 produces the low frequency row group. By pressing any key in the keyboard corresponding DTMF signal is available in its output pin at pin no.7. For producing the appropriate signals it is necessary that a crystal oscillator of 3.58MHz is connected across its pins 3 & 4 so that it makes a part of its internal oscillator. This encoder IC requires a voltage of 3V. For that IC is wired around 4.5V battery.

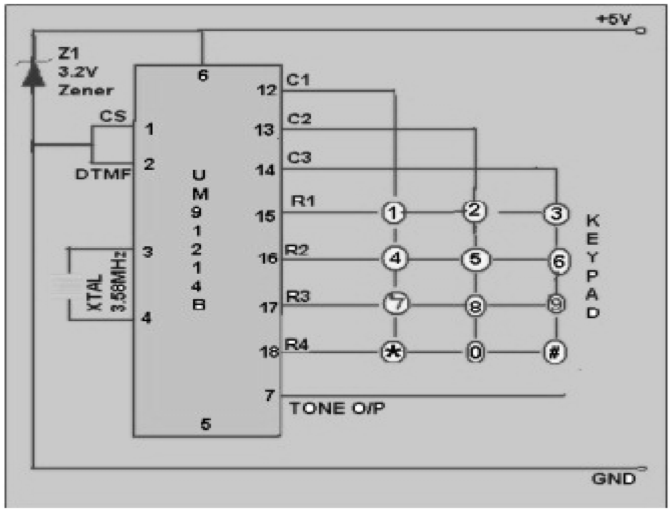


**Figure 4-13: DTMF Encoder**

The row and column frequency of this IC is as on the figure 4-13. And 3V backup Vcc for this IC is supplied by using 3.2V zener diode. By pressing the number 5 in the key pad the output tone is produced which is the resultant of addition of two frequencies, at pin no. 13 & pin no.16 of the IC and respective tone which represents number '5' in key pad is produced at pin no.7 of the IC This signal is sent to the local control system through telephone line via exchange.

## *4.8 Receiver circuit*

The receiver circuit decodes the DTMF signal and IC MT-8870/KT-3170 serves as DTMF decoder. This IC takes DTMF signal coming via telephone line and converts that signal into respective BCD number. It uses same oscillator frequency used in the remote section so same crystal oscillator with frequency of 3.85M Hz is used in this IC. The MT-8870 is a full DTMF Receiver that integrates both band split filter and decoder functions into a single 18-pin DIP. Its filter section uses switched capacitor technology for both the high and low group filters and for dial tone rejection. Its decoder uses digital counting techniques to detect and decode all 16 DTMF tone pairs into a 4-bit code. External component count is minimized by provision of an on-chip differential input amplifier, clock generator, and latched tri-state interface bus. Minimal external components required include a low-cost 3.579545 MHz crystal, a timing resistor, and a timing capacitor. The MT-8870 can also inhibit the decoding of fourth column digits.

MT-8870 operating functions include a band split filter that separates the high and low tones of the received pair, and a digital decoder that verifies both the frequency and duration of the received tones before passing the resulting 4-bit code to the output bus.
The low and high group tones are separated by applying the dual-tone signal to the inputs of two 6$^{th}$ order switched capacitor band pass filters with bandwidths that correspond to the bands enclosing the low and high group tones. The filter also incorporates notches at 350 and 440 Hz, providing excellent dial tone rejection. Each filter output is followed by a single-order switched capacitor section that smoothes the signals prior to limiting. Signal limiting is performed by high gain comparators provided with hysteresis to prevent detection of unwanted low-level signals and noise. The MT-8870 decoder uses a digital counting technique to determine the frequencies of the limited tones and to verify that they correspond to standard DTMF (Dual Tone Multiple Frequency) frequencies. When the detector recognizes the simultaneous presence of two valid tones (known as signal condition), it raises the Early Steering flag (ESt). Any subsequent loss of signal condition will cause ESt to fall. Before a decoded tone pair is registered, the receiver checks for valid signal duration (referred to as character- recognition-condition). This check is performed by an external RC time constant driven by ESt. A short delay to allow the output latch to settle, the delayed steering output flag (StD) goes high, signaling that a received tone pair has been registered. The contents of the output latch are made available on the 4-bit output bus by raising the three state control input (OE) to logic high. Inhibit mode is enabled by a logic high input to pin 5 (INH). It inhibits the detection of 1633 Hz.
The output code will remain the same as the previous detected code. On the MT- 8870 models, this pin is tied to ground (logic low).The input arrangement of the MT-8870 provides a differential input operational amplifier as well as a bias source (VREF) to bias the inputs at mid-rail. Provision is made for connection of a feedback resistor to the op-amp output (GS) for gain adjustment. The internal clock circuit is completed with the

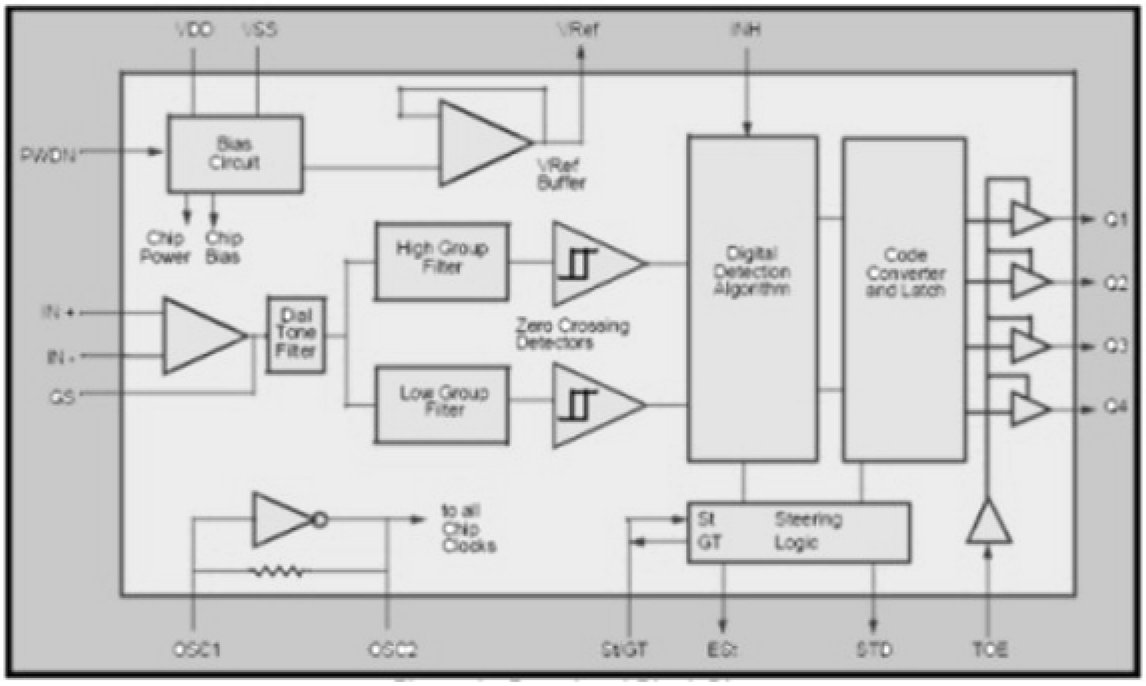


**Figure 4-14: DTMF (Dual Tone Multiple Frequency) Decoder**

addition of a standard 3.579545 MHz crystal. The input arrangement of the MT-8870 provides a differential input operational amplifier as well as a bias source (VREF) to bias the inputs at mid-rail. Provision is made for connection of a feedback resistor to the op-amp output (GS) for gain adjustment. The internal clock circuit is completed with the addition of a standard 3.579545 MHz crystal.

## *4.9 Seven segment Display*

This unit displays the received device code from the telephone line dialed from remote section. This unit consists of a BCD to seven segment decoder IC-7447 and a seven segment display. A seven segment display has seven LEDs connected in a sequence to give a regular shape and a LED to display the dot for decimal point. It has 10 pins. Out of this two pins are common for all LEDs and remaining are another polarity terminals of the LED. When common anode seven segment display is used, two common terminal pins are connected to +5V or logic high state and another terminal are kept at logic low state. Then respective LED glows. Here common anode seven segment display is used.

Because of this here we need a BCD to seven segment decoder which gives logic low output for the respective BCD input. Therefore we used a TTL IC-7447.The device selected from the Remote Section for control purpose, its code is displayed in this seven segment display.

## *4.10 4 to 16 line Decoder*

IC-74154 is a 4-16 line decoder, it takes the 4 line BCD input and selects respective output one among the 16 output lines. It is active low output IC so when any output line is selected it is indicated by active low signal, rest of the output lines will remain active high. This 4-line-to-16-line decoder utilizes TTL circuitry to decode four binary-coded inputs into one of sixteen mutually exclusive outputs when both the strobe inputs, G1 and G2, are low. The demultiplexing function is performed by using the 4 input lines to address the output line, passing data from one of the strobe inputs with the other strobe input low.

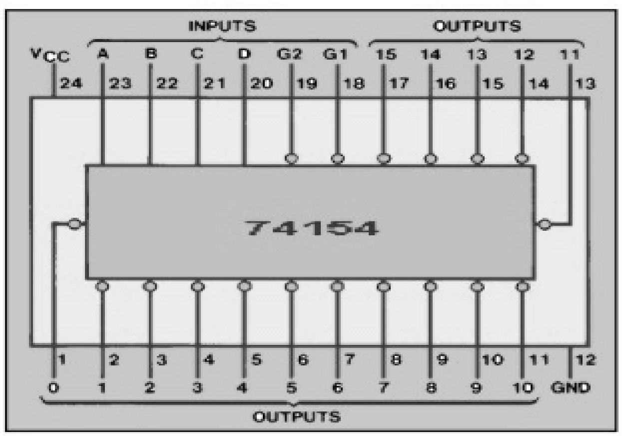


**Figure 4-15: 4 to 16 line Decoder**

When either strobe input is high, all outputs are high. These demultiplexers are ideally suited for implementing high-performance memory decoders. All inputs are buffered and input clamping diodes are provided to minimize transmission-line effects and thereby simplify system design.

## *4.11 Device switching unit (Relays)*

A relay is an electrically operated switch. The relay contacts can be made to operate in the pre-arranged fashion. For instance, normally open contacts close and normally closed contacts open. In electromagnetic relays, the contacts however complex they might be, they have only two positions i.e. OPEN and CLOSED, whereas in case of electromagnetic switches, the contacts can have multiple positions.

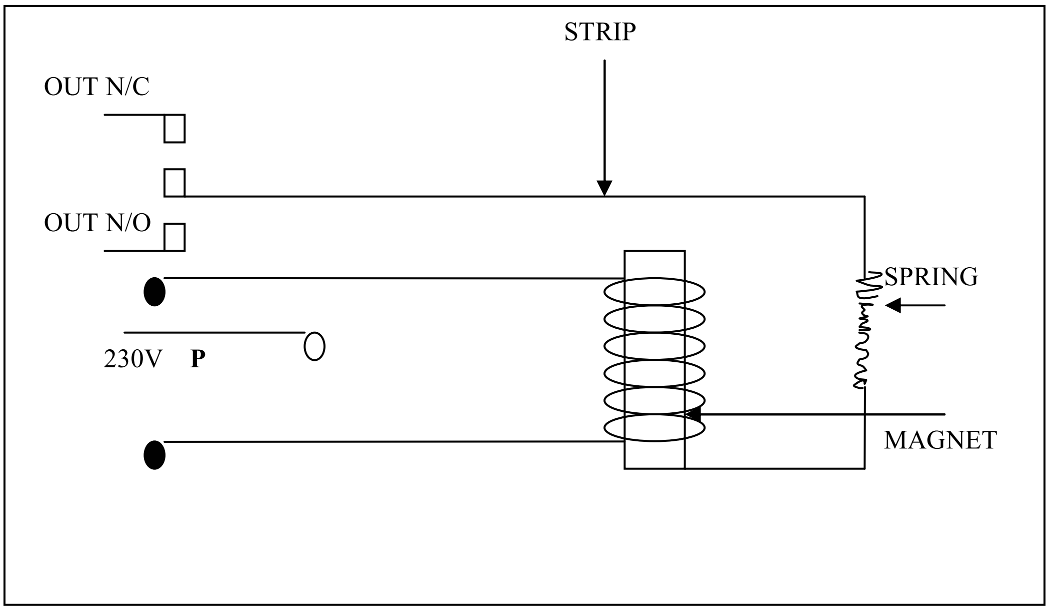


**Figure 4-16: Device switching unit**

## *4.12 Circuit Diagram of the system:*

**Figure 17**

# CHAPTER 5

# RESULTS, DISCUSSIONS AND CONCLUSIONS

## *5.1 Results and Discussions*

The front panel of the machinerun.vi is shown in the figure 5-1. It is the user interface of the system. When we press the run button it stores the different word spoken by the user then the user is asked to speak the teat word. Finally the test word is recognized

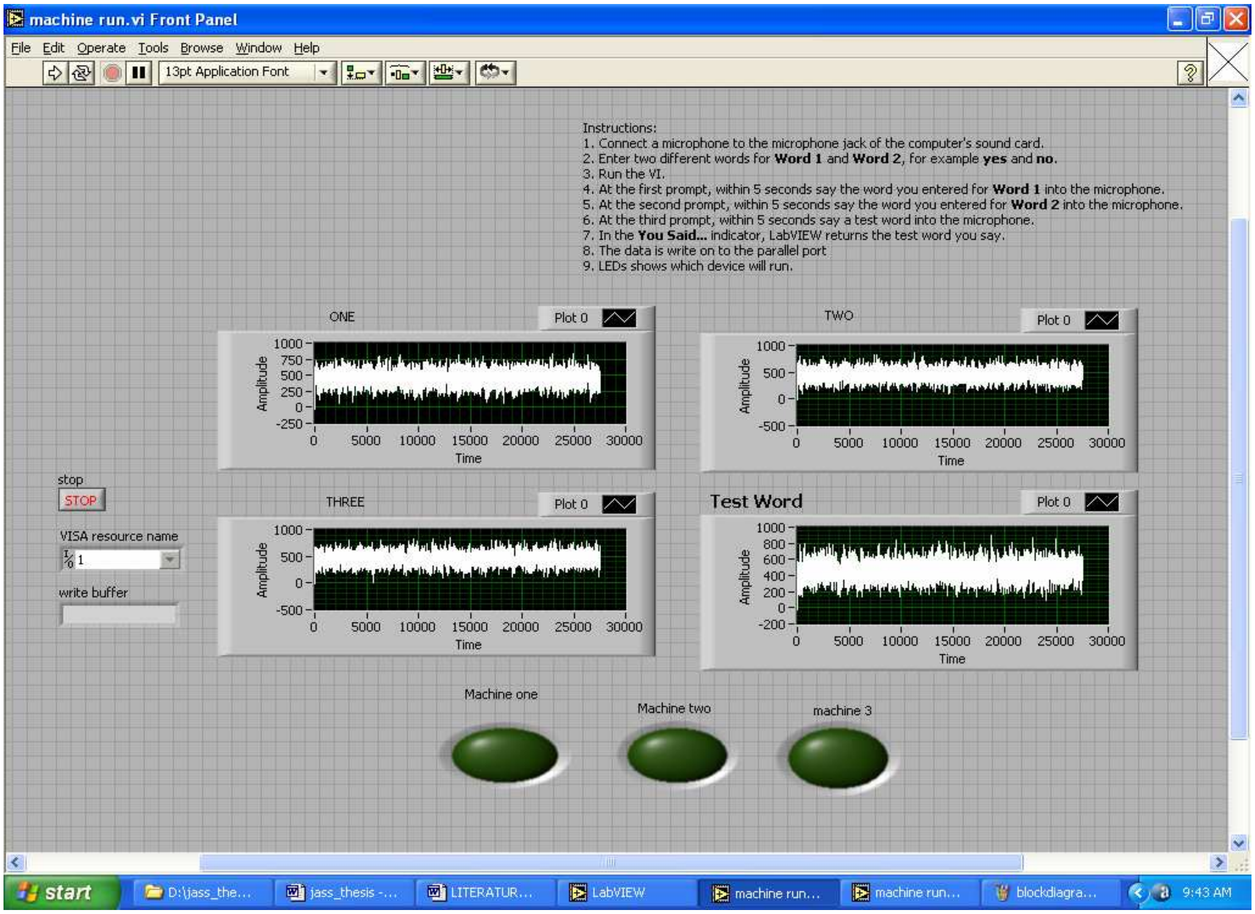


**Figure 5-1: Machine Run VI**

with the help of recognition program build in the LabVIEW. The machines corresponding to the word recognize is allowed to run and the signal to the DTMF hardware is sent using the parallel port of the computer. The figure 5-2 displays the recording of the word number one asked to speak and program waits for five seconds for the user to speak. Program also displays the waveform of the spoken word.

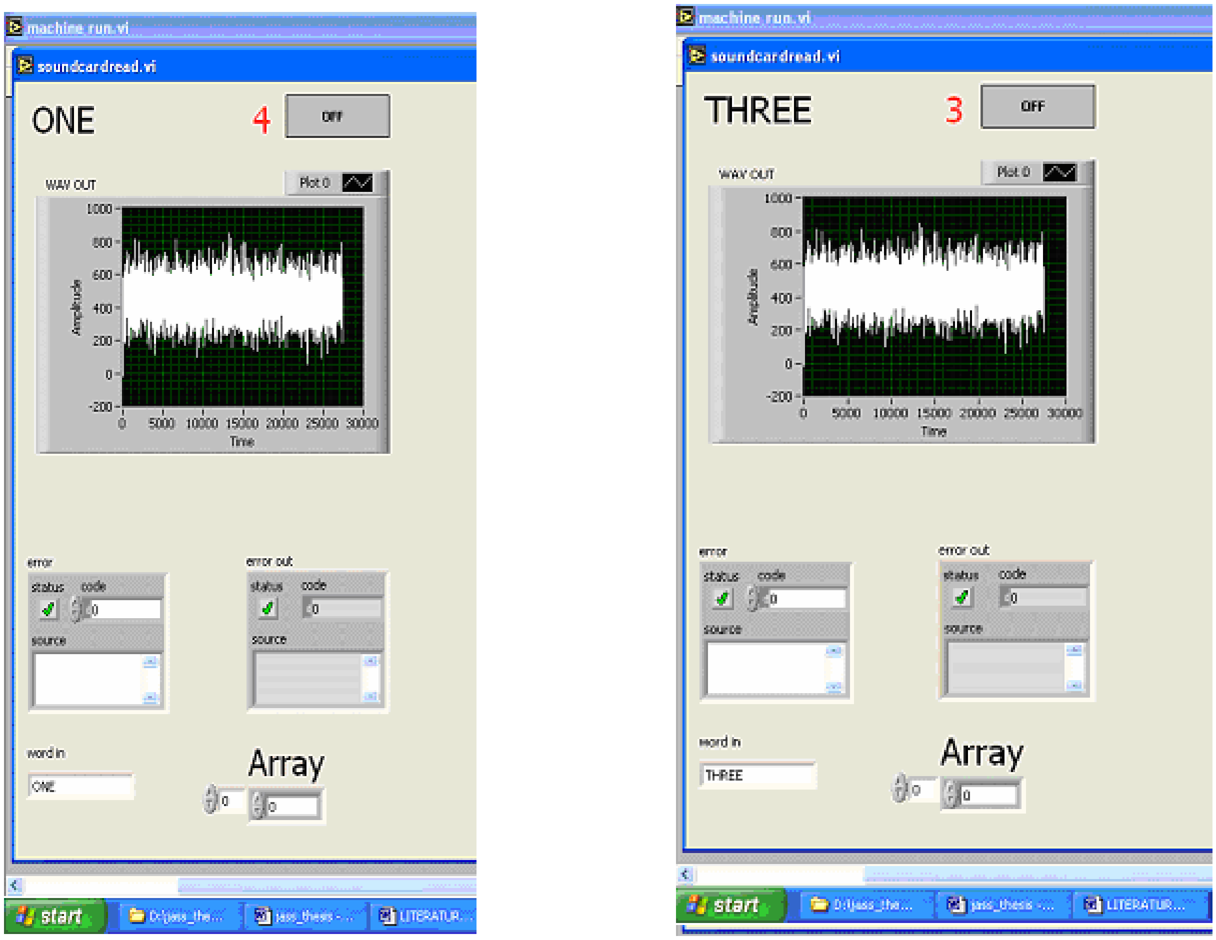


**Figure 5-2: Waveform of the spoken word 'one' and 'three'.**

Similarly the second and third word is also stored in the same fashion. After recording the all three words the test word is stored. And the test word is recognized with the help of cross correlation performed with the test word and the spoken word. After recognition of the word spoken the data is written on the parallel port as well as a LED corresponding to the machine number also glows on the front panel.

The front panel of the program in the figure 5-3 shows that machine three will run (i.e. LED three glows.)

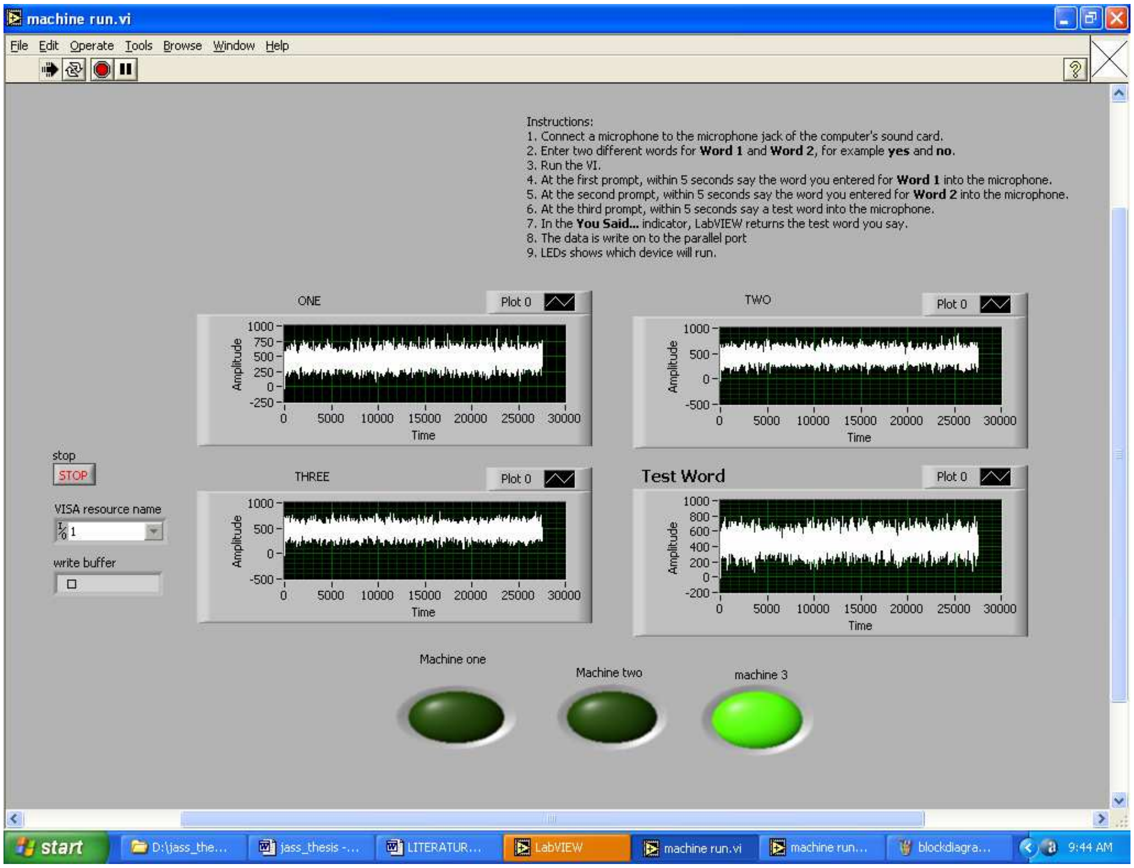


**Figure 5-3: Program for the running of the machine 3**

## *5.2 Conclusions*

The availability of the LabVIEW tool makes us to implement the voice-based remote control of machines. LabVIEW has its strong inbuilt library to implement the voice recognition system. There are various techniques available for the voice recognition. Some techniques are simple and some are complex ones. In our system we have utilized the calculation of the cross correlation.

The use of DTMF signal to send the signal from the control point to the remote location enhances the system capabilities as the system utilizes the existing telephonic networks and hence makes the system cheaper. Thus the technique of controlling the remote machines based on voice processing using LabVIEW can be very useful in most of the industrial and process plants and also provide facility to the disabled person to control the devices.

# FUTURE SCOPE

There exists great future scope in this thesis work.

1.) The devices control using the DTMF signal can also be controlled using the TCP/IP protocols as it increases the range of control.

2.) The wireless techniques can be used where telephone lines does not exist.

3.) LabVIEW also supports the use of blue tooth and other advanced techniques.

4.) If the user environment is noise then noise can be eliminated using the LabVIEW signal-processing tool.

# REFERENCES


[1] **Atkinson J.,** "Telephony", London: Sie Isaac Pitman, 1958

[2] **Bell Graham**, "Telephone", Hyderabad: Orient Longoman, 1984

[3] **Coskun, I. Ardam, H**. **"**A remote controller for home and office appliances by telephone*"*, In Consumer Electronics, IEEE Transactions on, Nov 1998

[4] **C Seelbach** "A Perspective on Early Commercial Applications of Voice-Processing Technology for Telecommunications and Aids for the Handicapped", In the proceedings of the National Academy of Sciences, Vol 92, 9989-9990,1995 by National Academy of Sciences.

[5] **Douligeris, C.,** "Intelligent Home System," IEEE Communication Magzine, Oct.1993

[6] **Numes, R,** "An architecture for a home automation system," Proceedings of the IEEE International Conference on Electronics, Circuits, and Systems, Sept 1998

[7] **Michael Duartkand Brian P. Bud** "The Virtual Laboratory for the Disabled*"*, In Frontiers in Education Conference, 2001. 31st Annual

[8] **Rabiner, L.R**, **"**Applications of voice processing to telecommunications", In the proceedings of the IEEE, Feb 1994

[9] **Rabiner, L.R "**The role of voice processing in telecommunications*", I*n the Interactive Voice Technology for Telecommunications Applications, 1994, Second IEEE Workshop

on, 26-27 Sep 1994

[10] **Raji, R.S.** "Smart networks for control" Spectrum, IEEE, Jun 1994

[11] **R Nakatsu** and **Y Suzuki,** "What Does Voice-Processing Technology Support today?," In the proceedings of the National Academy of Sciences, Vol 92, 10023-10030, 1995 by National Academy of Sciences

[12] **Samuel Pierre and Marthe Kassouf**, "Toward a telecommunication platform for supporting distributed virtual laboratories, Department of Electrical Engineering, National University of Singapore

[13] **Schneider, S.Swanson** and **J. Peng-Yung Woo** "Remote telephone control system", Consumer Electronics, IEEE Transactions on, May 1997

[14] **Wacks K.P**., "International development of home automation standards,"Digest of technical papers, International Conference on Consumer Electronics, 1992.

[15] **Wong, E.M.C** "A phone-based remote controller for home and office automation", In Consumer Electronics, IEEE Transactions on , Feb 1994.

[16] http:/ www.arm.com

[17] http:/ www.datasheets.archieve.com

[18] http:/ www.discovercircuits.com

[19] http:/ www.electroguys.com

[20] http:/ www.epanorama.net

[21] http:/ www.seanet.com

[22] http:/ www.webelectricmagzine.com