A USER-FRIENDLY SOFTWARE FOR GENERATING PATIENT-SPECIFIC MASKING NOISE IN RELIEVING THE INCURABLE TINNITUS

by

Murat FIRAT

B.S. in Electronic & Communication Engineering I.T.U., 1992

Submitted to the
Biomedical Engineering Institute
in partial fulfillment of
the requirements for the degree of

Master of Science
in
Biomedical Engineering

Boğaziçi University

1996

ACKNOWLEDGEMENTS

I am very grateful to Assoc. Prof. Dr. Halil Özcan GÜLÇÜR who supervised me in this project, with the utmost tolerance and patience. I am deeply happy to have the chance and honour to know Prof. Dr. Necmi TANYOLAÇ, who introduced to me The Biomedical Engineering Institute of B.U. and thus gave a new direction to my life which broadened my vision.

I am very fortunate of having Söğüt Yeşer GÜLEÇ, my fiancé, near me who encouraged me for completing the project and helped and shared my hard work.

I would like to thank to my family who had endured me in my hard working days. Very special thanks are due to Ph. Mehmet AKALIN, the General Manager of ORGANON TEKNIKA Turkey, for his precious support and unlimited tolerance. I also deeply appreciate the invaluable assistance provided by Cengiz ÇELİKYURT, M.D. I am indebted to Assoc. Prof. Dr. Yener YÖRÜK for giving me the precious time for finishing my work.

ABSTRACT

Tinnitus is a widely suffered complaint concerning the human auditory system. The tinnitus-stricken patient is generally characterized with hearing loss and suffers from an almost never-ending sound in his ears. These whistle, wind, water fowl, water flow, hissing, etc.-like sounds are always subjective and thus, their detection is quite difficult. There are plenty of tinnitus treatment methods that all focus on controlling these sounds, and are incapable of curing the disease.

Masking the tinnitus sound is one of the most successful non-invasive methods of coping with the tinnitus. In order for the masking method to be acceptable and successful, the masking sound should be individually tailored for the patient. This requires precise determination of the tinnitus frequencies, which is not an easy task and is time-consuming, since objective means of measurement are not available and trial-error methods must be used. Clearly there is a definite need for a system that could speed up the tedious work needed for the determination of the tinnitus frequencies and for the generation of the matching masking sound.

In this thesis work, such a system, consisting of a PC equipped with a sound card and specially developed software has been proposed and its cost-effectiveness has been demonstrated. In this system, in order to detect the tinnitus frequency, the sensitivity of the ears is measured and with the help of this audiometric data, during the tinnitus detecting phase, the volume level is adjusted. The pure tone audiometric test is applied for frequencies between 125 - 8000 Hz, at six different frequencies. This measurement is performed separately for both ears.

After the pure tone audiometry, the process of finding the correct frequency, the "pitch-matching" starts. This process begins with the application of pure tone at 860 Hz. With patient's guidance, the software successively approaches to the tinnitus frequency using a technique similar to the one used in the Newton-Raphson method. The frequency band of the pitch-matching applied is between 125 - 8000 Hz. When this process is completed, a narrow-band white noise with a pure tone at the pitch frequency is synthesized.

The software includes a database for the personal information and the measurement data of the patients. This database file can also be edited by most popular database software.

Keywords: Incurable tinnitus, tinnitus masking, controlling tinnitus

TEDAVİ EDİLEMEYEN TİNNİTUSUN DENETİMİNDE MASKELEME SESİ ÜRETMEK İÇİN KULLANIMI KOLAY BİR YAZILIM

ÖZET

"Tinnitus", insan işitme sistemiyle ilgili, çok yaygın bir rahatsızlıktır. Hastalığa yakalanan kişi, kulağında sürekli bir çınlama duyar ve işitme kaybı yaşar. Bu rahatsızlık verici sesler tamamen kişiye özeldir. Bu nedenle de, herkeste farklı olabilen, ıslık, rüzgar, su akışı, çağlayan, tıslama, v.b. gibi sesleri saptamak çok zordur. *Tinnitusun* tedavisinde birçok yöntemden faydalanılmaktadır, ancak bu yöntemlerin hepsi hastalığı iyileştirmeye yeterli değildir; sadece hastayı rahatlatmayı hedefleyen yöntemlerdir.

Maskeleme de, yan etkisi olmayan bir *tinnitus*'la mücadele yöntemidir. Maskeleme yönteminin kabul edilebilir olması ve başarıya ulaşabilmesi her hasta için ayrı ayrı çalışmayı gerekli kılar. Bu çalışma *tinnitus* frekansının kesin tanımlamasını içerir, kolay olmayan ve her şeyden önemlisi uzun süre alan bu çalışmada kullanılabilecek hazır objektif bir yöntem yoktur ve kullanılacak yöntemler hata-yönlenlendirmesi tabanlı olmak zorundadır. Kısaca bu öznel olmayan zorlu çalışmayı hızlandıracak, *tinnitus* frekansının olabildiğince hızlı bulunup gerekli maskeleme sesini üretebilecek bir sisteme ihtiyaç vardır.

Bu tez çalışmasında, ses kartı donanımına sahip bir PC üzerinde özel bir yazılım geliştirilmiş ve böyle bir sistemin ne kadar etkin olabileceği gösterilmeye çalışılmıştır. Bu yazılım hastaya ilk önce 125 - 8000 Hz arasında basit bir odiyometrik bir test uygular. Daha sonra, *tinnitus* frekansının yakalanması sırasında bu işitme hassasiyeti bilgileri kullanılarak ses yüksekliği ayarlaması otomatik olarak yapılacaktır.

Programın ana bölümü olan *tinnitus* frekansının bulunması evresi, Newton-Rhapson yöntemine dayalı bir algoritmaya sahiptir. Başlangıçta 860 Hz ile verilen saf sinüzoidal dalga, hastanın yönlendirmesi ile hastayı rahatlatan doğru ses bulunana dek frekans değiştirecektir. 125 ile 8000 Hz arasında yapılan *tinnitus* frekansı taraması, üç kez ard arda tekrarlanır. İşlemin sonunda, içinde *tinnitus* tepe frekansının da bulunduğu beyaz gürültü üretilir.

Yazılım ayrıca, içinde hastanın kişisel bilgilerinin ve ölçüm sonuçlarının saklanabileceği bir veri-tabanı bölümünü de içerir. Yazılımın ürettiği ve işlediği bu veri-tabanı dosyası aynı zamanda diğer çok kullanılan veri-tabanı yazılımları tarafından da açılıp işlenebilir özelliğe sahiptir.

Anahtar kelimeler: Tedavi edilemeyen tinnitus, tinnitusun maskelenmesi, tinnitusun denetimi

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 Motivation, Background and Objectives

Tinnitus, bearing a rather strange name, is in fact a common complaint all over the world concerning the human auditory system. To give an example, nearly 37 million Americans are today suffering from tinnitus (Rosenhall, 1991). Despite the insufficient data concerning our country, there is evidence that many people suffer from tinnitus also in Turkey. According to the MRC National Study of Hearing in Great Britain, a considerable amount of the patients are disturbed so much that they even had difficulty in getting to sleep (Pedersen, 1989).

We have little tangible information about the tinnitus and its mode of generation. There is insufficient data about the origin of this complaint which makes us rely almost entirely on the patients's subjective statements.

For the physicians, tinnitus is such a problematic complaint that they generally do not want to deal with these patients. According to an investigation, more than 95% of otologists have difficulties in treating tinnitus (Goodey, 1988).

Since tinnitus is subjective and difficult to appreciate, its treatement can offer only control, not cure. There are several methods improved for controlling the tinnitus such as, hypnotherapy, electrical stimulation, acupuncture, drug therapy, bio-feedback, masking and combined methods.

The most effective of the above-mentioned methods is the masking method. There is a growing need for this method in clinical applications where it is beginning to be used frequently. Considering that this complaint is subjective and hard to appreciate, it requires time-consuming patient-specific study. Therefore, the diagnosis, determination of tinnitus frequencies and the generation of the masking sounds should be automated as much as possible. This is the motive of this thesis work.

1.2 Hardware and Software Used

A PC equipped with an Intel 486 micro-processor and a 16-bit sound card[1] is used as the main tool for the determination of tinnitus frequencies and for the generation of masking sounds. Microsoft *Visual Basic* has been used to develop the software under the MS Windows environment.

1.3 Outline of The Thesis

Chapter one introduces the motivation, background and objectives behind this study. A brief summary of human auditory system and the formation of tinnitus is given in chapter two. Chapter three presents the masking and masker concepts. The tinnitus diagnoser (TD) developed in the present thesis work is described in chapter four. Chapter five gives a discussion of the main results and conclusions. The program list of the TD is given in Appendix A, whereas in Appendix B, the creative ADPCM Wav File Type Format is given. The block diagrams of the SB audio cards are given in Appendix C.

[1] SB AWE 32

2. TINNITUS

2.1 Introduction

It was in ancient Greece when tinnitus was frequently described. This concept of subjective noises were expressed in words like; "echos", "bombos" and "psophos". These words did not have any specific etymological meaning, but were onomatopoetical descriptions of the subjective noises.

We know that the ancient Greek was the leading language in Europe for centuries and that the technical terms we use even today are of Greek origin. However, this does not apply to tinnitus. The word tinnitus seems to be inherited from Latin which took over the rule of Greek and became a universal language. The term tinnitus is believed to be originated from the words; "sonitus", "susurrus" and "sibilus". These Latin words were used to explain subjective sensations. "Sonitus" meant sound or noise, e.g. the noise of footsteps. "Susurrus" meant whispering, buzzing, humming. "Sibilus" meant whistling, hissing. And "tinnitus" meant a jingle or clashing sound. (Feldmann, 1991)

Since the ancient Egypt ages, the knowledge about the human auditory system limits all the approaches to diagnose and treat tinnitus.

2.2 Human Auditory System

The "main" elements of the auditory system are the ears. Ears have the biggest role in the process of hearing. Hearing occurs when the sound waves stimulate the hearing sensory organ. The stimulus, or the sound wave energy is a molecular vibration which can be captured by the ear. The human ear can hear the sounds with frequencies between 16 - 16,000 Hz.

2.2.1 Anatomy of the ear

As shown in Figure 2-1, the ear is composed of three parts: outer ear, middle ear cleft, and external ear. The external ear has two parts; the auricle and the external auditory canal which is about 25 mm long. The middle ear consists of eustachian tube, and tympanic cavity. The inner ear is in the temporal bone. It is composed of osseous labyrinth and membranous labyrinth. The inner ear is the main organ of hearing. The sounds which vibrate the promontory bone cause to create pressure changes in the cochlear liquid and vibrates the organ of corti. This mechanical stimulus is transformed into temporary potentials at the cochlea and transferred by the auditory nerves to the centre of hearing.

2.3 The Formation of Tinnitus

Tinnitus have been studied in three aspects: physiologic, psychophysical-psychoacoustical, and clinical aspects.

a) *Physiologic aspect:* The feedback of the auditory system to the acoustical stimulus.

b) *Psychophysical and psychoacoustical aspect:* This aspect searches the relation between the physical and psychological worlds, and the part of the physical world described as acoustics.

c) *Clinical aspect:* The aspect of tinnitus which deals with the clinical history of it. The primary complaints about tinnitus such as hearing loss, vertigo and ear blockage are studied in this branch. Clinical experience shows that tinnitus is a herald of injury. (Shulmann, 1991)

Any problem at the auditory system may end with tinnitus. Tinnitus may occur either unilaterally, bilaterally, in the head, outside the head, in the back of the head or in different combinations.

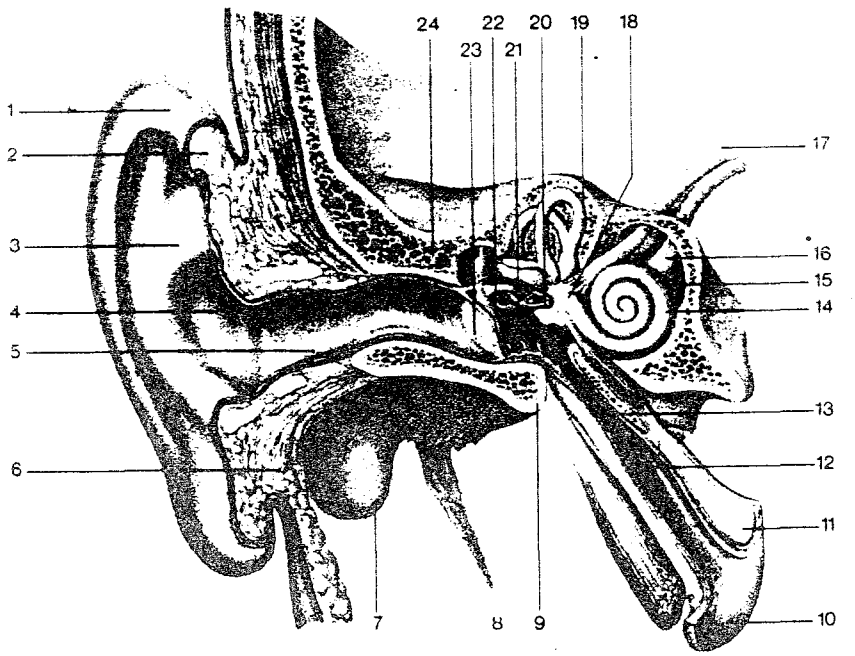


Figure 2-1 Anatomy of the ear. 1. Helix; 2. Cartilage; 3. Antelix; 4-5. External auditory meatus; 6. Ear lobe; 8. Sitloid; 9. Tympanic bone; 10. Pharynx; 11-12-13. Eustachian tube; 14. Cochlea; 15-16. Cochlear nerve; 17. Vestibular nerve; 18. Vestibule; 19. Semi-circular canals; 20. Hearing bones; 21. Tympan; 23. Tympanic membrane; 24. Mastoid cavity.

3. A RELIEF TO THE PATIENT: MASKING AND MASKER

3.1 Introduction

The first masking idea goes back to the ancient Greece. The Aristotelian scholars had the knowledge that "a buzzing in the ear ceases, if one makes a sound, because a greater sound drives out the less" (Feldmann, 1991). However, this knowledge was not used for about 2000 years. Itard (1821) was the first scientist who found that the relation between tinnitus and external noise could be used as a remedy. After nearly 200 years of masking experience, we know that it is unable to cure the tinnitus. It may only offer a relief for those who cannot sleep because of their tinnitus. A patient is not disturbed by a sound produced outside of him. Extraneous noise can be tolerated by the human nervous system. (Feldmann, 1991).

Everything physicians and scientists know about tinnitus was taught to them by the patients. Water sounds are the most effective maskers of tinnitus. Many patients observed that they cannot hear their tinnitus while taking a shower (Vernon, 1991).

3.2 How Masking Works

In the masking procedure we have two kind of signals; masking signal and the masked signal. These signals are defined by their frequency spectra, their intensities and their presentation types. In a normal hearing ear if we know the parameters of masking and the masked signals, we can predict what the subject will hear. Yet, the situation for a tinnitus patient is different because his hearing is not normal. In this case, the masking signal can be defined but its sensational equivalent which is the masked signal cannot be known. (Çelikyurt, 1994).

Man can neglect the monotonous noises which come from the outer environment of the body. Therefore, if the masking signal covers the masked signal then tinnutus can also

be neglected as an outer noise. Yet, to get an acceptable masking sound, the volume level of the signal must be at the hearing limits of the ears.

Normally, the masked signal and the masking signal must be present at the same ear, because there will be no effect of the signal at one ear to the other ear. The most absurd situation of the tinnitus, differing it from the normal psychoacoustic situation, is the non-stable phase of the tinnitus after masking. The non-stable phases after the masking can be:

1. **Steady state** : Normal masking process continues.
2. **Residual inhibition** : Fading effect occurs at the tinnitus and the masker can be switched off.
3. **Growing refractoriness** : Reappearing against the masking sound. A higher intensity of the masker sound can solve the problem.

The best masking sound is the sound which completely covers the tinnitus at the minumum acceptable level. Using pure tone for masking is not a good idea as using a narrow band white noise centered to the tinnutus frequency.

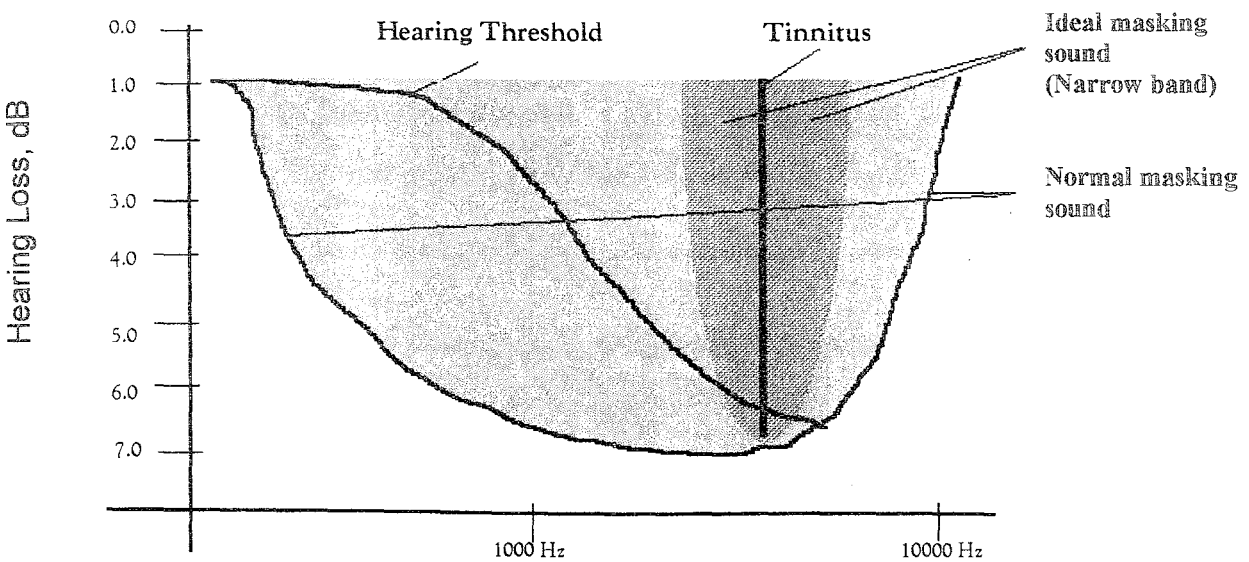


Figure 3-1 The ideal tinnitus masking sound has to cover only the pitch-frequency of the tinnitus and also to provide a sound that is an acceptable substitute for tinnitus.

The aim of masking is to provide a relief to the patient, not a residual inhibition. If residual inhibition occurs, this will be only a bonus not a planned action.

4. A TOOL FOR MASKING TINNITUS

4.1 Introduction to The Tinnitus Diagnoser (TD)

In this thesis work a computer-aided tool has been developed for facilitating physicians efforts to find the correct tinnitus pitch frequency and to produce the narrow-band noise for the masking process. The software consists of four main parts:

1. A database for patient records,
2. an audiometric tester,
3. a pitch frequency matcher, and
4. a white noise generator.

4.1.1 A database for patient records

The database embedded in the software not only keeps general information about patients such as name, surname, age and gender, but also stores auditory test results and tinnitus pitch frequencies. The database can be accessed via the "*Dosya* " menu. When "*eski hasta*" option under this menu is chosen, all the information related to the patients treated previously can be accessed. The user can then keep records about his / her patients.

Table 4-1
The personal information table "*patient data*" of the database of TD.

Pnumber	Name	Sex	Age	Education
.	.	.	.	.
.	.	.	.	.

Table 4-2
The table for the audiometric test results "audiodata" of the database of TD.

Pnumber	Lb1	Lc1	Lb2	Lc2	....	.....	Rb3	Rb4
.	.	.	.	.	.		.	.
.	.	.	.	.	.		.	.

Table 4-3
The table for the records of tinnitus frequencies "tinnitusdata".

Pnumber	Ptfreq1	Ptfreq2	Ptfreq3	Centralfreq	Bandwidth
.	.	.	.	.	.
.	.	.	.	.	.

Another characteristic of this database is that one can easily access the data in the file "*tpatnts.mdb*" created by the software by using other popular database softwares like *Microsoft Access*. Information stored in the database can be manipulated using these software or converted into other forms or tables and data can be exported to other spreadsheet software.

To add a new record and start the procedure for a new patient, the user should select the "*yeni hasta*"(new patient) option under the "*Dosya*" menu. In such a case, the software automatically assignes a new patient number. The user has to enter the other data related to this patient.

Only after all the information about the new patient is entered will the software allow the user to proceed with the audiometric test and the pitch-matching. The software automatically records the results of the audiometric tests and the pitch-matching in the database.

4.1.2 An audiometric tester

In order to make an efficient pitch-matching, one has to have adequate information related to the level of hearing loss of the patient. The tone which will be given to the patient should be selected carefully according to this information. The tones given should be at an appropriate level for the patient to hear it clearly.

In this part of the program, a simple audiometric test is applied to the patient. The hearing sensitivity of both ears of the patient is tested roughly at frequencies between 125 - 8000 Hz. A rough plot is drawn with the information obtained in this test and the graphic information is saved in a database to be used in the pitch-matching process of the software at a later stage.

Table 4-4
The frequencies of the sounds used in the audiometric test.

Sound number	0	1	2	3	4	5
Frequency	125 Hz	500 Hz	1000 Hz	2000 Hz	4000 Hz	8000 Hz

For the audiometric test to start, after the patient data are entered into the database, the "*Başla*" (Start)option in the "*Audio*" menu should be selected. The screen in this option is designed for the patient to test his/her own audiometric sensitivity. The only thing the patient has to do is to press the "*Duydum*" (I hear it) button, when he hears the sounds generated by the computer in different frequencies between 125 and 8000 Hz. The test is applied first to the left, then to the right ear. For both ears, a hearing loss graph is plotted. The shaded areas in these graphs show the patient's hearing loss in different frequencies. The y-axis is numbered from 0 to 100 and shows the volume level of the sound applied to the patient. The x-axis is the frequency axis. During the test, signals at 125, 500, 1000, 2000, 4000 and 8000 Hz are applied to the patient which are shown with numbers 0 to 5 along the x-axis. After the results of this audiometric tests are saved into the database, one can move to the next step.

4.1.3 A pitch frequency matcher

This is the main section of the software. The pitch frequencies of the audiometrically tested patient are identified in this section. The screen in this part is user-friendly in the sense that it is designed to be used easily by the patient himself. The two big buttons on the screen are the only controls the patient uses to find his tone that relieves him. At the beginning of the test, the computer begins to generate a pure tone with the frequency of 860 Hz. Then the patient should try to find out the correct tone by pressing the "*daha ince*" (increase the pitch) or "*daha kalın*" (decrease the pitch) buttons. Meanwhile, the computer will adjust the volume level with the help of the audiometric test performed previously. After reaching the proper frequency, the software allows the user to make two more trials for finding out the correct pitch frequency.

4.1.4 The white noise generator

After detecting the pitch frequency of the tinnitus, the program automatically generates a white noise with a pure tone at the pitch frequency of the patient. The last step is the generation of the narrow band masking sound. This step requires the use of a proper narrow band filter whose central frequency is equal to the pitch frequency. This is accomplished by the help of the *Mathlab* software.

4.2 Characteristics of TD

In the software developed, the patient information and the results are stored in a file in the *Microsoft Access* format. In order to detect the hearing loss of the patient the software applies an audiometric using six distinct frequencies at 125, 500, 1000, 2000, 4000 and 8000 Hz, respectively. During the pitch-matching process, the area between 125 - 8000 Hz is scanned. This interval can be enlarged via the software, if necessary.

4.3 The tool itself: The TD software

4.3.1 Visual Basic and *event-driven* programming

When the Basic programming language was created in 1963 by John Kemny and Thomas Kurtz at Dortmouth College, their purpose was only to teach programming concepts, emphasizing program clearity at the expense of speed and clearity. Several characteristics of the earliest versions of Basic are notable. Each line of a program began with a line number, and statements were generally not intended (as is often done today to enhance readability and to clarify structure). But these characteristics tended to encourage the creation of hard to follow "spaghetti code" - so called because the logical flow of the program often wound and branch around, as do strands of limp spaghetti on a plate.

The progress of Basic has closely followed the personal computer revolution. In 1982, Microsoft QuickBasic revolutionized Basic and legitimized it as a serious development language for the MS-DOS environment.

Today the Microsoft Windows revolution is well under way, providing a powerful and standardized environment to take advantage of the capabilities of Intel Corporations' newest microprocessors. While Windows was a breakthrough for the user, programmers had to retrain themselves and learn an entirely new set of programming concepts to achieve proficiency at developing applications for Windows. *Visual Basic* for Windows has many features that make it an ideal language for developing applications for Microsoft Windows.

Visual Basic for Windows programs are *event-driven,* which is a concept at the heart of all Microsoft Windows applications. *Event-driven* or *object-oriented* programming see its environment as a stack of objects which has basic properties and functions. When these objects are activated or choosen, they will carry out their pre-defined jobs.

Basically, this new programming concept is similar to us more than the old computer programming technics. As an example, we all have calculators on our desks. The calculator is an object with some special properties, if we push its buttons, it multiplies, adds, calculates numbers. Here we have an object and it has some predefined functions,

but to activate those functions we have to use the buttons on the calculator. Visual Basic has a programming concept just like this. If one designs a window with some objects, which has certain predefined properties and functions, the user can activate and make them run by selecting or pressing these.

4.3.2 Properties and the algorithms used in the software

The TD software consists of three main parts, all having their own distinct algorithms. During the development of the software *Visual Basic*[2] has been used because of its unique properties described in the previous section. For the sound processing tasks, "*Goldwave 2.0*"[3] software is used due to its useful and powerful sound playing capabilities.

In the database section *MS Access* file format has been used. This choice facilitates manipulation of these files or their export to other formats using the *MS Access* . This database is also used as a bridge between the three main parts of the TD. Since the program has three parts, the database also has three main tables for the three different types of records.

The first table is "*Patientdata*" and contains the data related to the personal information of the patients: "*Name*", "*Sex*", "*Age*", "*Education*" and "*Patient number*". The *patient number* is the key field of every table. All the tables in the database contain this data. This enables easy matching between the three tables. The second table is "*Audiodata*" and the parameters of the audiometric tests are recorded in this table. The third table "*Tinnitusdata*" and it has the three matched frequencies of the patient's tinnutus pitch frequency.

When the "*Yeni Hasta*" (new patient) option is chosen from the "Dosya" menu a little window will be opened in the middle of the screen. The user is expected to enter the personal information except the patient number. This number is automatically assigned by the software itself.

---

[2] 3.0 Pro version of Microsoft's Visual Basic has been used in thes thesis work.

[3] 2.00 Version of GoldWave Copyright © 1993-1994 Chris S. Craig

The "*Eski Hasta*" (old paitent) option is for looking back at the old records. This option can be futher developed and can be used for reporting or making statistical analysis. As explained before, the format of the files used for these database records is in the MS Access file format. Therefore they can easily be manipulated by the help of most popular database software without the need to develop the "*Eski hasta*" option.

The audio section of the TD has been designed to be simple with big, easy to use buttons. The reason of this simplicity is that in this part, the patient will use the program and find the sensitivity of his ears. When the audio form is activated, the *Goldwave (GW)* software begins to run at the background. There are six different frequencies opened: 125 Hz, 500 Hz, 1000 Hz, 2000 Hz, 4000 Hz and 8000 Hz. After the start button is pressed, 125 Hz sinusoidal wave starts to play.

The timer is the most important object in this part of the software. This is because with the timer object the level of the volume is continuously increased every 1.5 seconds. While listening to this sound, the patient must press the "*duydum*" button immidiately when he just hears the played sound. After this, the software resets the volume level to zero and begins to playback the next pure tone, continuously increasing its volume. This audiometric test should be applied to each ear.

At each ear, and after the patient pressed the "*duydum*" button the TD plots a graph which represents the hearing level at the test frequencies. The coefficients of the equations of this line are stored in a database in the "*Audiodata*" file. After completing the tests for each ear the user can pass to the last step, namely, pitch-matching.

The pitch-matching step is also designed for the self-use of the patient. Since the tinnitus sound is specific to the patient and since objective means of measuremnt of these sounds are not abvailable, it is the patient who should help herself or himself for finding out the right sound for his/her relief. At the start of this phase, the TD opens the database again and prepares itself for recording of the detected pitch frequencies.

The system uses only two pure tones in this part. The frequency of these tones are changed by changing the playback speed of the sound card. To scan the frequency band between 125 Hz - 8000 Hz, two sinusoidal wave files having 215 Hz and 2000 Hz

recording frequencies are used. In this technique, to obtain say 8000 Hz pure tone, the sampling rate of the 2000 Hz pure tone signal should be increased four times[4].

There is a simple algorithm to determine the pitch frequency: First the system begins playing 860 Hz (which is the top frequency that can be produced using 215 Hz) and by the help of the patient who presses "*Daha ince*" and "*Daha kalın*" buttons, the program gets closer and closer to the pitch frequency. The TD lets the patient two more trials for acquiring the best result.

The most interesting part of the pitch matching process is the automatic volume level adjustment according to the audio test data. At the begining of the process, the TD opens the database and looks for the "*audiodata*" table for the records of the audio test and using this data the TD can adjust the output volume at every frequency.

During the audio test, the algorithm pruduces six points that have frequency and volume level data: (f_0,v_0), (f_1,v_1), (f_2,v_2), (f_3,v_3), (f_4,v_4), (f_5,v_5). Using these points one can draw five lines. Since two coefficients are needed for each line, we have 10 different data for one ear and 10 data for the other. Thus with the help of these coefficients, the TD draws the proper line at each frequency and uses the proper volume level.

After the pitch-matching stage, the TD saves the data to the database and closes it. The TD can then generate a white noise that matches with the pitch frequency of the tinnitus. To be acceptable by the patient this sound data must be filtered using a narrow band filter. This can be done by the help of an external device or can be filtered by the help of the software.

4.3.3 Modules and objects of the software

The TD is composed of many modules and each module has many objects. A module is a single file, in which the procedures and declarations are stored together and recognized throughout an application.

The TD has five modules which are described below:

---

[4] This results in a four-fold increase in the effective playback speed.

CONSTANT.TXT : This file contains constants and declerations that are provided by Microsoft in the Visual Basic for Windows 3 package.

FILEMSG.BAS: The module for displaying messages.

TINNITUS.BAS: The type decleration of the *TASKENTRY.*

There are two VBX files in the software:

THREED.VBX: This file supplies the three dimensional controls used in the TD.

GRAPH.VBX: This file lets the TD plot graphics.

Five different forms are used in the whole application which are:

TIN1.FRM: This is the main form which appears at the start.

T_AUDI.FRM: Audiometric test form.

ABOUT.FRM: The form which presents the information about TD.

ESKİHST.FRM: Old data searching form.

YENİHST.FRM: New data entering form.

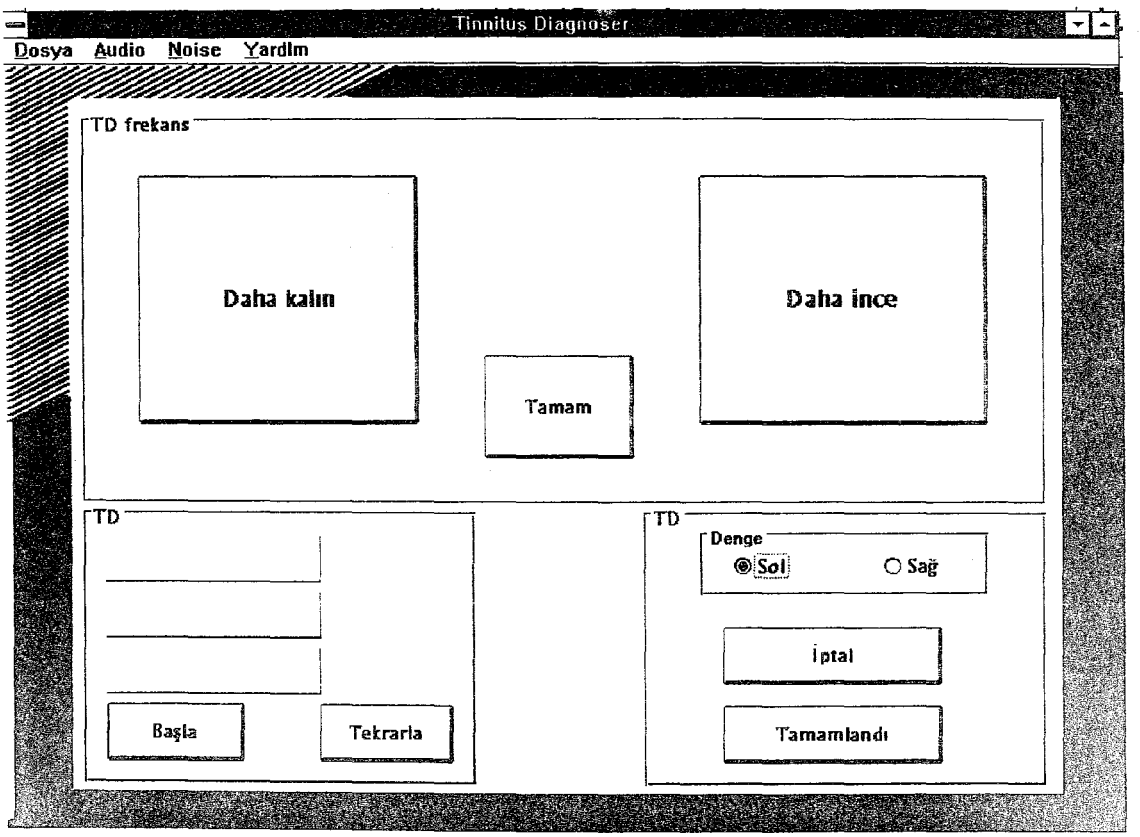


Figure 4-1 The main window of the Tinnitus Diagnoser.

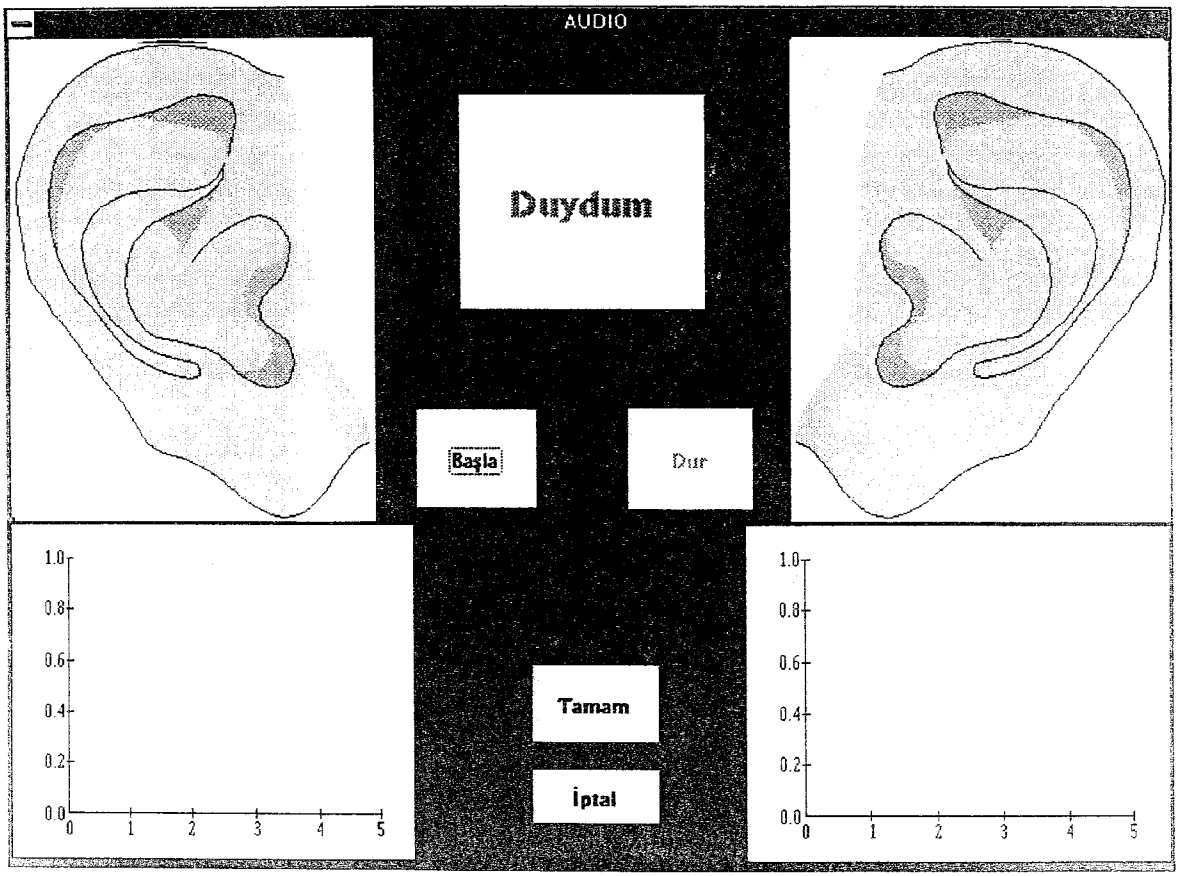


Figure 4-2 The window seen during the audiometric test process.

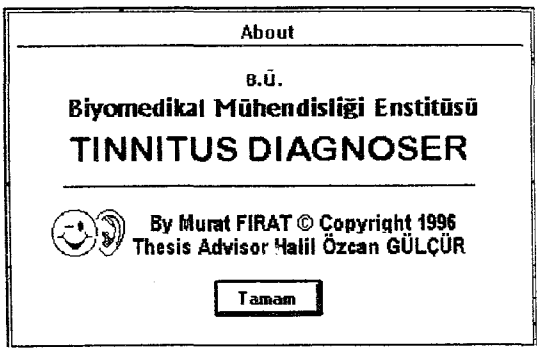


Figure 4-3 The introductory window to TD.

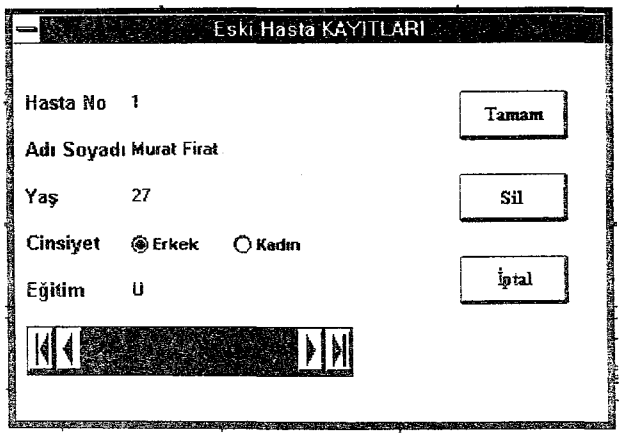


Figure 4-4 The window which appears during the search of the old data.

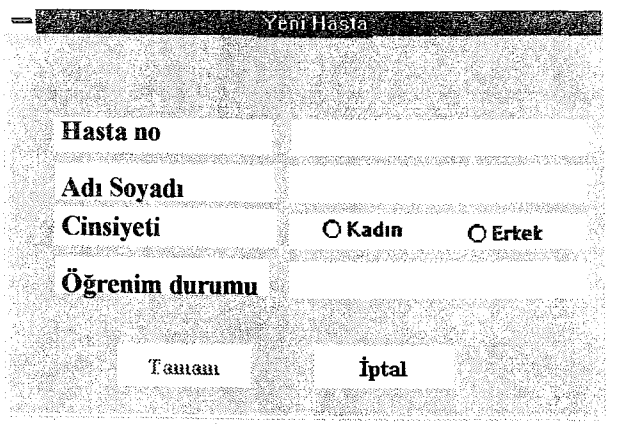


Figure 4-5 New data entering window.

4.4 The flow charts

The main flowchart of the TD is given in Fig.4.6. The flow-charts for the ''new patient data'' entries and for the ''old patient data'' searches are presented in Figs. 4.7 and 4.8, respectively. The flow-chart of the audiometry part is presented in Fig. 4.9 and the flow-chart of pitch-matching process is given in Fig. 4.10

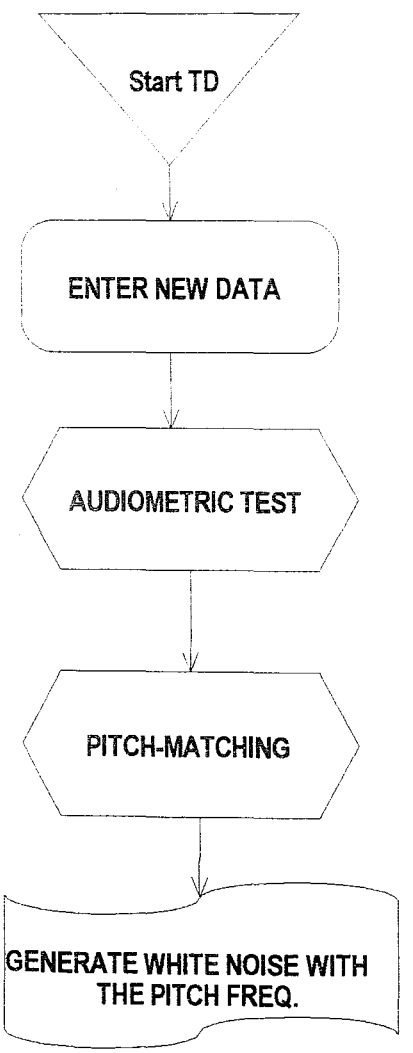


Figure 4-6 The main flow-chart of TD.

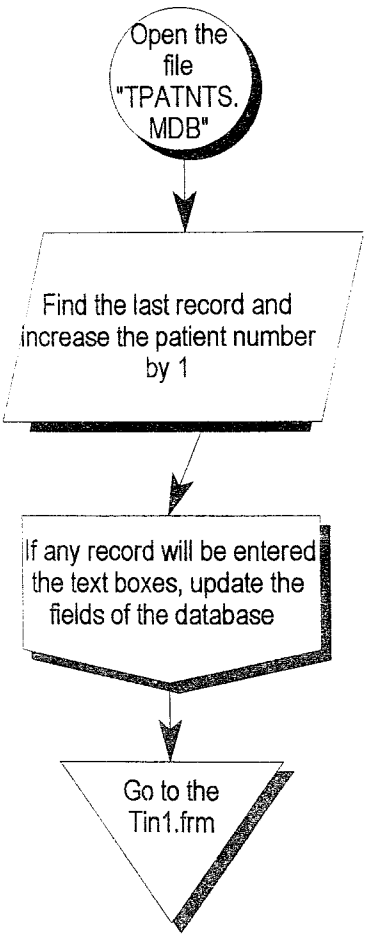


Figure 4-7 Flow-chart for new data entry.

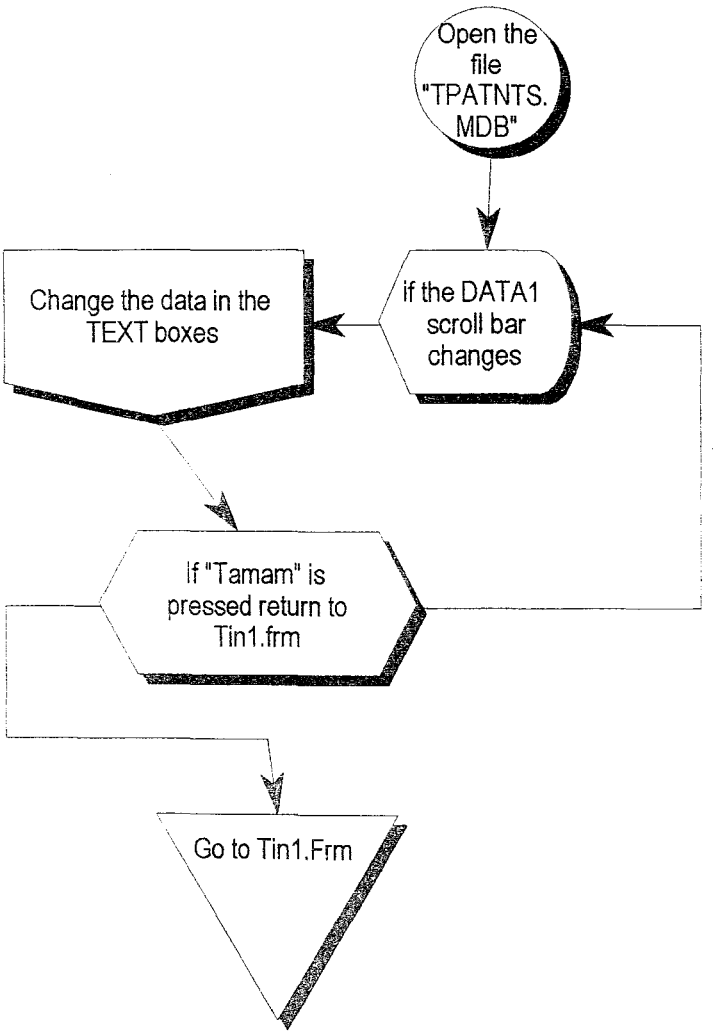


Figure 4-8 Flow-chart for old data searches.

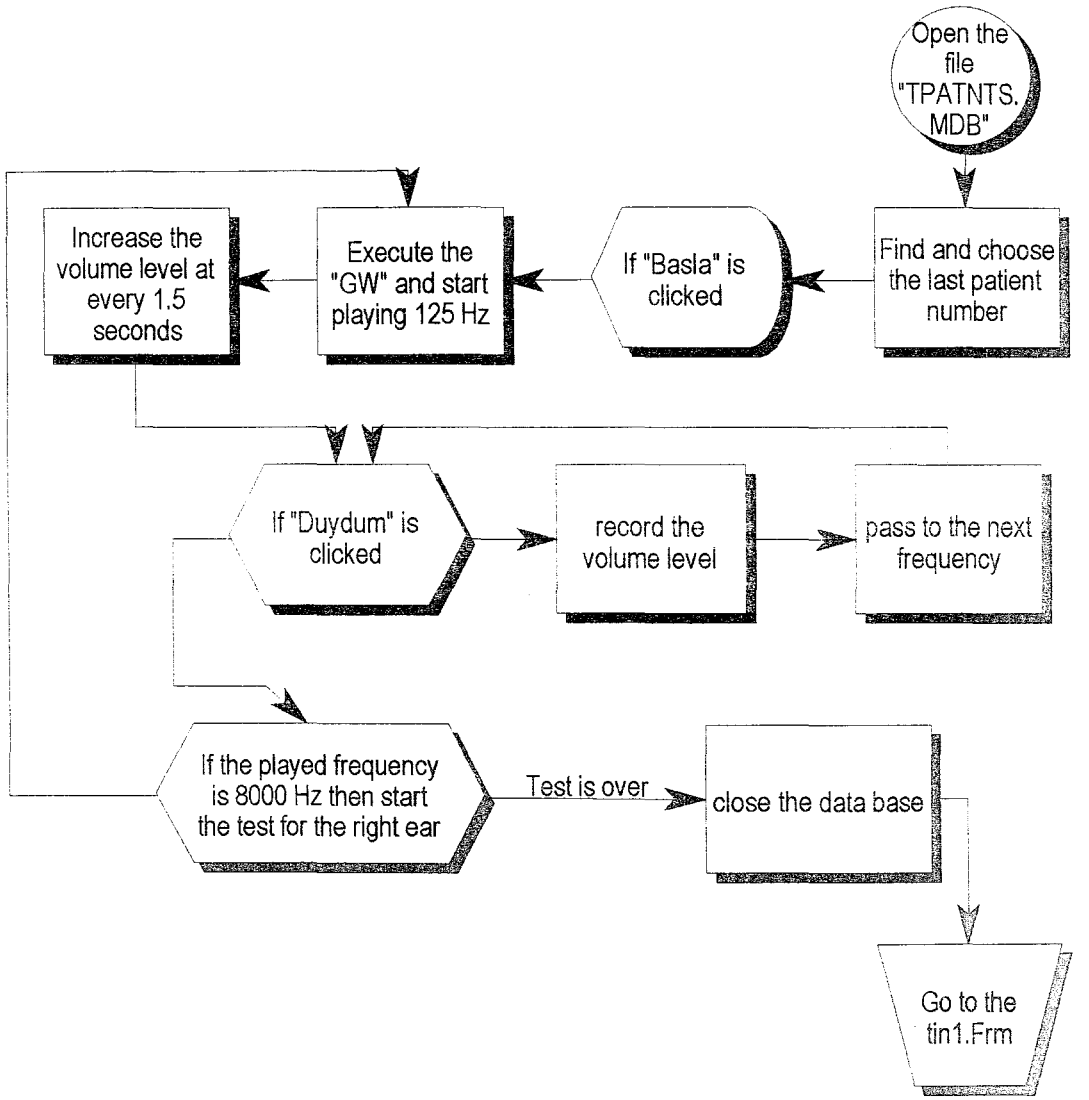


Figure 4-9 The flow-chart of the audiometry part.

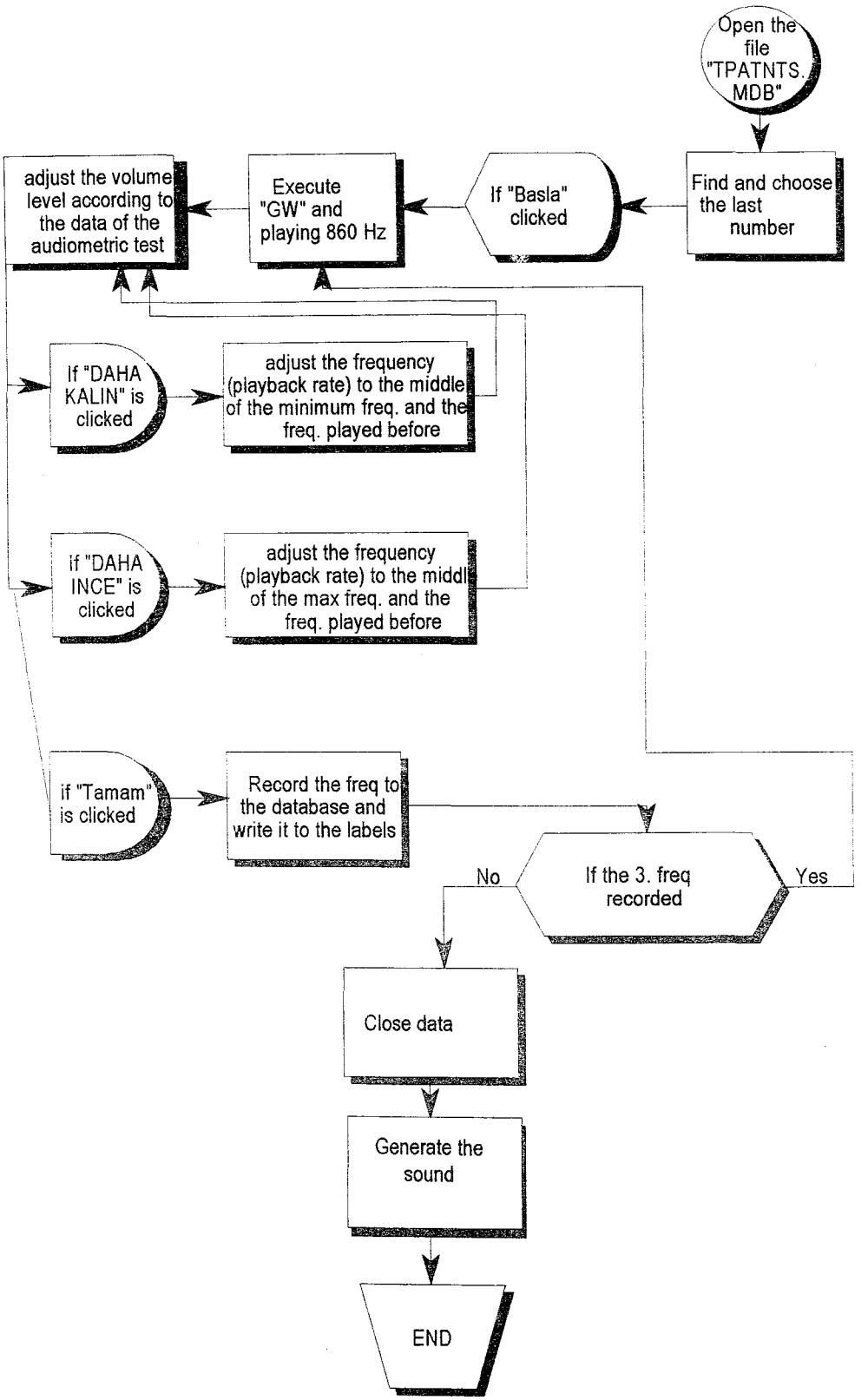


Figure 4-10 The flow-chart of pitch-matching.

4.5 Explanations

4.5.1 Sub-routines and functions used in TD

Tin1.FRM

get_handler () : finds the handler of the TD, GW, DC

loop_playback () : sets GW for continuous playback

open_wav (wv as string) : opens the wave file (wv)

play() : starts playing

reset_bal () : resets balance control of GW

reset_dc () : resets the device control window

reset_sp () : resets playback rate control

reset_volm () : resets the volume control

set_spe (I as integer) : sets the speed control to new position

set_volm (f as long) : sets volume control to new position

stp_pl () : stops playing

T_audy.FRM

ass_data (e,vl) : finds and records the audiometric data to the database

Audio_test (index) : plays the proper wav file

Function ***bn (I as integer)***: finds the coefficient b of the line equation between the two audiometric test points.

vol_up () : sets the volume to one level higher.

4.5.2 Other objects used in the forms

Tin1.FRM

Command3D1 : decreases the frequency of the playing tone

Command3D2 : increases the frequency of the playing tone

Command 3D3 : sets the selected frequency as pitch frequency

Command3D4 : cancels the process

Command3D5 : completes the process

Command3D6 : starts the process

Command3D7 : restarts the process

Option3D1 : sets the balance to the left ear

Option3D2 : sets the balance to the right ear

Label1-Label6 : the matched pitch frequencies are printed on them

T_audy.FRM

Command1 : sets the volume level and records it to the database, starts drawing of the graphs

Command2 : starts the TEST

Command3 : stops TEST

Command4 : cancel the TEST

Command5 : completes the TEST and updates records

Timer1 : provides the timer interval between the increasing steps of the volume level

Labels, picture boxes, images are the objects on which the user has no control.

ESKIHST.FRM

Command3D1 : unloads the form

Command3D2 : unloads the form

Data1 : opens the database file and becomes the data resource of the other related objects

Option boxes, textboxes are the part of the database management.

ESKIHST.FRM

Command3D1 : completes the new data enterance and saves them to the database

Command3D2 : cancels the recording

Option boxes, textboxes and labels are for entrance to new patient data to the database.

5. DISCUSSION AND CONCLUSIONS

It is believed that severe, disabling tinnitus ranks together with severe pain and balance disorders, as the third worst thing that can happen to a human beeing. As the incurable tinnutus patients are desperate, offering them a relief is equivalent to returning them back to life. Masking is one of the ways which is the most studied upon. As Jack Vernon puts it: "We should never deny relief to patients because of our own limitations" (Vernon, 1991).

In this study, a user-friendly software for easy pitch-matching of the tinnitus frequency and for the generation of the matching tinnitus masking sound has been developed. When the subjectivity and patient-specifity of the illness are taken into consideration, the help and direct participation of the patient is vital in determining the tinnitus frequency correctly. A considerable masking sound has to be in the volume level that the patient can hear without being disturbed. For an effective pitch-matching, the patient's hearing loss and sensitivity should be known and the the processes should be ordered according to these information.

5.1 Clinical Experience Results

To test the performance of the TD, a small clinical study was conducted. One of the older patients of Dr. Cengiz Çelikyurt, who had made a previous study on tinnitus in the B.U. Biomedical Engineering Institute, was called upon for this purpose. The patient was a female of 62 years old, suffering from tinnitus for 10 years. The reason for not choosing a new patient was to have the ideas and comments of an experienced patient on TD and to avoid waste of time by trying to explain everything to a new patient.

After a brief introduction to TD, it is observed that the patient quickly and quite comfortably found her tinnitus pitch frequency. The patient could easily and repetitiously make trials of pitch-matching. Although the patient was not computer-literate, she could control the device with the use of only three buttons. This implies that with a very limited and short training, everybody can use the TD.

5.2 Conclusions

The software developed in this thesis allows the physician to make a relative audiometric test. After this audiometric test, the software uses the test data while the pitch-matching is in process. At the end of this phase, a narrow-band white noise with a pure tone at the tinnitus frequency is generated.

The tools in the Sound Blaster Developer's Kit for driving the sound card (Sound Blaster AWE 32), were not directly usable with Visual Basic. Therefore, for playing the sounds at the background another software, *Goldwave version 2.0* has been used. Rather than generating different pure tones at different frequencies directly, they have been generated indirectly by adjusting the play-back rate of pre-stored pure tones

In the present study, a database option has also been added for recording the patient-specific tinnitus data and patients' personal information. Thanks to this database, the physician can have access to the old data of his or her patients. If needed, the physician can edit the database file which is created using Microsoft Access.

The software developed can be used for generating custom-made masking sounds. For further improvement of the software, a band-pass filtering function can be added to generate the narrow-band masking sounds.

The small clinical test showed that the pitch-matching, which is a time- consuming process for the physician and a boring one for the patients, will become a short and an easy procedure with the use of the TD. Eventually, the time wasting masking process will be performed with ease and the patients will have the opportunity to find a relief within a short period of time as a result of using the TD device.

APPENDIX A

LIST of SOFTWARE

ABOUT FORM

```
Sub Command1_Click ()
   'Cancel About form
   Unload About
End Sub

Sub Form_Load ()
   'Center form
   Left = (Screen.Width - Width) / 2
   Top = (Screen.Height - Height) / 2
   'Place both happy faces
   Image1.Left = 360
   Image1.Top = 1680
   Image2.Left = 360
   Image2.Top = 1680
   'Make first face visible
   Image1.ZOrder
   Image3.ZOrder 0
End Sub

Sub Timer1_Timer ()
   'Wink
   If Timer1.Interval <> 300 Then
      Timer1.Interval = 300
      Image2.ZOrder
      Image3.ZOrder
   'Undo wink
   Else
      Timer1.Interval = 2000
      Image2.ZOrder 1
      Image3.ZOrder 1
      Image1.ZOrder
   End If
End Sub
```

ESKİHST FORM

```
Sub Command3D1_Click ()
   Unload EskiHst
End Sub

Sub Command3D2_Click ()
   Unload EskiHst
End Sub

Sub Data1_Reposition ()
   'Center form
   Left = (Screen.Width - Width) / 2
   Top = (Screen.Height - Height) / 2
```

```
        Dim i As Integer
      i = Data1.Recordset("Sex")
   Select Case i
     Case 0:  Option3D1(0).Value = 1
           Option3D1(1).Value = 0
     Case 1:  Option3D1(1).Value = 1
           Option3D1(0).Value = 0
     Case 2:  Option3D1(0).Value = 0
           Option3D1(0).Value = 0
     End Select
End Sub

Sub Form_Load ()
   Dim i As Integer
   Data1.DatabaseName = "tpatnts.MDB"   ' Set Database file.
   Data1.Refresh   ' Open the Database.
   i = Data1.Recordset("Sex")
   Select Case i
     Case 0:  Option3D1(0).Value = True
           Option3D1(1).Value = False
     Case 1:  Option3D1(1).Value = True
           Option3D1(0).Value = False
   End Select
End Sub
```

T_AUDI form

```
Dim hwnd_gw, hwnd_dc, hwnd_au, vol, tab_count, stat_count, wave, left_right, end_au As Integer
Dim TDDb As Database, TDADn As Dynaset, TDDn As Dynaset, TDnum As Field, AUnum As
Field, Lc1 As Field, lc2 As Field, Lc3 As Field, Rc1 As Field, Rc2 As Field, Rc3 As Field
Dim Lc4 As Field, Lc5 As Field, Rc4 As Field, Rc5 As Field, Lb4 As Field, Lb5 As Field, Rb4 As
Field, Rb5 As Field
Dim Lb1 As Field, Lb2 As Field, Lb3 As Field, Rb1 As Field, Rb2 As Field, Rb3 As Field
Dim g(6), v(6)
Declare Function winexec Lib "Kernel" (ByVal lpCmdLine As String, ByVal nCmdShow As Integer)
As Integer
'Form handle numaralarını öğrenmek için
Declare Function GetWindow Lib "user" (ByVal hWnd As Integer, ByVal wCmd As Integer) As
Integer
'GetWindow, wCmd sabitleri
Const GW_CHILD = 5
Const GW_HWNDFIRST = 0
Const GW_HWNDLAST = 1
Const GW_HWNDNEXT = 2
Const GW_HWNDPREV = 3
Const GW_OWNER = 4
'Form başlıklarını öğrenmek için
Declare Function GetWindowText Lib "User" (ByVal hWnd As Integer, ByVal lpString As String,
ByVal aint As Integer) As Integer
'Form başlık uzunluğunu öğrenmek için
Declare Function GetWindowTextLength Lib "user" (ByVal hWnd As Integer) As Integer
'Form gösterim şeklini değiştirmek için
Declare Function ShowWindow Lib "User" (ByVal hWnd As Integer, ByVal nCmdShow As Integer)
As Integer
'Formu yok etmek için
Declare Function DestroyWindow Lib "User" (ByVal hWnd As Integer) As Integer
'Formu en üstte tutmak için
```

```
Declare Sub SetWindowPos Lib "User" (ByVal hWnd As Integer, ByVal hWndInsertAfter As
Integer, ByVal X As Integer, ByVal Y As Integer, ByVal cx As Integer, ByVal cy As Integer, ByVal
wFlags As Integer)
Const HWND_TOPMOST = -1
Const HWND_NOTOPMOST = -2
Const SWP_NOACTIVATE = &H10
Const SWP_SHOWWINDOW = &H40
'Form başlığını değiştirmek için
Declare Sub SetWindowText Lib "User" (ByVal hWnd As Integer, ByVal lpString As String)
Declare Function GetWindowTask Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function PostAppMessage Lib "User" (ByVal hTask As Integer, ByVal wMsg As Integer,
ByVal wParam As Integer, lParam As Any) As Integer
'pencereyi en üste getirmek için
Declare Sub BringWindowToTop Lib "User" (ByVal hWnd As Integer)
Sub ass_data (e, vl)
Dim b As Double
  If e = 0 Then
    Select Case vl
      Case 1: v(0) = vol
      Case 2: v(1) = vol
          Lb1 = bn(2)
          Lc1 = v(0) - Lb1 * 125
      Case 3: v(2) = vol
          Lb2 = bn(3)
          lc2 = v(1) - Lb2 * 500
      Case 4: v(3) = vol
          Lb3 = bn(4)
          Lc3 = v(2) - Lb3 * 1000
      Case 5: v(4) = vol
          Lb4 = bn(5)
          Lc4 = v(3) - Lb4 * 2000
      Case 6: v(5) = vol
          Lb5 = bn(6)
          Lc5 = v(4) - Lb5 * 4000
    End Select
  Else
    Select Case vl
      Case 1: v(0) = vol
      Case 2: v(1) = vol
          Rb1 = bn(2)
          Rc1 = v(0) - Rb1 * 125
      Case 3: v(2) = vol
          Rb2 = bn(3)
          Rc2 = v(1) - Rb2 * 500
      Case 4: v(3) = vol
          Rb3 = bn(4)
          Rc3 = v(2) - Rb3 * 1000
      Case 5: v(4) = vol
          Rb4 = bn(5)
          Rc4 = v(3) - Rb4 * 2000
      Case 6: v(5) = vol
          Rb5 = bn(6)
          Rc5 = v(4) - Rb5 * 4000
    End Select
  End If
End Sub

Sub Audio_test (index As Integer)
  Select Case index
```

```
        Case 1: close_wav
                open_wav ("c:\goldw\a125.wav")
        Case 2: close_wav
                open_wav ("c:\goldw\a500.wav")
        Case 3: close_wav
                open_wav ("c:\goldw\a1000.wav")
        Case 4: close_wav
                open_wav ("c:\goldw\a2000.wav")
        Case 5: close_wav
                open_wav ("c:\goldw\a4000.wav")
        Case 6: close_wav
                open_wav ("c:\goldw\a8000.wav")
    End Select
    reset_volm
    vol = 0
    play
    Timer1.Enabled = True
End Sub

Function bn (i As Integer) As Double
        bn = ((v(i - 1) - v(i - 2)) / (g(i - 1) - g(i - 2)))
End Function

Sub close_wav ()
    BringWindowToTop hwnd_gw
    SendKeys "^{F4}", -1
End Sub

Sub get_handler ()                          'get handler of the 3 window
  Dim aktif_pencere_hwnd, uz, i As Integer, a, b, c, baslik As String
    aktif_pencere_hwnd = GetWindow(Ana_form.hWnd, 0)
    While aktif_pencere_hwnd <> 0
       uz = GetWindowTextLength(aktif_pencere_hwnd)
       baslik = Space(uz + 1)
       uz = GetWindowText(aktif_pencere_hwnd, baslik, uz + 1)
       If uz > 0 Then
          a = Left$(baslik, 15)
          b = Left$(baslik, 14)
          c = Left$(baslik, 5)
          If a = "Device Controls" Then
             hwnd_dc = aktif_pencere_hwnd
          End If
          If b = "GoldWave v2.00" Then
             hwnd_gw = aktif_pencere_hwnd
          End If
          If c = "AUDIO" Then
             hwnd_au = aktif_pencere_hwnd
          End If
          End If
       aktif_pencere_hwnd = GetWindow(aktif_pencere_hwnd, 2)
    Wend
End Sub

Sub loop_playback ()              'setup the device control for loop
    BringWindowToTop hwnd_dc 'playback
    SendKeys "{TAB 6}", -1
    SendKeys "{ENTER}", -1
    SendKeys "{TAB 4}", -1
    SendKeys " ", -1
```

```
    SendKeys "{ENTER}", -1
    tab_count = 5
End Sub

Sub open_wav (wv As String)
    Dim aktif_pencere_hwnd, uz, hnd As Integer, a, baslik As String
    BringWindowToTop hwnd_gw
    SendKeys "%f", -1
    SendKeys "o", -1
    SendKeys wv, -1
    aktif_pencere_hwnd = GetWindow(Audio.hWnd, 0)
    While aktif_pencere_hwnd <> 0
        uz = GetWindowTextLength(aktif_pencere_hwnd)
        baslik = Space(uz + 1)
        uz = GetWindowText(aktif_pencere_hwnd, baslik, uz + 1)
        If uz > 0 Then
            a = Left$(baslik, 15)
            If a = "Open Sound File" Then
                hnd = aktif_pencere_hwnd
            End If
        End If
        aktif_pencere_hwnd = GetWindow(aktif_pencere_hwnd, 2)
    Wend
    BringWindowToTop hnd
    SendKeys "~", -1
End Sub

Sub play ()                'begin playing the wave file
    reset_dc
    BringWindowToTop hwnd_dc
    SendKeys "{ENTER}", -1
    tab_count = 0
    stat_count = 1
End Sub

Sub reset_bal ()                'set the balance to left
    reset_dc
    BringWindowToTop hwnd_dc
    SendKeys "{TAB 3}", -1
    SendKeys "{HOME}", -1
    tab_count = 3
End Sub

Sub reset_dc ()            'bring device control to top
    Dim ts As Integer
    BringWindowToTop hwnd_dc
    If stat_count = 0 Then
        ts = 6 - (tab_count Mod 6)
    Else
        ts = 8 - (tab_count Mod 8)
    End If
    Do Until ts = 0
    SendKeys "{TAB}", -1
    ts = ts - 1
    Loop
    tab_count = 0
End Sub

Sub reset_volm ()          'set the volume to 0
```

```
    reset_dc
    BringWindowToTop hwnd_dc
    SendKeys "{TAB 2}", -1
    SendKeys "{HOME}", -1
    tab_count = 2
End Sub

Sub stop_pl ()
    Dim i As Integer
    If stat_count = 1 Then
        Timer1.Enabled = False
        BringWindowToTop hwnd_dc
        i = tab_count Mod 8
        Select Case i
            Case 0: SendKeys " ", -1
            Case 1: SendKeys "{TAB 7}", -1
                    SendKeys " ", -1
            Case 2: SendKeys "{TAB 6}", -1
                    SendKeys " ", -1
            Case 3: SendKeys "{TAB 5}", -1
                    SendKeys " ", -1
            Case 4: SendKeys "{TAB 4}", -1
                    SendKeys " ", -1
            Case 5: SendKeys "{TAB 3}", -1
                    SendKeys " ", -1
            Case 6: SendKeys "{TAB 2}", -1
                    SendKeys " ", -1
            Case 7: SendKeys "{TAB 1}", -1
                    SendKeys " ", -1
        End Select
        stat_count = 0
        tab_count = 0
    End If
End Sub

Sub vol_up (i As Integer)
    Dim v
    reset_dc
    BringWindowToTop hwnd_dc
    SendKeys "{Tab}", -1
    For v = 0 To i - 1
        SendKeys "{RIGHT}", -1
    Next v
    tab_count = 1
End Sub

Sub Command1_Click ()
    Dim a, task_handle As Integer
    Timer1.Enabled = False
    stop_pl
    If left_right = 0 Then
        If wave < 7 Then
            Graph1.ThisPoint = wave
            Graph1.GraphData = vol
            Graph1.DrawMode = 2
            ass_data left_right, wave
            wave = wave + 1
            If wave = 7 Then
                left_right = 1
```

```
            reset_dc
            BringWindowToTop hwnd_dc
            SendKeys "{TAB 3}", -1
            SendKeys "{END}", -1
            tab_count = 3
            wave = 1
            image1.ZOrder 1
            image2.ZOrder
          End If
          Audio_test (wave)
        End If
      Else
        If wave < 7 Then
          Graph2.ThisPoint = wave
          Graph2.GraphData = vol
          Graph2.DrawMode = 2
          ass_data left_right, wave
          wave = wave + 1
          Audio_test (wave)
        End If
      End If
      end_au = end_au + 1
      If end_au = 12 Then
        Timer1.Enabled = False
        FileMsg "Aud01.MSG", 1
        On Local Error Resume Next
        a = DestroyWindow(hwnd_gw)
        task_handle = GetWindowTask(hwnd_gw)
        If task_handle <> 0 Then
          a = PostAppMessage(task_handle, &H12, 0, 0)
        End If
        Command1.Enabled = False
        Command4.Enabled = False
        image2.ZOrder 1
      End If
End Sub


Sub Command2_Click ()
    Dim a
    a = winexec("C:\td\goldwave.exe c:\td\a125.wav", 0)
    SendKeys "y", -1
    get_handler
        SetWindowPos hwnd_dc, HWND_TOPMOST, 0, 0, 0, 0, (SWP_NOACTIVATE Or
SWP_SHOWWINDOW)
    a = ShowWindow(hwnd_dc, 0)
    reset_dc
    loop_playback
    reset_volm
    reset_bal
    wave = 1
    left_right = 0
    Command1.Enabled = True
    Command4.Enabled = True
    Command2.Enabled = False
    image1.ZOrder
    Audio_test (wave)
End Sub
```

```
Sub Command3_Click ()
   Dim a, task_handle
   On Local Error Resume Next
   a = DestroyWindow(hwnd_gw)
   task_handle = GetWindowTask(hwnd_gw)
   If task_handle <> 0 Then
      a = PostAppMessage(task_handle, &H12, 0, 0)
   End If
   Unload Audio
End Sub

Sub Command4_Click ()
   Timer1.Enabled = False
   stop_pl
   On Local Error Resume Next
   a = DestroyWindow(hwnd_gw)
   task_handle = GetWindowTask(hwnd_gw)
   If task_handle <> 0 Then
      a = PostAppMessage(task_handle, &H12, 0, 0)
   End If
   Command1.Enabled = False
   Command4.Enabled = False
   Command2.Enabled = True
   image1.ZOrder 1
   image2.ZOrder 1
End Sub

Sub Command5_Click ()
   Dim a, task_handle
   On Local Error Resume Next
   a = DestroyWindow(hwnd_gw)
   task_handle = GetWindowTask(hwnd_gw)
   If task_handle <> 0 Then
      a = PostAppMessage(task_handle, &H12, 0, 0)
   End If
   'Ana_form.Frame3D1.Enabled = True
   Ana_form.Frame3D4.Enabled = True
   Ana_form.Frame3D5.Enabled = True
   TDADn.Update
   TDADn.Close
   TDDb.Close
   Unload Audio
End Sub

Sub Form_Load ()
   g(0) = 125
   g(1) = 500
   g(2) = 1000
   g(3) = 2000
   g(4) = 4000
   g(5) = 8000
   Timer1.Enabled = False
   'Center form
   Left = (Screen.Width - Width) / 2
   top = (Screen.Height - Height) / 2
   get_handler
      SetWindowPos hwnd_au, HWND_TOPMOST, 0, 0, 0, 0, (SWP_NOACTIVATE Or
SWP_SHOWWINDOW)
   Set TDDb = OpenDatabase("c:\td\TPATNTS.MDB")
```

```
  Set TDDn = TDDb.CreateDynaset("SELECT * FROM Patientdata")
  Set TDADn = TDDb.CreateDynaset("SELECT * FROM Audiodata")
  Set TDnum = TDDn.Fields("Pnumber")
  Set AUnum = TDADn.Fields("Pnumber")
  Set Lb1 = TDADn.Fields("Lb1")
  Set Lb2 = TDADn.Fields("Lb2")
  Set Lb3 = TDADn.Fields("Lb3")
  Set Lb4 = TDADn.Fields("Lb4")
  Set Lb5 = TDADn.Fields("Lb5")
  Set Lc1 = TDADn.Fields("Lc1")
  Set lc2 = TDADn.Fields("Lc2")
  Set Lc3 = TDADn.Fields("Lc3")
  Set Lc4 = TDADn.Fields("Lc4")
  Set Lc5 = TDADn.Fields("Lc5")
  Set Rb1 = TDADn.Fields("Rb1")
  Set Rb2 = TDADn.Fields("Rb2")
  Set Rb3 = TDADn.Fields("Rb3")
  Set Rb4 = TDADn.Fields("Rb4")
  Set Rb5 = TDADn.Fields("Rb5")
  Set Rc1 = TDADn.Fields("Rc1")
  Set Rc2 = TDADn.Fields("Rc2")
  Set Rc3 = TDADn.Fields("Rc3")
  Set Rc4 = TDADn.Fields("Rc4")
  Set Rc5 = TDADn.Fields("Rc5")
  TDDn.MoveLast
  TDADn.MoveLast
  TDADn.AddNew
  AUnum = TDnum
  TDDn.Close
End Sub

Sub Timer1_Timer ()
  vol_up (1)
  vol = vol + 1
End Sub
```

TIN1 Form

general,declerations

```
Option Explicit

Declare Function winexec Lib "Kernel" (ByVal lpCmdLine As String, ByVal nCmdShow As Integer)
As Integer
'Form handle numaralarını öğrenmek için
Declare Function GetWindow Lib "user" (ByVal hWnd As Integer, ByVal wCmd As Integer) As
Integer
'GetWindow, wCmd sabitleri
Const GW_CHILD = 5
Const GW_HWNDFIRST = 0
Const GW_HWNDLAST = 1
Const GW_HWNDNEXT = 2
Const GW_HWNDPREV = 3
Const GW_OWNER = 4
'Form başlıklarını öğrenmek için
Declare Function GetWindowText Lib "User" (ByVal hWnd As Integer, ByVal lpString As String,
ByVal aint As Integer) As Integer
```

```
'Form başlık uzunluğunu öğrenmek için
Declare Function GetWindowTextLength Lib "user" (ByVal hWnd As Integer) As Integer
'Form gösterim şeklini değiştirmek için
Declare Function ShowWindow Lib "User" (ByVal hWnd As Integer, ByVal nCmdShow As Integer)
As Integer
'Formu yok etmek için
Declare Function DestroyWindow Lib "User" (ByVal hWnd As Integer) As Integer
'Formu en üstte tutmak için
Declare Sub SetWindowPos Lib "User" (ByVal hWnd As Integer, ByVal hWndInsertAfter As
Integer, ByVal X As Integer, ByVal Y As Integer, ByVal cx As Integer, ByVal cy As Integer, ByVal
wFlags As Integer)
Const HWND_TOPMOST = -1
Const HWND_NOTOPMOST = -2
Const SWP_NOACTIVATE = &H10
Const SWP_SHOWWINDOW = &H40
Declare Function GetWindowTask Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function PostAppMessage Lib "User" (ByVal hTask As Integer, ByVal wMsg As Integer,
ByVal wParam As Integer, lParam As Any) As Integer
'pencereyi en üste getirmek için
Declare Sub BringWindowToTop Lib "User" (ByVal hWnd As Integer)

Declare Function GetWindowTask Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function PostAppMessage Lib "User" (ByVal hTask As Integer, ByVal wMsg As Integer,
ByVal wParam As Integer, lParam As Any) As Integer
Declare Sub BringWindowToTop Lib "User" (ByVal hWnd As Integer)

Dim hwnd_gw, hwnd_dc, hwnd_td, vol, vol_count, tab_count, s_count, w_count, stat_count, wave,
left_right, end_au As Integer
Dim w(4), s(57), s_max, s_min, pitch_n As Integer, file As String
Dim TDDb As Database, TDTDn As Dynaset, TDDn As Dynaset, TDnum As Field, Tnum As Field,
F1 As Field, F2 As Field, F3 As Field
Dim TDADn As Dynaset, AUnum As Field, Lc1 As Field, Lc2 As Field, Lc3 As Field, Lc4 As Field,
Lc5 As Field, Rc1 As Field, Rc2 As Field
Dim Rc3 As Field, Rc4 As Field, Rc5 As Field, Rb1 As Field, Rb2 As Field, Rb3 As Field, Rb4 As
Field, Rb5 As Field
Dim Lb1 As Field, Lb2 As Field, Lb3 As Field, Lb4 As Field, Lb5 As Field

Sub gen1 ()                    'Generating the masking soun
    SendKeys "{Esc}", -1
    'open an other file
    SendKeys "%f", -1
    SendKeys "n", -1
    SendKeys "{Tab}", -1
    SendKeys "{Down}", -1
    SendKeys "{Tab}", -1
    SendKeys "{Up 2}", -1
    SendKeys "{Tab}", -1
    SendKeys "{Up 2}", -1
    SendKeys "{Enter}", -1
    'convolv them
    SendKeys "%c", -1
    SendKeys "x", -1
    SendKeys "wave1" + "+9" + "n" + "+0" + "+=" + "wave2" + "+9" + "n" + "+0", -1
    SendKeys "{Enter}", -1
    Timer2.Interval = 2500
    Timer2.Enabled = True
End Sub

Sub gen2 ()                    'Generating the masking sound2
```

```
    Dim a, task_handle As Integer
    SendKeys "{Esc}", -1
    'save the file as
    SendKeys "%f", -1
    SendKeys "a", -1
    SendKeys file, -1
    SendKeys "{Enter}", -1
    SendKeys "y", -1
    'close file and exit
    On Local Error Resume Next
    a = DestroyWindow(hwnd_gw)
    task_handle = GetWindowTask(hwnd_gw)
    If task_handle <> 0 Then
       a = PostAppMessage(task_handle, &H12, 0, 0)
    End If
    TDTDn.Close
    TDDb.Close
    MsgBox "SES DOSYASI :" + file + " Üretildi."
End Sub

Sub open_wav (wv As String)
    Dim aktif_pencere_hwnd, uz, hnd As Integer, a, baslik As String
    BringWindowToTop hwnd_gw
    SendKeys "%f", -1
    SendKeys "o", -1
    SendKeys wv, -1
    aktif_pencere_hwnd = GetWindow(Ana_form.hWnd, 0)
    While aktif_pencere_hwnd <> 0
       uz = GetWindowTextLength(aktif_pencere_hwnd)
       baslik = Space(uz + 1)
       uz = GetWindowText(aktif_pencere_hwnd, baslik, uz + 1)
       If uz > 0 Then
          a = Left$(baslik, 15)
          If a = "Open Sound File" Then
            hnd = aktif_pencere_hwnd
          End If
        End If
        aktif_pencere_hwnd = GetWindow(aktif_pencere_hwnd, 2)
    Wend
    BringWindowToTop hnd
    SendKeys "~", -1
End Sub

Sub play ()                    'begin playing the wave file
    reset_dc
    BringWindowToTop hwnd_dc
    SendKeys "{ENTER}", -1
    tab_count = 0
    stat_count = 1
End Sub

Sub reset_sp ()          'set the playback rate to origin
    reset_dc
    BringWindowToTop hwnd_dc
    SendKeys "{Tab 4}", -1
    SendKeys "{HOME}", -1
    SendKeys "{RIGHT 6}", -1
    tab_count = 4
    s_count = 0
```

```
End Sub

Sub reset_volm ()          'set the volume to 0
   reset_dc
   BringWindowToTop hwnd_dc
   SendKeys "{TAB 2}", -1
   SendKeys "{HOME}", -1
   tab_count = 2
   vol_count = 0
End Sub

Sub set_spe (i As Integer)                    'set the playback rate
   Dim v
   reset_dc
   BringWindowToTop hwnd_dc
   SendKeys "{Tab 4}", -1
   SendKeys "{HOME}", -1
   For v = 0 To i
      SendKeys "{RIGHT}", -1
   Next v
   tab_count = 4
End Sub

Sub set_volm (f As Long)                      'set the volume
   Dim volum, vol_dif, s  As Integer
   If Option3D1.Value = True Then
      If f < 500 Then volum = Fix(Lb1 * f + Lc1)
      If f > 500 And f < 1000 Then volum = Fix(Lb2 * f + Lc2)
      If f > 1000 And f < 2000 Then volum = Fix(Lb3 * f + Lc3)
      If f > 2000 And f < 4000 Then volum = Fix(Lb4 * f + Lc4)
      If f > 4000 Then volum = Fix(Lb5 * f + Lc5)
    Else
      If f < 500 Then volum = Fix(Rb1 * f + Rc1)
      If f > 500 And f < 1000 Then volum = Fix(Rb2 * f + Rc2)
      If f > 1000 And f < 2000 Then volum = Fix(Rb3 * f + Rc3)
      If f > 2000 And f < 4000 Then volum = Fix(Rb4 * f + Rc4)
      If f > 4000 Then volum = Fix(Rb5 * f + Rc5)
   End If

   'If volum <> vol_count Then
      'vol_dif = volum - vol_count
      reset_dc
      BringWindowToTop hwnd_dc
      SendKeys "{TAB 2}", -1
      SendKeys "{HOME}", -1
      For s = 0 To volum
            SendKeys "{Right}", -1
      Next s
      'Else
       '  For s = vol_dif To 0 Step -1
        '     SendKeys "{Left}", -1
         ' Next s
      'End If
         tab_count = 2
       '  vol_coun = vol_count + s
   'End If
End Sub

Sub stp_pl ()                    'stop playing
```

```
    Dim i As Integer
    BringWindowToTop hwnd_dc
    i = tab_count Mod 8
    Select Case i
          Case 0: SendKeys " ", -1
          Case 1: SendKeys "{TAB 7}", -1
                SendKeys " ", -1
          Case 2: SendKeys "{TAB 6}", -1
                SendKeys " ", -1
          Case 3: SendKeys "{TAB 5}", -1
                SendKeys " ", -1
          Case 4: SendKeys "{TAB 4}", -1
                SendKeys " ", -1
          Case 5: SendKeys "{TAB 3}", -1
                SendKeys " ", -1
          Case 6: SendKeys "{TAB 2}", -1
                SendKeys " ", -1
          Case 7: SendKeys "{TAB 1}", -1
                SendKeys " ", -1
    End Select
    tab_count = 0
    stat_count = 0
End Sub

Sub Command3D1_Click ()
    Dim freq As Integer
    Dim s_d As Integer
    s_max = s_count
    s_count = s_count - Fix((s_count - s_min) / 2)
    s_d = s_count
    If w_count = 1 And s_count > 56 Then s_d = s_count - 57
    If w_count = 1 And s_count < 56 Then
       stp_pl
       BringWindowToTop hwnd_gw
       SendKeys "^{F6}", -1
       play
       w_count = 0
     End If
    Select Case w_count
       Case 0: freq = 215 * s(s_d)
       Case 1: freq = 2000 * s(s_d)
    End Select
    stp_pl
    set_spe (s_d)
    set_volm (freq)
    play
End Sub

Sub Command3D2_Click ()
    Dim s_d, freq As Integer
    s_min = s_count
    s_count = s_count + Fix((s_max - s_count) / 2)
    s_d = s_count
    If w_count = 1 And s_count > 56 Then s_d = s_count - 56
      Select Case w_count
        Case 0: freq = 215 * s(s_d)
        Case 1: freq = 2000 * s(s_d)
    End Select
    stp_pl
```

```
    set_spe (s_d)
    set_volm (freq)
    play
End Sub
Sub Command3D3_Click ()
    Dim s_d, freq As Integer
        s_d = s_count
        If w_count = 1 And s_count > 56 Then
            s_d = s_count - 57
        End If
        Select Case w_count
            Case 0: freq = Fix(215 * s(s_d))
            Case 1: freq = Fix(2000 * s(s_d))
        End Select
        Select Case pitch_n
            Case 1: Label1.Caption = "1. Frekans :"
                    F1 = freq
                    Label2.Caption = freq
            Case 2: Label3.Caption = "2. Frekans :"
                    F2 = freq
                    Label4.Caption = freq
            Case 3: Label5.Caption = "3. Frekans :"
                    F3 = freq
                    Label6.Caption = freq
        End Select
        pitch_n = pitch_n + 1
        stp_pl
        If pitch_n = 4 Then
            MsgBox "Tinnutus frekansi bulundu"
            Command3D5.Enabled = True
            Command3D7.Enabled = True
            Command3D3.Enabled = False
        Else
            Command3D7_click
        End If
    End Sub

Sub Command3D4_Click ()
    Dim a, task_handle
    On Local Error Resume Next
    a = DestroyWindow(hwnd_gw)
    task_handle = GetWindowTask(hwnd_gw)
    If task_handle <> 0 Then
        a = PostAppMessage(task_handle, &H12, 0, 0)
    End If
    TDDb.Close
    menDosyaÇıkış.Enabled = True
End Sub

Sub Command3D5_Click ()
    Dim a, task_handle
    TDTDn.Update
    TDADn.Close
    TDTDn.Close
    TDDb.Close
    On Local Error Resume Next
    a = DestroyWindow(hwnd_gw)
    task_handle = GetWindowTask(hwnd_gw)
    If task_handle <> 0 Then
```

```
    a = PostAppMessage(task_handle, &H12, 0, 0)
  End If
  menDosyaÇıkış.Enabled = True
  Command3D5.Enabled = False
  menNoise.Visible = True
  menNoise.Enabled = True
End Sub


Sub Command3D6_click ()
  Dim a
  Set TDDb = OpenDatabase("c:\td\TPATNTS.MDB")
  Set TDDn = TDDb.CreateDynaset("SELECT * FROM Patientdata")
  Set TDADn = TDDb.CreateDynaset("SELECT * FROM Audiodata")
  Set TDTDn = TDDb.CreateDynaset("SELECT * FROM Tinnitusdata")
  Set TDnum = TDDn.Fields("Pnumber")
  Set AUnum = TDADn.Fields("Pnumber")
  Set Tnum = TDTDn.Fields("Pnumber")
  Set Lb1 = TDADn.Fields("Lb1")
  Set Lb2 = TDADn.Fields("Lb2")
  Set Lb3 = TDADn.Fields("Lb3")
  Set Lb4 = TDADn.Fields("Lb4")
  Set Lb5 = TDADn.Fields("Lb5")
  Set Lc1 = TDADn.Fields("Lc1")
  Set Lc2 = TDADn.Fields("Lc2")
  Set Lc3 = TDADn.Fields("Lc3")
  Set Lc4 = TDADn.Fields("Lc4")
  Set Lc5 = TDADn.Fields("Lc5")
  Set Rb1 = TDADn.Fields("Rb1")
  Set Rb2 = TDADn.Fields("Rb2")
  Set Rb3 = TDADn.Fields("Rb3")
  Set Rb4 = TDADn.Fields("Rb4")
  Set Rb5 = TDADn.Fields("Rb5")
  Set Rc1 = TDADn.Fields("Rc1")
  Set Rc2 = TDADn.Fields("Rc2")
  Set Rc3 = TDADn.Fields("Rc3")
  Set Rc4 = TDADn.Fields("Rc4")
  Set Rc5 = TDADn.Fields("Rc5")
  Set F1 = TDTDn.Fields("Ptfreq1")
  Set F2 = TDTDn.Fields("Ptfreq2")
  Set F3 = TDTDn.Fields("Ptfreq3")
  TDDn.MoveLast
  TDADn.MoveLast
  TDTDn.AddNew
  Tnum = TDnum
  TDDn.Close
  a = winexec("C:\td\goldwave.exe c:\td\t215.wav", 0)
  get_handler
      SetWindowPos hwnd_dc, HWND_TOPMOST, 0, 0, 0, 0, (SWP_NOACTIVATE Or
SWP_SHOWWINDOW)
  a = ShowWindow(hwnd_dc, 0)
  loop_playback
  reset_volm
  reset_bal
  reset_sp
  open_wav ("c:\td\t2000.wav")
  s_count = 57
  w_count = 1
  s_max = 114
```

```
    s_min = 0
    pitch_n = 1
    Command3D6.Enabled = False
    Command3D4.Enabled = False
    Command3D5.Enabled = False
    Command3D7.Enabled = False
    menDosyaÇıkış.Enabled = False
    Frame3D1.Enabled = True
    set_volm (860)
    play
End Sub


Sub Command3D7_click ()
    If w_count = 0 Then
        BringWindowToTop hwnd_gw
        SendKeys "^{F6}", -1
    End If
    reset_volm
    reset_sp
    s_count = 0
    w_count = 1
    menNoise.Enabled = False
    Command3D3.Enabled = True
    Command3D6.Enabled = False
    Command3D4.Enabled = False
    Command3D5.Enabled = False
    Command3D7.Enabled = False
    set_volm (860)
    play
End Sub

Sub Form_Load ()
    Dim n, i As Integer
    'Position form
    Left = (Screen.Width - Width) / 2
    Top = (Screen.Height - Height) / 2
    get_handler
        SetWindowPos hwnd_td, HWND_TOPMOST, 0, 0, 0, 0, (SWP_NOACTIVATE Or
SWP_SHOWWINDOW)
    s(0) = .43
    For i = 1 To 57
        n = i Mod 4
        If n = 1 Then
            s(i) = s(i - 1) + .07
        Else
            s(i) = s(i - 1) + .06
        End If
    Next i
End Sub

Sub menAudioBaşla_Click ()
    Audio.Show
End Sub

Sub menDosyaÇıkış_Click ()
    End
End Sub
```

```
Sub menEskiHasta_Click ()
   EskiHst.Show
End Sub

Sub menNoiseGenerate_Click ()
   Dim a, task_handle, p_freq As Integer, pnumb As String, TDname As Field
   Set TDDb = OpenDatabase("c:\td\TPATNTS.MDB")
   Set TDTDn = TDDb.CreateDynaset("SELECT * FROM Tinnitusdata")
   Set Tnum = TDTDn.Fields("Pnumber")
   Set F1 = TDTDn.Fields("Ptfreq1")
   Set F2 = TDTDn.Fields("Ptfreq2")
   Set F3 = TDTDn.Fields("Ptfreq3")
   TDTDn.MoveLast
   p_freq = (F1 + F2 + F3) / 3
   a = Len(Str$(Tnum))
   pnumb = Left$(Tnum, a - 1)
   file = "c:\td\wav\" + pnumb + ".wav"
   On Local Error Resume Next
   a = DestroyWindow(hwnd_gw)
   task_handle = GetWindowTask(hwnd_gw)
   If task_handle <> 0 Then
      a = PostAppMessage(task_handle, &H12, 0, 0)
   End If
   a = winexec("C:\td\goldwave.exe c:\td\wn.wav", 0)
   get_handler
       SetWindowPos hwnd_dc, HWND_TOPMOST, 0, 0, 0, 0, (SWP_NOACTIVATE Or
SWP_SHOWWINDOW)
   a = ShowWindow(hwnd_dc, 0)
   BringWindowToTop hwnd_gw
   'open a new wave file
   SendKeys "%f", -1
   SendKeys "n", -1
   SendKeys "{Tab}", -1
   SendKeys "{Down}", -1
   SendKeys "{Tab}", -1
   SendKeys "{Up 2}", -1
   SendKeys "{Tab}", -1
   SendKeys "{Up 2}", -1
   SendKeys "{Enter}", -1
   'generate the sinusoidal tone
   SendKeys "%c", -1
   SendKeys "x", -1
   SendKeys "sin" + "+9" + "2*pi*f*t" + "+0", -1
   SendKeys "{Tab 3}", -1
   SendKeys Str$(p_freq), -1
   SendKeys "{Enter}", -1
   Timer1.Interval = 2000
   Timer1.Enabled = True
End Sub

Sub menYardımHakkında_Click ()
   About.Show MODAL
End Sub

Sub menYeniHasta_Click ()
   YeniHst.Show
End Sub
Sub Option3D1_Click (Value As Integer)
```

```
    reset_dc
    BringWindowToTop hwnd_dc
    If stat_count = 1 Then
        SendKeys "{TAB 2}", -1
        tab_count = 2
    Else
        SendKeys "{TAB 3}", -1
        tab_count = 3
    End If
    SendKeys "{HOME}", -1
End Sub
Sub Option3D2_Click (Value As Integer)
    reset_dc
    BringWindowToTop hwnd_dc
    If stat_count = 1 Then
        SendKeys "{TAB 2}", -1
        tab_count = 2
    Else
        SendKeys "{TAB 3}", -1
        tab_count = 3
    End If
    SendKeys "{END}", -1
End Sub


Sub Timer1_Timer ()
    Timer1.Enabled = False
    gen1
End Sub

Sub Timer2_Timer ()
    Timer2.Enabled = False
    gen2
End Sub
```

YeniHst Form

```
Option Explicit
Dim t As Integer, TDDb As Database, TDDn As Dynaset, TDnum As Field, TDname As Field,
TDage As Field, TDsex As Field, TDedu As Field

Sub Command1_Click ()
    TDnum = t
    TDDn.Update
    TDDn.Close
    TDDb.Close
    Ana_form.menAudioBaşla.Enabled = True
    Unload YeniHst
End Sub

Sub Command2_Click ()
    TDDn.Close
    TDDb.Close
    Unload YeniHst
End Sub

Sub Form_Load ()
    'Center form
```

```
    Left = (Screen.Width - Width) / 2
    Top = (Screen.Height - Height) / 2
    Set TDDb = OpenDatabase("c:\td\TPATNTS.MDB")
    Set TDDn = TDDb.CreateDynaset("SELECT * FROM Patientdata")
    Set TDname = TDDn.Fields("Name")
    Set TDage = TDDn.Fields("Age")
    Set TDsex = TDDn.Fields("Sex")
    Set TDedu = TDDn.Fields("Education")
    Set TDnum = TDDn.Fields("Pnumber")
    TDDn.MoveLast
    t = TDnum + 1
    TDDn.AddNew
    Label2.Caption = t
End Sub

Sub Option3D1_Click (Value As Integer)
    TDsex = 0
    Command1.Enabled = True
End Sub

Sub Option3D2_Click (Value As Integer)
    TDsex = 1
    Command1.Enabled = True
End Sub

Sub Text1_Change ()
    TDname = Text1.Text
End Sub

Sub Text2_Change ()
    TDage = Text2.Text
End Sub

Sub Text4_Change ()
    TDedu = Text4.Text
End Sub
```

Filemsg BAS

```
Sub FileMsg (FileName$, Section%)
    'Determine path for message file
    MsgFile$ = App.Path + "\" + FileName$

    'Be sure file exists
    Fil$ = Dir$(MsgFile$)
    If Fil$ = "" Then
        Msg$ = "File " + MsgFile$ + " not found"
        MsgBox Msg$, 48, "FILEMSG"
        Exit Sub
    End If

    'Create newline string
    NL$ = Chr$(13) + Chr$(10)

    'Open message file for reading
```

```
NumFile% = FreeFile
Open MsgFile$ For Input As #NumFile%

'Find specified section
Do Until EOF(NumFile%)
   Line Input #NumFile%, FileTxt$
      If Left$(FileTxt$, 1) = ">" Then
      If Val(Mid$(FileTxt$, 2)) = Section Then
         Exit Do
      End If
   End If
Loop
'Did we reach end of file during search?
If EOF(NumFile%) Then
   Msg$ = "Message section" + Str$(Section) + " not found"
   MsgBox Msg$
   Exit Sub
End If

'Extract message box type and title
FileTxt$ = RTrim$(LTrim$(Mid$(FileTxt$, 2)))
FileTxt$ = Mid$(FileTxt$, InStr(FileTxt$, ",") + 1)
TypeNum% = Val(FileTxt$)
Title$ = LTrim$(Mid$(FileTxt$, InStr(FileTxt$, ",") + 1))

'Loop through all sections of block
Do

   'Clear message string
   Msg$ = ""

   'Read message section
   Do Until EOF(NumFile%)
      Line Input #NumFile%, FileTxt$
      If Left$(FileTxt$, 1) = ">" Then
         Exit Do
      End If
      Msg$ = Msg$ + FileTxt$ + NL$
   Loop

   'Chop off any ending blank lines
   Do While Right$(Msg$, 4) = NL$ + NL$
      Msg$ = Left$(Msg$, Len(Msg$) - 2)
   Loop

   'Display message block
   If Msg$ <> "" Then
      MsgBox Msg$, TypeNum%, Title$
   End If

'Continue block if > was by itself
Loop While LTrim$(RTrim$(FileTxt$)) = ">"

'We've finished with file
Close NumFile%
End Sub
```

Tinnitus BAS

```
Option Explicit

Type TASKENTRY
  dwSize As Long
  hTask As Integer
  hTaskParent  As Integer
  hInst As Integer
  hModule As Integer
  wSS As Integer
  wSP As Integer
  wStackTop As Integer
  wStackMinimum As Integer
  wStackBottom As Integer
  wcEvents As Integer
  hQueue As Integer
  szModule As String * 10
  wPSPOffset As Integer
  hNext As Integer
End Type

'dwSize  Specifies the size of the TASKENTRY structure, in bytes.
'hTask Identifies the task handle for the stack.
'hTaskParent Identifies the parent of the task.
'hInst Identifies the instance handle of the task. This value is equivalent to the task's
'DGROUP segment selector.
'hModule Identifies the module that contains the currently executing function.
'wSS Contains the value in the SS register.
'wSP Contains the value in the SP register.
'wStackTop Specifies the offset to the top of the stack (lowest address on the stack).
'wStackMinimum Specifies the lowest segment number of the stack during execution of the task.
'wStackBottom  Specifies the offset to the bottom of the stack (highest address on the stack).
'wcEvents  Specifies the number of pending events.
'hQueue  Identifies the task queue.
'szModule  Specifies the name of the module that contains the currently executing function.
'wPSPOffset  Specifies the offset from the program segment prefix (PSP) to the beginning of
'the executable code segment.
'hNext Identifies the next entry in the task list. This member is reserved for internal use
'by Windows.
```

Constant Text

```
' Visual Basic global constant file. This file can be loaded
' into a code module.
'
' Some constants are commented out because they have
' duplicates (e.g., NONE appears several places).
'
' If you are updating a Visual Basic application written with
' an older version, you should replace your global constants
' with the constants in this file.
'
''''''''''''''''''''''''

' General
```

```
' Clipboard formats
Global Const CF_LINK = &HBF00
Global Const CF_TEXT = 1
Global Const CF_BITMAP = 2
Global Const CF_METAFILE = 3
Global Const CF_DIB = 8
Global Const CF_PALETTE = 9

' DragOver
Global Const ENTER = 0
Global Const LEAVE = 1
Global Const OVER = 2

' Drag (controls)
Global Const CANCEL = 0
Global Const BEGIN_DRAG = 1
Global Const END_DRAG = 2

' Show parameters
Global Const MODAL = 1
Global Const MODELESS = 0

' Arrange Method
' for MDI Forms
Global Const CASCADE = 0
Global Const TILE_HORIZONTAL = 1
Global Const TILE_VERTICAL = 2
Global Const ARRANGE_ICONS = 3

'ZOrder Method
Global Const BRINGTOFRONT = 0
Global Const SENDTOBACK = 1

' Key Codes
Global Const KEY_LBUTTON = &H1
Global Const KEY_RBUTTON = &H2
Global Const KEY_CANCEL = &H3
Global Const KEY_MBUTTON = &H4    ' NOT contiguous with L & RBUTTON
Global Const KEY_BACK = &H8
Global Const KEY_TAB = &H9
Global Const KEY_CLEAR = &HC
Global Const KEY_RETURN = &HD
Global Const KEY_SHIFT = &H10
Global Const KEY_CONTROL = &H11
Global Const KEY_MENU = &H12
Global Const KEY_PAUSE = &H13
Global Const KEY_CAPITAL = &H14
Global Const KEY_ESCAPE = &H1B
Global Const KEY_SPACE = &H20
Global Const KEY_PRIOR = &H21
Global Const KEY_NEXT = &H22
Global Const KEY_END = &H23
Global Const KEY_HOME = &H24
Global Const KEY_LEFT = &H25
Global Const KEY_UP = &H26
Global Const KEY_RIGHT = &H27
Global Const KEY_DOWN = &H28
Global Const KEY_SELECT = &H29
```

```
Global Const KEY_PRINT = &H2A
Global Const KEY_EXECUTE = &H2B
Global Const KEY_SNAPSHOT = &H2C
Global Const KEY_INSERT = &H2D
Global Const KEY_DELETE = &H2E
Global Const KEY_HELP = &H2F

' KEY_A thru KEY_Z are the same as their ASCII equivalents: 'A' thru 'Z'
' KEY_0 thru KEY_9 are the same as their ASCII equivalents: '0' thru '9'

Global Const KEY_NUMPAD0 = &H60
Global Const KEY_NUMPAD1 = &H61
Global Const KEY_NUMPAD2 = &H62
Global Const KEY_NUMPAD3 = &H63
Global Const KEY_NUMPAD4 = &H64
Global Const KEY_NUMPAD5 = &H65
Global Const KEY_NUMPAD6 = &H66
Global Const KEY_NUMPAD7 = &H67
Global Const KEY_NUMPAD8 = &H68
Global Const KEY_NUMPAD9 = &H69
Global Const KEY_MULTIPLY = &H6A
Global Const KEY_ADD = &H6B
Global Const KEY_SEPARATOR = &H6C
Global Const KEY_SUBTRACT = &H6D
Global Const KEY_DECIMAL = &H6E
Global Const KEY_DIVIDE = &H6F
Global Const KEY_F1 = &H70
Global Const KEY_F2 = &H71
Global Const KEY_F3 = &H72
Global Const KEY_F4 = &H73
Global Const KEY_F5 = &H74
Global Const KEY_F6 = &H75
Global Const KEY_F7 = &H76
Global Const KEY_F8 = &H77
Global Const KEY_F9 = &H78
Global Const KEY_F10 = &H79
Global Const KEY_F11 = &H7A
Global Const KEY_F12 = &H7B
Global Const KEY_F13 = &H7C
Global Const KEY_F14 = &H7D
Global Const KEY_F15 = &H7E
Global Const KEY_F16 = &H7F

Global Const KEY_NUMLOCK = &H90

' Variant VarType tags

Global Const V_EMPTY = 0
Global Const V_NULL = 1
Global Const V_INTEGER = 2
Global Const V_LONG = 3
Global Const V_SINGLE = 4
Global Const V_DOUBLE = 5
Global Const V_CURRENCY = 6
Global Const V_DATE = 7
Global Const V_STRING = 8


' Event Parameters
```

```
' ErrNum (LinkError)
Global Const WRONG_FORMAT = 1
Global Const DDE_SOURCE_CLOSED = 6
Global Const TOO_MANY_LINKS = 7
Global Const DATA_TRANSFER_FAILED = 8

' QueryUnload
Global Const FORM_CONTROLMENU = 0
Global Const FORM_CODE = 1
Global Const APP_WINDOWS = 2
Global Const APP_TASKMANAGER = 3
Global Const FORM_MDIFORM = 4

' Properties

' Colors
Global Const BLACK = &H0&
Global Const RED = &HFF&
Global Const GREEN = &HFF00&
Global Const YELLOW = &HFFFF&
Global Const BLUE = &HFF0000
Global Const MAGENTA = &HFF00FF
Global Const CYAN = &HFFFF00
Global Const WHITE = &HFFFFFF

' System Colors
Global Const SCROLL_BARS = &H80000000          ' Scroll-bars gray area.
Global Const DESKTOP = &H80000001              ' Desktop.
Global Const ACTIVE_TITLE_BAR = &H80000002     ' Active window caption.
Global Const INACTIVE_TITLE_BAR = &H80000003   ' Inactive window caption.
Global Const MENU_BAR = &H80000004             ' Menu background.
Global Const WINDOW_BACKGROUND = &H80000005    ' Window background.
Global Const WINDOW_FRAME = &H80000006         ' Window frame.
Global Const MENU_TEXT = &H80000007            ' Text in menus.
Global Const WINDOW_TEXT = &H80000008          ' Text in windows.
Global Const TITLE_BAR_TEXT = &H80000009       ' Text in caption, size box, scroll-bar arrow
box..
Global Const ACTIVE_BORDER = &H8000000A        ' Active window border.
Global Const INACTIVE_BORDER = &H8000000B      ' Inactive window border.
Global Const APPLICATION_WORKSPACE = &H8000000C ' Background color of multiple
document interface (MDI) applications.
Global Const HIGHLIGHT = &H8000000D            ' Items selected item in a control.
Global Const HIGHLIGHT_TEXT = &H8000000E       ' Text of item selected in a control.
Global Const BUTTON_FACE = &H8000000F          ' Face shading on command buttons.
Global Const BUTTON_SHADOW = &H80000010        ' Edge shading on command buttons.
Global Const GRAY_TEXT = &H80000011            ' Grayed (disabled) text. This color is set to 0 if
the current display driver does not support a solid gray color.
Global Const BUTTON_TEXT = &H80000012          ' Text on push buttons.

' Enumerated Types

' Align (picture box)
Global Const NONE = 0
Global Const ALIGN_TOP = 1
Global Const ALIGN_BOTTOM = 2

' Alignment
Global Const LEFT_JUSTIFY = 0  ' 0 - Left Justify
```

```
Global Const RIGHT_JUSTIFY = 1 ' 1 - Right Justify
Global Const CENTER = 2        ' 2 - Center

' BorderStyle (form)
'Global Const NONE = 0         ' 0 - None
Global Const FIXED_SINGLE = 1   ' 1 - Fixed Single
Global Const SIZABLE = 2        ' 2 - Sizable (Forms only)
Global Const FIXED_DOUBLE = 3   ' 3 - Fixed Double (Forms only)

' BorderStyle (Shape and Line)
'Global Const TRANSPARENT = 0    '0 - Transparent
'Global Const SOLID = 1          '1 - Solid
'Global Const DASH = 2           ' 2 - Dash
'Global Const DOT = 3            ' 3 - Dot
'Global Const DASH_DOT = 4       ' 4 - Dash-Dot
'Global Const DASH_DOT_DOT = 5 ' 5 - Dash-Dot-Dot
'Global Const INSIDE_SOLID = 6 ' 6 - Inside Solid

' MousePointer
Global Const DEFAULT = 0        ' 0 - Default
Global Const ARROW = 1          ' 1 - Arrow
Global Const CROSSHAIR = 2      ' 2 - Cross
Global Const IBEAM = 3          ' 3 - I-Beam
Global Const ICON_POINTER = 4   ' 4 - Icon
Global Const SIZE_POINTER = 5   ' 5 - Size
Global Const SIZE_NE_SW = 6     ' 6 - Size NE SW
Global Const SIZE_N_S = 7       ' 7 - Size N S
Global Const SIZE_NW_SE = 8     ' 8 - Size NW SE
Global Const SIZE_W_E = 9       ' 9 - Size W E
Global Const UP_ARROW = 10      ' 10 - Up Arrow
Global Const HOURGLASS = 11     ' 11 - Hourglass
Global Const NO_DROP = 12       ' 12 - No drop

' DragMode
Global Const MANUAL = 0    ' 0 - Manual
Global Const AUTOMATIC = 1 ' 1 - Automatic

' DrawMode
Global Const BLACKNESS = 1      ' 1 - Blackness
Global Const NOT_MERGE_PEN = 2  ' 2 - Not Merge Pen
Global Const MASK_NOT_PEN = 3   ' 3 - Mask Not Pen
Global Const NOT_COPY_PEN = 4   ' 4 - Not Copy Pen
Global Const MASK_PEN_NOT = 5   ' 5 - Mask Pen Not
Global Const INVERT = 6         ' 6 - Invert
Global Const XOR_PEN = 7        ' 7 - Xor Pen
Global Const NOT_MASK_PEN = 8   ' 8 - Not Mask Pen
Global Const MASK_PEN = 9       ' 9 - Mask Pen
Global Const NOT_XOR_PEN = 10   ' 10 - Not Xor Pen
Global Const NOP = 11           ' 11 - Nop
Global Const MERGE_NOT_PEN = 12 ' 12 - Merge Not Pen
Global Const COPY_PEN = 13      ' 13 - Copy Pen
Global Const MERGE_PEN_NOT = 14 ' 14 - Merge Pen Not
Global Const MERGE_PEN = 15     ' 15 - Merge Pen
Global Const WHITENESS = 16     ' 16 - Whiteness

' DrawStyle
Global Const SOLID = 0          ' 0 - Solid
Global Const DASH = 1           ' 1 - Dash
Global Const DOT = 2            ' 2 - Dot
```

```
Global Const DASH_DOT = 3     ' 3 - Dash-Dot
Global Const DASH_DOT_DOT = 4 ' 4 - Dash-Dot-Dot
Global Const INVISIBLE = 5    ' 5 - Invisible
Global Const INSIDE_SOLID = 6 ' 6 - Inside Solid

' FillStyle
' Global Const SOLID = 0          ' 0 - Solid
Global Const TRANSPARENT = 1      ' 1 - Transparent
Global Const HORIZONTAL_LINE = 2  ' 2 - Horizontal Line
Global Const VERTICAL_LINE = 3    ' 3 - Vertical Line
Global Const UPWARD_DIAGONAL = 4  ' 4 - Upward Diagonal
Global Const DOWNWARD_DIAGONAL = 5 ' 5 - Downward Diagonal
Global Const CROSS = 6            ' 6 - Cross
Global Const DIAGONAL_CROSS = 7   ' 7 - Diagonal Cross

' LinkMode (forms and controls)
' Global Const NONE = 0         ' 0 - None
Global Const LINK_SOURCE = 1    ' 1 - Source (forms only)
Global Const LINK_AUTOMATIC = 1 ' 1 - Automatic (controls only)
Global Const LINK_MANUAL = 2    ' 2 - Manual (controls only)
Global Const LINK_NOTIFY = 3    ' 3 - Notify (controls only)

' LinkMode (kept for VB1.0 compatibility, use new constants instead)
Global Const HOT = 1    ' 1 - Hot (controls only)
Global Const SERVER = 1 ' 1 - Server (forms only)
Global Const COLD = 2   ' 2 - Cold (controls only)


' ScaleMode
Global Const USER = 0       ' 0 - User
Global Const TWIPS = 1      ' 1 - Twip
Global Const POINTS = 2     ' 2 - Point
Global Const PIXELS = 3     ' 3 - Pixel
Global Const CHARACTERS = 4  ' 4 - Character
Global Const INCHES = 5     ' 5 - Inch
Global Const MILLIMETERS = 6 ' 6 - Millimeter
Global Const CENTIMETERS = 7 ' 7 - Centimeter

' ScrollBar
' Global Const NONE    = 0 ' 0 - None
Global Const HORIZONTAL = 1 ' 1 - Horizontal
Global Const VERTICAL = 2   ' 2 - Vertical
Global Const BOTH = 3       ' 3 - Both

' Shape
Global Const SHAPE_RECTANGLE = 0
Global Const SHAPE_SQUARE = 1
Global Const SHAPE_OVAL = 2
Global Const SHAPE_CIRCLE = 3
Global Const SHAPE_ROUNDED_RECTANGLE = 4
Global Const SHAPE_ROUNDED_SQUARE = 5

' WindowState
Global Const NORMAL = 0    ' 0 - Normal
Global Const MINIMIZED = 1 ' 1 - Minimized
Global Const MAXIMIZED = 2 ' 2 - Maximized

' Check Value
Global Const UNCHECKED = 0 ' 0 - Unchecked
```

```
Global Const CHECKED = 1   ' 1 - Checked
Global Const GRAYED = 2    ' 2 - Grayed

' Shift parameter masks
Global Const SHIFT_MASK = 1
Global Const CTRL_MASK = 2
Global Const ALT_MASK = 4

' Button parameter masks
Global Const LEFT_BUTTON = 1
Global Const RIGHT_BUTTON = 2
Global Const MIDDLE_BUTTON = 4

' Function Parameters
' MsgBox parameters
Global Const MB_OK = 0              ' OK button only
Global Const MB_OKCANCEL = 1        ' OK and Cancel buttons
Global Const MB_ABORTRETRYIGNORE = 2  ' Abort, Retry, and Ignore buttons
Global Const MB_YESNOCANCEL = 3     ' Yes, No, and Cancel buttons
Global Const MB_YESNO = 4           ' Yes and No buttons
Global Const MB_RETRYCANCEL = 5     ' Retry and Cancel buttons

Global Const MB_ICONSTOP = 16       ' Critical message
Global Const MB_ICONQUESTION = 32   ' Warning query
Global Const MB_ICONEXCLAMATION = 48  ' Warning message
Global Const MB_ICONINFORMATION = 64  ' Information message

Global Const MB_APPLMODAL = 0       ' Application Modal Message Box
Global Const MB_DEFBUTTON1 = 0      ' First button is default
Global Const MB_DEFBUTTON2 = 256    ' Second button is default
Global Const MB_DEFBUTTON3 = 512    ' Third button is default
Global Const MB_SYSTEMMODAL = 4096  'System Modal

' MsgBox return values
Global Const IDOK = 1               ' OK button pressed
Global Const IDCANCEL = 2           ' Cancel button pressed
Global Const IDABORT = 3            ' Abort button pressed
Global Const IDRETRY = 4            ' Retry button pressed
Global Const IDIGNORE = 5           ' Ignore button pressed
Global Const IDYES = 6              ' Yes button pressed
Global Const IDNO = 7               ' No button pressed

' SetAttr, Dir, GetAttr functions
Global Const ATTR_NORMAL = 0
Global Const ATTR_READONLY = 1
Global Const ATTR_HIDDEN = 2
Global Const ATTR_SYSTEM = 4
Global Const ATTR_VOLUME = 8
Global Const ATTR_DIRECTORY = 16
Global Const ATTR_ARCHIVE = 32

'Grid
'ColAlignment,FixedAlignment Properties
Global Const GRID_ALIGNLEFT = 0
Global Const GRID_ALIGNRIGHT = 1
Global Const GRID_ALIGNCENTER = 2

'Fillstyle Property
Global Const GRID_SINGLE = 0
```

```
Global Const GRID_REPEAT = 1


'Data control
'Error event Response arguments
Global Const DATA_ERRCONTINUE = 0
Global Const DATA_ERRDISPLAY = 1

'Editmode property values
Global Const DATA_EDITNONE = 0
Global Const DATA_EDITMODE = 1
Global Const DATA_EDITADD = 2

' Options property values
Global Const DATA_DENYWRITE = &H1
Global Const DATA_DENYREAD = &H2
Global Const DATA_READONLY = &H4
Global Const DATA_APPENDONLY = &H8
Global Const DATA_INCONSISTENT = &H10
Global Const DATA_CONSISTENT = &H20
Global Const DATA_SQLPASSTHROUGH = &H40

'Validate event Action arguments
Global Const DATA_ACTIONCANCEL = 0
Global Const DATA_ACTIONMOVEFIRST = 1
Global Const DATA_ACTIONMOVEPREVIOUS = 2
Global Const DATA_ACTIONMOVENEXT = 3
Global Const DATA_ACTIONMOVELAST = 4
Global Const DATA_ACTIONADDNEW = 5
Global Const DATA_ACTIONUPDATE = 6
Global Const DATA_ACTIONDELETE = 7
Global Const DATA_ACTIONFIND = 8
Global Const DATA_ACTIONBOOKMARK = 9
Global Const DATA_ACTIONCLOSE = 10
Global Const DATA_ACTIONUNLOAD = 11


'OLE Client Control
'Actions
Global Const OLE_CREATE_EMBED = 0
Global Const OLE_CREATE_NEW = 0          'from ole1 control
Global Const OLE_CREATE_LINK = 1
Global Const OLE_CREATE_FROM_FILE = 1    'from ole1 control
Global Const OLE_COPY = 4
Global Const OLE_PASTE = 5
Global Const OLE_UPDATE = 6
Global Const OLE_ACTIVATE = 7
Global Const OLE_CLOSE = 9
Global Const OLE_DELETE = 10
Global Const OLE_SAVE_TO_FILE = 11
Global Const OLE_READ_FROM_FILE = 12
Global Const OLE_INSERT_OBJ_DLG = 14
Global Const OLE_PASTE_SPECIAL_DLG = 15
Global Const OLE_FETCH_VERBS = 17
Global Const OLE_SAVE_TO_OLE1FILE = 18

'OLEType
Global Const OLE_LINKED = 0
Global Const OLE_EMBEDDED = 1
```

```
Global Const OLE_NONE = 3

'OLETypeAllowed
Global Const OLE_EITHER = 2

'UpdateOptions
Global Const OLE_AUTOMATIC = 0
Global Const OLE_FROZEN = 1
Global Const OLE_MANUAL = 2

'AutoActivate modes
'Note that OLE_ACTIVATE_GETFOCUS only applies to objects that
'support "inside-out" activation.  See related Verb notes below.
Global Const OLE_ACTIVATE_MANUAL = 0
Global Const OLE_ACTIVATE_GETFOCUS = 1
Global Const OLE_ACTIVATE_DOUBLECLICK = 2

'SizeModes
Global Const OLE_SIZE_CLIP = 0
Global Const OLE_SIZE_STRETCH = 1
Global Const OLE_SIZE_AUTOSIZE = 2

'DisplayTypes
Global Const OLE_DISPLAY_CONTENT = 0
Global Const OLE_DISPLAY_ICON = 1

'Update Event Constants
Global Const OLE_CHANGED = 0
Global Const OLE_SAVED = 1
Global Const OLE_CLOSED = 2
Global Const OLE_RENAMED = 3

'Special Verb Values
Global Const VERB_PRIMARY = 0
Global Const VERB_SHOW = -1
Global Const VERB_OPEN = -2
Global Const VERB_HIDE = -3
Global Const VERB_INPLACEUIACTIVATE = -4
Global Const VERB_INPLACEACTIVATE = -5
'The last two verbs are for objects that support "inside-out" activation,
'meaning they can be edited in-place, and that they support being left
'in-place-active even when the input focus moves to another control or form.
'These objects actually have 2 levels of being active.  "InPlace Active"
'means that the object is ready for the user to click inside it and start
'working with it.  "In-Place UI-Active" means that, in addition, if the object
'has any other UI associated with it, such as floating palette windows,
'that those windows are visible and ready for use.  Any number of objects
'can be "In-Place Active" at a time, although only one can be
'""InPlace UI-Active".

'You can cause an object to move to either one of states programmatically by
'setting the Verb property to the appropriate verb and setting
'Action=OLE_ACTIVATE.

'Also, if you set AutoActivate = OLE_ACTIVATE_GETFOCUS, the server will
'automatically be put into "InPlace UI-Active" state when the user clicks
'on or tabs into the control.

'VerbFlag Bit Masks
```

```
Global Const VERBFLAG_GRAYED = &H1
Global Const VERBFLAG_DISABLED = &H2
Global Const VERBFLAG_CHECKED = &H8
Global Const VERBFLAG_SEPARATOR = &H800

'MiscFlag Bits - Or these together as desired for special behaviors

'MEMSTORAGE causes the control to use memory to store the object while
'        it is loaded.  This is faster than the default (disk-tempfile),
'        but can consume a lot of memory for objects whose data takes
'        up a lot of space, such as the bitmap for a paint program.
Global Const OLE_MISCFLAG_MEMSTORAGE = &H1

'DISABLEINPLACE overrides the control's default behavior of allowing
'        in-place activation for objects that support it.  If you
'        are having problems activating an object inplace, you can
'        force it to always activate in a separate window by setting this
'        bit
Global Const OLE_MISCFLAG_DISABLEINPLACE = &H2

'Common Dialog Control
'Action Property
Global Const DLG_FILE_OPEN = 1
Global Const DLG_FILE_SAVE = 2
Global Const DLG_COLOR = 3
Global Const DLG_FONT = 4
Global Const DLG_PRINT = 5
Global Const DLG_HELP = 6

'File Open/Save Dialog Flags
Global Const OFN_READONLY = &H1&
Global Const OFN_OVERWRITEPROMPT = &H2&
Global Const OFN_HIDEREADONLY = &H4&
Global Const OFN_NOCHANGEDIR = &H8&
Global Const OFN_SHOWHELP = &H10&
Global Const OFN_NOVALIDATE = &H100&
Global Const OFN_ALLOWMULTISELECT = &H200&
Global Const OFN_EXTENSIONDIFFERENT = &H400&
Global Const OFN_PATHMUSTEXIST = &H800&
Global Const OFN_FILEMUSTEXIST = &H1000&
Global Const OFN_CREATEPROMPT = &H2000&
Global Const OFN_SHAREAWARE = &H4000&
Global Const OFN_NOREADONLYRETURN = &H8000&

'Color Dialog Flags
Global Const CC_RGBINIT = &H1&
Global Const CC_FULLOPEN = &H2&
Global Const CC_PREVENTFULLOPEN = &H4&
Global Const CC_SHOWHELP = &H8&

'Fonts Dialog Flags
Global Const CF_SCREENFONTS = &H1&
Global Const CF_PRINTERFONTS = &H2&
Global Const CF_BOTH = &H3&
Global Const CF_SHOWHELP = &H4&
Global Const CF_INITTOLOGFONTSTRUCT = &H40&
Global Const CF_USESTYLE = &H80&
Global Const CF_EFFECTS = &H100&
Global Const CF_APPLY = &H200&
```

```
Global Const CF_ANSIONLY = &H400&
Global Const CF_NOVECTORFONTS = &H800&
Global Const CF_NOSIMULATIONS = &H1000&
Global Const CF_LIMITSIZE = &H2000&
Global Const CF_FIXEDPITCHONLY = &H4000&
Global Const CF_WYSIWYG = &H8000&                    'must also have CF_SCREENFONTS &
CF_PRINTERFONTS
Global Const CF_FORCEFONTEXIST = &H10000
Global Const CF_SCALABLEONLY = &H20000
Global Const CF_TTONLY = &H40000
Global Const CF_NOFACESEL = &H80000
Global Const CF_NOSTYLESEL = &H100000
Global Const CF_NOSIZESEL = &H200000

'Printer Dialog Flags
Global Const PD_ALLPAGES = &H0&
Global Const PD_SELECTION = &H1&
Global Const PD_PAGENUMS = &H2&
Global Const PD_NOSELECTION = &H4&
Global Const PD_NOPAGENUMS = &H8&
Global Const PD_COLLATE = &H10&
Global Const PD_PRINTTOFILE = &H20&
Global Const PD_PRINTSETUP = &H40&
Global Const PD_NOWARNING = &H80&
Global Const PD_RETURNDC = &H100&
Global Const PD_RETURNIC = &H200&
Global Const PD_RETURNDEFAULT = &H400&
Global Const PD_SHOWHELP = &H800&
Global Const PD_USEDEVMODECOPIES = &H40000
Global Const PD_DISABLEPRINTTOFILE = &H80000
Global Const PD_HIDEPRINTTOFILE = &H100000

'Help Constants
Global Const HELP_CONTEXT = &H1          'Display topic in ulTopic
Global Const HELP_QUIT = &H2             'Terminate help
Global Const HELP_INDEX = &H3            'Display index
Global Const HELP_CONTENTS = &H3
Global Const HELP_HELPONHELP = &H4       'Display help on using help
Global Const HELP_SETINDEX = &H5         'Set the current Index for multi index help
Global Const HELP_SETCONTENTS = &H5
Global Const HELP_CONTEXTPOPUP = &H8
Global Const HELP_FORCEFILE = &H9
Global Const HELP_KEY = &H101            'Display topic for keyword in offabData
Global Const HELP_COMMAND = &H102
Global Const HELP_PARTIALKEY = &H105     'call the search engine in winhelp

'Error Constants
Global Const CDERR_DIALOGFAILURE = -32768

Global Const CDERR_GENERALCODES = &H7FFF
Global Const CDERR_STRUCTSIZE = &H7FFE
Global Const CDERR_INITIALIZATION = &H7FFD
Global Const CDERR_NOTEMPLATE = &H7FFC
Global Const CDERR_NOHINSTANCE = &H7FFB
Global Const CDERR_LOADSTRFAILURE = &H7FFA
Global Const CDERR_FINDRESFAILURE = &H7FF9
Global Const CDERR_LOADRESFAILURE = &H7FF8
Global Const CDERR_LOCKRESFAILURE = &H7FF7
Global Const CDERR_MEMALLOCFAILURE = &H7FF6
```

```
Global Const CDERR_MEMLOCKFAILURE = &H7FF5
Global Const CDERR_NOHOOK = &H7FF4

'Added for CMDIALOG.VBX
Global Const CDERR_CANCEL = &H7FF3
Global Const CDERR_NODLL = &H7FF2
Global Const CDERR_ERRPROC = &H7FF1
Global Const CDERR_ALLOC = &H7FF0
Global Const CDERR_HELP = &H7FEF

Global Const PDERR_PRINTERCODES = &H6FFF
Global Const PDERR_SETUPFAILURE = &H6FFE
Global Const PDERR_PARSEFAILURE = &H6FFD
Global Const PDERR_RETDEFFAILURE = &H6FFC
Global Const PDERR_LOADDRVFAILURE = &H6FFB
Global Const PDERR_GETDEVMODEFAIL = &H6FFA
Global Const PDERR_INITFAILURE = &H6FF9
Global Const PDERR_NODEVICES = &H6FF8
Global Const PDERR_NODEFAULTPRN = &H6FF7
Global Const PDERR_DNDMMISMATCH = &H6FF6
Global Const PDERR_CREATEICFAILURE = &H6FF5
Global Const PDERR_PRINTERNOTFOUND = &H6FF4

Global Const CFERR_CHOOSEFONTCODES = &H5FFF
Global Const CFERR_NOFONTS = &H5FFE

Global Const FNERR_FILENAMECODES = &H4FFF
Global Const FNERR_SUBCLASSFAILURE = &H4FFE
Global Const FNERR_INVALIDFILENAME = &H4FFD
Global Const FNERR_BUFFERTOOSMALL = &H4FFC

Global Const FRERR_FINDREPLACECODES = &H3FFF
Global Const CCERR_CHOOSECOLORCODES = &H2FFF


'3D Controls
'------------------------------------------------------
'Alignment (Check Box)
Global Const SSCB_TEXT_RIGHT = 0       '0 - Text to the right
Global Const SSCB_TEXT_LEFT = 1        '1 - Text to the left

'Alignment (Option Button)
Global Const SSOB_TEXT_RIGHT = 0       '0 - Text to the right
Global Const SSOB_TEXT_LEFT = 1        '1 - Text to the left

'Alignment (Frame)
Global Const SSFR_LEFT_JUSTIFY = 0     '0 - Left justify text
Global Const SSFR_RIGHT_JUSTIFY = 1    '1 - Right justify text
Global Const SSFR_CENTER = 2           '2 - Center text

'Alignment (Panel)
Global Const SSPN_LEFT_TOP = 0         '0 - Text to left and top
Global Const SSPN_LEFT_MIDDLE = 1      '1 - Text to left and middle
Global Const SSPN_LEFT_BOTTOM = 2      '2 - Text to left and bottom
Global Const SSPN_RIGHT_TOP = 3        '3 - Text to right and top
Global Const SSPN_RIGHT_MIDDLE = 4     '4 - Text to right and middle
Global Const SSPN_RIGHT_BOTTOM = 5     '5 - Text to right and bottom
Global Const SSPN_CENTER_TOP = 6       '6 - Text to center and top
Global Const SSPN_CENTER_MIDDLE = 7    '7 - Text to center and middle
```

```
Global Const SSPN_CENTER_BOTTOM = 8     '8 - Text to center and bottom

'Autosize (Command Button)
Global Const SS_AUTOSIZE_NONE = 0       '0 - No Autosizing
Global Const SSPB_AUTOSIZE_PICTOBUT = 1  '0 - Autosize Picture to Button
Global Const SSPB_AUTOSIZE_BUTTOPIC = 2  '0 - Autosize Button to Picture

'Autosize (Ribbon Button)
'Global Const SS_AUTOSIZE_NONE     = 0  '0 - No Autosizing
Global Const SSRI_AUTOSIZE_PICTOBUT = 1  '0 - Autosize Picture to Button
Global Const SSRI_AUTOSIZE_BUTTOPIC = 2  '0 - Autosize Button to Picture

'Autosize (Panel)
'Global Const SS_AUTOSIZE_NONE   = 0   '0 - No Autosizing
Global Const SSPN_AUTOSIZE_WIDTH = 1    '1 - Autosize Panel width to Caption
Global Const SSPN_AUTOSIZE_HEIGHT = 2   '2 - Autosize Panel height to Caption
Global Const SSPN_AUTOSIZE_CHILD = 3    '3 - Autosize Child to Panel

'BevelInner (Panel)
Global Const SS_BEVELINNER_NONE = 0     '0 - No Inner Bevel
Global Const SS_BEVELINNER_INSET = 1    '1 - Inset Inner Bevel
Global Const SS_BEVELINNER_RAISED = 2   '2 - Raised Inner Bevel

'BevelOuter (Panel)
Global Const SS_BEVELOUTER_NONE = 0     '0 - No Outer Bevel
Global Const SS_BEVELOUTER_INSET = 1    '1 - Inset Outer Bevel
Global Const SS_BEVELOUTER_RAISED = 2   '2 - Raised Outer Bevel

'FloodType (Panel)
Global Const SS_FLOODTYPE_NONE = 0      '0 - No flood
Global Const SS_FLOODTYPE_L_TO_R = 1    '1 - Left to light
Global Const SS_FLOODTYPE_R_TO_L = 2    '2 - Right to left
Global Const SS_FLOODTYPE_T_TO_B = 3    '3 - Top to bottom
Global Const SS_FLOODTYPE_B_TO_T = 4    '4 - Bottom to top
Global Const SS_FLOODTYPE_CIRCLE = 5    '5 - Widening circle

'Font3D (Panel, Command Button, Option Button, Check Box, Frame)
Global Const SS_FONT3D_NONE = 0         '0 - No 3-D text
Global Const SS_FONT3D_RAISED_LIGHT = 1  '1 - Raised with light shading
Global Const SS_FONT3D_RAISED_HEAVY = 2  '2 - Raised with heavy shading
Global Const SS_FONT3D_INSET_LIGHT = 3   '3 - Inset with light shading
Global Const SS_FONT3D_INSET_HEAVY = 4   '4 - Inset with heavy shading

'PictureDnChange (Ribbon Button)
Global Const SS_PICDN_NOCHANGE = 0      '0 - Use 'Up'bitmap with no change
Global Const SS_PICDN_DITHER = 1        '1 - Dither 'Up'bitmap
Global Const SS_PICDN_INVERT = 2        '2 - Invert 'Up'bitmap

'ShadowColor (Panel, Frame)
Global Const SS_SHADOW_DARKGREY = 0     '0 - Dark grey shadow
Global Const SS_SHADOW_BLACK = 1        '1 - Black shadow

'ShadowStyle (Frame)
Global Const SS_SHADOW_INSET = 0        '0 - Shadow inset
Global Const SS_SHADOW_RAISED = 1       '1 - Shadow raised
```

```
'-------------------------------------
'Graph Control
'-------------------------------------
'General
Global Const G_NONE = 0
Global Const G_DEFAULT = 0

Global Const G_OFF = 0
Global Const G_ON = 1

Global Const G_MONO = 0
Global Const G_COLOR = 1

'Graph Types
Global Const G_PIE2D = 1
Global Const G_PIE3D = 2
Global Const G_BAR2D = 3
Global Const G_BAR3D = 4
Global Const G_GANTT = 5
Global Const G_LINE = 6
Global Const G_LOGLIN = 7
Global Const G_AREA = 8
Global Const G_SCATTER = 9
Global Const G_POLAR = 10
Global Const G_HLC = 11

'Colors
Global Const G_BLACK = 0
Global Const G_BLUE = 1
Global Const G_GREEN = 2
Global Const G_CYAN = 3
Global Const G_RED = 4
Global Const G_MAGENTA = 5
Global Const G_BROWN = 6
Global Const G_LIGHT_GRAY = 7
Global Const G_DARK_GRAY = 8
Global Const G_LIGHT_BLUE = 9
Global Const G_LIGHT_GREEN = 10
Global Const G_LIGHT_CYAN = 11
Global Const G_LIGHT_RED = 12
Global Const G_LIGHT_MAGENTA = 13
Global Const G_YELLOW = 14
Global Const G_WHITE = 15
Global Const G_AUTOBW = 16

'Patterns
Global Const G_SOLID = 0
Global Const G_HOLLOW = 1
Global Const G_HATCH1 = 2
Global Const G_HATCH2 = 3
Global Const G_HATCH3 = 4
Global Const G_HATCH4 = 5
Global Const G_HATCH5 = 6
Global Const G_HATCH6 = 7
Global Const G_BITMAP1 = 16
Global Const G_BITMAP2 = 17
Global Const G_BITMAP3 = 18
Global Const G_BITMAP4 = 19
Global Const G_BITMAP5 = 20
```

```
Global Const G_BITMAP6 = 21
Global Const G_BITMAP7 = 22
Global Const G_BITMAP8 = 23
Global Const G_BITMAP9 = 24
Global Const G_BITMAP10 = 25
Global Const G_BITMAP11 = 26
Global Const G_BITMAP12 = 27
Global Const G_BITMAP13 = 28
Global Const G_BITMAP14 = 29
Global Const G_BITMAP15 = 30
Global Const G_BITMAP16 = 31

'Symbols
Global Const G_CROSS_PLUS = 0
Global Const G_CROSS_TIMES = 1
Global Const G_TRIANGLE_UP = 2
Global Const G_SOLID_TRIANGLE_UP = 3
Global Const G_TRIANGLE_DOWN = 4
Global Const G_SOLID_TRIANGLE_DOWN = 5
Global Const G_SQUARE = 6
Global Const G_SOLID_SQUARE = 7
Global Const G_DIAMOND = 8
Global Const G_SOLID_DIAMOND = 9

'Line Styles
'Global Const G_SOLID = 0
Global Const G_DASH = 1
Global Const G_DOT = 2
Global Const G_DASHDOT = 3
Global Const G_DASHDOTDOT = 4

'Grids
Global Const G_HORIZONTAL = 1
Global Const G_VERTICAL = 2

'Statistics
Global Const G_MEAN = 1
Global Const G_MIN_MAX = 2
Global Const G_STD_DEV = 4
Global Const G_BEST_FIT = 8

'Data Arrays
Global Const G_GRAPH_DATA = 1
Global Const G_COLOR_DATA = 2
Global Const G_EXTRA_DATA = 3
Global Const G_LABEL_TEXT = 4
Global Const G_LEGEND_TEXT = 5
Global Const G_PATTERN_DATA = 6
Global Const G_SYMBOL_DATA = 7
Global Const G_XPOS_DATA = 8
Global Const G_ALL_DATA = 9

'Draw Mode
Global Const G_NO_ACTION = 0
Global Const G_CLEAR = 1
Global Const G_DRAW = 2
Global Const G_BLIT = 3
Global Const G_COPY = 4
Global Const G_PRINT = 5
```

```
Global Const G_WRITE = 6

'Print Options
Global Const G_BORDER = 2

'Pie Chart Options          '
Global Const G_NO_LINES = 1
Global Const G_COLORED = 2
Global Const G_PERCENTS = 4

'Bar Chart Options          '
'Global Const G_HORIZONTAL = 1
Global Const G_STACKED = 2
Global Const G_PERCENTAGE = 4
Global Const G_Z_CLUSTERED = 6

'Gantt Chart Options        '
Global Const G_SPACED_BARS = 1

'Line/Polar Chart Options   '
Global Const G_SYMBOLS = 1
Global Const G_STICKS = 2
Global Const G_LINES = 4

'Area Chart Options         '
Global Const G_ABSOLUTE = 1
Global Const G_PERCENT = 2

'HLC Chart Options          '
Global Const G_NO_CLOSE = 1
Global Const G_NO_HIGH_LOW = 2

'------------------------------------
'MCI Control (Multimedia)
'------------------------------------
'NOTE:
'Please use the updated Multimedia constants
'in the WINMMSYS.TXT file from the \VB\WINAPI
'subdirectory.

'Mode Property
'Global Const MCI_MODE_NOT_OPEN = 11
'Global Const MCI_MODE_STOP = 12
'Global Const MCI_MODE_PLAY = 13
'Global Const MCI_MODE_RECORD = 14
'Global Const MCI_MODE_SEEK = 15
'Global Const MCI_MODE_PAUSE = 16
'Global Const MCI_MODE_READY = 17

'NotifyValue Property
'Global Const MCI_NOTIFY_SUCCESSFUL = 1
'Global Const MCI_NOTIFY_SUPERSEDED = 2
'Global Const MCI_ABORTED = 4
'Global Const MCI_FAILURE = 8

'Orientation Property
'Global Const MCI_ORIENT_HORZ = 0
'Global Const MCI_ORIENT_VERT = 1
```

```
'RecordMode Porperty
'Global Const MCI_RECORD_INSERT = 0
'Global Const MCI_RECORD_OVERWRITE = 1

'TimeFormat Property
'Global Const MCI_FORMAT_MILLISECONDS = 0
'Global Const MCI_FORMAT_HMS = 1
'Global Const MCI_FORMAT_MSF = 2
'Global Const MCI_FORMAT_FRAMES = 3
'Global Const MCI_FORMAT_SMPTE_24 = 4
'Global Const MCI_FORMAT_SMPTE_25 = 5
'Global Const MCI_FORMAT_SMPTE_30 = 6
'Global Const MCI_FORMAT_SMPTE_30DROP = 7
'Global Const MCI_FORMAT_BYTES = 8
'Global Const MCI_FORMAT_SAMPLES = 9
'Global Const MCI_FORMAT_TMSF = 10
'-----------------------------------
' ERROR CONSTANT DECLARATIONS (MAPI CONTROLS)
'-----------------------------------
Global Const SUCCESS_SUCCESS = 32000
Global Const MAPI_USER_ABORT = 32001
Global Const MAPI_E_FAILURE = 32002
Global Const MAPI_E_LOGIN_FAILURE = 32003
Global Const MAPI_E_DISK_FULL = 32004
Global Const MAPI_E_INSUFFICIENT_MEMORY = 32005
Global Const MAPI_E_ACCESS_DENIED = 32006
Global Const MAPI_E_TOO_MANY_SESSIONS = 32008
Global Const MAPI_E_TOO_MANY_FILES = 32009
Global Const MAPI_E_TOO_MANY_RECIPIENTS = 32010
Global Const MAPI_E_ATTACHMENT_NOT_FOUND = 32011
Global Const MAPI_E_ATTACHMENT_OPEN_FAILURE = 32012
Global Const MAPI_E_ATTACHMENT_WRITE_FAILURE = 32013
Global Const MAPI_E_UNKNOWN_RECIPIENT = 32014
Global Const MAPI_E_BAD_RECIPTYPE = 32015
Global Const MAPI_E_NO_MESSAGES = 32016
Global Const MAPI_E_INVALID_MESSAGE = 32017
Global Const MAPI_E_TEXT_TOO_LARGE = 32018
Global Const MAPI_E_INVALID_SESSION = 32019
Global Const MAPI_E_TYPE_NOT_SUPPORTED = 32020
Global Const MAPI_E_AMBIGUOUS_RECIPIENT = 32021
Global Const MAPI_E_MESSAGE_IN_USE = 32022
Global Const MAPI_E_NETWORK_FAILURE = 32023
Global Const MAPI_E_INVALID_EDITFIELDS = 32024
Global Const MAPI_E_INVALID_RECIPS = 32025
Global Const MAPI_E_NOT_SUPPORTED = 32026

Global Const CONTROL_E_SESSION_EXISTS = 32050
Global Const CONTROL_E_INVALID_BUFFER = 32051
Global Const CONTROL_E_INVALID_READ_BUFFER_ACTION = 32052
Global Const CONTROL_E_NO_SESSION = 32053
Global Const CONTROL_E_INVALID_RECIPIENT = 32054
Global Const CONTROL_E_INVALID_COMPOSE_BUFFER_ACTION = 32055
Global Const CONTROL_E_FAILURE = 32056
Global Const CONTROL_E_NO_RECIPIENTS = 32057
Global Const CONTROL_E_NO_ATTACHMENTS = 32058
```

```
'-----------------------------------------------
' Outline
'-----------------------------------------------
' PictureType
Global Const MSOUTLINE_PICTURE_CLOSED = 0
Global Const MSOUTLINE_PICTURE_OPEN = 1
Global Const MSOUTLINE_PICTURE_LEAF = 2

'Outline Control Error Constants
Global Const MSOUTLINE_BADPICFORMAT = 32000
Global Const MSOUTLINE_BADINDENTATION = 32001
Global Const MSOUTLINE_MEM = 32002
Global Const MSOUTLINE_PARENTNOTEXPANDED = 32003
```

APPENDIX B

WAV FILE FORMAT

Creative ADPCM Wav Type Format

WAVE_FORMAT_CREATIVE_ADPCM, the name for a new WAV format tag, 0x0200 has been registered with Microsoft. The wave format header is *typedef*'ed with

```
typedef struct creative_adpcmwaveformat_tag
    {
    WAVFORMATEX     ewf;
    WORD            vRevision;
    } CREATIVEADPCMWAVEFORMAT;
```

An exploded view of the structure is layed out below

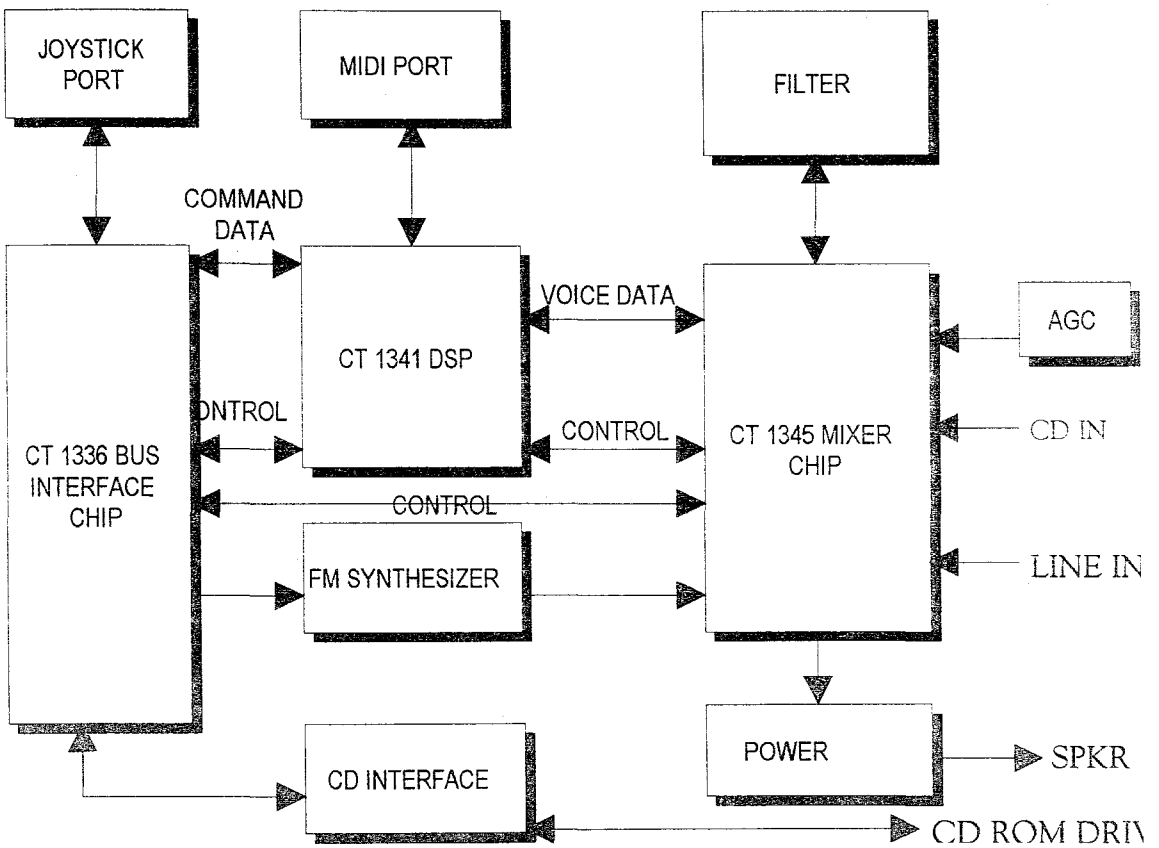
Structure Members	Description
vFormatTag	WAVE_FORMAT_CREATIVE_ADPCM
nChannels	Number of channels. 1 for mono, 2 for stereo.
nSamplesPerSec	Sampling frequency of the data. Shoul be restricted to 8000, 11025, 22050 and 44100 Hz.
nAvgBytesPerSec	Average data rate
nBlockAlign	Block alignment. 1 for both mono and stereo data.
wBitsPerSample	Number of bits per sample. The value is 4.
cbExtraSize	Number of bytes of extra information in the extended WAVE 'fmt' header. The value is 2.
wRevision	Revision of algorithm. The value is 0 for the current definition.

This information should be in the latest release of the "Microsoft Multi Media Standards Update". Where there are differences, this document is the final arbiter.

APPENDIX C

BLOCK DIAGRAMS OF THE SOUND BLASTER AUDIO CARDS

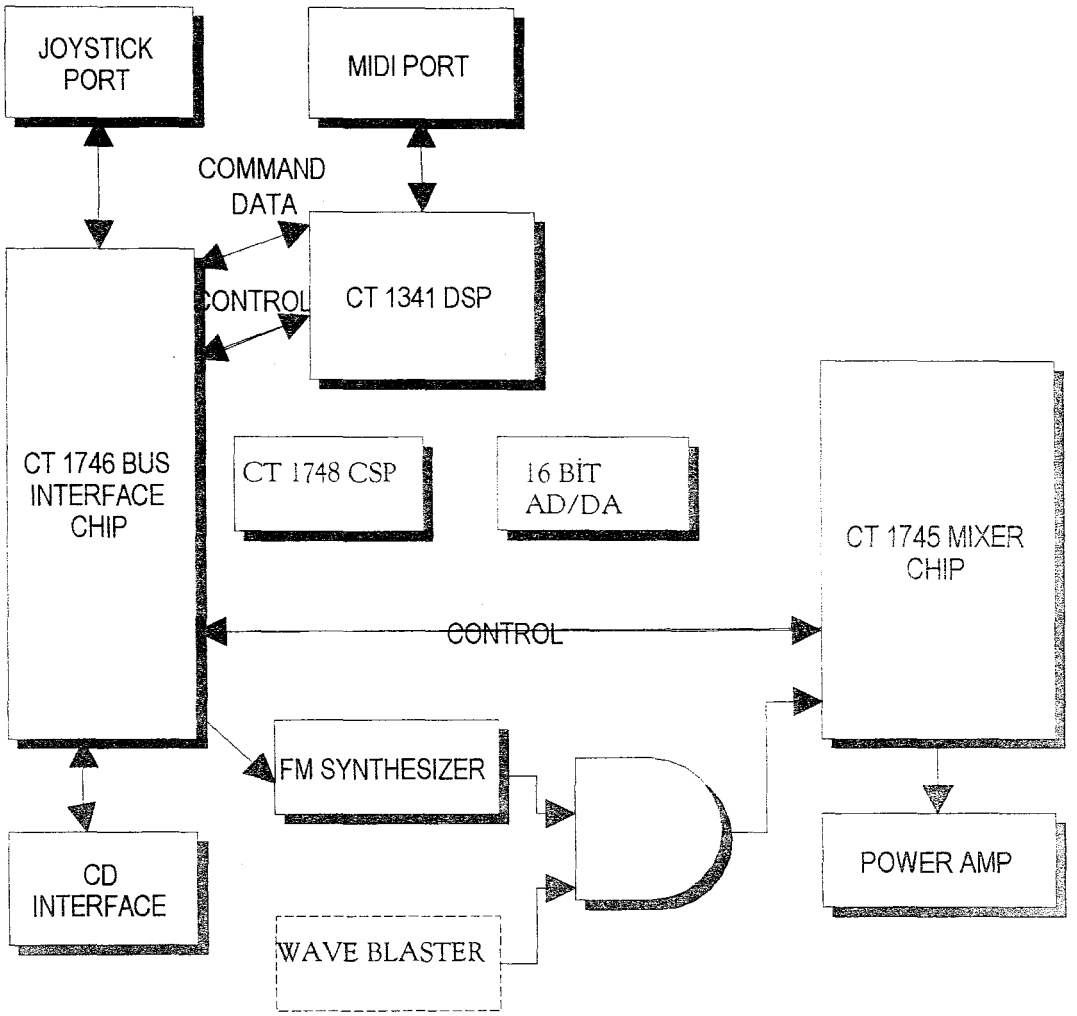
SOUND BLASTER PRO



Sound blaster pro is a 8-bit streo sampling card with stereo mixer chip and CD-ROM interface built in. The DSP has been gradually enhanced to cover wider sampling range.

There are two versions of sound blaster pro. The different is in FM synthesizer chip used. The earlier versions uses a two-operator FM chip known as OPL3, while the later version uses a four,operator FM synthesizer chip known as OPL3. The version with OPL3 is commonly known as Sound Blaster Pro 2.

SOUND BLASTER 16 ASP



Sound Blaster 16 Advanced Signal Processing is a 16-bit stereo sampling card. It also features an enhanced stereo mixer chip. MPU-401 UART mode MIDI interface has also been added. One of the key component is the advanced Signal Processor which can perform high-speed mathematical processing on the data from/to AD/DA.

Sound blaster 16 is the same as Sound Blaster 16 Advanced Signal Processing except the ASP chip is an optional upgrade.

Wave Blaster is an optional expansion doughter board for both Sound Blaster 16 ASP and Sound Blaster 16.

APPENDIX D

FORMAT AND CONTENTS OF DISKETTES

CONTAINING COMPUTER SOFTWARE

Diskettes 1 and 2

These are the installation diskettes, which contains the executable files. The first diskette contains a "Setup.exe" file which creates the required directory on the harddisk and installs required files to appropriate directories. This file needs MS Windows.

After executing "setup.exe" the user must create the directory "wav" under the "td"

c:\td\wav\

The TD software will be written in "c:\td\" after the installation.

Diskette 3

This diskette contains the source codes and forms which are developped in MS Visual Basic Version 3.0

It contains eight files, these are:

Constant.txt
Filemsg.bas
Tinnitus.bas
Tin1.frm
T_Audi.frm
About.frm
Eskihst.frm
Yenihst.frm

REFERENCES

1. Craig, J.C., *Microsoft Visual Basic Workshop*, Microsoft Press, Washington, 1993.

2. Çelikyurt, Cengiz, *"A Novel PC-Based System For Evaluation and Management of Tinnitus Using Masking"*, MS Thesis, Boğaziçi University, İstanbul, 1994.

3. *Developer Kit for Sound Blaster Series*, Hardware Programming Reference, Creative Technology Ltd., second edition, Singapore, 1993.

4. *Developer Kit for Sound Blaster Series*, Library Reference, Creative Technology Ltd., second edition, Singapore, 1993.

5. *Developer Kit for Sound Blaster Series*, Programmer's Guide, Creative Technology Ltd., second edition, Singapore, 1993.

6. Feldman, F., "Masking of tinnitus-Historical remarks," in Feldman, H., (ed.), *Proceedings 3rd International Tinnitus Seminar,* pp. 210-213, Harsch Verlag, Munster, 1987.

7. Feldman, F., "Masking phenomena in tinnitus," in Feldman, H., (ed.), *Proceedings 3rd International Tinnitus Seminar,* pp. 210-213, Harsch Verlag, Munster, 1987.

8. Goodey, D.J., "Drugs in the treatment of tinnitus", *In Tinnitus*, pp: 64-73, Igaku-shoin, Tokyo, 1988.

9. Hazel, J.W.P., Woods, S.M., Cooper, H.R., Stephance, S.D.G., Corcoran, A.L., Coles, R.R.A, Baskill, J.L., Sheldrake, J.B., "A clinical study of tinnitus maskers," *British Journal of Audiology,* Vol. 19, pp. 65-146, 1985.

10. Karagülle, I., Z. Pala, *Basic Bilenler için Visual Basic*, Türkmen Kitabevi, İstanbul, 1996.

11. Lim, J. S., *Two-Dimensional Signal and Image Processing*, Prentice-Hall, Inc., New Jersey, 1990.

12. *Microsoft QuickC, C For Yourself*, Microsoft Corporation, U.S., 1988.

13. Mitchell, C., "Masking tinnitus with pure tones," *Audiology,* Vol. 22, pp. 73-87, 1983.

14. Oppenheim, A.V., A.S. Willsky, *Signals and Systems*, Prentice Hall, Inc., New Jersey, 1983.

15. Pedersen, K.E., U. Rosenhall, M. B. Moller, "Changes in pure tones thresholds in individuals aged 70-71. Results from a longitudinal study", *Audiology* 28: 194-204, 1989.

16. Rosenhall, U., Karlsson, A.N., "Tinnitus in old age", *Scand Audiol* 20:165-171, 1991.

17. Shulman, Abraham, "Tinnitus: Diagnosis / Treatment", pp: 20, 24-26, 50, 51, 67, 237, Lea & Febiger, Philadelphia, 1991.

18. Smith, P.A., Parr, V.MM., Lutman, M.E., Coles, R.R.A., "Comparative study of four noise spectra as potential tinnitus maskers," *British Journal of Audiology,* Vol. 25, pp. 25-34, 1991.

19. *Sound Blaster AWE32 with Advanced WAVEffects Synthesis*, User's Guide, Creative Technology Ltd., Singapore, 1993.

20. Tatar, Halim, *Visual Basic: Standart ve Profesyonel Uyarlama*, Seçkin Yayınevi, Ankara, 1995.

21. Tyler, R., and Conrad-Armes, D., "Masking of tinnitus compared to masking of pure tones," *J Speech Hear Research,* Vol. 27, pp. 106-111, 1984.

22. Tyler, R., and Conrad-Armes, D., "Tinnitus pitch: Acomparison of three measurment methods" *British Journal of Audiology,* Vol. 17, pp. 101-108, 1983.

23. Vernon, J., *"The* common errors of masking for relief of tinnitus," in Feldman H. (ed.), *Proceedings Third International Tinnitus Seminar,* pp. 229-238, Harsch Verlag, Munster, 1987