COVOX INC.

675-D Conger Street, Eugene, OR 97402
503/342-1271 Telex 706017 (Av Alarm UD)

AY3 8913

S O U N D   M A S T E R

USER MANUAL

PRINTED CIRCUIT SPEECH/MUSIC CARD

FOR APPLE II, II+, IIe
AND APPLE DOS 3.3

February, 1985

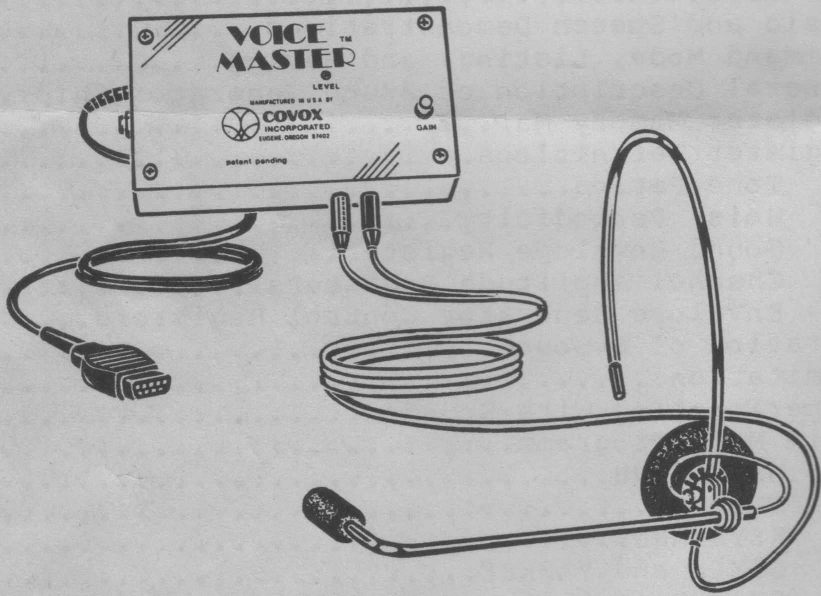
With Demonstration Disk with
Music and Speech in Three Languages.

SOUND MASTER USER MANUAL

PRINTED CIRCUIT MUSIC CARD FOR APPLE II, II+, IIe AND APPLE DOS 3.3

TABLE OF CONTENTS

675 D Conger Street
Eugene, Oregon, 97402



VOICE MASTER

(See page 27 for description)

INTRODUCTION

See important installation instructions following this Introduction.

Most of the popular low cost personal computers contain a fairly sophisticated sound generating integrated circuit that allows the computer to produce a rich variety of music forms, sound effects, and even human-like speech and singing. The Apple II series of computers is not equipped with such a device. However, the Apple has a "mother board" which accommodates a variety of special purpose plug-in boards. The Sound Master described here is a small special purpose board that makes up for the Apple's musical shortcomings. (Apple II, II+, and IIe are registered trade marks of Apple Computer, Inc.)

The Sound Master plugs into one of the slots in the Apple. The demonstration and utility programs on the disk, all written in BASIC, are written for slot number 4. But another slot can be used by changing the addresses of various POKE statements. The Sound Master contains a small power amplifier which can drive the Apple's internal loudspeaker directly, or an external speaker can be powered. Plugging in an external speaker automatically disconnects the internal speaker.

It is possible to use two or more Sound Masters at the same time in order to increase the number of tones and other sounds that the computer can produce simultaneously, and stereophonic effects become possible.

Programming to write music is not difficult. But it can be somewhat tedious. This manual, and the programs that are provided, are designed to ease the task--the services of an experienced programmer are not required.

Digitally recorded speech is also contained on the disk, put there with the aid of a companion Covox product, the Voice Master. Speech material consists of three vocabularies, one in English, one in Chinese, and one in Spanish. The words are the numbers and mathematical symbols which can be used to create number programs such as a talking calculator or clock. Vocabularies are general enough to permit their use in a variety of user written BASIC programs.

Software authors: The speech demonstration serves to indicate that a variety of programs can be written which employ voice response from pre-recorded vocabularies. Furthemore, these programs do not require the use of the

Voice Master--only the Sound Master. With Voice Master, original vocabularies and programs can be produced for private use or for sale. A royalty free license is granted to use pre-recorded vocabularies or other sounds under the conditions that speech output make use of the Covox Sound Master and that you acknowledge the source of the digitally recorded speech in a way that permits others to obtain Voice Masters for themselves. Also of interest to software authors is work in process at Covox aimed at reducing the "raspiness" of the speech as well as reducing the memory needed to store a word. Voice Master also achieves word recognition. Furthermore, a circuit in Voice Master extracts fundamental voice pitch so that music can be created, played, written, and and produced simply by humming into the microphone.

INSTALLATION

Insert the printed circuit card into slot number 4. This is the fourth slot from the left looking from the keyboard. The number 4 may be printed on the mother board near the slot. Be sure to have the phone jack on the Sound Master to the rear, and the attached wire in the front. Volume is at the top of the PC card for thumb control. The internal speaker in the Apple is normally plugged into a small socket near the right front looking from the keyboard. Unplug this and plug it into the matching socket on the Sound Master. The wire attached to the Sound Master is now plugged into the socket from which the original wire was removed. The Sound Master can be left in place.

MUSIC AND SPEECH DEMONSTRATIONS

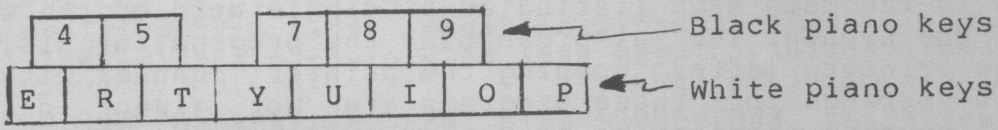
Starting with the computer turned off, insert the disk. Turn on the computer and the monitor. (If the computer already is on, push three keys at the same time for the Apple IIe: the control key, the open Apple key, and the reset. Keys for the Apple II and II+ may differ.) The system automatically boots and displays a MENU:

A) SOUND
B) KEYBOARD
C) PLAY-DIXIE
D) PLAY-YANKEE
E) SPEECH
F) TALKING-CALCULATOR
G) END

Try the keyboard letter C. The program automatically loads from disk without requiring the return key to be depressed (as is the case for all of the main MENU items).

After the tune has played, the MENU reappears. Then try D for another tune. You can play these over and over again.

Next try B for KEYBOARD. This turns the computer into a small organ with the keyboard arrangement as below:



To increase pitch by an octave, strike the up cursor arrow. To decrease by an octave, strike the down cursor arrow. Strike the cursor a second time to get another octave shift. A total of 8 octaves is covered with the octave number being displayed above the keyboard letters and numbers on the screen. The sound is a little like a piano. Later on you will learn how to change the quality of the sound.

Return to the main MENU and try choice A for SOUND. This gives you a utility program (in BASIC) which eases the tedium of experimenting with sound effects, tone qualities, and multiple tones and noises. Discussing this in detail is left until the several registers in the sound generator integrated circuit have been described. However, some demonstrations can be given now. The display for the utility program has a LOAD option at the bottom right of the screen. Strike the L key and you get a second menu that lists 4 nonmusical sound effects. Try them out. Select an option. When the utility display returns, press th P key to hear the sound effect you have selected. These sounds were designed with the aid of the utility SOUND program.

Now go back to the main MENU by striking key M. Try E for a demonstration of speech. This illustrates what you can store on a disk and play back if you also have the Voice Master speech digitizer. The Sound Master provides the sound output means for speech as well as for music. The demonstration program gives the same short sentence in three different languages. Three numbers vocabularies are also stored on the disk for general purpose use as well as for the demonstration program to make the computer into a talking calculator. Go back to the main menu and select F for the TALKING-CALCULATOR. You will first be asked to select a lanugage. After this has been loaded in, you are ready to start computing.

COMMAND MODE, LISTING, AND SAVING

If you are in the keyboard command mode (which you can get by selecting the END option when in the main MENU, or the ESC key otherwise), you can load any of the demonstration programs as LOAD DIXIE, etc. (If unsure about cleaning out old programs, use the command NEW.) Then RUN in the conventional manner. Program names are given in the main MENU listing as a single word or the word after the hyphen. You also can LIST the program, or LIST it to the printer (after opening the printer channel with PR #1). Listings of these programs are not given in this manual because they are so easy to obtain.

Although written in BASIC, the principal function of SPEECH is to load in the machine language program that handles speech reproduction and also to load in the selected numbers vocabulary. CALCULATOR loads the same machine language programs but the BASIC part is longer. Any BASIC program can be loaded from the master disk and saved in a newly formatted disk. Machine language programs can also be saved on a different disk, but in a different way. Perhaps the easiest way to put certain programs and vocabularies on a backup disk is to copy the entire Voice Master disk to a blank disk using standard Apple disk utility COPY A, and then delete the programs that are not dlesired. The disk catalog reveals all programs on the disk (except for DOS 3.3 which is automaticlly copied from one disk to another using tracks 1 and 2). Important speech files are PART 1 and PART 2, and the three numbers vocabularies ENGLISH, SPANISH, and CHINESE. The disk provided with Voice Master has locked files. You cannot accidentally erase a program or file unless you first unlock it.

GENERAL DESCRIPTION OF THE SOUND GENERATOR CHIP

The Sound Master depends primarily on a sophisticated music integrated circuit (General Instrument Corp. type AY-3-8913). There also is a power amplifier integrated circuit which can directly drive a small speaker, a trimmer volume control, and support circuitry which interfaces to one of the slots in the Apple. Sounds can go to the built in Apple speaker or to an external speaker. The Sound Master can be left in place when not in use (unless the slot is needed for something else).

The Sound Master can produce up to three tones at three different frequencies at the same time. The tones are square waves such as are familiar from the "beeps" of a digital alarm clock or a microwave oven. A square wave is characterized by a fundamental pure tone along with a collection of odd numbered harmonics having progressively smaller intensities. For example, if a square wave has a periodicity of 500 per second, then the fundamental component is at 500 Hz, a third harmonic is at 1500 Hz which is 1/9 as powerful as the fundamental, a fifth harmonic at 2500 Hz which is 1/25 as powerful, and so on for components at 3500, 4500, 5500, etc. Hz. In the Sound Master, the relative intensities of the various harmonics are modified by a first order low pass filter in the audio amplifier with a bandwidth of 3000 Hz.

Noise generators are also provided which can be combined with each tone (i.e., linearly added) or employed in place of one or more of the tones. Noise is produced as a sequence of pulses whose widths vary randomly and whose average rate can be specified. All three noise generators always have the same average pulse rate, but random components are different. Thus the sum of two or three noise channels will sound somewhat different from the sound of the noise from a single channel. The noise can be made to sound somewhat like whispered vowel sounds or like fricatives in speech (f, sh, s, th). Or the sound can represent the ocean surf or a jet engine. Combinations of tones and noises can produce a rich variety of effects.

Each of the three tones or noises or tone-noise combinations can separately be controlled in amplitude. The range allows for 16 amplitude levels, which is enough in most cases to give the impression of a continuous variation in amplitude. However, the 16 amplitude levels are not at uniform intervals. Rather, the intervals in magnitude are smaller for low amplitude sounds than they are for larger amplitudes. This is done so as to match the characteristics of the human ear such that the subjective measure of loudness increases by the same amount (3 decibels) with each level increment.

In order to simplify programming for some kinds of sounds that may be of interest, the integrated circuit sound generating chip contains several pre-programmed amplitude fluctuation characteristics, these being referred to as Envelope Generator Control. The envelope is usually made to change slowly compared with tone frequencies. A detailed discussion will be presented later.

There are several slots in the Apple computer. Each can be supplied with a Sound Master. This allows for the creation of sounds and music with more than 3 tones and noises (because each slot can be addressed separately). With two sound boards, each with an amplifier, stereophonic sounds can be produced. Three sounds and noises go to one speaker, with these being different from those applied to the other speaker. It is conceivable that 4 or 5 Sound

Masters could be plugged in at once, giving sounds with 12 or 15 different tone-noise combinations and 4 or 5 different envelope characteristics.

REGISTER MEMORY MAP

The sound generator integrated circuit contains 14 active registers whose contents control all functions of the device. Each register can be reached from the main computer program through memory locations. The procedure to change the contents of a register is to POKE the sound generator register address into the proper computer memory location, and then POKE in the number to be put into the register. This second POKE goes through the next higher memory address. Many of the registers are longer than 8 bits so that it will require two POKE(s) to insert the proper register control value.

For example, let us put a number in the channel B tone generator so as to produce a tone in the middle of the audio range. The two 8 bit register numbers in the sound generator integrated circuit for channel B are numbers 2 and 3. The number to be put into these registers to create a particular tone frequency in this example will be 1 (unity) for coarse period control, and 1Ø2 for fine tuning control (with all numbers being decimal for programming in BASIC). The Voice Master "base" or starting address for slot number 4 is 49344. The instructions to command the correct note frequency are thus

```
POKE 49344,2      (Get Register number 2)
POKE 49345,1Ø2    (Store fine tuning)
POKE 49344,3      (Get Register number 3)
POKE 49345,1      (Store coarse tuning)
```

Note that we use the same memory address in order to get both register addresses 2 and 3, and the same one-higher memory address for transferring both coarse and fine tuning values. Once a number has been put into one of the 14 control registers, the number will remain there until changed by POKE(ing) in another number (or with Reset).

In the preceding example, if you don't known what is in registers 2 and 3, then you can find out by printing their values as

```
POKE 49344,2            (Get Register number 2)
PRINT PEEK (49346)      (Print contents)
POKE 49344,3            (Get Register number 3)
PRINT PEEK (49346)      (Print contents)
```

where it is to be noted that we PEEK at an address for either coarse or fine tuning values that is 2 higher than the base address for the slot.

The several commands cited in these examples can of course be numbered BASIC statements in a program.

Addresses used for POKE depend upon the Apple slot number. These are listed in TABLE NO 1. Some slots are customarily used for standard peripherals such as a disk or printer, and this is noted also. (The last column, RESET, wil be explained later.) For programs written in BASIC, the decimal values for memory locations are employed. If programming is in assembly language, the hexidecimal numbering system may be preferred. The demonstration programs in BASIC are for slot number 4.

TABLE NO. 1

Slot Number	Memory Register ADDRESS Decimal/Hex	Memory Register CONTENTS Decimal/Hex	RESET location Decimal/Hex
1(Printer)	49296/C090	49297/C091	49408/C100
2	49312/C0A0	49313/C0A1	49664/C200
3(80 Col)	49328/C0B0	49329/C0B1	49920/C300
4	49344/C0C0	49345/C0C1	50176/C400
5	49360/C0D0	49361/C0D1	50432/C500
6(Disk)	49376/C0E0	49377/C0E1	50688/C600
7	49392/C0F0	493931C0F1	50944/C700

Note: One of the reasons for the popularity of the Apple II, II+, and IIe is this ability to attach a variety of peripherals to it, using slots with decoded addresses.

REGISTER DEFINITIONS

Tone Period: R0,R1,R2,R3,R4,R5

Tone channels A, B, and C have their periods controlled by 3 separate 12 bit numbers. The 12 bits are defined in terms of an 8 bit number for fine tuning and 4 bits of a second 8 bit number for coarse tuning. Important data are given in TABLE NO. 2. (This table is repeated in the Appendix, also including other registers and the number ranges that can be used.)

TABLE NO. 2

Register	Function	Value Range Decimal	Hex
A: RØ	Fine tune	Ø-255	(ØØ-FF HEX)
R1	Coarse tune	Ø-15	(ØØ-ØF Hex)
B: R2	Fine tune	Ø-255	(ØØ-FF Hex)
R3	Coarse tune	Ø-15	(ØØ-ØF HEX)
C: R4	Fine tune	Ø-255	(ØØ-FF HEX)
R5	Coarse tune	Ø-15	(ØØ-ØF HEX)

The actual tone fundamental frequency is the clock frequency divided by 16 and then divided again by the decimal equivalent of the 12 bit number. The number in the formula can thus vary between Ø and 4Ø95 (decimal). The formula is

f = frequency = (clock frequency)/(16*number)

If, for example, the clock runs at 1.ØØ MHz and the register contains the decimal number 35Ø, then

f = 1,ØØØ,ØØØ/(16*35Ø) = 178.5 Hz

Note: TABLE NO. 3 in the Appendix is a table of coarse and fine tuning numbers as they relate to frequency in Hertz for the standard musical scale. No need to compute values--just look them up in a table.

Special note on the number zero: In the formula, using zero gives a tone frequency of infinity. This is not what actually happens. The effect is the same as using the number 4Ø96 which gives the lowest possible frequency. Thus the number range Ø-4Ø95 as put into the registers must be interpreted as a slightly nonuniform sequence as: 4Ø96,1,2,3,...,4Ø93,4Ø94,4Ø95. This same phenomenon occurs in other registers that specify frequency or period. What appears to happen is that a 5 bit binary number (for example) such as ØØØØØ, with the number limited to 5 bits, becomes interpreted through a form of overflow as a 6 bit number 1ØØØØØ.

Noise Periodicity: R6

Register R6 sets the average rate of the random pulses of all three of the noise generators to the same value. (However, the random components of the three are independently random.) In this case, only a single 8 bit memory location need be used, and only the lower 5 bits of the number, Ø-31 decimal. (The number zero actually acts like the number 32 as discussed above.)

Average frequency = Clock frequency/(16*number)

Sound Enable Register: R7

The tone generators and/or the noise generators that are made active are controlled by Register R7. A 6 bit number is employed which gives up to 64 combinations in the range Ø-63. In particular, bits BØ,B1,B2 turn on and off tone generators A, B, and C, respectively (with 1 turning off, and Ø turning on). Bits B3, B4, and B5 turn on and off the noise generators A, B, and C. If tone B and noise B are both active, then the net waveform is simply the sum of the two. Similarly, the overall net waveform is the sum of (as many as) 6 separate parts. Bits and their significance are indicated below, along with a specific numerical example.

B5	B4	B3	B2	B1	BØ	
C	B	A	C	B	A	
←Tones →			←Noise →			
1	Ø	Ø	1	Ø	1	←--example
C-off	B-on	A-on	C-off	B-on	A-off	

Although the 6 bits act like 6 separate switches turning on or off the 6 different signal components individually, the number put into Register R7 must be a decimal number in BASIC (or Hex in assembly). The number 1ØØ1Ø1 is 41 in decimal. TABLE NO. 4 in the Appendix writes out all 64 different possible combinations, both decimal and hex.

Channel Amplitude Registers: R8,R9,R1Ø

Each of the registers uses only 4 bits and controls the volume (amplitude) of the tone/noise channel. R8 is for channel A, R9 is for channel B, and R1Ø is for C. Normal values control the three tones or tone-noise combinations over a range Ø-15 with level Ø being off and level 15 giving the maximum value. As mentioned previously, sound levels are not uniform but rather follow a logarithmic plan. In terms of power ratios, each step in level differs by 3 decibels.

Envelope Generator Control Registers: R11,R12, R13

The sound generator integrated circuit has several pre-programmed envelope shapes and periodicities. Although their functions can be obtained using registers as have thus far been described, programming is greatly simplified if one of the pre-progammed waves will serve.

One or more of the tones and/or noises can be envelope controlled by setting the amplitude control register for the tone and/or noise (A,B,C) to 16. Changes in amplitude for that sound will thereafter be controlled entirely by the envelope generator. If two or more tone/noise channels have their amplitudes set at 16, then all of them will be controlled the same way because there is only one envelope generator.

Think of the envelope generator as a volume control which can be moved up and down from maximum to zero. The control can move up and down only once to produce a single sound event (although it can last for several seconds), or it can move up and down cyclically to produce a continuing sound with a fluctuation rate that can extend to thousands of fluctuations per second. Or it can move up from zero once (only) and then maintain a constant sound that is not further changed by the envelope generator.

Three registers are used in the envelope generator. One defines the shape of the amplitude fluctuations from the envelope generator, and the other sets the basic period of the shortest cycle of the equivalent volume control. Register number 13 uses 4 bits to determine shape. Although this implies as many as 16 different waveform types, several of these are duplicates and only 8 are unique. Registers 11 and 12 determine the period with a full 16 bit number variation (Ø-65535 decimal). This period, which is the reciprocal of an equivalent frequency, is

Period = 1/f = 256*number/(clock frequency)

As before, using the number zero in both tuning registers results in the longest possible period corresponding to the number 65536. The formula shows that this period is 16 times longer than that for one of the 3 tones if the number is the same. The longest possible period for a 1 MHz clock frequency is 16.8 seconds. The minimum period is found with a number value of unity to be 256 microseconds. This compares with a minimum tone period of 16 microseconds. As a rule, the envelope period will be long enough to include several periods of the tone. But this is not necessary with numerous unusual sounds resulting when periods are comparable.

TABLE NO. 5 in the Appendix defines the several envelope functions that are available.

Note on binary control of shape: A 4 bit register number 13 determines envelope type. (The upper 4 bits of the 8 bit number are not used.) BØ signifies "hold". B1 means "alternate". B2 sets "attack" time. B3 is for "continue". The binary numbers ØØxx and Ø1xx where "x" is a "don't care" bit repeat two of the patterns specified. Patterns shown in the table all have bit 3 at 1 to give a decimal equivalent range of 8-15. For details, see the manufacturer's specifications for the AY-3-8913.

DURATION OF A SOUND

How can a sound be turned on and off? One way is to use envelope generator shapes 9 or 15. All other shapes repeat or sustain sounds. Tones and/or noise with amplitudes set below 16 are not subject to envelope control and will persist unless the appropriate amplitude is reduced to zero. This also applies to a continuing sound being envelope controlled--this control must be removed by reducing the register value for the pertinent tone magnitude to zero. An alternative is to POKE the enable register number 7 with the number 63, which is binary 111111 which turns off all 3 tones and all 3 noises. (This sometimes results in a click.) If all three tones are not on, or if not all of them are to be turned off, then an appropriate value other than 63 would be employed. It will not matter whether or not a channel is enabled if the amplitude for the channel is zero. TABLE NO. 4 in the Appendix can be consulted to determined the number that should be used.

Reset provides another way to turn things off. Each slot has its own reset address, given in the right column of TABLE NO. 1 for memory and register addresses. Turn-off of slot 4 (only) results from POKE 5Ø176. The disadvantage to this procedure is that all registers in use must again be loaded in order to re-establish a sound. With other methods, changes can be limited to fewer register. A "global" reset with CONTROL and RESET keys also works, but not within a program.

The registers in the sound generator integrated circuit remain latched after POKE(ing) in the numbers. They do not change unless and until different numbers are POKE(d) in. Therefore, except for two envelope control functions (9 and 15), turning sounds on and off requires that one or more registers be changed by the user's program. As discussed above, turn off can be achieved with a RESET, by putting 63 (or other suitable number) into the enable register, or by putting zero values in A,B, and/or C amplitude registers. There are some unusual ways to achieve zero amplitude which must be mentioned. One is to increase tone frequencies to such high values that, although physically present, they cannot be heard by a person. Or perhaps change tone frequencies to sub-audible ones.

Creating tones with time gaps between them must come from the user's BASIC program. Changing tones without zero amplitude gaps must also come from the user's program. Tone durations and gap durations, whether applied to all sounds the same or selectively to individual sounds, must come from the user's program. Each change must involve one or more POKE(s). Time intervals between changes can most easily be established with the familiar timing loop:

```
FOR K=1 TO 100: NEXT
```

LIMITATIONS

Without considerable detailed amplitude programming, only square wave tones can be produced. Thus the harmonic content of the basic sound is fixed. With careful control of timing, triangular and certain other waveshapes can be produced. For example, during each half cycle of a tone the amplitude control can be caused to vary. The rapid action required to do this may exceed the capabilities of BASIC. Programming in machine (assembly) language may be required in order to make things go fast enough. Careful adjustments of tone frequency and envelope period can also produce a modified elementary shape. Generally, one should not plan on undertaking these detailed procedures. Rather, limit control of sound quality through use of multiple tones, noise, and various standard envelope functions.

A number of musical instruments can be represented with the aid of the envelope generator. A rapid onset and slow decay sounds like a plucked string. Periodic envelope fluctuations may sound a little like a human singing voice.

The noise waveform is variable only in terms of average pulse rate. It is essentially broad band noise and, although some tonal quality can be perceived, tonality is not as pronounced as, for example, a whispered vowel sound.

EXPERIMENTING WITH SOUNDS

Use the main MENU to select SOUND by striking key A. The screen will present the layout as shown in Figure 1. You can load all registers on this screen. The layout can be understood with a simple example. After VOICE B on the screen, there appears "2)". This means register number 2 which is the fine tuning register for voice or channel B. The blinking cursor near the bottom of the display is labeled CHOICE. Type in the number 2 for register 2 and return. The blinking cursor now goes to the number following the word FINE in the VOICE B line. At the outset, this register contains Ø. Type in the desired number for the fine tuning register (range Ø-255) and return. Then type in 3 and return in order to get to register 3 for coarse tuning of voice B. Then type in a number (range Ø-15) and return. And so on for all of the registers that you want to change from zero, always including the enable register number 7.

FIGURE 1

```
VOICE A     0) FINE -0     1) COARSE -0

VOICE B     2) FINE -0     3) COARSE -0

VOICE C     4) FINE -0     5) COARSE -0

NOISE       6)      -0

ENABLE      7)      -0

AMPLITUDE A  8)     -0

AMPLITUDE B  9)     -0

AMPLITUDE C 10)     -0

ENVELOPE 11) FINE -0    12) COARSE -0

ENVELOPE SHAPE       13) -0

CHOICE:  ▭

P = PLAY  R = RESET  M = MENU  L = LOAD
```

After everything is set, then simply strike the P for PLAY and return. While a sound is being produced, it can be changed by changing a register (register number followed by the number to be installed in the register as before). The change occurs after striking P and return. Push the R key and return to reset all registers to zero. Press L to get pre-recorded samples. Press M to go back to the original main MENU.

The simplest and easiest sound to produce is a single tone. Put the number 62 in the enable register number 7 so as to enable only channel A. Next put 15 in register number 8 in order to have maximum volume for tone A. Now put numbers in registers Ø and 1 for tuning. A frequency of 1ØØØ Hz requires the number N that satisfies the equation

1ØØØ = 1,ØØØ,ØØØ/(16*N)   thus N = 1,ØØØ,ØØØ/(16x1ØØØ)

from which N = 62.5. Use 62 or 63 because only whole numbers are allowed. (Note: We spare you the task of making these calculations with a chromatic scale table in the Appendix, TABLE NO. 3.) Because the number is less than 255, only the fine tuning register RØ will have the number 62. The coarse tuning register R1 contains zero. Now strike P and return in order to hear the tone. While playing, select register Ø again (fine tune for tone A) and then enter some different number, such as 125, in order to hear a tone of about half the frequency of the first tone (after striking P and return).

This simple example illustrates how you can write a tune with one note at a time. Start by POKE(ing) 15 into register number 8 in order to get all tones with a constant maximum amplitude. We must also enable through register 7 for tone A only, using the number 62 (or 6Ø or 58, etc. because enabling channels B and/or C won't matter if their amplitudes are zero anyway). But remember that all of this must be done with memory addressing. The first part of the program becomes

```
1Ø RESTORE
2Ø POKE 49344,8
3Ø POKE 49345,15
4Ø POKE 49344,7
5Ø POKE 49345,62
```

From here on, we simply POKE one number after another into register RØ in order to get a sequence of different frequencies. We can get these numbers from DATA statements. A final data statement number such as - 1 (a negative number) stops the process.

```
60 READ A
70 IF A < 0 GOTO 130
80 POKE 49344,0
90 POKE 49345,A
100 FOR K=0 TO 100: NEXT K
110 GOTO 60
120 DATA 62,75,84,.......,-1
130 END
```

It is not much more difficult to change note durations while the tune is playing. Also, a second harmonizing tone is not too difficult to add.

Let us next give some examples of non-musical sounds which require use of the envelope function generator. Registers 8,9, and 10 (amplitudes of the three channels) are all set at 16. Enable register 7 contains 56, which enables all three tone channels. A peculiar sound like a science fiction space station results with three closely spaced notes. Try, for example, the numbers 190, 191, and 192 for the three fine tuning tone registers. But we are not yet finished because we must provide the right kind of envelope (otherwise we get the default value of zero). Let us have this space station approach and then depart as if it is orbiting about us. Do this by selecting the triangle shape number 14 in register number 13. Use a coarse envelope period of 200 in register number 12. The orbiting speed increases if we reduce the number below 200.

A somewhat similar constant space "whine" results with fine tuning values 95, 97, and 98, enable at 56 as before, and all tone amplitudes at 15 so that the envelope function generator is not employed.

The sound of ocean surf requires noise instead of tones. Use all three noise channels with enable register number 7 containing the number 7 and noise period of 16 in register R6. All amplitudes will be at 16 so that the envelope function generator is employed. Use the same triangular envelope shape 14 as for the space station. But use a shorter envelope period with coarse tuning number 48 in register R12. If you do this using one or two of the noise channels instead of all three, you will also get an ocean sound, but with three it is a bit more convincing.

It is most interesting to listen to the ocean sound when all you do is change the envelope shape. Try changing register R13 from 14 for the triangle to 9 (or zero) for a suddenly rising sound followed by a gradual decay to zero.

The result is a pretty good "explosion". You can convert the explosion to a gunshot sound by speeding up the period by putting 1Ø into R12 in place of 48.

MAIN MENU PROGRAMS

Listings for the several programs in the main MENU can be displayed on the video monitor or listed to the printer given here. Special statements must first be explained. Some Apples have an 8Ø column board installed. In order to turn off this board, the statement CHR$(21) is used (which is ignored if the board is not installed. An alternative with an installed board is simply to request the 4Ø column mode with CHR$(17). Another often used statement is CHR$(4). This readies the disk for loading a BASIC program from the disk. At one point, you will see the statement CHR$(4) without anything following it except for another CHR$(4) followed by the load command. This is a "peculiarity" that was discovered in routine program writing. Occasionally (not often), the disk would not get selected with a single CHR$(4) statement. This problem appears to be corrected with the extra statement.

<u>MAIN MENU</u>. This program is run when you first use the demonstration disk, or whenever M is selected in some subsidiary program. The video display offers a selection of demonstration sounds, A,B,C,D,E, and F. The selection G returns to BASIC (like ESC which won't work when the program is running). Once a selection has been made, the proper program is automatically loaded from disk. How it all works is fairly self-evident from the listing.

<u>SOUND</u> This program is for experimentation as previously discussed. Figure 1 shows how the video display is organized. Also in this SOUND program are the several special (nonmusical) sounds listed in a sub menu that appears upon selecting L. In the listing, this corresponds to line number 58 and the subrouting starts at line 8ØØ and gives a choice of 4 sounds. Parameters will be recognized in DATA statements. The program is written to skip over data elements to the ones desired.

<u>KEYBOARD</u>. This program involves some moderately complex programming in order to produce an easily interpreted video display. The music part itself is fairly straightforward. Statement No. 1Ø defines the base address of slot number 4. Statements 31-35 load registers with numbers for envelope control. Statement 35 shows the envelope period number to be 1Ø (which can easily be changed for experimentation). Statements 11Ø-225 determine the note key that was depressed in a series of IF-THEN statements, where each

statement is followed by corresponding display modifications (blinking cursor, etc.). The subroutine at 3000 computes and plays the notes, with the envelope shape established in the last statement of the subroutine. The reader may discover that, although all three tone channels are supplied with the same number for envelope control and all are caused to produce the same note, only channel A is enabled in statement 34 (number 62 is put in R7). This was done to facilitate experimentation in enabling more than one channel and perhaps produce the second note with a different frequency so as to get a chord organ effect. We will not discuss the machanics of creating the display because this subject is not the principal one here.

DIXIE and YANKEE. These programs are fairly short even though multiple notes are played which go on and off at different times. This should convince the user that such programs are not all that complex. There are, however, a lot of data statements and this implies some tedium in song writing. Saving a melody involves saving the BASIC program, or at least the DATA statements in this program.

SPEECH. This program loads in machine language programs PART 1 and PART 2 and several pre-recorded messages in three different languages. The program must be RUN; automatic RUN occurs only when the program is called from the main MENU. How this very simple program handles the loading and selection of messages can be seen from a listing. How more general vocabularies can be manipulated will be better understood after reading the section on applications.

CALCULATOR. This program loads in PART 1 and PART 2 as above. It then requests the language selection. After the vocabulary is loaded, it can be used as a talking calculator. The BASIC program can be listed for study. It is considerably more complex than SPEECH because keystrokes must be identified with particular vocabulary index numbers. The following section on applications will provide additional clarification.

USING THE NUMBERS VOCABULARY

A vocabulary cannot be utilized until PART 1 and PART 2 have been loaded. PART 2 is the main machine language program which is located "behind" BASIC (i.e., in RAM locations with the same addresses as the Apple BASIC ROM). PART 1 is short and located just below DOS. It's function is to turn off BASIC so that PART 2 behind BASIC can be accessed.

Programs in BASIC that you write for your own use require that both PART 1 and PART 2 as well as one of the vocabularies be present. One way to do this is to load the BASIC program CALCULATOR and select the desired vocabulary. Then get into BASIC with the ESC key. Then be sure to NEW. This leaves you with PART 1 and PART 2 and the selected vocabulary. Then type in or load in your BASIC program and proceed. (Loading in a BASIC program automatically executes NEW.)

Another (more fundamental) way is with keyboard binary load commands:

```
BLOAD PART 1
BLOAD PART 2
CALL 37376
&FIND"ENGLISH"
```

(or &FIND "CHINESE" or "SPANISH"). Then load in the BASIC program. The CALL activates the wedged in commands.

The spoken numbers and symbols are numbered as:

Ø	oh	1Ø	.	point
1	one	11	+	plus
2	two	12	-	minus
3	three	13	*	times
4	four	14	/	divided by
5	five	15	=	equals
6	six	16	$	dollars
7	seven	17	c	cents
8	eight	18	&	and
9	nine	19	E	exponent

Assuming that you have loaded in both the vocabulary and the assembly language programs PART 1 and PART 2 (and exercised NEW if any doubts remain), then type &SPEAK 5 and return. The Sound Master puts forth the spoken "five". Type &SPEAK 12 and get "minus". (If you &SPEAK n where n is the number for a word that was never recorded, you will get a short "beep". The Voice Master speech digitization program can record up to 64 words.) Write the following program and get the entire vocabulary spoken in sequence, with a short delay between each utterrance.

```
10 FOR J=0 TO 19
20 &SPEAK J
30 FOR K=0 TO 50: NEXT K
40 NEXT J
```

You can save your BASIC program in the usual way. In order to create a talking program, just follow the example given above.

The assembly language program that handles voice response works very easily with BASIC because a number of new commands (statements) have been "wedged" into BASIC itself. The "ampersand", &, is the character that is used with Apple BASIC in order to make the wedge work. There are a few other commands that give flexibility to talking programs. One of these is &PAUSE n which is essentially a pre-programmed delay loop with n being the number of millisceonds delay (e.g., 1000 gives one second). The command &SPEED n changes the playback rate much like a slowed or speeded recording. The range is 0-9 with the normal or "default" value being 6. Sometimes a slightly slowed playback speed enhances the sounds. The command &VOLUME n has n in the range 0-15 with the default value 15. This sets the maximum loudness level of the output. For any n value, speech is limited to this level with all lesser levels being the same as if the full range is employed. VOLUME is useful for special effects such as an echo.

This should give you the general idea as to how to include a talking vocabulary in your own programs. If you want different words, or words in a different accent, dialect, different person, or different language, then you will need to have a Voice Master. Vocabulary preparation with this device is really a very simple operation.

A DATA READING PROGRAM

This sample program will read off (whole integer) numbers previously typed into data statements so that you can check for accuracy. As you run your finger along the printed listing of the program, the numbers will be spoken out. If there is an error, then note the error for later correction and continue. You can press Control and C keys if you want to stop or rest at any time. Then type CONT to proceed. There is a limitation to the program in that you cannot change data statements without resetting to the start of the data.

You must have three programs in the machine at the same time, along with the necessary English, Chinese, or Spanish vocabulary. First, you must have PART 1 and PART 2, the machine language programs previously described. Then you must have the program which contains the data statements that you wish to read out. And finally, you must have your own short program which does the reading as in the following example.

Assume that you have DATA statements to be checked, and these are statement numbers 49Ø and 5ØØ. We will add one more DATA statement which inserts a special number that signals when the task has been completed. A convenient special number is a fairly large negative number. We presume that the DATA to be checked ends at a statement number that is less than the beginning number of your special program. Assume the special program starts at statement number 15ØØ and you RUN from this number.

```
490 DATA 0,303,10,30,99,102,200
500 DATA 1,23,64,67,128,23,0,20
501 DATA -999
     .......
     .......
1500 &FIND "ENGLISH"
1510 READ J:IF J<-900 THEN END
1520 PRINT J
1530 A$=STR$(J)
1540 FOR K=2 TO LEN(A$)
1550 B$=MID$(A$,K,1)
1560 B=VAL(B$)
1570 &SPEAK B
1580 NEXT K
1590 &PAUSE 250
1600 GO TO 1510
```

You could elaborate on this program to permit editing during operation when you detect an error. This requires keeping track of the number of data statements that have been checked. After the error is corrected, the program is run again from statement 151Ø with a subroutine inserted so as to quickly scan through all the data statements up to the one following the correction. This is left as a reader exercise.

The talking calculator has many of the same operations as the foregoing data checking program. In particular, numbers must be changed to strings so that individual digits can be removed. Then the individual string elements are changed back to numbers which can be spoken directly with &SPEAK.

TABLE NO. 1: SLOT ADDRESSES

Slot Number	Memory Register ADDRESS Decimal/Hex	Memory Register CONTENTS Decimal/Hex	RESET location Decimal/Hex
1(Printer)	49296/CØ9Ø	49297/CØ91	494Ø8/C1ØØ
2	49312/CØAØ	49313/CØA1	49664/C2ØØ
3(8Ø Col)	49328/CØBØ	49329/CØB1	4992Ø/C3ØØ
4	49344/CØCØ	49345/CØC1	5Ø176/C4ØØ
5	4936Ø/CØDØ	49361/CØD1	5Ø432/C5ØØ
6(Disk)	49376/CØEØ	49377/CØE1	5Ø688/C6ØØ
7	49392/CØFØ	493931CØF1	5Ø944/C7ØØ

TABLE NO. 2: REGISTER NAMES AND NUMERICAL VALUE LIMITS

VOICE A	RØ:	FINE Ø-255 (ØØ-FF)	R1:	COARSE Ø-15 (ØØ-ØF)
VOICE B	R2:	FINE Ø-255 SAME	R3:	COARSE Ø-15 SAME
VOICE C	R4:	FINE Ø-255 SAME	R5:	COARSE Ø-15 SAME
NOISE	R6:	Ø-31 (ØØ-1F)		
ENABLE	R7:	Ø-63 (ØØ-3F)		
CHAN. A AMPL.	R8:	Ø-15 (ØØ-ØF) 16 (FF) FOR ENVELOPE ENABLE		
CHAN. B AMPL.	R9:	Ø-15 (ØØ-ØF) SAME		
CHAN. C AMPL.	R1Ø:	Ø-15 (ØØ-ØF) SAME		
ENV. PERIOD	R11:	FINE Ø-255 (ØØ-FF)	R12:	COARSE Ø-255 (ØØ-FF)
ENV. SHAPE	R13:	8-15 (Ø8-ØF)		

TABLE NO. 3: CHROMATIC SCALE

NOTE	HERTZ	VALUE	COARSE	FINE	HEX C/F
A	27.5	2273	8	225	08 / E1
A#	29.1352	2145	8	97	08 / 61
B	30.8677	2025	7	233	07 / E9
C	32.7032	1911	7	119	07 / 77
C#	34.6478	1804	7	12	07 / 0C
D	36.7081	1703	6	167	06 / A7
D#	38.8909	1607	6	71	06 / 47
E	41.2035	1517	5	237	05 / ED
F	43.6535	1432	5	152	05 / 98
F#	46.2493	1351	5	71	05 / 47
G	48.9994	1276	4	252	04 / FC
G#	51.9131	1204	4	180	04 / B4
A	55	1136	4	112	04 / 70
A#	58.2705	1073	4	49	04 / 31
B	61.7354	1012	3	244	03 / F4
C	65.4064	956	3	188	03 / BC
C#	69.2957	902	3	134	03 / 86
D	73.4162	851	3	83	03 / 53
D#	77.7817	804	3	36	03 / 24
E	82.4069	758	2	246	02 / F6
F	87.3071	716	2	204	02 / CC
F#	92.4986	676	2	164	02 / A4
G	97.9989	638	2	126	02 / 7E
G#	103.826	602	2	90	02 / 5A
A	110	568	2	56	02 / 38
A#	116.541	536	2	24	02 / 18
B	123.471	506	1	250	01 / FA
C	130.813	478	1	222	01 / DE
C#	138.591	451	1	195	01 / C3
D	146.832	426	1	170	01 / AA
D#	155.563	402	1	146	01 / 92
E	164.814	379	1	123	01 / 7B
F	174.614	358	1	102	01 / 66
F#	184.997	338	1	82	01 / 52
G	195.998	319	1	63	01 / 3F
G#	207.652	301	1	45	01 / 2D

TABLE NO. 3: CHROMATIC SCALE (continued)

NOTE	HERTZ	VALUE	COARSE	FINE	HEX C/F
A	220	284	1	28	01/1C
A#	233.082	268	1	12	01/0C
B	246.942	253	0	253	00/FD
C	261.626	239	0	239	00/EF
C#	277.183	225	0	225	00/E1
D	293.665	213	0	213	00/D5
D#	311.127	201	0	201	00/C9
E	329.628	190	0	190	00/BE
F	349.228	179	0	179	00/B3
F#	369.994	169	0	169	00/A9
G	391.995	159	0	159	00/9F
G#	415.305	150	0	150	00/96
A	440	142	0	142	00/8E
A#	466.164	134	0	134	00/86
B	493.883	127	0	127	00/7F
C	523.251	119	0	119	00/77
C#	554.365	113	0	113	00/71
D	587.33	106	0	106	00/6A
D#	622.254	100	0	100	00/64
E	659.255	95	0	95	00/5F
F	698.457	89	0	89	00/59
F#	739.989	84	0	84	00/54
G	783.991	80	0	80	00/50
G#	830.609	75	0	75	00/4B
A	880	71	0	71	00/47
A#	932.328	67	0	67	00/43
B	987.767	63	0	63	00/3F
C	1046.5	60	0	60	00/3C
C#	1108.73	56	0	56	00/38
D	1174.66	53	0	53	00/35
D#	1244.51	50	0	50	00/32
E	1318.51	47	0	47	00/2F
F	1396.91	45	0	45	00/2D
F#	1479.98	42	0	42	00/2A
G	1567.98	40	0	40	00/28
G#	1661.22	38	0	38	00/1E

TABLE NO. 3: CHROMATIC SCALE (continued)

NOTE	HERTZ	VALUE	COARSE	FINE	HEX C/F
A	176Ø	36	Ø	36	Ø Ø /2 4
A#	1864.66	34	Ø	34	Ø Ø /2 2
B	1975.53	32	Ø	32	Ø Ø /2 0
C	2Ø93	3Ø	Ø	3Ø	Ø Ø /1 E
C#	2217.46	28	Ø	28	Ø Ø /1 C
D	2349.32	27	Ø	27	Ø Ø /1 B
D#	2489.Ø2	25	Ø	25	Ø Ø /1 9
E	2637.Ø2	24	Ø	24	Ø Ø /1 8
F	2793.83	22	Ø	22	Ø Ø /1 6
F#	2959.96	21	Ø	21	Ø Ø /1 5
G	3135.96	2Ø	Ø	2Ø	Ø Ø /1 4
G#	3322.44	19	Ø	19	Ø Ø /1 3
A	352Ø	18	Ø	18	Ø Ø /1 2
A#	3729.31	17	Ø	17	Ø Ø /1 1
B	3951.Ø7	16	Ø	16	Ø Ø /1 0
C	4186.Ø1	15	Ø	15	Ø Ø /0 F
C#	4434.92	14	Ø	14	Ø Ø /0 E
D	4698.64	13	Ø	13	Ø Ø /0 D
D#	4978.Ø3	13	Ø	13	Ø Ø /0 D
E	5274.Ø4	12	Ø	12	Ø Ø /0 C
F	5587.65	11	Ø	11	Ø Ø /0 B
F#	5919.91	11	Ø	11	Ø Ø /0 B
G	6271.93	1Ø	Ø	1Ø	Ø Ø /0 A
G#	6644.88	9	Ø	9	Ø Ø /0 9

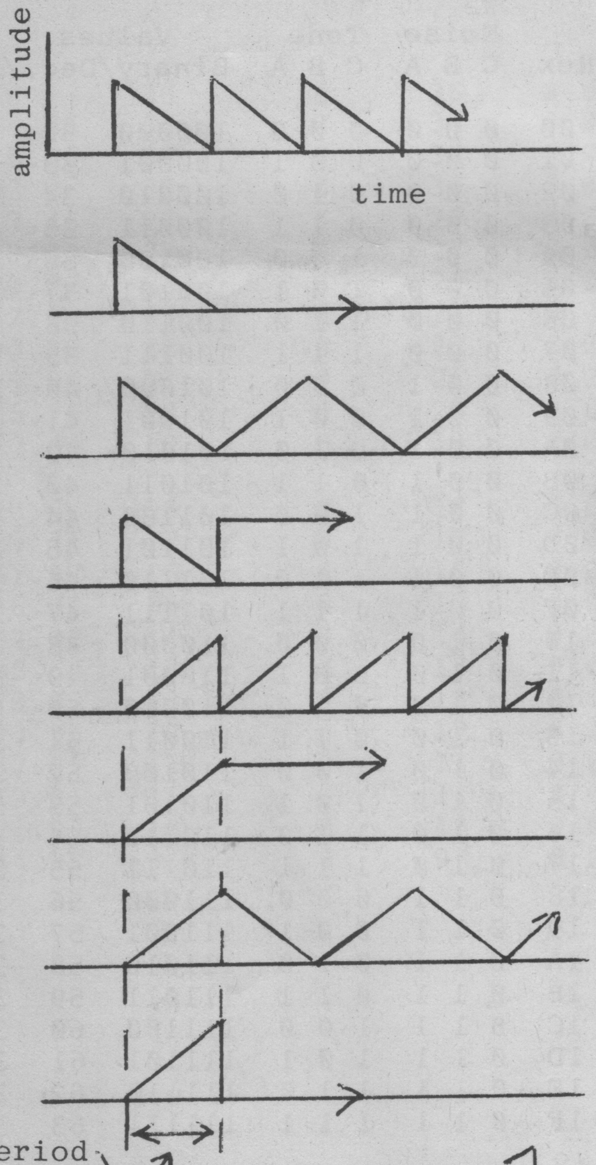
TABLE NO. 4: TONE ENABLE REGISTER VALUES

Off channel is 1. On channel is 0.

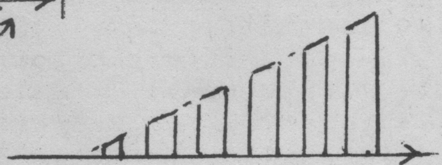
Values Binary	Dec	Hex	Noise C	Noise B	Noise A	Tone C	Tone B	Tone A	Values Binary	Dec.	Hex	Noise C	Noise B	Noise A	Tone C	Tone B	Tone A
000000	00	00	0	0	0	0	0	0	100000	32	20	1	0	0	0	0	0
000001	01	01	0	0	0	0	0	1	100001	33	21	1	0	0	0	0	1
000010	02	02	0	0	0	0	1	0	100010	34	22	1	0	0	0	1	0
000011	03	03	0	0	0	0	1	1	100011	35	23	1	0	1	0	1	1
000100	04	04	0	0	0	1	0	0	100100	36	24	1	0	0	1	0	0
000101	05	05	0	0	0	1	0	1	100101	37	25	1	0	0	1	0	1
000110	06	06	0	0	0	1	1	0	100110	38	26	1	0	0	1	1	0
000111	07	07	0	0	0	1	1	1	100111	39	27	1	0	0	1	1	1
001000	08	08	0	0	1	0	0	0	101000	40	28	1	0	1	0	0	0
001001	09	09	0	0	1	0	0	1	101001	41	29	1	0	1	0	0	1
001010	10	0A	0	0	1	0	1	0	101010	42	2A	1	0	1	0	1	0
001011	11	0B	0	0	1	0	1	1	101011	43	2B	1	0	1	0	1	1
001100	12	0C	0	0	1	1	0	0	101100	44	2C	1	0	1	1	0	0
001101	13	0D	0	0	1	1	0	1	101101	45	2D	1	0	1	1	0	1
001110	14	0E	0	0	1	1	1	0	101110	46	2E	1	0	1	1	1	0
001111	15	0F	0	0	1	1	1	1	101111	47	2F	1	0	1	1	1	1
010000	16	10	0	1	0	0	0	0	110000	48	30	1	1	0	0	0	0
010001	17	11	0	1	0	0	0	1	110001	49	31	1	1	0	0	0	1
010010	18	12	0	1	0	0	1	0	110010	50	32	1	1	0	0	1	0
010011	19	13	0	1	0	0	1	1	110011	51	33	1	1	0	0	1	1
010100	20	14	0	1	0	1	0	0	110100	52	34	1	1	0	1	0	0
010101	21	15	0	1	0	1	0	1	110101	53	35	1	1	0	1	0	1
010110	22	16	0	1	0	1	1	0	110110	54	36	1	1	0	1	1	0
010111	23	17	0	1	0	1	1	1	110111	55	37	1	1	0	1	1	1
011000	24	18	0	1	1	0	0	0	111000	56	38	1	1	1	0	0	0
011001	25	19	0	1	1	0	0	1	111001	57	39	1	1	1	0	0	1
011010	26	1A	0	1	1	0	1	0	111010	58	3A	1	1	1	0	1	0
011011	27	1B	0	1	1	0	1	1	111011	59	3B	1	1	1	0	1	1
011100	28	1C	0	1	1	1	0	0	111100	60	3C	1	1	1	1	0	0
011101	29	1D	0	1	1	1	0	1	111101	61	3D	1	1	1	1	0	1
011110	30	1E	0	1	1	1	1	0	111110	62	3E	1	1	1	1	1	0
011111	31	1F	0	1	1	1	1	1	111111	63	3F	1	1	1	1	1	1

TABLE NO. 5: ENVELOPE SHAPE DETERMINATIØN

Register 13 value Decimal	Hex	Shape
8	Ø8	
9 (orØ,1,2,3)	Ø9	
1Ø	1A	
11	1B	
12	1C	
13	1D	
14	1E	
15 (or 4,5,6,7)	1F	



Basic period



Example: Tone modulated by No. 15

VOICE MASTER APPLICATIONS

The purpose of Sound Master is to output information stored in computer main memory and disk, whereas Voice Master is used to input such information. The demonstration vocabularies described in this manual were produced with Voice Master. Individual vocabularies contain up to 64 words. Any number of vocabularies may be stored on disk. Creating a vocabulary is a very easy process with wedged in BASIC commands. Software authors who create talking programs must use a Voice master in their work, but their customers need only the Sound Master. (Voice Master connects to the paddle port of Apple IIe or via a suitable adapter installed on earlier models.)

The Voice Master has major applications in addition to that of preparing and saving vocabularies. One is word recognition. Another applies to speech and music training and education. A third allows creation and writing of music by humming or whistling into the microphone.

Word recognition begins with storage of up to 32 word templates for the words to be recognized. This is done by simply speaking the word (twice). Template sets may be stored in arbitrary numbers on disk. Recognition proceeds by speaking the word. The 32 templates are searched and a best fit is selected, subject to chosen threshold levels. Recognition does not require use of Sound Master.

A bar-like display with speech or singing input presents information for speech training and analysis. The plot is similar to a real time spectrum. Another program tracks voice fundamental pitch while speaking or singing. Yet another program causes musical notes to scroll by on the screen as the operator hums or whistles and the resulting tune can be edited. The entire score can be saved on disk or written out to a printer. Displays by themselves do not require use of Sound Master. But in a "performance" mode when Sound Master is in place, music is produced while humming, and this music can be in a different key, with different sound qualities, and in the form of multi-note harmony. Music capabilities of Voice Master can make a composer and writer out almost anyone, whether or not the person can read a note of music.

Covox, Inc. has created a family of fresh new techniques based on original researches. Covox users are advised to remain informed on new software and hardware developments from our laboratories.

COVOX INC.
675-D Conger Street
Eugene, OR 97402