COVOX

VOICE MASTER®

USER MANUAL

FOR APPLE II+, IIe, IIc
SOFTWARE VERSION 4.0
(II+ requires 64K and paddle adapter)

SUPPORTS SOUND MASTER (II+ and IIe)

Includes:

SPEECH RECORDING AND PLAYBACK
SPEECH WORD RECOGNITION
APPLICATION EXAMPLES ON DISK
PROGRAM LIST EXAMPLES WITH
VOICE CONTROL OF EXTERNAL SWITCHES

WITH AMPLITUDE EDITOR

675 Conger Street
Eugene, Oregon 97402

First Printing November, 1986
Second Printing August, 1987

CONTENTS

APPENDICES

QUICK REFERENCE FOR CABLE CONNECTIONS

The main captive cable from your Voice Master plugs into the joystick port. (For Apple II+, an optional joy stick adapter is needed.) The headset has two mini stereo-type jacks on the end of one cable. The red one goes to MIKE, the black one to EAR (if used), both located next to each other on the Voice Master unit. That's it!

All sound output normally comes from the internal speaker of the Apple II+/IIe/IIc. The additional cable is for operating the earphone on the headset. For Apple IIc, connect one end of the mini stereo plug to the jack located to the foward left side of the computer. The other end goes to the EAR IN jack of the Voice Master, located opposite the headset input jacks. (An external mini speaker can also be plugged into the IIc external audio port for improved sound quality.) For Apple IIe and II+, a Covox Sound Master board is required. Connect the cord to the jack on the Sound Master, the other end to EAR IN on the Voice Master.

LIMITED WARRANTY STATEMENT

COVOX, Inc. guarantees the VOICE MASTER to be free from defective materials and workmanship for a period of one year from the date of purchase. COVOX, Inc. will replace defective parts and make repairs under this warranty when the defect occurs under normal use, provided the unit is returned to the factory via prepaid transportation. The warranty provides that examination of the returned product must disclose a manufacturing defect to be judged by COVOX, Inc. The warranty does not extend to any product which has been subject to misuse, neglect, accident, improper installation, or where the panel legends or other markings have been removed or defaced, and is given in lieu of any other warranty implied or expressed, and will not cover any consequential damages.

Information in this manual and associated software are provided on an "as is" basis. No warranty, either expressed or implied, is made by COVOX, Inc. pertaining to suitability for any specific application or commercial use. It is the purchasers responsibility to make appropriate evaluations for such purposes. COVOX, Inc. disclaims liability for direct, indirect, or incidental damages arising from the use of this product, including but not being limited to interruption of service, loss of business or potential profits, legal actions, or other consequential damages.

Control of environmental factors by means of voice could expose the user to some risk. Word recognition remains an unreliable technology due to uncontrollable variations in the way that normal speech is produced in an uncertain and noisy acoustic environment. Covox, Inc. specifically disclaims liability as stated in the preceding paragraph when applied to word recognition.

PATENTS AND COPYRIGHTS

The software supplied with VOICE MASTER is copyrighted. It may not be copied, reproduced, translated, or reduced to any readable medium or code for other than personal use without prior written permission of COVOX, Inc.

The hardware/software system comprising the COVOX VOICE MASTER is subject to existing patent applications. Unauthorized duplication for commercial purposes or to otherwise avoid payment of appropriate royalties or license fees will be deemed to be a violation of proprietary rights under patent and trademark laws.

The names COVOX, VOICE MASTER, and VOICE HARP, and the COVOX "logo" are registered trademarks and are the property of COVOX, Inc.

RESTRICTIONS ON SOFTWARE USE

Software may generally not be used in programs which are sold or otherwise distributed in violation of copyright laws. There is one exception. Speech that has been produced with Voice Master software may be put into other programs along with playback software, without royalty charges provided (1) software is not for commercial sale, and (2) the source of the speech must be given on the disk jacket, instruction book, and in the disk program itself in sufficient detail to permit a user to acquire a Voice Master. Those wishing to use recognition software and/or edited playback software in programs for sale are advised to contact Covox, Inc. for licensing information.

INTRODUCTION

If you are new to Voice Master, you may wish to experiment with some of the many demonstration programs contained on the Voice Master disk, such as a talking calculator, blackjack game, and others. If this interests you, then turn to the section on "DEMONSTRATION PROGRAMS" before reading the first parts of this manual (but after finishing this INTRODUCTION). You will be guided from there. The Voice Master disk will auto-load to "MENU" for the demonstration programs--simply put the disk in disk drive number 1 and turn on the computer. Then make selections from "MENU". But if you want to follow the procedure in this manual, you will be asked at times to load in essential Voice Master programs in a way that the auto-load function on the Voice Master disk will not do. In this case, select from "MENU" the "RETURN TO BASIC" option. We chose to organize the manual with demonstration programs given later on so that the manual itself would continue to serve as a reasonably compact programmers' reference guide. We expect that the serious programmer will make backup disks that do not contain all of the demonstration programs (if any of them).

The DOS on the Voice Master disk is version 3.3. However, utilities not required for Voice Master programs have been removed in order to make sufficient room on the single disk to hold important applications examples. Utilities not supplied may be found on the disk that you originally received with your computer.

If your interest is in the music capabilities of Voice Master, a different manual than this one applies. Music programs are not software related to those described in this manual. Software relating to speech on the Voice Master disk is very extensive. In fact, it is so extensive that we were forced to put music software on the reverse side of the disk. It can be loaded directly from the reverse side (with BLOAD), or you can follow instructions on "MENU" from the speech side of the disk.

The Voice Master disk contains essential utility software as well as a number of demonstration programs. We presume that the reader is familiar with the BASIC programming language. But it is not presumed that knowledge of this language is extensive. Thus a more or less detailed discussion of demonstration programs is not presented at the outset. Rather, we want to give essential Voice Master programming information as rapidly and thoroughly as possible in the first part of this manual. The demonstration programs and other less impelling topics can then be covered.

Voice Master has three main functions, speech recording and playback, word recognition, and music writing from voice input. This last topic is covered in a separate manual and will not be considered further here. Speech recording and playback can be had in combination with word recognition so as to implement a two way dialog with the computer. A speech recording can also be modified with forms of editing to improve quality and intelligibility on playback (or to create sounds not like those recorded).

Voice Master may find its greatest use in recording speech for later playback. Voice Master hardware is not required for playback from pre-recorded vocabularies. High quality speech can be realized with various forms of editing.

There are different variations of the Covox speech editor. The one contained on the Voice Master disk is an amplitude editor. A more sophisticated (optional) version called "Speech Construction Set" allows "cut and paste" operations with time slices in the millisecond range.

Audio output capability of the Apple is limited. The internal speaker is capable only of being toggled by a constant voltage such that the driving signal consists of a rectangular wave of constant amplitude. Surprisingly intelligible speech can be produced. With full editing using the "Speech Construction Set", it becomes difficult to believe that the audio system is not high quality. Even with the limited amplitude editing capability provided on the Voice Master disk, where "tricks" are used to fool the ear, good results are obtained.

Speech quality can be further improved if a range of amplitude values is imposed. A low cost plug-in card called "Sound Master" provides for 16 amplitude levels. It also permits a broad range of musical expression to be enjoyed, similar to that available from music "chips" that are standard in certain other low cost personal computers. Note, however, that Sound Master is not applicable to the Apple IIc because no expansion ports are provided.

Recorded speech for later playback retains amplitude information whether or not the Sound Master is present. It is the responsibility of the user to install the correct software. The word recognition function is independent of Sound Master.

Voice Master software utilizes DOS 3.3. There is one playback (only) program that can function with ProDOS. Conversion of this particular program to ProDOS form can be accomplished with the conversion routine on the ProDOS systems disk. An Appendix provides further information.

In preparing a general manual for the Apple II family, we have had to contend with systems variations and models II+, IIe, and IIc, with and without extended memories (for II+ and IIe) and with and without Sound Master. Each variation requires somewhat different Voice Master software. We have tried to explain this profusion of systmes in simple terms.

The foregoing discussion reveals the rationale for the organization of this manual--first speech playback, then speech recording (including attaching the Voice Master and microphone "technique"), then editing (amplitude type), and then word recognition. Finally, demonstration programs are described. Appendices present memory locations and other details.

Note a "bonus": Demonstration programs and/or vocabularies not described in this manual may be included on the Voice Master disk. This extra software will usually be found on the back side of the disk. Use the normal CATALOG command to determine disk contents. Examples: Numbers vocabularies in German and Chinese.

SPEECH PLAYBACK

This section explains how to load essential machine language programs directly, without the auto-load function. (Auto-load requires that you turn on the computer system with the disk installed. You are presented with "MENU" from which a selection can be made.)

But first, we urge that you lock the keyboard to capital letters.

Because Apple II system have several model numbers and configurations, four different programs are provided on the Voice Master disk. All four support functions of recording, playback, and word recognition. (Six more are for playback only as described in an Appendix.) All 4 load as:

```
BLOAD PARTAxx
BLOAD PARTBxx
CALL 35072
```

where 35072 is $8900 (Hex) and where "xx" values are:

```
xx = X for 64K systems without Sound Master
xx = (nothing) for 64K systems with Sound Master
xx = EX for 128K systems without Sound Master
xx = E for 128K systems with Sound Master
```

The Voice Master disk contains several pre-recorded word vocabularies which are used with the various demonstration programs. One of these vocabularies (for a talking calculator) has spoken numbers and symbols. Select this vocabulary with a keyboard loading command as:

```
&FIND"ENGLISH"
```

where it is implied that there may be another vocabulary for the same words, but in a different language. Note the ampersand "&". Voice Master commands have been "wedged" into Applesoft BASIC and all such commands begin with this symbol.

A pre-recorded vocabulary is loaded into the lower 64K memory bank if the version of Voice Master software that you choose to employ is for a 64K system, whether or not your actual system has extended memory. A vocabulary automatically loads to the upper 64K memory bank if the version of Voice Master software allows for extended memory, irregardless of memory size when the original vocabulary was created.

Also remember that a Voice Master command with "&" is meaningless to Applesoft BASIC unless this BASIC has been augmented with Voice Master software.

It is the user's responsibility to install software that does or does not presume that the Sound Master is present. Speech output is routed through the internal speaker for non-Sound Master software versions, whether or not the Sound Master is present. If the software version for Sound Master is installed but no physical Sound Master is plugged into one of the slots (specifically named if not the default slot number 4), then no sound will be produced at all. You can use Sound Master when it is plugged into a slot other than number 4 with the keyboard command

```
&SLOT n
```

where n is the slot number (in the range 1-7). The default value (i.e., that presumed if no &SLOT is specified) is slot number 4. The current slot number can be determined by peeking memory location 35075. If software has been installed which does not use the Sound Master, this location will contain the number 255. The &SLOT command is not applicable to Apple IIc. (The &SLOT command can be a BASIC statement. This suggests the possibility of using two or more Sound Masters with different audio circuits so that speech can be caused to be produced at different locations.)

Next get ready to hear sounds from the computer's built-in speaker or on earphones or on a speaker that is plugged into the Sound Master. (Additional information on the headset is given later in this manual.) On Apple IIc, it is suggested that you use earphones or an external speaker because the one in the computer is very small with only marginal performance for speech.

Now type

```
&SPEAK 5
```

and you will hear the spoken word "five" from the vocabulary called "ENGLISH". Do the same for other numbers and symbols in the vocabulary. There are 17, numbered 0 to 16. If you SPEAK 20, or any other number above 16 (but less than 64), you will hear a tone beep. This indicates that a word for that index number was not recorded. The range of indices is 0-63 and playback can be in any order.

Now type

```
&SPEED 4
&SPEAK 5
```

and you will hear "five" slowed down. The sampling rate during playback has been slowed. The range of &SPEED values is 0-10 and 6 is the default value (which exists in the absence of a specific &SPEED command). The &SPEED index, like all other Voice Master commands, can be computed. This means that a symbol or string with a value specified elsewhere can be used instead of an actual number. This ability to compute is the same as for normal Applesoft commands. A &SAMPLE command controls the sampling rate during recording, and it also has a range of values 0-10. The &SPEED during playback must be the same as the &SAMPLE during recording if the reproduced sound is to be at a normal rate.

Before proceeding, return &SPEED to the normal (default) value by typing &SPEED 6. Then type

```
&VOLUME 5
```

and then &SPEAK 5. The word comes back with lower volume—but only if you have a Sound Master in place and have specified the correct slot number if other than the 4. (Not applicable for Apple IIc.) The volume range is 0-15 with 15 being the maximum value, and also the default value. Return to the default condition by typing &VOLUME 15.

If you next type &RESET, your vocabulary is erased. But the machine language program remains. You can reload a different vocabulary as

```
&FIND"SPANISH"
```

and your words will be the same, but in a different language. (You don't actually have to &RESET because loading in a different vocabulary with the &FIND command does this automatically.) Note: &RESET has more specific significance in recording. It also specifies where in memory the vocabulary is stored. This will be discussed in greater detail in the section on "RECORDING".

Let us next write a simple program that speaks out all of the words in the vocabulary, including some tone beeps. We will have the program load in the vocabulary as well. For now, presume that Voice Master software (parts A and B) has been loaded by keyboard command. We will shortly show how this too can be loaded in with BASIC statements so that a single RUN command can do everything.

```
10 &FIND"ENGLISH"
20 FOR J=0 TO 18
30 &PAUSE 4
40 &SPEAK J
50 NEXT J
60 END
```

The &PAUSE command is essentially a time-wasting FOR-NEXT loop, and in fact can easily be replaced with such a loop. The index number after &PAUSE is the number of one-tenth second delay increments. For example, &PAUSE 10 gives a one second delay.

A word vocabulary is placed in main memory beginning at a particular page number. A page is a block of memory 256 bytes long (with a starting address given by the upper 8 bits of the 16 bit address). There are a total of 256 pages of memory in the lower bank of memory (256*256=65536 bytes) and another 256 pages in the upper bank for Apple IIc and memory augmented IIe. (Memory augmented versions of Apple II+ beyond 64K may not perform properly with Voice Master.)

The command

```
&RESET n
```

defines the location of a vocabulary when the vocabulary is originally created where index n is the starting page number. It can be in the range 16-114 with the 64K memory version, or 16-176 with extended memory. The default value (when &RESET is not specifically given when a vocabulary is produced) is n=64. This puts the starting address at 64*256=16384. The first few hundred bytes contain individual word memory limits and other data. The nominal rate of memory useage is about 1000 bytes for each full second of speech. Short words may require less than 1000 bytes, and long words or phrases may require more.

A "base address" is defined here as the address in memory where vocabulary information begins (16384 for the default case). All parameters and word boundary limits are specified in terms of this base address. Switching a given base address from low to high 64K banks (for n in the range 16-114) is automatic according to the particular Voice Master program that resides in main memory. But other changes as, for example, moving speech from page number 60 to page number 70, are not possible without a special (user written) program that avoids overwriting parts of the vocabulary as memory locations are shifted..

We have now defined the following "wedged-in" Voice Master commands:

&FIND &SPEAK &VOLUME &PAUSE &SPEED &SLOT &RESET

These act like ordinary BASIC commands. But the computer will not recognize them unless the proper Voice Master machine language program resides in the computer's main memory.

And that's really all there is to playback from pre-recorded vocabularies (edited or not edited) except for information on how to load parts A and B from a BASIC program. (There is another playback program which does not contain wedges. This is discussed in an Appendix.)

As stated, you cannot use Voice Master commands in a program unless Voice Master software has first been loaded. You should not attempt to load, save, list, or RUN a program that contains Voice Master commands without this software in memory. Thus, your BASIC program must load in Voice Master software before it encounters any Voice Master commands, that is, after statement number 70 in the following example:

```
...
10 D$=CHR$(4)
...
...
50 PRINT D$"BLOAD PARTA"
60 PRINT D$"BLOAD PARTB"
70 CALL 35072
...
...
100 &FIND"ENGLISH"
...
```

When running a BASIC program, you can stop the program with the CONTROL/C key at any time and change playback characteristics such as &SPEED or &VOLUME with keyboard commands (or equivalent POKE's to memory locations as discussed an Appendix). Then type CONT to continue.

When playback is in progress, you can press the space bar in order to restart playback from the beginning. This can help to evaluate the beginning parts of a recording. It also serves to produce novel stuttering sounds.

BACKUP

The Voice Master disk jacket is not notched, or if it is, the notch is covered. Without a notch, it is not possible to write anything to the disk. It is write protected for the benefit of the user, and not because copying is discouraged. To the contrary, it is suggested that you make at least one copy. You could of course make or open a notch and then record to the Voice Master disk. This won't do much good because there is very little empty space on the disk. Also, you could lose the disk by accident and than be forced to wait for a replacement.

BASIC programs copy easily with LOAD-SAVE sequences (load from the Voice Master disk and save to a formatted disk). Vocabularies can be loaded from a disk with &FIND and saved to another disk with &PUT, and similarly for recognition with &TFIND and &TPUT. (These additional save and load commands are explained in later sections of this manual.) Copying machine language programs is not quite so straightforward but can be done with some third party software. Voice Master programs with wedges are in two parts. The "A" parts are loaded directly from disk. But the "B" parts share memory addresses with read-only memory which requires a separate loading step.

A backup of the entire disk can be made with an Apple utility called "FID". Third party software is also available. After loading, follow instructions for a one or two disk system.

Disk space is limited on the Voice Master disk. You will probably want to make some special disk backup copies containing only a small part of what is on the disk. The easiest way to do this is to delete programs and files from a full backup copy. A useful disk must contain the elements of DOS (not cataloged and not easily deleted from the disk) and both parts of one of the two-part (A,B) Voice Master programs (or perhaps the playback-only program). In order to calibrate, you can use the wedged-in &CALIB command (described later) or the separate "BAR" program. If the playback-only program is the one that you intend to use, then calibration is not a factor.

A catalog of programs on the Voice Master disk can be examined on the video display in the usual manner.

CALIBRATION AND MICROPHONE TECHNIQUE

Getting speech into the computer for recording or word recognition normally depends on proper operation of a voice operated switch, sometimes referred to as "VOX". A command to record should not normally cause recording to start until a reasonably loud signal is measured. And when the speech sample ends, a short period of low amplitude levels indicates that the recording process should end. (An Appendix presents a more detailed explanation of VOX operation.) If speech is in a noisy background, then recording starts as soon as the command to record occurs and does not end until the buffer has been filled (which takes about 8 seconds for recording and 2 seconds for recognition). A filled buffer can return an error signal and require that you re-enter your speech.

In a noisy environment, one should first attempt to adjust gain, voice loudness, and microphone placement in an effort to make the VOX operate properly. If this is not possible, then start talking the moment that the recording (or recognizing) command is given and press any key the moment you stop speaking. Normally, however, this won't be required.

To manually stop the recording or playback process (including recording when inputting speech for word recognition or in order to create a recognition template), press any key (except the space bar during playback). This puts an error code number 251 into memory location 25 in page zero. (Other conditions associated with inputting speech place characteristic numbers in this same location as will later be described.)

If the computer is waiting for input and it is not noisy and you wish to do something without worrying about the computer sensing a sound, put it on "hold". Use Control A for the 64K version (CTL and "A" keys pressed together), or the Open-Apple key for the 128K version. In order to go back to the active mode, press the same key(s) again.

A time-out function exists in programs involving speech input when the VOX is operating and waiting for meaningful input. After a certain length of time, the wait is terminated and the program returns to the pre-input command state. Time-out duration is set in memory location number 31 (page zero). Change time-out with POKE 31,n where n determines the number of approximately half-second increments (10 for 5 seconds, etc., but not more than 255). When a time-out occurs, memory location number 25 (page zero) contains the number 250. The default value for n is 60. (The exact time-out varies with the sampling rate.)

When the computer is waiting for speech input, a question mark (?) appears in the lower right hand corner. This mark is steady in the absence of sounds, but jitters about during speech input. Clicks and other short and/or weak sounds may show a brief flicker, but may not start the recording process. If the system is operating properly, then at the end of a speech sample, the ? should become stable, and a very short time thereafter the program should leave the input state. Pressing any key when the screen display shows ? in the upper right corner puts the number 251 into memory location 25. The particular key that was pressed can be determined in a BASIC program with the statement GET A$.

There is a red monitor light on the Voice Master itself. This should flicker during speech peaks to indicate an adequate speech level. But in the absence of speech, or for low level sounds, it should not glow at all.

Proper operation of the VOX requires that the Voice Master be calibrated. Once this is done, it may not have to be repeated. But it should be checked occasionally in case inadvertent jarring, temperature effects, or aging have changed the effective setting. There are two different ways to calibrate, one with a machine language program called "BAR", and another with a wedged-in command &CALIB. (One of the options on "MENU" is CALIBRATION, which selects the wedged command. "BAR" can be loaded directly as will be described, or it can be selected from the "DEMO" program, which is in turn selected from the main "MENU".) In either case, a suitable microphone is plugged into the Voice Master jack labeled "MIKE" and the Voice Master itself is plugged into the joystick port. Voice Master comes with an electret microphone having two (not three) connecting wires, and a suitable biasing voltage is also applied. An alternative is a low or medium impedance dynamic microphone, provided sound level is high enough. Or sounds can come from a radio or tape deck. (The Voice master microphone is combined with an earphone as a headset. The microphone plug is normally red in color. On some units, this was reversed, with red on the earphone. If in doubt, reverse the plugs. No harm results. The earphone will in fact act like a dynamic microphone, but sound level is too low to be useful in this application.)

We first describe the use of "BAR". This program is independent of Voice Master programs and so it can be loaded directly after power up as

```
BLOAD BAR
CALL 16405
```

Turn up the gain on Voice Master and talk into the microphone. A system of dancing bars should appear. There are 16 of these representing a measure of sound frequency content, plus two more bars on the right side of the display. The furthest to the right measures speech amplitude. Next to this is a bar that indicates fundamental voice pitch. To the right of the amplitude bar is a number that indicates the height of this bar. You can experiment with various sounds. The bar graph system is used in part for word recognition.

Adjust the gain so that the average maximum level is about 16, which is where the amplitude bar changes from asterisks to plus signs. The red indicator on Voice Master should glow at levels in the range of 16 or more. In the absence of speech the level as indicated on the display should be zero. If not, or if more than a soft sound is required in order to make the number rise above zero, then calibration is required. Calibration sets the VOX level. If set above zero, the VOX will always be on. If too far below zero, a large signal may be required in order to record speech and distortion can result.

Unplug the microphone so that input sound level is zero. The microphone jack physically shorts the input to ground. Use a small screwdriver (or the "tool" supplied with Voice Master) in the "CALIBRATE" hole on Voice Master. Adjust for an index of zero just below where "1" appears.

Now replace the microphone plug. Gain should be set for average maximum of 16 for sounds such as "ah" and the level should be 1 or 3 for nasal sounds such as "m". Microphone placement will help to get proper values. Locate the microphone not too far from your nose if nasal sounds need strengthening. If external noise is a problem, talk closer to the microphone or talk louder and reduce gain. Changing the calibration setting to reduce effects of noise is not the proper thing to do.

The second method for calibrating requires that one of the Voice Master programs with wedges be in main memory. Then use the special wedged-in command

```
&CALIB
```

When this command is issued, the question mark in the lower right corner appears as in normal recording. But recording never takes place. Proper calibration has the question mark motionless in the absence of speech. When gain is set for normal flickering of the indicator light on Voice Master during average speech peaks, the ? should remain motionless with no speech input, or at most give only an occasional brief flicker. If it becomes too active, the recording process will begin. This is a rapid method for calibrating which will usually be quite satisfactory. The command can in fact be put into a program as a program statement. There will be a time-out to continue the program with a duration depending on the value placed in memory location 31 as previously described. You can press any key to exit the &CALIB command before time-out occurs.

Another check on proper calibration is to record a word and then play it back to see if the word fills the time space without blanks or noise at the ends, which indicates the VOX operates in the absence of speech. Also, weak word parts should not be eliminated, which would indicate that the VOX is too insensitive to respond to weak but necessary speech sounds.

A direct check on amplitude levels is had with the amplitude "EDITOR" program described later. In a way, this program provides the final and most definitive evaluation of amplitudes. Experimenting with your recording technique with the aid of "EDITOR" is perhaps the best way to get the most from the system.

EARPHONE

No specific information has yet been given on use of the headset provided with Voice Master. It has an earphone as well as a microphone. The two plugs are plugged into "MIKE" (red plug) and "EAR" (black plug) on Voice Master. The microphone boom swings on a hinge at the earpiece. You can bend the boom, but don't twist it. Swing completely around for left or right side placement on the head. The microphone under the foam piece should be pointing inward (towards the mouth) in all positions. If in doubt, peel the foam back a little to show a screw, which is on the microphone side.

There are three ways to get audio output from Apple II systems. The internal speaker, which is toggled with a square wave, is the first. The second is from the audio output of Sound Master. The third, for the Apple IIc only, is from the external audio jack on the side of the keyboard. There is a jumper cable supplied with Voice Master which has miniature phone plugs at both ends. One end can be plugged into the Voice Master jack labeled EAR IN. The other end can go to the Sound Master on Models II+ or IIe, if installed, or to the external audio jack on Model IIc. Then both miniature plugs on the headset supplied with Voice Master can be plugged into the Voice Master. If no Sound Master is installed on Models II+ or IIe, then audio comes only from the internal speaker. A user-made cable can connect the Voice Master to the audio lead that normally goes to the internal speaker. Of course, a separate audio power amplifier, or telephone connection, can be adapted to suit special needs.

RECORDING

With essential Voice Master software installed, have the microphone ready and type

```
&LEARN 5
```

Upon pressing RETURN, speak a word or phrase. But don't stop prematurely if you don't want the recording to stop. You can then &LEARN 27, &LEARN 2, etc., in any order, using an index number in the range 0-63.

At any time, you can check the quality of a recorded word with &SPEAK 5, etc. If not satisfactory, then simply re-&LEARN the designated indexed word. The computer program automatically adjusts the memory to fit the repeated word.

If you make a complete vocabulary, you can check it word by word, or write a short FOR-NEXT loop to speak the words in sequence. You can also record in sequence with a similar loop, using &PAUSE so you can catch your breath between recorded words. You might want to &SPEAK the word immediately after &LEARN(ing) it.

There are only three more wedged-in commands to worry about for use in recording in addition to &LEARN. One saves the vocabulary to disk as:

```
&PUT"filename"
```

which saves to disk number 1, as the default disk. To save to disk drive number 2, then write &PUT"filename,D2". (The same procedure applies with &FIND from a second disk drive.)

There are two more commands that affect the way that words are recorded. One of these is &SAMPLE, which controls the rate at which speech is sampled.

Each word in a vocabulary can be recorded with a different &SAMPLE value, or all can be the same. If a rate other than the default value is desired, then just before each word or group of words to be recorded at the desired rate, type as a keyboard command (or as a BASIC statement in a recording program):

```
&SAMPLE n
```

where the index n is in the range 0-10 with 6 the default value. The values correspond to those used with &SPEED as previously described and as tabulated in an Appendix. A high sampling rate yields somewhat improved speech quality as compared to the default value. But more memory is then required to store the speech. A rate lower than the default value results in more distortion, but the memory that is required can be reduced. A technique that might be tried to reduce distortion while not increasing memory needs is to speak a word rapidly using a somewhat elevated sampling rate, and then reproduce it with a lower &SPEED value than the &SAMPLE value.

Finally there is &RESET as has previously been described. Only one such command is allowed per complete vocabulary. A vocabulary in main memory is deleted if this command is given:

```
&RESET n
```

where n in the range 16-114 (64K version) or 16-176 (128K version) specifies the page number where the vocabulary begins--it is the BASE address previously discussed. The default value, which applies if no &RESET is specified, is &RESET 64. If the Voice Master program is meant for a 128K system, the &RESET value will apply to the page number in the upper bank of 64K. But this same vocabulary is loaded into the lower 64K bank if Voice Master software is for a 64K system.

As described briefly in sections on playback, a recording with the ? showing in the lower right corner can be put on hold with Control A (64K version) or the Open Apple key (128K version). Recording can be terminated with any key, the result being that 251 is put into memory location 25. A time-out due to an excessively long duration input puts 250 into location 25 and the sounds preceding the time-out are not recorded. Recording in a noisy environment will usually cause the recording to start as soon as the command &LEARN is executed, and will continue until the buffer is full. A satisfactory recording can still be made if speech starts upon execution of &LEARN and a terminating key is pressed as soon as the speech word or phrase has been completed.

In the section on playback, it was stated that &SPEAK(ing) a number that was never recorded (with &LEARN) results in a tone beep. In addition, this condition places the number 249 into memory location 25. Other numbers are placed in location 25 as a result of different conditions in recording and recognition. These are listed in the section on recognition.

AMPLITUDE EDITOR

The quality and intelligibility of recorded speech can be improved with the special program called "EDITOR". This program, written in BASIC, also loads in a short machine language routine, "WORD EDIT 64K" OR "WORD EDIT 128K".

With the proper Voice Master program in memory, type

LOAD EDITOR

and RUN. A shorter way is to type RUN EDITOR. There are two ways to get words into memory for editing. Words may be recorded one by one while running "EDITOR", or a previously recorded vocabulary can be loaded from disk. In either case, the final result can be saved back to disk memory. The "EDITOR" program presents a menu from which the appropriate selection can be made.

Effective use of "EDITOR" is enhanced if you understand the nature of the speech coding. Voice Master converts speech to a rectangular wave which is sampled (at the specified &SAMPLE) and placed in memory as a sequence of "1's" and "0's", usually several of each in sequence. Speech is played back by reversing the process. Voice Master also measures speech amplitude. Preceding each 15 bytes of fast samples (at the specified &SAMPLE), a byte is added for amplitude data. (Four of the 8 bits are used, giving 16 levels of amplitude, including zero.) Playback first sets the amplitude value in the Sound Master, assuming the Sound Master version is installed. Then the 15 following bytes, converted to a square wave similar to that originally sampled, are sent to the audio output with the proper amplitude. Amplitude can be changed with every amplitude byte (every 15 bytes of high speed data), or even set to zero. But because the 15 bytes of sampled data remain, the original signal can be recovered (with exceptions to be described).

Of course, if you do not have a Sound Master in place, all amplitudes will be the same--that is, maximum or zero. There is a method for modifying this, however, so as to reduce the intensity of high frequency sounds even when no Sound Master is installed. Presence or absence of Sound Master has no bearing on the nature of the speech initially presented for editing.

Editing with Sound Master: The use of the "EDITOR" will be discussed first for the case when the Sound Master is installed. Then the special methods and techniques which can improve speech without the presence of Sound Master can be explained. The principal one of these special manipulations is ignored when a Sound Master is functioning. Similarly, amplitude adjustments which are effective with Sound Master are ignored when Sound Master is not present. Thus, one vocabulary can perform well in both environments.

The "EDITOR" program shows the amplitude levels throughout the word in convenient graphical form, with cursors to keep track of where you are in the edit process. From the menu for "EDITOR", select number 1 (by pressing the number 1 key) to &LEARN a current word, with a chosen index number for the word. Then record the word and edit it. Or select number 5 to load a speech file, then type in the file name, then proceed to edit specific numbered words.

To edit the speech, select number 3. The speech amplitude data then appears on the screen. The complete speech pattern can be scrolled right or left with right and left arrow keys respectively. Scrolling is necessary if the recorded word or phrase is too long to fit on the screen (40 amplitude samples for 40*15=600 bytes, which is approximately 2/3 of a second). Scrolling can also help edit parts of words rather than complete words because what is heard begins at the left side of the screen. This will not correspond to the beginning of the word if some scrolling has been done.

There is a vertical bar on the screen which is the edit bar. This bar occurs at the amplitude sample to be edited. Move the bar left or right with the "J" and "K" keys respectively.

When the edit bar is over the amplitude sample to be edited, move the cursor mark up or down with corresponding arrows on Apple Models IIe and IIc, or with the "I" key for up and the "M" key for down for the II+ (which does not have up-down arrow keys).

The selected amplitude value is installed by pressing the space bar. The following 15 fast bytes will be played back at this amplitude level, including a level of zero if this value is selected.

Editing is done after selecting number 3 for EDIT A WORD from the menu. Recording a word and other tasks are also done from the main menu. If a word has been edited and you return to the menu by pressing the "Q" key (for quit), then the edited amplitude values are permanently changed in main memory and may not be recovered (except possibly if they originally came from disk memory). However, you can restore the original value at the edit bar (one value at a time) before returning to the main menu by pressing the "R" key.

But an escape is still possible. If you edit and return to the menu, you can always go back and re-edit to original values, provided you remember what these were. Getting back to truly original values will not usually be very important to your editing. Editing has only changed the amplitude bytes preceding each 15 bytes of fast samples. You have not changed these samples themselves.

There remain some additional edit options that will permanently and irrevocably change data in main memory once you return to the menu. Prior to this, they too can be cancelled with the "R" key. These additional edit functions change the 15 bytes of fast data.

First is the "B" key. This is a fast way to zero an amplitude. Whereas amplitudes set to zero as previously described can be recovered, the method with the "B" key zeros the fast bytes in a way that cannot be cancelled once you leave the edit mode.

Another special edit option is the "X" key. This removes every fourth positive square wave half cycle from the 15 bytes. Pressing "X" repeatedly repeats the fourth half cycle removal process until nothing is left. Reverse to the starting point with "R". Changes are not recoverable after leaving the edit mode. A number on the screen indicates how many "X" pressing you have made, but the count does not show a number above 3 even though the act exceeds this number. Some fricative sounds can be improved in quality and naturalness with the "X" key.

The "Z" key makes another change in the 15 fast bytes somewhat akin to a high pass filter. Also recoverable while in the edit mode with "R", the change is permanent after leaving this mode. Some fricatives can be improved with "X" or "Z" or a combination. These edit methods may not be so useful with voiced sounds.

The final special key is "S". Changes with this key are partially recoverable. When the "S" key is pressed, two things happen on the screen. First, the amplitude level is automatically set to 7--about half value. In addition, the asterisk representing the amplitude value is replaced with the letter "S". You can change the amplitude value up or down from 7, but the "S" remains. If software using Sound Master is installed, the only effect of this is to set amplitude at value 7, or whatever else you set it at. But if a non-Sound Master version is installed, sounds are reduced in amplitude by a substantial amount. What happens is that each half square wave in the 15 bytes following an amplitude sample is made much narrower. This reduces the sound energy without changing the fact that the square wave switches between two fixed values. The principal use for "S" is to soften sibilants ("ss" and "sh") when not using a Sound Master.

When you are in the edit mode, press the ESC key and you get a description of the various edit functions. Getting this list does not set amplitude values--"R" still works. Two charts are shown below. One gives the selections available from the main "EDITOR" menu. The other gives the edit commands as displayed with ESC.

CATALOG on the menu displays disk contents. CHANGE DRIVE facilitates use of two disk drives. RETURN TO MAIN MENU goes back to the menu that first appeared when you booted the Voice Master disk. Other selections should be self explanatory.

Editing as described is all fine and good, but rather meaningless unless you can listen to the results of your efforts. When in the edit mode, press the "P" key to hear the entire word being edited, including the effects of the editing already done. The "O" key (letter) plays the word from the left edge of the screen to the edit bar. You can also hear a word by selecting SPEAK A WORD from the menu.

Not much more can be said about the mechanics of editing--gaining practical experience is more valuable. Try recording a word such as "six". Reduce amplitude of the beginning "s" part (use the "S" key") and see if it improves the word. Do the same for the ending "s" sound. Next try reducing amplitudes following the end of the voiced "i" sound so as to enhance the sudden amplitude drop. The word might be a little easier to understand. Fricatives such as "f" and "th" also can be improved by reducing amplitudes and/or with "S" and "Z" keys.

But you can do more. Try changing "six" to "ticks" by putting a zero amplitude gap just before the voiced "i" sound and by shortening the leading "s" sound, but not weakening it. Try making the "six" into "sick" by eliminating the final sibilant "s". Your objective is to gain skill in improving words and changing them as you wish. And you will learn quite a bit about the nature of speech itself.

The edit program does not directly allow beginning and ending parts of words to be deleted so as to reduce memory storage requirements. Putting amplitudes to zero does not also remove this part of the speech. Such a procedure could in many cases shorten the words so that less memory would be required for storage. This manipulation is possible by modifying memory locations for words with suitable PEEK(s) and POKE(s), but the process is not simple (especially for 128K systems). A better procedure is to use the more extensive "Speech Construction Set" (a separate optional software program). With this program, words can be shortened, even during a prolonged sound, voice pitch can be changed, and pitch periods can be repeated to achieve noise reduction. An extremely versatile capability for creating and changing words is provided by the "Speech Construction Set".

Those who want to directly experiment with speech data files can do so with the aid of memory location information in an Appendix.

Editing Without Sound Master: This is really a special case of general editing because the same editing commands remain available except for amplitude itself. That is, changing amplitude values through cursor positioning does not apply. You have at your disposal only the "B", "X", "Z", and "S" keys. The "B" key is important because it provides the only means for forcing amplitude to zero (although repeated "X"'s might approximate this). The "X" and "Z" keys affect sound quality, especially for fricatives. The "S" key is perhaps the most valuable one, especially for creating improved "ss" and "sh" sounds.

If you have a Sound Master, you may wish to edit speech so that it takes advantage of its presence, but at the same time speech without Sound Master retains good quality. the "S" is ignored with Sound master, while the actual amplitude level of the "S" on the display is ignored when Sound Master is not in place (including when the "S" is at the zero level). To get zero amplitudes in both cases, the "B" key must be used. The"X" and "Z" keys can be used, but with some care.

The sophisticated (optional) program, "Speech Construction Set", depends in part on the same amplitude editing procedures discussed here. If you gain skill with the amplitude editor, handling the "Speech Construction Set" will not be difficult.

AMPLITUDE EDITOR

B	BLANK DATA AT CURSOR
I	RAISE AMPLITUDE VALUE
J	MOVE CURSOR LEFT
K	MOVE CURSOR RIGHT
M	LOWER AMPLITUDE VALUE
O	PLAY TO CURSOR
P	PLAY ENTIRE WORD
Q	QUIT TO EDITOR MENU
R	RESTORE AT CURSOR
S	SILENCE A SIBILANT
X	REMOVE EVERY 4TH CYCLE
Z	LOW PASS AT CURSOR
←	SCROLL LEFT
→	SCROLL RIGHT
↓	LOWER AMPLITUDE (M)
↑	RAISE AMPLITUDE (I)

MENU

1	LEARN A WORD
2	SPEAK A WORD
3	EDIT A WORD
4	CHANGE WORD NUMBER
5	LOAD A SPEECH FILE
6	SAVE A SPEECH FILE
7	CATALOG
8	CHANGE DRIVE
9	RETURN TO MAIN MENU

CONCEPTS IN RECOGNITION

If a speech word is reduced to a set of comparatively simple characteristics, and if each characteristic is transformed to a graphical variation of time, then this set of time functions forms a "template" which characterizes the word. If several different words are formed into templates, the result is a catalog which can be used in the study of some unknown word. Recognition is based on the best fit, or match, of the unknown template with one in the catalog. This requires that the unknown be compared with each template in the catalog. If no comparison gives a good match, then it is implied that the unknown word is not in the catalog. If two or more good matches are found, then a decision involves uncertainty and advising the operator of this situation might be warranted.

The forgoing applies to virtually all kinds of pattern recognition such as speech, vision, smells, etc. Differences arise in the nature of the characteristics used to form templates and in the error criteria used to measure the closeness of a match. It may not be necessary to complete a match with every member of the catalog if one or more cues contained in the characteristics can narrow down the choices at an early stage. A process that sequentially narrows choices is sometimes called a tree pattern search. Voice Master recognition does involve a limited form of tree search in that a poor match may be indicated before the process for a given template has been completed with the process then jumping to the next template. Another form of tree search applies when sub-vocabularies of words and sub-catalogs of templates are employed. Voice Master allows for sub-vocabularies.

The dancing pattern of the bar graph provides the basic characteristic used in Voice Master recognition (although additional cues not shown on the bar graph may be used as well). Pattern shapes are measured at (approximately) 20 millisecond intervals and each individual pattern is designated with a set of 8 numbers. The total number of 8-number sets depends on the length of the word. Adjacent patterns are subjected to a running average in order to reduce random variations. Then the set of patterns for the entire word is time normalized with the end result being 12 8-number sets. Templates for each word in the catalog as well as the template for the unknown are processed in the same way. The total number of bytes in each template is 12*8=96 (plus four more for memory location data).

Pattern matching could commence at this time by simply taking differences between corresponding numbers for templates in the catalog and those for the unknown. A closeness score can be computed as the sum of the differences in magnitudes (or root-mean-square magnitudes). Certain weightings might be applied to the patterns according to relative importance of their various parts. The lowest score then indicates the best estimate for the unknown. A large lowest score indicates no good match. Two or more low scores indicate uncertainty. (In order to maintain proper comparative measures, stored templates must be normalized.)

In the Voice Master recognition algorithm, a variation of the matching process called "dynamic time warping" is employed. This procedure accounts for some minor differences in the way a word is said. The cues as functions of time can be moved slightly, as if the template were a rubber sheet. A word such as "hello" will then continue to give a good match even though the last syllable may be stretched out compared to that used in making the catalog template for the word.

The Voice Master allows for up to 32 templates per catalog. These may be broken into 4 sets of 8 templates. Each 8 may in turn be broken into subgroups. A tree-like search results if the first recognition from a restricted set of words then points, or vectors, to a second set of words, and so on. Words in each set can be made very distinctive with an error being unlikely. In this way, two very similar words can be recognized reliably, provided that they occur in different subgroups and that neither subgroup will be addressed by the incorrect word.

There are two error criteria: No match good enough, or two or more good matches giving uncertainty. Both of these error criteria can be changed in a user written program.

RECOGNITION PROGRAMMING

One of the Voice Master programs must be in main memory. In order to make a template for a catalog, type

```
&TRAIN n
```

where n is the index number given to the template, in the range 0-31. Unless your interest is limited to only the existence or non-existence of a particular word, you will want to have a catalog of 2 or more templates. Thus &TRAIN additional words.

Suppose you have &TRAIN(ed) a few words in the range 0-7. Now you present a word to the microphone for recognition. Type

```
&RECOG
```

and all 32 templates are scanned for a best fit. Scanning all templates takes time. The scan can be limited to the first 8 templates with

```
&RECOG 1
```

or to the second group 8-15 with &RECOG 2, and so on to &RECOG 4. You can scan two template groups, the first and third, for example, with &RECOG 1,3 (or in reverse order with &RECOG 3,1).

Note: Template numbers that were never &TRAIN(ed) are quickly passed by in the scanning process. If, for example, only templates 0-7 were &TRAIN(ed), then &RECOG with scanning of all 32 templates would take about the same amount of time as &RECOG 1. Speed-up with partitioning is most effective when templates outside the sub-group of interest have been &TRAIN(ed).

What happens when you &RECOG? The index number of the best match is put into memory location 25 in page zero. If the best match was, for example, for word index number 3, then the decimal number 3 will appear on the screen with PRINT PEEK(25).

What if you get no good match? A different number appears. A table of possibilities follows, including codes for recording and playback as well as those for recognition. Several of the items in the table are also discussed in the section on "CALIBRATION AND MICROPHONE TECHNIQUE".

Loc. 25	Situation
248	Tone beep produced. &RECOG when nothing was &TRAIN(ed). Repeated &TRAIN word too long.
249	Tone beep produced. &SPEAK a word never &LEARN(ed).
250	Time-out. Number of half-second increments in Loc. 3.
251	Any key pressed during &LEARN, &TRAIN, &RECOG, or &SPEAK. Read key: GET A$. Exception: Space bar during playback resets &SPEAK to start of word.
252	Speech memory full (&LEARN only).
253	Speech input buffer full. About 8 seconds for &LEARN. About 2 seconds for &RECOG and &TRAIN.
254	Min. error. No &RECOG because 2 or more words too similar.

255	Max. error. No &RECOG because no word close enough to qualify. Word for recognition longer than any in the template set.

You can erase an entire set of templates, with

```
&BLANK
```

You can blank one particular word with

```
&BLANK n
```

where n is the index number of the word (range 0-31).

You can recover a particular &BLANK(ed) template with

```
&UNBLANK n
```

or all &BLANK(ed) templates with

```
&UNBLANK
```

These various manipulations can all be handled within a program. For example, suppose you produce 16 templates, 0-15, and want the program to consider only 9-13 at first, and then consider all except 11 and 12. A sequence of program steps (with error handling statements) could be as follows:

```
100 FORJ=0TO8:&BLANKJ:NEXTJ
110 &BLANK 14:&BLANK 15
120 &RECOG 1,2
130 &UNBLANK
140 A=PEEK(25):IF A>253 THAN 110
 ...
 ...
200 &BLANK 11:&BLANK 12
210 &RECOG 1,2
220 &UNBLANK
230 A=PEEK(25)
240 IF A=250 THEN 400
 ...
 ...
```

In this example, note that a request to repeat the recognition is made if the MIN/MAX error occurs (error numbers 254 and 255 in Loc. 25 as shown in the table). The second recognition jumps elsewhere if a time-out occurs.

The nature of the number in location 25 can be most useful. A simple comparison might be: Is or is not the word (or other sound) in the catalog? Then you don't care what is in location 25 unless it is the number 254 or 255.

If you pressed a key to put code 251 into location 25, then with a GET A$ you can find out which key was pressed. In this way you have created a means to mix voice and keyboard commands in the same program. With judicious handling of &RECOG and various error and indicating numbers in location 25, forms of artificial intelligence can be demonstrated.

You will no doubt want to make and save a template set and later recover it from disk memory. The commands for saving to disk and loading from disk are, respectively

```
&TPUT"filename"
&TFIND"filename"
```

where disk number 1 or 2 can be specified with, for example, &TPUT"filename,D2". If disk number 1 is the default disk, it need not be specified. However, once you have specified a different drive number, that partiuclar drive remains active until you specifically change it.

The template set always contains 32 templates, even though many may never have been &TRAIN(ed).

If a template does not provide satisfactory recognition, then it can be re-&TRAIN(ed). But first it must be &BLANK(ed). If you do not first &BLANK, then the result will be the average of two &TRAIN(ings). This is not desireable if one attempt to &TRAIN was poor. But it is good practice to &TRAIN each good word twice so as to average out some random errors. Averaging over three or more &TRAIN(ings) may not improve recognition and can have a negative effect by muting some of the more important characteristics of a word template. But for some words, especially those without fast changing parts, multiple &TRAIN(ings) can help.

If you &TRAIN a word more than once, and you hear a tone beep, then you have entered a word that differs in duration from the initial &TRAIN(ed) word by 50% or more. This indicates something is abnormal, the number 248 is placed in location 25 as an error condition, and the word just entered is not averaged. (If you are writing an original program, you might want to prompt the user to re-&TRAIN, or &BLANK and then re-&TRAIN.) If you re-&TRAIN and continue to get beeps, perhaps your original word is at fault and you should start over again.

Error Criteria, Thresholds, and Hints: Two kinds of errors that prevent recognition were have been discussed. One error results when two or more words are too similar (254 in Loc. 25). The second is when no word in the template set is close enough to give a reasonably convincing match (255 in Loc. 25). In the process of dynamic time warp template matching, differences are accumulated between the unknown template resulting from &RECOG and each and every &TRAIN(ed) template being scanned. The result is a set of numbers, equal to the size of the stored template set, with values ranging from a minimum (closest match) to a maximum (poorest match). If the overall minimum score is not small, then no good match has been found. This not-small score becomes a maximum number criterion. On the other hand, if two templates show nearly the same minimum scores, then one cannot with confidence state which is the best match (allowing for noise and other uncertainties).

For most practical applications, a single number error criterion can combine both kinds of errors. The criterion is established with a wedged-in command

```
&ACCEPT n
```

where n is in the range 0-4 with 0 the most lax and 4 the tightest. The default value is 2. Values for n and associated minimum and maximum numerical differences are:

n	Min	Max	
0	0	200	
1	5	150	
2	10	125	(default)
3	15	100	
4	20	100	

When a recognition is made, the best matched template word, and the accumulated error score, are to be found in memory locations in part of the memory used by the Voice Master machine language code. The second best match and score are also placed in memory. These locations are:

35070	Index number of closest match.
35079,35080	Error score for above (low-high bytes).
35081	Index number of second closest match.
35082,35083	Error score for above (low-high bytes).

The error criteria themselves are stored in memory as:

35084,35085	Minimum value (low-high bytes).
35086,35087	Maximum value (low-high bytes).

By PEEK(ing) the second best word and score, you can detemine which word is confusing the recognition algorithm. That way, you can determine whether you need to pick another word that won't be confused, or define a new sub group.

The experienced programmer may wish to separately specify minimum and maximum error values rather then the values established by the single &ACCEPT n command. Simple POKE(s) can serve to make changes.

Two other parameters are worth describing here. The general recording procedure establishes the minimum word length that will be presumed to represent a valid word. This is determined by the number of contiguous amplitude samples with non-zero values. The ruling number is in location 35088 (nominal value 12). Another parameter determines how long after the end of a word the computer must wait in order to decide when the word has in fact ended. This involves a count of contiguous amplitudes having zero values. The parameter is in location 35089 (nominal value 12). Additional data on memory locations is in an Appendix.

Template Making: A discussion on techniques for making good templates and achieving good recognition scores is warranted. The machine is not as good a recognizer as is an attentive human listener. Don't ask the machine to choose between close alternatives if you would not expect a human to do well at this task. Think of how often you must ask to have numbers repeated over the telephone. Can you expect a machine to do better? The machine does have some advantages, however. It never tires. It never complains. It maintains a constant set of rules and procedures. It is constantly attentive.

Unlike a human, the machine operates with a strictly limited set of rules. One of these determines the starting and ending points of a word. Unlike the human, a simple machine does not slide a sound back and forth in time to align it with a comparison--at least not much sliding is allowed. Thus in making templates or recognizing, try to have your words start with certainty from a low noise background and end with equal certainty. Be aware of extraneous noises just before and just after speaking, such as lip smacks, tongue and teech clicks, and breath noises. These sounds are acoustically similar to plosive bursts and/or fricatives and could be mistaken for such. Noises at the start and/or end of a word can be especially troublesome, in part because they misrepresent just when the word is supposed to start and end. (A plosive is a brief burst of sound as in the letters "t", "k", and "p".)

Voice Master contains an automatic volume control mechanism. But this is not as versatile as the equivalent in the human ear. Attempt to make your words at constant level and at an adequate level to get well above any background noise. If nasals are too weak by comparison, then perhaps speak with the microphone closer to the nose.

Attempt to always say your words the same way and in a natural manner. If natural, it is less likely that there will be large differences from one word to the next.

&TRAIN words in the same manner and in the same environment as you expect to confront when attempting actual &RECOG. It is natural for a person to change the way speech is produced to fit the environmental situation.

A template has some random perturbations superimposed on it. Making a template that is the average of several such templates tends to smooth out these perturbations. But averaging too many tends to blur the distintive features, especially fast changing ones. The final average template will be compared to a single template from &RECOG which is not averaged or smoothed. If no fast changing cue is left in the average &TRAIN(ed) template, then the relevant cue will not help in recognition. Thus limit the number of repeated &TRAIN(s) to perhaps 2. (Some words can benefit with more avaraging than others.)

Be aware of how you release final plosives like "t" and "p". It is often optional in ordinary speech to release such a plosive or not to release it. Consider, for example, the final "t" in the word "eight". You may not even pronounce the "t", or you may replace it with a weak "tah". Or you might put the "t" close to the end of the vowel part of the word, or put it some distance away. Whatever speech characteristic you do employ, maintain consistency between the template you made and your pronunciation for recognition.

Problems can become especially severe when you try to differentiate between words such as "ache" and "ate" (or "eight") and "ape". Use of such similar sounding words in a single vocabulary will give trouble to a human listener as well as to a machine. In general, try to avoid rhyming words with final plosives; these are similar in both vowel and ending parts. (A possible countermeasure is to purposely emphasize the final plosive on one word and not on the other.)

Multi-syllable and acoustically different words give the best results. Consider the telephone operator who pronounces the number "five" in a rather special way so as to differentiate it from "nine", or the pilot who says "zero" instead of "oh", and "niner" instead of "nine". Do not attempt to recognize the letters "c", "d", "e", "b", when pronounced as such because they sound quite alike. An Appendix to this report gives the international phonetic alphabet and also numbers as spoken by pilots and telephone operators.

Generally, the larger is the vocabulary and the more similar sounding are the words, the larger will be the error rate. If accuracy is a problem, make use of the sub-groups in the &RECOG command. In a two step recognition process, the first recognition can be limited to only a few distinctly different words. Perhaps one of these recognitions then brings up a second group of words which are equally distinctive among themselves. And so on for third and fourth sub-groups. The way that different word groups can be arranged is almost limitless, partly with the number after the command as &RECOG n, and partly by blanking and unblanking certain words in the total template set. And, of course, recognition of a particular word can cause a second complete template set to be loaded from disk memory (or RAM-disk). A series of recognition steps can in principle involve an almost unlimited number of different words or other sounds. Programs with menus are well tailored to two step recognition methods, especially "pull-down" menus. A literal example applies to a fast food restaurant where you first select categories such as "sandwich", "drinks", or "deserts". For a second selection you choose the type of drink, or the type of sandwich.

A final suggestion is to make words in the vocabulary all have about the same durations. A word is time normalized to 12 patterns which make up the template. A short word is stretched and a long word is compressed. The recognition algorithm measures word durations as well as template shapes and recognition is refused if durations of a template and an input unknown differ by more than 50%. But even with allowable differences thus limited, errors can occur from time distortion, especially where short plosive sounds occur.

DEMONSTRATION PROGRAMS ON DISK

Note: The vocabularies used with the demonstration programs have been amplitude edited. Some have in addition been edited with "Speech Construction Set". The quality is thus likely to be somewhat better than can be realized with directly recorded vocabularies which have not been edited.

Put the Voice Master disk in the drive and turn on the computer. There occurs an auto-load and a "MENU" appears. A number of choices are presented.

For music applications, select "COMPOSER" and refer to the music manual. No further relationship with the speech material presented here will be discussed here.

Select "EXIT TO BASIC" and you get back to BASIC with a Voice Master program installed. The particular program that is selected will automatically utilize the upper 64K memory bank, if installed, and Sound Master in slot number 4, if installed. If you want to use a program that does not utilize all of the resources available, then you must directly load in the two part program as described in the section of the manual "SPEECH PLAYBACK".

Selecting "CALIBRATE" is the same as executing the Voice Master wedged-in keyboard command &CALIB except that instructions also appear on the screen as a one page display with the ? appearing in the lower right hand corner.

The selection "EDITOR" installs the same program discussed in the "AMPLITUDE EDITOR" section.

The remaining selections, all in the nature of demonstration programs, are discussed in the balance of this section of the manual. Many require some speech input and this requires the ability to properly use the microphone. But otherwise they can be enjoyed without having to first understand the programming previously discussed in this manual. It is suggested, however, that you first read "CALIBRATION AND MICROPHONE TECHNIQUE". You may be able to use "CALIBRATE" selected from the "MENU" instead of "BAR" or &CALIB from direct loading. But read the section anyway. To load "BAR" or to directly use the wedged-in command &CALIB, you will first have to exit to BASIC from "MENU" by selecting option "0". (Another alternative is to install "BAR" as a selection from "DEMO" described below.)

DEMO: A sub menu appears. Press the "R" key to prepare to record a word or phrase. Press it again and then speak. After recording, play it back with the "P" key. Play it back with an echo effect that is created with repeated playback at decreasing amplitudes by pressing the "E" key (but only when Sound Master is installed and properly utilized by the software). Play it back at different speeds with the "V" key. These several variations are implemented with specific Voice Master commands. You can exit from "DEMO" back to "MENU" with the "Q" key.

The "D" key gives a short recognition demonstration. You are prompted to enter 4 words, "Covox", "Voice Master", "Computer", and "Finish". After inputting these (to &TRAIN and make recognition templates), then repeat back the various words to see if the computer printout indicates recognition. Say "Finish" in order to return to the menu for "DEMO". If your word "Finish" doesn't work, then press the RETURN key. You can use another set of 4 words as you may wish, such as right, left, go, and stop, with the last word showing on the screen as "Return". (This demonstration makes a one-pass template. Accuracy improves with a double average as discussed elsewhere.)

The "S" key for "SPECTRUM DISPLAY" gets "BAR" as discussed in the section on "CALIBRATION AND MICROPHONE TECHNIQUE". You can experiment with this display to see how patterns change with your speech. Information contained in the bar patterns is used in part for word recognition. The furthest right bar measures amplitude and can be used in calibration. Next to this bar is one that measures fundamental voice pitch. It is used with the music programs and also with the optional "Speech Construction Set". It is not used in any programs on the Voice Master disk except for "BAR" (and indirectly in certain pre-recorded vocabularies). Try raising and lowering your voice pitch and watch this bar behave. The remaining bars measure different periodicities and the display is similar to a frequency spectrum except that high frequencies are on the left. You can try a sliding whistle to see how frequency is discriminated. The "ss" sound shows as a high frequency one. The vowel "ee" is almost as high, and "ooo" is low.

Still on the "DEMO" sub-menu, the "Q" key is presssed to return to the main "MENU".

CLOCK: You are told how to set time. Then you are prompted to record words needed for the clock, or use a pre-recorded vocabulary called "CVOICE". After the clock is started, you can get the time spoken out upon pressing a key. There also is an alarm clock feature. If you do elect to make your own vocabulary, follow menu instructions. But be careful in naming your new vocabulary—you don't want to give it the same name as one that is already on the disk. (Note: The clock is not accurate because software loops in BASIC determine the time base. You may wish to modify the program to access an optional real-time clock, if you Apple is so equipped.)

Any number of different vocabularies can be chosen, provided each has a distinctive name. If you attempt to save a vocabulary with a name that already represents a vocabulary on the same disk, you will replace the vocabulary such that the original vocabulary on the disk cannot be recovered. If you must use the same name, and you don't want to destroy a vocabulary, then put in on a different disk. In saving a vocabulary, you must replace the original Voice Master disk with a copy or a properly formatted disk that is not write protected (i.e., there must be a notch in the jacket). You can easily swap disks if you need to load a Voice Master program.

CALCULATOR: You are prompted to record numbers, math symbols, equals, and so on. Then you can operate the computer the same way as a simple four function electronic pocket calculator. The computer speaks the keys as they are pressed and then speaks the answer after you press the equals sign. As with "CLOCK", you can make and save your own vocabulary. Two pre-recorded vocabularies are on the Voice Master disk, one called "ENGLISH", and the other "SPANISH". Choose your language! (These vocabularies have been edited.)

BLACKJACK: This is a standard form of the gambling game as played at Las Vegas (but without doubles). You train the computer to your words. From then on, you need not touch the keyboard. A pre-recorded (edited) vocabulary called "DEALER" speaks cards and other data to you. You are given a sum of money to start. You say numbers such as "one", "five" to give 15. You can erase this bet by voice and replace it. Then say "bet" and the dealer deals and reads visible cards. You then say "stand" or "hit me". The idea is to get as close to "21" as you can without going over. Aces can count as one or eleven. After your final "stand", the dealer says either that you won or he won or a draw and your accumulated capital is updated. Say "cards" and see what cards have been played (for "card counting" practice). Say "cards" again and return to the game.

If you don't like the dealer's voice, you can make your own vocabulary and save it with a BASIC program on the disk called "BJVOICE". Make sure that you give your vocabulary some name other than "DEALER" (unless you load it from a disk other than the one containing the pre-recorded vocabulary "DEALER".

PULL-DOWN: This program is designed to work on an Apple IIe with an 80 column card (or standard Apple IIc). It is meant to illustrate the potential of Voice Master recognition with pull-down menus. It will automatically load in a voice template file, "P-D.TEMPLATE", which contains dummy voice patterns. Before attempting to use voice recognition, press Control and "D" in order to train the pull-down vocabulary to your own voice. (Note: This program is written entirely in BASIC and thus is a little slow.)

Please note that many of the demonstration programs are more than just simple demonstrations. They are useful and practical examples of what the enterprising author of software can accomplish, whether it be in education, business, or entertainment.

SELECTED PROGRAMMING EXAMPLES

You can list the various demonstration programs on the Voice Master disk in order to study programming methods and techniques. However, these programs have not been written with the objective of making them easy to interpret. The purpose of this section of the manual is to give some examples that are easier to understand.

Talking Numbers: We will next write a program for a very simple talking keyboard. It will speak numbers 0-9 from "ENGLISH" as you press the numbered keys, followed by RETURN. End the program with a number greater than 9. You must first be sure that both parts of one of the two-part Voice Master programs are in main memory. Then you must &FIND"ENGLISH" because we are going to use the spoken digits from this vocabulary. You might prefer to &FIND"SPANISH". The program is:

```
40 INPUT N
50 IF N>9 THEN 80
60 &SPEAK N
70 GOTO 40
80 END
```

Next is a program that speaks numbers in DATA statements (only positive integers in the range 0-9):

```
40 RESTORE
50 READ N
60 IF N<0 THEN 130
70 &PAUSE 2
80 &SPEAK N
90 GOTO 50
100 DATA 0,5,6,3,1
110 DATA 9,7,8,4,5
120 DATA -1
130 END
```

In this example, we have used a negative number to end the program so that an out of data error statement does not occur.

Now for a slightly more realistic data talking program. We presume numbers are in the range 0-999, positive integers only. This range includes all applications where (positive) decimal values are contained in 8 bit memory cells (sub range 0-255). This program will be even more practical if we set it up to read data from some different program--perhaps one you got from a magazine listing and that you want to check for accuracy by listening to the spoken numbers as you follow along the printed listing with your eyes.

We will presume that the program to be checked, including its DATA statements, does not have statement numbers as high or higher than 10000. We will write our reading and talking program to start at 10000 and then GOTO this number in order to activate the program. You must of course have a Voice Master program in main memory. Also you must have a suitable vocabulary, such as "ENGLISH". Then you must have the program whose DATA statements are to be checked, or at least the DATA statement part of this program (with line numbers less than 10000). Finally, you must have your special checking program in memory, and this must be appended to the program to be checked. Simply type in your program after the program to be checked has been loaded, or else get your program from disk and append it.

If you load the Voice Master program and "ENGLISH" from the keyboard, then do this first. Otherwise your own program can do this loading as previously described. Next load the program to be tested and write your testing program at the keyboard, or append it from disk memory. A suitable program (not compacted for ease of understanding) is the following (where we use FOR-NEXT loops instead of &PAUSE for delay between words, just for novelty):

```
10000 RESTORE
10010 READ N
10015 FORJ=1 TO 500:NEXT J
10020 IF N<0 THEN 10200
10030 FOR J=1 TO 100:NEXT J
10040 IF N>9 THEN 10070
10050 &SPEAK N
10060 GOTO 10010
10070 IF N>99 THEN 10120
10080 M1=INT(N/10)
10090 &SPEAK M1
10100 N=N-10*M1
10110 GOTO 10050
10120 M2=INT(N/100)
10130 &SPEAK M2
10140 N=N-100*M2
10150 M3=INT(N/10)
10160 IF M3>0 THEN 10080
10170 &SPEAK 0
10180 DATA -1
10190 GOTO 10010
10200 END
```

The procedure is to break the number into separate integers and &SPEAK each by itself. Different program pathways apply depending on the particular mix of integers. The checking program ends when a negative final DATA statement appears. This is guaranteed to occur after all other DATA statements and thus provides a positive ending command. The reader will recognize that this program could be written with fewer lines by using colons to put two or more statements on a line.

A second and more efficient program converts each number to a string and then extracts one string element at a time for &SPEAK(ing). The example program is:

```
10000 RESTORE
10010 READ N
10015 FOR J=1 TO 400:NEXTJ
10020 IF N<0 THEN 10110
10030 K$=STR$(N)
10040 FOR J=1 TO LEN(K$)
10050 J$=MID$(K$,J,1)
10060 IF J$="-" THEN 10110
10070 N=VAL(J$)
10080 &SPEAK N
10090 NEXTJ
10100 DATA -1
10120 END
```

This program will speak multi-digit numbers up to the point where the form is changed to floating point. The program ends when a negative symbol is READ.

<u>Two Approaches to a Talking Keyboard</u>: Define a string as

```
A$ = " ABCDEFGHIJKLMNOPQRSTUVWXYZ.,?!'"
```

Use the Voice Master to create a vocabulary where vocabulary index numbers are 1 for "A", 2 for "B", 3 for "C", and so on. We will use index 0 to represent the space bar, which shows as a leading blank in A$ above, and the word will be "space". Don't confuse vocabulary numbers with the string count, which always starts with 1. By choice, we have started the word vocabulary with zero. Thus expect a J-1 somewhere in the program.

We can input a single character at the keyboard with the GET statement as GET B$. We then scan the long string A$ and count each element from left to right until we get a match. The count number is then used in &SPEAK N-1. Also, we can print out the identified B$, adding it to a continuing string so as to show what is typed on the screen while also speaking out the letter.

A short program that speaks the letters as the keys are pressed follows:

```
10 A$=" ABCDEFGHIJKLMNOPQRSTUVWXYZ.,?!'"
20 GET B$
30 FOR J =1 TO 32
40 IF MID(A$,J,1)=B$ THEN 60
50 NEXTJ
60 PRINT B$;
70 &SPEAK J-1
80 GOTO 20
90 END
```

Note that printing is in sequence as a kind of simplified word processor. This approach can be a little slow, especially for characters near the end of the string. We could speed it up somewhat by creating a vocabulary with the more frequently appearing letters of the alphabet in the first part of the string (as done.here with space, which is the most frequently seen "character" of them all).

Another method uses the designated ASCII symbol and reference number. Assume that the letter "A" is typed. We get the number 65 as ASC("A") = 65. If we designate B$ as the string representing the typed symbol using GET as before, ASC(B$) = 65 clearly states that the letter is "A". Now simply subtract 64. We get ASC(B$)-64=1 which produces the voiced "A" with &SPEAK 1. More generally, a suitable program segment could be

```
400 GET B$
410 PRINT B$;
420 N=ASC(B$)-64
430 &SPEAK N
```

(A modified procedure is required for the punctuation marks, as well as space, because the proper number to be subtracted is not 64.)

Of course, any practical talking keyboard must contain a number of features to avoid various typing errors or inconsistencies, avoid most non-printing characters, and so on. Also, if the string length is limited, means for handling a sequence of strings must be provided if continuing text is to be presented on the screen.

<u>The Cash Register Vocabulary</u>: A suggested vocabulary for implementing a

talking cash register is given below. The number "73" can be spoken as "seventy" followed by "three". In this way, a relatively small vocabulary can handle a large number range.

Index numbers 0 to 20 same as the spoken number.

Index	Speak	Index	Speak	Index	Speak
21	30	22	40	23	50
24	60	25	70	26	80
27	90	28	hundred	29	dollars
30	cents	31	amount is	32	and
33	debit	34	dollar	35	thank you

If we were so inclined, the words "thousand", "million", "billion", as well as "exponent", "point" (decimal) and so on could be added while still remaining well below the Voice Master limit of 64.

There are many ways we can say $ 4.95. One is "The amount is four dollars and ninety five cents". At a supermarket check stand, you are more likely to hear only "four ninety five". But there are variations. $ 4.00 will be said as "four dollars". And $ 1.00 will be said as the singular "dollar". $ 0.15 will be read as "15 cents".

Language Translation: Make a template set of the numbers one through five with indices 1,2,3,4,5. Call it "NBRS". Make a playback vocabulary of the numbers in Spanish as uno, dos, tres, quatro. Use indices 1,2,3,4. Call it "SNRS". Start by recognizing a number when spoken in English. Then PEEK(25) to speak the number in Spanish. The following example uses the English "five" to end the program and various error conditions call for a correct^d input:

```
10 BLOAD PART AX
15 BLOAD PART BX
20 CALL 35072
30 &TFIND"NBRS"
40 &FIND"SNRS"
50 &RECOG 1
60 A=PEEK(25)
70 IF A>5 THEN 50
80 IF A=5 THEN 110
90 &SPEAK A:&PAUSE 2
100 GOTO 50
110 END
```

EXTERNAL SENSING AND CONTROL

Apple II+ and IIe computers have long been used in science and engineering for measuring things in the external environment. Some serve in manufacturing to control entire processes by making measurements and then commanding things to change according to these measurements. Apple computers have been popular in such applications because of their versatility. They possess a number of slots into which an unlimited variety of circuits can be inserted to accomplish an unlimited variety of tasks. The idea of such slots is not new. It began with mini-computers in the late 1950's. It was adopted by the original Altair personal computer with the so-called "S-100" bus, which is still in use. The "STD" bus promoted by the Pro-Log company standardized interconnections for industry (with the so-called IEEE bus coming along later). The slot concept was also wisely adopted by IBM for their personal computer and has now become an international standard for IBM type machines. The Apple computer was the first personal computer to be aggressively marketed on a broad scale, especially to educational institutions. Part of the success of the Apple company can be attributed to the plug-in card concept because this encouraged many third party developers to build compatible cards which then opened up new applications which, of course, also required use of the Apple computer.

With Apple IIc, the philosophy of the Apple designers changed (for reasons that still remain obscure) and no slots were made available. One can still design hardware for external system inputs and outputs, but this tends to be more expensive because it must function from a serial (RS 232) port. Speed of data access is also limited. As of the date of this writing, Apple IIe computers continue to sell as well or better than the IIc, even though priced considerably higher. This attests to the value placed on versatility and ability to be expanded to suit many needs.

Why discuss this here? It is because Voice Master speech can announce things that are sensed or measured outside the computer. In addition, Voice Master can be used to control external events by voice command. Optional software is available which allows for control of a variety of household appliances through standard household 115 volt AC circuits (or 230 volts in many countries). But this requires specialized hardware in addition to Voice Master and it is not suited to reporting on external measurements (i.e., it is one-way, although the computer can speak out what it is doing).

What we wish to do here is to describe what can be done with minimum hardware. The discussion will relate mostly to Apple II+ and IIe. Some rather limited applications can use Apple IIc, and these will be stated where appropriate.

We will be concerned with three input/output ports which are available on all Apple II+ and IIe computers (only some of which also are available on Apple IIc). Specifically, we consider the joy-stick/paddle port with 9 pins on the connector (requires a special adapter cable for Apple II+), as well as the expanded version of this port available within the case of the computer on a 16 pin connector, the cassette tape input and output with two miniature phone jacks (not available on Model IIc), and the built-in loudspeaker. Different ports require different interface hardware and software in order to be utilized for similar tasks.

If the computer is asked to &SPEAK through the built-in speaker, then this port is not available for other uses. But if output goes through the Sound Master, or a speech output capability is not desired, then it is available and it is easy to use for producing tones of specified frequencies. You must redirect the wires that connect to the internal speaker to outside the case for Apple II+ and IIe. But for the IIc, simply insert a miniature phone plug into the external audio jack on the side of Model IIc (and turn the volume to maximum).

If Voice Master is needed in order to either &LEARN or &RECOG, then three of the available input lines must be used for this purpose. These three lines all appear on the 9 pin joy-stick/paddle connector, as well as on the 16 pin connector within the case of Models II+ and IIe. (The 16 pin connector does not exist on Model IIc.) On Apple IIc, the total number of input lines that are available on the 9 pin connector is only 5, with three of these needed for Voice Master, if used. On Models II+ and IIe, these 5 inputs are also available, plus another 3 on the 16 pin connector. The cassette read line gives another input (also not available on Model IIc).

The cable from the Apple II+/IIe to the 9 pin connector that goes to the joy stick/paddle system, (or "mouse"), connects to a 16 pin connector on the computer's main board. (As stated above, a special adapter cable is used with the II+ to convert the 16 pin connector to the proper 9 pin type). On this connector, but not on the 9 pin connector, there are 4 "annunciators" and one strobe, none of which are available on Apple IIc. None of these outputs is used by Voice Master, nor is the cassette write line.

Output Control: Let us first create a single on-off switch which we can operate by voice command. We will use the wires that normally go to the speaker (or Model IIc external audio jack). The speaker responds primarily to changes between high and low values of a single voltage. But because the speaker receives this voltage via a capacitor, it will never receive a sustained constant positive voltage. Thus whatever external switch we implement must operate on the basis of changes which are equivalent to pulses. In order to make the speaker signal change state, simply address a memory location as if you want to write to it, as PEEK(49200). The number in this location is unimportant. Repeatedly address this line to create a square wave. Each time you address with a PEEK, the state of the speaker voltage changes from zero to maximum or vice versa. You have no control over which one comes first, only that you can alternate between the two. A BASIC program with a loop to create the square wave follows:

```
10 INPUT N
20 PEEK(49200)
30 FOR J=1 TO N:NEXT J
40 GOTO 20
```

The frequency is set by specifying the number N at the beginning. An on-off switch can be created if the switched device can discriminate between a repeating square wave and no square wave at all, or perhaps between square waves at consideralby different rates.

Note that we have addressed a memory location with PEEK. A location can also be addressed with POKE. However, this actually involves two memory address actions which results in a single very narrow pulse--too narrow to be useful.

The frequency will be low because BASIC is slow. If too low, the coupling capacitor between the speaker wires and the computer will distort the square wave (especially if load resistance is small). Higher frequencies, to the tens of thousands of periods per second, are possible with a machine language equivalent to this program.

Lets create a program to command production of one of several tonal frequencies, each lasting for a period that can be set separately. In this case, &RECOG returns the number N and we must provide a place in the loop for changing frequency. We will presume that index numbers for templates have values of 0,1,2, etc. to define frequencies, with index 17 for quitting the game.

```
10 INPUT M:REM SETS TONE DURATION
20 &RECOG:REM GETS N
30 N=PEEK(25)
40 IF N=17 GOTO 120:REM END
50 IF N>17 THEN 20:REM ERROR. TRY AGAIN
```

```
60 A=0:REM SET COUNTER FOR DURATION
70 PEEK(49200):REM TOGGLE SPEAKER
80 FOR J=1 TO N:NEXT J:REM SQUARE WAVE PERIOD
90 A=A+1:REM ADVANCE COUNTER
100 IF A=M THEN 20
110 GOTO 60
120 END
```

This program makes more sense in a practical situation if there exist several on-off devices which individually respond to different tone frequencies. We could use one tone to turn on device "A", and another to turn it off. And similarly, we could use tone pairs to operate other switches. The easiest way for a device to measure frequency is to have it measure period instead. This can be done by counting clock pulses from the time that the square wave goes high until it again goes low. Other methods for sensing frequency also are applicable, although they may not work too well at the very low frequencies that are generated with the BASIC program. Creating different frequency bursts to control different switches is really the basis for many forms of touch tone dialing.

Let us next look at the four "annunciator" outputs which are available on the 16 pin connector inside the case of Apple II+ and IIe. This will require that you remove the cable going to the 9 pin connector (if it exists), or else make a cable that taps into the 16 pin connector while also allowing the cable to the 9 pin connector to be attached. Each switch associates with two memory locations. By referencing the first location with a memory read command (PEEK), the annunciator line is turned off (voltage low). By referencing the second memory location of a pair, the annunciator is turned on (voltage high). Lets consider just one of the four annunciators and leave it to the reader to dream up applications software utilizing all four. In our example, we will use annunciator line number 2 which uses memory locations 49244 and 49255. We presume template reference numbers as follows:

0 Highly reliable command to start
1 Turn off
2 Turn on
3 Go back to idle state
4 Quit the program

We also can advise of conditions with &SPEAK from a stored vocabulary. The words to use are evident from the REMark statements in the program.

```
10 &RECOG 1
20 A=PEEK(25)
30 IF A<>0 then 10
40 &SPEAK0:REM ADVISE RELIABLE START
50 RECOG 1
60 A=PEEK(25)
70 IF A>4 THEN 90
80 ON A GOTO 110,140,170,190
90 &SPEAK5:REM TRY AGAIN (MIN/MAX ERRORS)
100 GOTO 50
110 PEEK(49244):REM TOGGLE OFF
120 &SPEAK1:REM ADVISE OFF
130 GOTO 50:REM GET ANOTHER COMMAND
140 PEEK(49255)
150 &SPEAK 2:REM ADVISE ON
```

```
160 GOTO 50
170 &SPEAK3:REM BACK TO IDLE
180 GOTO 10
190 &SPEAK 4:REM ADVISE END OF PROGRAM
200 END
```

There also is a strobe output on the 16 pin connector. It drops from 5 volts to zero for about 1/2 microsecond when you read memory location 49216. This is another convenient way to generate a sequence of short pulses. These pulses can be keyed to other outputs for special purposes. For example, a toggle of the speaker that associates with a strobe pulse can mean something different without this pulse, and similarly for each of the four annunciator outputs. In other words, we can turn a one-bit sgnal into a 2-bit signal by using the strobe, and thus obtain a combined command with 4 states instead of only 2. With the separate strobe signal, one can then have 8 separate commands on the 4 annunciator lines (although at the cost of additional gating hardware).

Finally, there is the cassette write line from one of the two miniature phone jacks (except on Model IIc). This is very much like the toggled speaker and thus best for tones of different frequencies. But the signal is small, being only about 0.025 volt and hence would likely require amplification. The memory location to PEEK is 49184.

Unfortunately, the only output line from Apple IIc among those discussed above is the speaker. Output control must otherwise go through one of the serial ports, which requires more complex hardware and software. Serial ports are not available on Models II+ and IIe except from optional plug-in cards for use with printers or disk drives or modems. Virtually all Apple II+ and IIe computers in use have a disk drive which is coupled via a serial port. But many (as in classrooms) do not have a card to operate a printer. (A reasonably good printer costs more than the computer it serves!)

As an aside: The ambitious machine language programmer should be able to create a standard serial output line using any one of the four annunciator bits, or the loudspeaker, or the cassette write. It is really just a matter of timing this line on and off in a prescribed sequence. Voice Master speech is such a serial signal when applied to the loudspeaker (in the absence of a Sound Master). BASIC is far too slow in almost all cases.

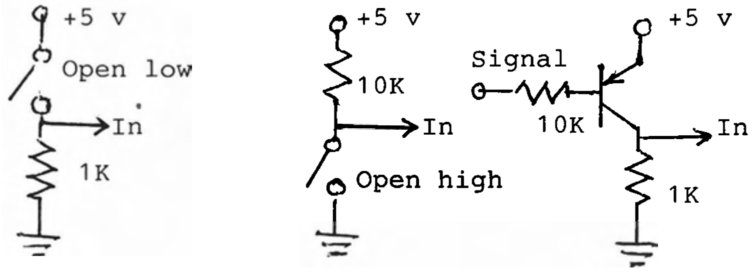
Inputs: Voice Master uses three of the inputs available on the 9 pin joy-stick/paddle port connector. Two of these are paddle port inputs and one is a switch input. On the 9 pin connector there are a total of 3 switch inputs and 2 paddle inputs. If you are using the Voice Master, only 2 switch inputs remain available, and these have exactly the same function as pressing open and closed Apple keys (except for Apple II+ which does not have these keys). If you are using one of these switches, be careful to avoid touching the keyboard: Pressing one of the Apple keys is being accomplished by the input signal which is not under your control, and pressing some other key at the same time can make the computer do strange things, including reset. If you do not use Voice Master as, for example, when the computer is giving you information from a pre-recorded vocabulary, then you have 3 switches and 2 paddle inputs (plus paddle reset) available on the 9 pin connector. If you get to the 16 pin connector, you get another 2 paddle inputs as well as the 4 annunciator lines and a strobe for outputs.

How do you read the signal at one of the switch inputs? You PEEK one of the memory locations:

49249	Switch 0 and open Apple
49250	Switch 1 and closed Apple
49251	Switch 2

A memory location read will get a binary number 0xxxxxxx if the voltage is low, and 1xxxxxxx if the voltage is high. Only the highest order bit is valid, the rest being undefined (a "don't care" condition). In BASIC when using PEEK, the low voltage case yields a decimal number less than 128, and the high voltage case gives a number of 128 or more (to 255).

We next give a program that senses switch closures (only 3). The switches could be part of a burglar alarm system. You read the state of the switch with a PEEK (or equivalent in machine language). You can also get data from a write to a memory location. However, as in the case of annunciator lines, a write actually addresses memory twice with the result that the signal for a closed switch would consist of a single very short pulse, not readily useable for sensing. The diagram suggests how a switch can be implemented. Examples with a physical switch show how to make a closure give either a low voltage or a high voltage.



```
10 A=PEEK(49249)
20 IF A>127 THEN &SPEAK0:REM A IS ON
30 B=PEEK(49250)
40 IF B>127 THEN &SPEAK1:REM B IS ON
50 C=PEEK(49251)
60 IF C>127 THEN &SPEAK2):REM C IS ON
70 GOTO 10
```

This program reports on closed switches. With another set of IF-THEN statements, it could report on open switches as well.

The paddle signals can also be used as single binary input lines. But a slight delay must be accepted because of the existence of a resistance-capacitance charging circuit. Getting an on-off value (with a PEEK to the proper memory address) may also require a short delay after setting with the reset line. Additional comments on using paddle lines as binary inputs are not given here.

The cassette read line can serve as a binary input. But this signal is only sensed as a change. Like the speaker or cassette output, it is best used to input a square wave whose frequency can be measured by counting.

Finally, consider the paddle inputs in terms of their originally intended functions. Each is meant to measure the value of a resistor in the range 0-150K ohms and return the value as a byte with a numerical value in the range 0-255. Each paddle line thus acts as an analog to digital converter. The analog reset is made by reading memory location 49264. This starts a voltage rising from zero towards + 5 volts and a counter starts to count. When the voltage rises to a fixed established value, the state of the rising voltage suddenly changes and the counter stops. The count value measures the amount of resistance. The equivalent of a resistor can be had from temperature or pressure or position measurements, to name but a few. A simple example is a mechanical moving part that is attached to a linear or rotary potentiometer. Position then directly translates to resistance. Measurements such as this, combined with switch and motor control over annunciator lines, form the basis for creating robots.

The paddle inputs can be read from Applesoft BASIC in a very simple and direct manner which we set down here for completeness. Simply use a command or statement A=PDL(0) which sets A at a value between 0 and 255 according to the resistance at paddle port number 0. One also has ports 1,2, and 3 on Apple II+ and IIe (but only 0 and 1 on IIc) which are addressed in the same way. BASIC takes care of putting in enough delay to assure that the voltage has increased to the fixed threshold level before the critical time duration value is acquired. One can, of course, use the value of PDL(J) to form a voiced message from a stored vocabulary. You could have temperature, wind velocity, wind direction, and humidity all spoken out as measured somewhere else with comparatively simple potentiometers.

APPENDICES

1: COMMAND SUMMARY.

Recording and Playback

&SPEAK n. Designated word or phrase in the range 0 to 63. Plays back through the internal speaker, TV monitor (of rf modulator used), external amplifier, or optional Sound Master. &SPEAK(ing) a phrase that does not exist gives a tone beep. The space bar resets to the start of the word being produced.

&LEARN n. Word or phrase with index n is recorded in main memory. Re-entered phrase replaces previous one. Stop in-process recording with any key. When waiting for input, put on hold with Control A (64K version) or Open Apple key (128K version). Press a second time to return to &LEARN. Too long an input state causes time-out.

&PUT"filename". Saves vocabulary on disk, starting on page n (see &RESET). Also saved are speed and volume settings.

&FIND"filename". Recovers named vocabulary previously &PUT. Retains same starting page address and speed and volume settings.

&RESET n. Number n either omitted or given in the range 16-114 (64K Version) or 16-176 (128K version). Clears vocabulary from main memory and prepares for introducing a new vocabulary from the microphone. Sets parameters to normal (default) values. The number n specifies the page in memory where the vocabulary starts (decimal 256*n). With n left unstated, the default value of 64 is inserted. Loading a vocabulary sets the page number for the original recording and clears any vocabulary previously in main or auxiliary memory. The number n applicable to a stored vocabulary cannot be changed.

&SPEED n. Changes playback speed with n in the range 0-10 with normal (default) value of 6. Speed of playback proportional to n.

&VOLUME n. Changes output volume if optional Sound Master used. There are 16 levels, 0 to 15, with 15 the loudest and 0 being zero. Amplitudes greater than 15 in the digitizer are limited to 15. Normal (default) value is 15. Not applicable for Apple IIc.

&PAUSE n. This acts like a software timing loop and produces a fixed delay. n is the number of one-tenth second delay increments.

&SAMPLE n. Changes the sampling rate at which the speech is digitized. n is in the range of 0 to 10 with the normal (default) value 6. A nonstandard sampling rate will not play back at a normal speaking rate unless &SPEED has the same index number.

&CALIB. Displays question mark (?) as in &LEARN and &TRAIN. Does not actually record. Subject to time-out. Use for calibration.

Recognition

&TRAIN n. Designated word in the range 0 to 31. Re-training the same number creates an average template. Terminate a TRAIN with any key. Control A (64K) or Open Apple key (128K) puts &TRAIN on hold. Time-out if word not produced in time or if too long.

&BLANK n. Clears the template for word n for n in the range 0-31. &BLANK without a number clears all templates. When re-training a word, &BLANK first in order to avoid averaging.

&UNBLANK n. Recovers the template previously &BLANK(ed), or all templates if index number n not used.

&RECOG n. Program waits for an input and attempts recognition by comparing the template made for the input word to those in memory. If no number is specified, all 32 templates are scanned (with untrained templates skipped without delay). For n=1,2,3, and 4, template numbers 0-7, 8-15, 16-23, and 24-31 are scanned respectively. The best fit template number is placed in memory location 25. Other numbers are put into location 25 for certain errors and other conditions. Terminate with any key. Time-out applicable.

&ACCEPT n. Index n in range 0-4 sets error criteria. Zero value accepts most words. Value 4 has close tolerances. Variations in error measures can be changed in memory locations.

&TPUT"filename". Saves a set of 32 templates in random access memory to disk memory.

&TFIND"filename". Transfers a set of 32 previously saved templates from disk to random access memory.

2: COMMENTS ON MEMORY USE

Programs which manipulate speech recording, playback, and recognition utilize several addresses in page zero of computer main memory. Most of these locations are saved by the Voice Master wedge when it is called, and then restored when the wedge programs are finished. Exceptions are locations 25 and 31 which are used by the wedge.

The Voice Master program is in two parts. The first is a 3.3K section located just below DOS starting at 37052 ($8900). The second part resides in Bank 2 starting at location 53248 ($D000) which is behind the Applesoft ROM. The speech input buffer and general work area begin at location 57344 ($E000) which is behind the ROM. Voice recognition templates are stored in Bank 1 at location 53248 ($D000) behind the ROM. Speech playback data is stored in either main memory or auxiliary memory depending upon the program used. If speech is in main memory, then the maximum amount of speech will be realized when it starts at location 4096 ($1000) and extends up to 35071 ($88FF). Maximum speech memory available in auxiliary memory extends from location 4096 ($1000) to location 53247 ($CFFF). Stored vocabulary words for playback are located as desired through use of &RESET n.

It must be realized that the memory map for a computer is quite specific to that computer. Voice Master programs are not transferrable from one computer to another without numerous adjustments being made, even between computers having the same type of microprocessor employed in the Apple II. Machine language programs for speech recording and playback as well as for disk storage and retrieval make frequent use of utility programs in the Applesoft ROM.

3: IMPORTANT MEMORY LOCATIONS

Many of the memory locations in this section refer to a "BASE" address, which is defined in the &RESET statement used in conjunction with making and saving a vocabulary. The BASE address is stored in memory location 35076.

Memory location 25 ($19 hex). Current phrase number recorded or spoken during speech recording and playback with a range 0-63. In recognition, contains index number of best match, or a number defining error or other parameter related to recording or recognition.

Memory location 29 ($1D). Volume setting to be subtracted from 15. Range 0-15. Sound Master only.

Memory location 31 ($1F). Contains paramter setting the time-out value equal to the number of one-tenth second intervals.

Memory location 35075 ($8903). Contains slot number that the Voice Master is plugged into if using Sound Master playback software. Otherwise it contains 255 ($FF).

Memory location 35076 ($8904). Contains the page number of memory where speech data begins.

Memory location 35077 ($8905). Number equals 0 if not using the extended memory version. Otherwise number equals 1.

Memory locations 35078-35087. First part of memory in which two-part Voice Master program resides. Contains numbers and parameters relating to error criteria in recognition.

Memory location 35088. Number determines shortest phrase that can be recorded. Normally set to 12. If too short, a recording can start from clicks or low level background noise. If too long, initial parts of speech can be missed.

Memory location 35089. Number determines the duration of low level sounds at the end of a word which terminates the recording. If too short, the recording can stop at a short time gap when not intended. If too long, delays occur. Nominal value is 12.

Memory location 35090. Default value of zero causes speech input for recognition or training to be rejected if the duration of the input differs from that of a template in memory by more than 50%. If the value is set to 1, the allowable difference in durations is reduced to 25% which makes recognition more selective but requires more consistency in speaking a word.

Memory locations BASE to BASE+255. These 256 bytes of memory define starting and ending addresses of where a recorded phrase is stored in memory. For example, to find where the 7th phrase is stored, multiply 7 times 4, which is 28, to add to BASE. PEEK this memory location to yield the low order byte of the start of phrase 7, and the next location is the high order byte. The next two locations are the low and high order bytes, respectively, for the ending address for phrase 7. Note: PEEK will not work if you use the 128K version because speech resides in the upper 64K of memory which cannot be PEEK(ed) or POKE(d) from BASIC.

Memory Locations BASE+256 and BASE+257. These two memory locations define the current top of speech memory.

Memory location BASE+259. Total number of recorded phrases. Range is 0-63.

Memory location BASE+265. Recording &SAMPLE setting. Same numbers as for &SPEED setting.

Memory location BASE+266. Playback &SPEED setting. Same as in recording rate (&SAMPLE) setting. With POKE(s) instead of the wedged-in command, any number in the range can be used to get intermediate &SPEED (or &SAMPLE) values. A limited form of singing is possible by changing playback $SPEED for a single recorded note. In the following table, the number in BASE+ 266 (or 267 for &SAMPLE) is the same as the number used in $SPEED n.

&SPEED n	Sample Rate (hertz)
0	4000
1	4400
2	5000
3	5300
4	5900
5	6500
6 (default)	7100
7	7900
8	8900
9	10,500
10	12,500

Memory locations BASE+267 to BASE+330. List of 64 bytes giving the order in which the phrases where &LEARN(ed). Example: &LEARN phrases 3,8,12,45, and 4, in that order, then the memory location starting at BASE+267 will contain 3, followed by 8, then 12, and so on. Memory location BASE+259 contains total number of phrases &LEARN(ed).

Memory locations BASE+331 to 35071 or to 53248. This is where the actual digitized speech is stored. (First and second addresses for 64 and 128K versions, respectively.) For the minimum value BASE=16, compute the total available memory for speech as: 35071-16*256=30975 bytes (64K system).

4: ORGANIZATION OF VOCABULARY

Data to the computer from Voice Master consists of three square waves. The principal one follows the rapid changes of the detailed speech waveform, with components to thousands of hertz. A second one is more slowly varying with a frequency (period) that changes with the average amplitude of the speech. The period is measured by counting sample values for the duration of a square wave period, thus implementing an analog-to-digital converter. A third square wave follows voice fundamental pitch, but this is not used except for the music feature of Voice Master and for the optional "Speech Construction Set". The fast square wave is sampled at the a rate of 7100 per second for the normal or default condition (&SAMPLE 6). Samples are formed as sequences of 1's and 0's, usually with several of each type in a row (but there can be just one). These samples are formed into a series of 8 bit groups, or bytes. But before each group of 15 bytes, a single byte is inserted to indicate average amplitude. In reproducing speech, this amplitude byte sets the gain of the Sound Master so as to reproduce the original square wave, but with a controlled amplitude. If Sound Master is not employed, amplitude bytes are ignored. Some errors occur because samples do not exactly line up with the original square wave edges. This error is reduced when &SAMPLE values are above the default value, but at the cost of additional memory for storage.

The beginning byte of a vocabulary consisting of one or more words (up to a total of 64 words) is at BASE+331. The starting address can be displayed (for 64K versions only) as

```
PRINT PEEK(256*n + 331)
```

where n is the page number used in &RESET (default value 64). Each word in this vocabulary has starting and ending addresses that are to be found in the range BASE to BASE+255, with the starting address for the first word recorded being that computed above. The first byte of a vocabulary word is an amplitude byte and this is followed by 15 fast bytes. Then another amplitude byte followed by 15 fast bytes. And so on to the end of the word. The amplitude byte by itself uses only 4 of the available 8 bits to give a range of 16 amplitude levels, including zero. The other 4 bits are available for other uses, including the "S" key command available with the amplitude editor. Each word in the vocabulary can be recorded with a different &SAMPLE value, but this number is not retained in a saved vocabulary. (It is implied by the degree of roughness in the sampling structure within each word).

5: SPEECH PLAYBACK-ONLY PROGRAMS

In order to provide a means for software authors to include Voice Master speech in their programs using a minimum amount of memory, or for those that desire speech playback under Apple ProDOS, six programs have been provided. These playback programs are limited to loading pre-recorded vocabulary files from disk, and speaking words or phrases from these vocabularies. The programs do not utilize "&" (wedged-in) commands; operation is with memory pokes and calls. This makes them suited for use in other programming languages besides BASIC. Individual programs are only a few hundred bytes in length.

The six programs are as follows:

PLAY	-- With Sound Master, 64K, DOS 3.3
PLAYX	-- Without Sound Master, 64K, DOS 3.3
PLAYE	-- With Sound Master, 128K, DOS 3.3
PLAYEX	-- Without Sound Master, 128K, DOS 3.3
PDPLAY	-- With Sound Master, 128K, ProDOS.
PDPLAYX	-- Without Sound Master, 128K, ProDOS.

Each of these six programs must be loaded with a corresponding boot program. For example, to load in the PLAY program, type BRUN PLAY/BOOT. Similarly, type BRUN PLAYEX/BOOT to load in PLAYEX. The boot program requires less than 100 bytes.

The four DOS 3.3 playback programs reside mainly in the language card starting at $D000 in bank 2. Since you cannot access this memory directly, a short part of the boot program is located at $9500 (hex) just below DOS. The two ProDOS playback versions are quite different and will be discussed later.

Playback under DOS 3.3

After using the proper boot program to load in your desired playback program, a vocabulary speech file can be loaded from disk memory. This is accomplished by poking the ASCII equivalent of the filename (with the most significant bit set to one) into a special memory location and then calling the load addresses. For example, assume you want to load a file called "ENGLISH". The following steps will accomplish this:

```
10 A$="ENGLISH"
20 FOR W=1 TO LEN (A$)
30 POKE 38272 + W - 1, ASC (MID$ (A$,W,1)) + 128
40 NEXT W
50 POKE 38272 + W - 1,141: REM REQUIRED ENDING BYTE
60 CALL 38150: REM LOAD FILE
```

In order to play back a particular phrase, first POKE location 25 with the desired phrase number then CALL 38148. For example, the following program will ask for a particular phrase number and play back that phrase:

```
10 INPUT "ENTER PHRASE NUMBER ";N
20 POKE 25,N
30 CALL 38148
40 GOTO 10
```

A BASIC program on the Voice Master disk, "PLAY/DEMO", demonstrates how to use these playback routines. Simply RUN PLAY/DEMO and a menu gives full instructions for loading the proper speech playback program and vocabulary. You can list the program for study purposes.

Memory location 38147 contains the slot number that the Sound Master is plugged into (if used), or contains a 255 if the non-Sound Master programs are loaded. (Note: This is not the same location that applies for programs with wedges.)

Playback speed (&SPEED in the version with wedges) can be changed with a POKE to the proper memory location, but only for the 64K version.

Playback Under ProDOS

The two routines, PDPLAY and PDPLAYX are meant for loading and playing back speech under ProDOS. They must first be loading into memory by the appropriate boot program. The bulk of the routine resides in bank 2 of the upper 64K memory bank. A short routine resides just under ProDOS in main memory starting at location $9400. You use these programs in a similar fashion as with the DOS 3.3 versions with a few exceptions. Speech is always stored in the upper 64K bank, and therefore your RAM Disk is disabled.

Before using these programs, you must first convert the two playback files, as well as the two boot files, from the DOS 3.3 format in which they are provided on your Voice Master disk, into ProDOS format using the convert utility supplied on a ProDOS system disk. In addition, you must convert your speech file into ProDOS format.

The following instructions show you how to load the speech file "ENGLISH" assuming your ProDOS prefix is called /USERS.DISK:

```
10 A$ = "/USERS.DISK/ENGLISH"
20 POKE 38080, LEN (A$) :REM SET LENGTH OF FILENAME
30 FOR W = 1 TO LEN (A$)
40 POKE 38080 + W, ASC (MID$ (A$,W,1)) + 128
50 NEXT W
60 CALL 37894 :REM LOAD SPEECH FILE
```

The following program shows you how to play back a word:

```
10 INPUT "ENTER WORD NUMBER ";N
20 POKE 25,N
30 CALL 37892
40 GOTO 10
```

Location 37891 contains the slot number that the Sound Master is plugged into (if used) or contains 255 if you are using PDPLAYX.

6: PHONETIC ALPHABET AND NUMBERS

Phonetic Alphabet:

Alpha Bravo Charlie Delta Echo Foxtrot Golf
Hotel India Juliette Kilo Lima Mike November
Papa Quebec Romeo Sierra Tango Uniform Victor
Whisky X-ray Yankee Zulu

Airman's Numbers:

Zero One Two Three Four Five Six Seven
Eight Niner

Telephone Operator's Numbers:

Oh One Two Thuh-ree Fow-wer Fie-yuv Six Seven
Eight Nine (or Nie-yun)

7: CALIBRATE AND GAIN CONSIDERATIONS

One of the most critical aspects of having successful voice recognition and recording is understanding the relationship between proper calibration and gain setting, and how they relate to two software counters: minimum acceptable duration (MAD) and maximum zero count (MZC).

The Voice Master uses a VOX, or voice operated switch, to automatically determine when a speech utterance begins and ends. This is accomplished, in part, by monitoring the average volume of the input. When this volume exceeds a threshold level, recording commences, and when it drops below the threshold, recording terminates. This is a simplified explanation and is illustrated graphically in Figure 1.

The threshold level is set by the calibration adjustment which requires a small screwdriver or "adjust tool" to facilitate. Note how the word in Figure 1 is improperly sampled if the calibration level is too high or too low. If too high, then the beginning and end portions of the desired word are chopped off. If too low, then recording begins immediately and will continue until the input buffer is full (2 seconds for &RECOG and &TRAIN, and 8 seconds for &LEARN).

The gain setting is closely related to the calibration setting. Assuming that the Voice Master is calibrated properly, the start and end of a word can still be chopped off if the gain setting is too low. Likewise, if the gain is set too high, then extraneous noise (e.g. background noise, breathing, or lip smacks) will be amplified so much that they will trigger the VOX. Figure 2 graphically illustrates how gain affects proper endpoint detection.

Next we will consider the two software counter values. The minimum acceptable duration value, MAD, corresponds to the shortest length of a spoken word that will be accepted. For example, Figure 3 shows a short, click-like sound that will be rejected if the length of the word between threshold points A and B is less than T1. The purpose of the MAD count is to prevent short bursts of noise from being considered as possible speech candidates. You can change this value with a POKE to location 35088. Too large a value for MAD will cause the Voice Master to reject short words such as "bet" or "two". Too small a value will let sounds such as key clicks from the keyboard trigger the recording routine.

The other software counter is the MZC. This value can be changed by a POKE to location 35089. The MZC determines the time the recording routine continues to sample data after the amplitude drops below the threshold. This silence period, shown as T2 in Figure 4, extends from point B to point C. After the recording stops, this period is subtracted from the input buffer so that only the speech from point A to point B is retained. If the MZC value is set too small, then any time a short pause occurs btween words, or parts of words, recording can cease prematurely and only the first part of the utterance greater than T1 is retained. If you experience problems when recording several words together in a single phrase, i.e. recording ceases too early, then increase the MZC value. However, if the MZC count is set too large, one of two things will happen: First, you will notice an increase in the time it takes to stop the recording process, which is not a problem when &LEARN(ing) but does noticeably slow down word recognition speed. If a sound exceeds the threshold level during the silence period T2, even if this sound burst is less than T1, then the MZC is reset to the starting value and that sound burst will become part of the speech sample. This wastes a lot of memory for speech storage and will create significant recognition errors.

One of the best methods of determining if the parameters discussed above are adjusted properly is to LEARN words and listen to the result. Use &SAMPLE 8 because this is the sample rate used for recognition and the &CALIB command. Listen carefully for abrupt chopping of the word, elimination of portions of the word (e.g. the "e" in "equals"), or for excessive noise or silence gaps at the beginning or end of the word. A more accurate means is to use the "EDITOR" program to visually inspect the endpoints.

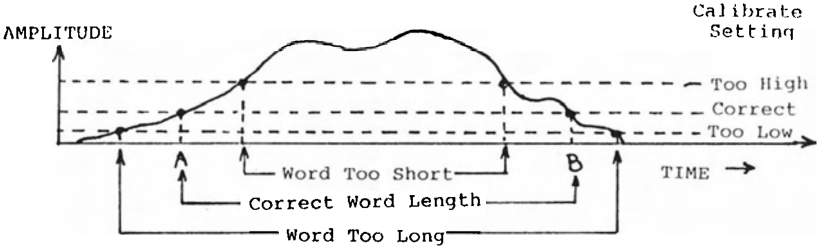


FIG 1 -- Effects of Calibration Setting

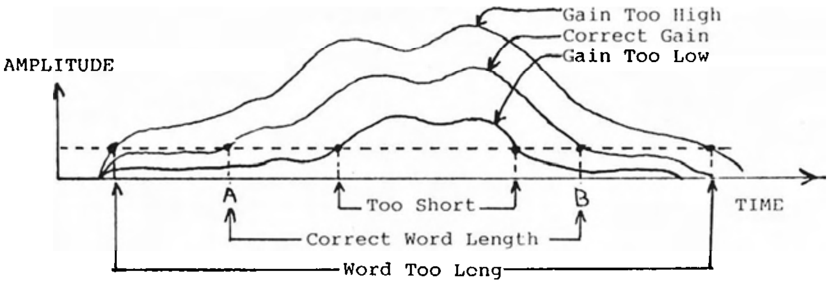


FIG 2 -- Effects of Varying Gain

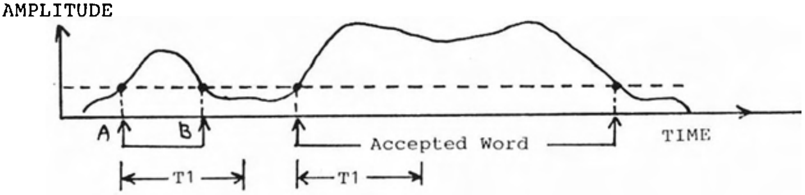


FIG 3 -- Effect of Minium Acceptable Duration (MAD)

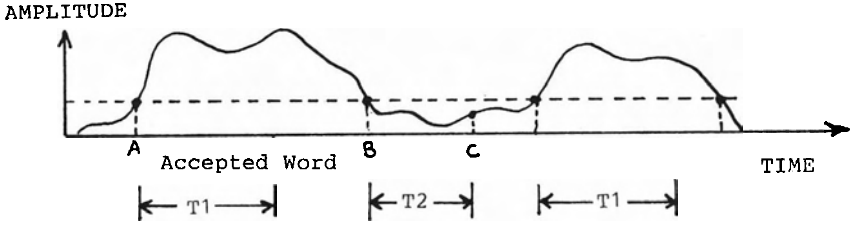


FIG. 4 -- Effect of Maximum Zero Count (MZC)

COVOX INC.
675-D Conger Street • Eugene, Oregon 97402 • U.S.A.
Area Code (503) 342-1271 • Telex 706017 (Av Alarm UD)