

TALKER 2.0

**Text to speech converter
For VS-100 Voice synthesizer.**

USER'S MANUAL



ALPHA *Products*

79-04 Jamaica Ave., Woodhaven. NY 11421

How to Use TALKER 2.0

Loading TALKER 2.0

TALKER 2.0 requires a minimum of one disk drive and 32K of memory. Model III: Rrefer to the VS 100 Manual to transfer the "TALKER" file onto a Model III disk.

Model I: All the features of TALKER 2.0 will work on the Model I VS-100 except the "background speech feature," which will only work on VS-100 (Model I) shipped after March 1984.

The file is called "TALKER;" it is a command file about 6K long. Since TALKER is self-relocatable, be careful not to load it several times, as you will end up filling memory with duplicate TALKERs. TALKER will remain in high memory until you reboot the computer.

After you transfer the TALKER file to your own disk (Model III must CONVERT), simply type "TALKER" from DOS READY.

TALKER will load itself into memory in the highest available space. A banner will appear on the screen and control is returned to DOS. We will see later that many options can be specified when loading. From this point, you may go into BASIC and experiment with the power of PRINT★ and PRINT!.

The Beauty of PRINT★ and PRINT!

(★ is an asterisk, read it "Print star")

Anytime TALKER is loaded, BASIC will recognize two new statements:

PRINT★ is a new "speak" statement. To experience the power of PRINT★ let's load TALKER, go into BASIC and type:

PRINT★ "It is working" or type: A=1234.56 : PRINT★ A

Note: PRINT★ can be abbreviated using ?★.

The new PRINT★ command will accept *any* valid string or numeric constant or variable. You can even include complex expressions or functions, such as MID\$, LEFT\$, SIN, etc.

PRINT! is another new BASIC statement: It prints on the screen exactly as a regular PRINT command, and in addition, the text is spoken. So PRINT! is the same as PRINT and PRINT★ together.

To get your favorite BASIC program to talk, simply LOAD, then LIST the program. Add a few "!" to the existing PRINT statements, where you want speech. (Remember that the beauty of PRINT! is that the ! does not affect the PRINT function. It *adds* speech without affecting the PRINT to the screen).

New control codes

In addition to standard text, many new control codes can be sent to TALKER 2.0 with PRINT★ or PRINT!. A control code is defined by one or two letters or numbers enclosed in slashes. Example: /4/ select pitch 4

The Automatic Echo Options

In addition to PRINT★ and PRINT!, TALKER 2.0 allows two more ways to get speech. Since it is easier to demonstrate than to explain, let's do the following:

If you start from scratch, Turn the computer on. At DOS READY type TALKER, go into BASIC, and then type:

PRINT★"/KL/" **KL** stands for **K**eyboard echo by **L**etter.

Nothing happened, but you just sent the control code KL to TALKER 2.0. From now on, any key you press will be echoed. Try pressing a few keys. This feature stays on even if you return to DOS or run some other programs. To turn the automatic keyboard echo off, simply type (from BASIC)

PRINT★"/KN/" (**K**eyboard echo **N**o)

Now type PRINT★"/KW/" (**K**eyboard echo by **W**ord).

From now on, everything you type will be said word by word. TALKER considers a word to be completed when you key a space, ENTER, CLEAR, the arrows, etc.

To terminate the echo function type PRINT★"/KN/" (**K**eyboard echo **N**o).

Automatic Video Echo

This function is similar to the keyboard echo. Anything that is sent to the screen is said. As with the keyboard echo, you have the option to have the text spelled, or said word by word. The control codes are:

/VL/ **V**ideo echo by **L**etter

/VW/ **V**ideo echo by **W**ord

/VN/ **V**ideo echo **N**o (turns echo off)

Now type PRINT★"/VW/" and try out the Video echo.

Note: The automatic video and keyboard echo stay in effect until turned off, (by a control code) even if you exit BASIC. Some machine language programs, such as Scripsit, have their own keyboard or display routines, so they bypass TALKER, and there will be no speech.

Background Speech

Did you notice that your computer does not slow down a bit and that the control is never lost. Your cursor is active even while the VS-100 is talking. This is what "background speech" is, and here is how it works:

When PRINT★ or PRINT! is in use, and also when the automatic echo is enabled, text is sent to TALKER for processing. There, the phoneme codes are generated and immediately sent to a buffer area. Every few milliseconds (each interrupt) the TALKER program checks to see if the VS-100 is free for the next phoneme. If it is free, the next phoneme code in the buffer is sent to the VS-100. Then the computer returns to what it was doing. If the VS-100 is not free, (still busy saying the previous phoneme), then nothing is sent. This all happens in a few microseconds (millionths of a second) and the user does not even notice it.

The buffer

When you load "TALKER" a standard buffer is set for 256 bytes, which is enough for about 25 seconds of continuous speech. The buffer is emptied little by little as the VS-100 speaks. The buffer is filled every time the PRINT★ or PRINT! or automatic echo is used.

What happens when the buffer is filled up? That means that you are sending text to be spoken faster than the VS-100 can speak. In this case, the computer waits until some space is freed in the buffer. The computer slows down but no speech is lost.

Introducing the Variable Buffer

In most applications, the VS-100 does not output speech continuously, and therefore the standard 256 byte buffer is sufficient. However, the buffer size can be chosen up to 10K. This is accomplished by adding a space and a number when you load TALKER. The number, from 1 to 40, is the number of 256 byte blocks reserved.

Example: From reboot type TALKER 40. When you press ENTER, TALKER will load and reserve 40x256=10K bytes of memory for the speech buffer. *This is enough for 20 minutes of continuous speech!*

You might want a buffer this long if you have a long ASCII file created by a word processor, or maybe downloaded from a database.

Example: You could type TALKER 40 /VW/. This command will load TALKER, reserve a 10K buffer *and* turn on the video echo by word feature. From then on, if you listed an ASCII file (from DOS, e.g. LIST TEXT) the text would be listed on the screen as usual, but it would also be sent to TALKER and converted to phoneme codes. The codes would then be sent to the buffer, and since we set up a 10K buffer, up to 20 minutes of speech could then be stored. The control would be returned to DOS, but the VS will keep talking until the end of the text.

You may clear the whole buffer at any time by hitting shift space.

Now you get an idea of the power of TALKER, but there is still much more. Here is an example:

After rebooting, type TALKER /KL//VW//PY/ from DOS READY. Talker is loaded, the **K**eyboard is echoed by **L**etter, the **V**ideo is echoed by **W**ord, and all **P**unctuation will be pronounced. (PY = **P**unctuation **Y**es)

From now on, in DOS or in BASIC, every key you press is said, and at the end of each word, the word is pronounced. With this function, a blind person could conceivably use the computer for many tasks, and could also access bulletin boards and databases.

More Control Codes

Now let's see the other control codes. Reboot, then from DOS READY type TALKER, then go into BASIC. Type in this short program:

10 CLEAR 300 : LINEINPUT A\$: PRINT★ A\$: GOTO 10

Run the program. This allows us to enter any text and control codes to try them out.

Type /SY/ (**S**pell **Y**es) From now on, all text sent to TALKER is spelled letter by letter instead of being pronounced normally. Enter some text and experiment.

Type /SN/ (**S**pell **N**o) Now it's speaking as usual again.

For example type: Radio Shack is spelled /SY/Radio Shack/SN/ usually.

Punctuation control

Type /PY/ (**P**unctuation **Y**es). Now all punctuation is said. Type some punctuation in and press enter.

Type /PN/ (**P**unctuation **N**o). Now the following punctuation marks are ignored: ! " ' () : - ; , . ?

Speech Delivery Control (Speed)

There are 14 speeds from A (fast) to N (very slow).

When TALKER is loaded, speed delivery is set to DP (**D**elivery **P**roportional) This means that each phoneme is spoken with its own duration. This is the normal mode of operation. Phonemes can be sent at fixed intervals by issuing the command /Dx/ where x is a letter from A (fastest) to N (super slow).

Type /DA/ and some text. How does it sound?

Type /DN/ and some text. See how the speech is slowed so much that you can hear the individual sound components forming each syllable. Now type /DP/ to return to the more natural proportional mode.

Pitch Control

The VS-100 is capable of speaking at any of 4 pitch levels. They are selected by sending any one of the following control codes: /1/ Lowest pitch, /2/ Standard default pitch, /3/ Higher pitch, and /4/ Highest pitch. Try typing /1/Hello /2/Hello /3/Hello /4/Hello.

You can simulate a dialogue using pitches 1 and 4.

Math Mode

This mode is selected with /MY/ (**M**ath **Y**es). To see its effect, type the following:

/MY/ 31*45-10/6 > 100

In this mode, the * - . / > < are pronounced with their mathematical meanings. To return to normal mode, type /MN/ (**M**ath **N**o). Now the "*" is pronounced "star", the "-" is ignored and the "/" becomes "slash". This feature is handy in educational programs dealing with mathematics.

The Quiet Mode

Speech can be totally turned on and off in two ways:

- Using the control codes /QY/ (**Q**uiet **Y**es) and /QN/ (**Q**uiet **N**o)
- In the direct mode (DOS or BASIC) by pressing 3 keys at the same time. Shift-space-Y is the same as Quiet Yes, and Shift-space-N is the same as Quiet No.

Note: If the VS-100 is talking, hold shift-space-Y down for a few seconds. Also, while /QY/ is in effect, all control codes are still recognized.

Blank compression

The blanks (spaces between words or at the end of lines) can be ignored, by issuing the /BY/ (**B**lank compress **Y**es) control code.

In this mode, the regular space between words is shortened, and any extra spaces are ignored. This will give a quicker and more natural sounding speech. If the code /BN/ (**B**lank compress **N**o) is issued, all spaces are converted into pauses during speech output.

Remember that many codes can be combined. For example, type: (Note: we are still running our little line 10 program)

/4//VL//KW//PY//BY//SY/ This is a test.

This is a valid PRINT★ line, which will do the following:

/4/ select pitch 4 (highest)

/VL/ Turn on the automatic video echo by letter

/KW/ Turn on the automatic keyboard echo by word

/PY/ Tell the text to speech translator to pronounce all the punctuation.

/BY/ Compress the blanks, and ignore extra spaces if more than one.

/SY/ Turn the spelling mode on (as opposed to standard word by word mode)

Now with these codes in effect, anything you type will be said several times. First when you type it in (keyboard echo), then when it goes to the screen (video echo), then when it is processed by PRINT★ (in our line 10 program.)

This is obviously an overkill, it is just to illustrate the possibilities of TALKER.

Remember that all the control codes can be put into effect at the time when you are loading TALKER. For example, when you first turn the computer on and from DOS READY type:

TALKER /4//VL//KW//PY//BY//SY/"TALKER is loaded"

When you press ENTER, All the codes will be put into effect and "Talker is loaded" will also be spoken. You can include any message when TALKER is loaded. e.g. TALKER SYSTEM READY.

Note: TALKER always says "OK" when it finishes loading, then the optional message is spoken.

Remember to turn the computer off or to hit reset to avoid loading TALKER more than once in memory.

TALKER 2.0

Control codes must be enclosed in slashes, e.g. /KL/.

Automatic Echo

- KL** Keyboard echo by **Letter**
KW Keyboard echo by **Word**
***KN** Keyboard echo **No** (off)
- VL** Video echo by **Letter**
VW Video echo by **Word**
***VN** Video echo **No** (off)
- Speech Delivery Speed**
- *DP** Delivery **Proportional**
Dx x is **A** to **N**. **A**=Fastest, **N**=Slowest

Spelling Mode

- SY** Spell **Yes**. Text is spelled.
***SN** Spell **No** (off)

Punctuation Mode

- *PN** Punctuation is ignored (**No**)
PY Punctuation is said (**Yes**)

Math Mode

- MY** Math **Yes** (* - / > < said)
***MN** Math mode **No** (off)

Reference Card

★ Default after loading TALKER

Space (blank) compression

- *BN** All spaces are pauses
BY Duplicate blanks are ignored, single blanks are shortened.

Pitch Selection

- /1/** Lowest pitch (male voice)
***/2/** Regular pitch
/3/ Higher pitch
/4/ Highest pitch (female voice)

Quiet Mode

- QY** Quiet **Yes** (no speech)
***QN** Quiet **No** (speech on)

Buffer Size: TALKER N

- N is from 0 to 40. Buffer is 256xN bytes long. If N is 0 voice is synchronized with display.
★ Default, if N not specified, is 1 (256 bytes).

Direct Key Controls

- Shift Space** Clears the speech buffer
Shift Space Y Same as **Quiet Yes**
Shift Space N Same as **Quiet No**



TALKER 2.0

Second generation text to speech software for the VS-100

We are proud to introduce TALKER 2.0, the latest version of our powerful text to speech translator. We added many new features which enhance the original TALKER 1.4. The pronunciation rules (over 400) have been fine tuned. Many new control codes add to the versatility of the VS-100.

NEW FEATURES

Control codes: One of the major enhancements of TALKER 2.0 is the addition of control codes that can be mixed with the text. A control code must be included between two slashes. Example: /SY/ (Spell Yes) Any number of control codes can be sent at the same time, but they must each be enclosed in slashes. Example: /4//SY//PY/ (Pitch 4, Spell Yes, Punctuation Yes).

Pitch Control: The TALKER program recognizes the following pitch control codes: /1/, /2/, /3/ and /4/. They correspond to the four pitch levels. The default after loading TALKER 2.0 is pitch 2. Example of pitch control: PRINT★ "/1/ Are you /3/ there?" will say "Are you" at pitch 1 and "there" at pitch 3. The pitch you specify remains in effect until it is changed by a different pitch control code.

Self-relocatable: When loaded, the TALKER program automatically relocates itself to the top of available memory and protects itself. TALKER 2.0 will not interfere with any high memory drivers provided that they are loaded first, and the top of memory is set.

Automatic PRINT★: (★ is an asterisk) Once TALKER is loaded, PRINT★ will be active whenever you are in BASIC. You don't have to execute the PRTSTAR lines as you did with TALKER 1.4.

PRINT!: PRINT! is a new command which speaks the line, *and* prints it on the screen like a regular PRINT statement. This allows you to get speech and screen display with a single statement.

PRINT★ and PRINT! now accept numerical expressions: You can PRINT★ a number, variable, or expression without converting it to a string first. The entire PRINT★ line is spoken, not just the first expression as in TALKER 1.4. (Examples: PRINT★ A+B;C-D/(E-F) or PRINT! X\$; TAB(15);Y\$)

Note 1: If you are using PRINT!, expressions are evaluated twice (once for speech, and once for printing). For this reason, if you use RND(exp) or INKEY\$ in a PRINT!, the value spoken will not be the same as the value printed. Use X=RND(99): PRINT! X

Note 2: For the speech part of PRINT!, TAB(exp) and USING are ignored, so speech may not be exactly the same as the screen.

Background speech: (Model III VS-100 only for now). Phonemes created by TALKER are placed in a buffer area and sent to the VS-100 during interrupts. Your computer is free for other processing, even while the VS-100 is speaking.

Variable buffer length: You can set the size of the background buffer when you load TALKER. The buffer can be anywhere from .25K to 10K long. The default is .25K (256 bytes). this represents about 25 seconds of speech and is sufficient for most applications. (If more text is sent when the buffer is full, the TALKER program will hold the system until some area clears up in the buffer. This way, no speech is lost. Area is reserved in 256 byte blocks. To set the buffer size, simply type a number between 1 and 40 after TALKER when you load it. (example: TALKER 8 will reserve a 2K buffer (8X256) for phoneme codes. Up to 20 minutes of continuous speech could be stored in the revolving buffer when it is set at 40 (10K).

Clearing the buffer: Shift @. Since the buffer can contain as much as 20 minutes worth of speech, it might be useful to clear the buffer out. This is done at any time by holding down the shift key, then pressing the @ key. This function works under DOS or BASIC. In BASIC, it will also cause the program to pause as usual. Remember to hit any key to resume execution.

Other new control codes:

/SY/ "Spell YES" -- Text sent to TALKER is spelled, letter by letter, instead of being read.

/SN/ "Spell NO" -- (default) Text sent to TALKER is read normally, word by word.

/PY/ "Punctuation YES" -- All punctuation is pronounced.

/PN/ "Punctuation NO" -- (default) Only the following punctuation is pronounced: # \$ % & * = + @

The following punctuation is ignored: ! " ' () : - ; , . / ?

/MY/ "Math YES" -- Punctuation will be said according to their mathematical meanings: "-" is "minus" as opposed to not being pronounced; "*" is "times" instead of "star"; "/" becomes "divided by". In the math mode, parenthesis and inequality signs "><" are pronounced. They are ignored when the math option is off (/MN/ for "Math NO").

/BY/ "Blank compress YES" -- Duplicate blanks are ignored and all pauses are shortened. This speeds up speech delivery and is handy once you get used to the VS "accent".

/BN/ "Blank compress NO" -- All blanks (spaces) are converted to pauses during speech. (default)

/DP/ "Delivery Proportional" -- Phonemes have different durations. In the proportional delivery mode, the computer sends the subsequent phonemes only when the current one has been said completely. This mode is the default and is the standard way to use the VS-100.

/Dx/ "Delivery speed x, where x is a letter from A to N -- In these modes, phonemes are sent by the computer at fixed intervals. /DA/ produces the fastest speech; /DN/ produces very slow speech. There are twelve speed gradations between A and N.

Automatic Keyboard Echo:

/KL/ "Keyboard echo by Letter" -- Every key typed on the keyboard is said by TALKER. This will occur in BASIC and DOS, and most other situations as long as the standard keyboard routine is used. Some machine language programs, such as Scripsit, have their own keyboard scan routines, thereby bypassing TALKER 2.0. (This also applies to /KW/, below.)

/KW/ "Keyboard echo by Word" -- Keys are echoed at the end of each word typed. A word is spoken when any space or BASIC control character (ASCII code less than 32) is typed. This means that TALKER will try to pronounce a group of letters once SPACE or ENTER or CLEAR or the arrows or BREAK, etc. is typed.

/KN/ "Keyboard echo NO" -- Turns off keyboard echo.

Automatic Video Display Echo:

/VL/ "Video echo by Letter" -- Each letter sent to the screen is echoed to the VS-100. Note that the standard video routine must be used for TALKER to catch the characters and send the sounds to the VS-100.

/VW/ "Video echo by Word" -- Each word sent to the screen is read by talker individually. Speech is triggered at the first space or BASIC control character following the word.

/VN/ "Video echo NO" -- Turns off video echo.

Defaults:

Note 1: After loading TALKER, all the controls are set as follows:

/2/ Pitch level 2 is selected

/SN/ Spelling NO. Text is read word by word.

/PN/ Punctuation NO. Most punctuation is ignored.

/MN/ Math mode NO. Math operators are not pronounced as such.

/BN/ Blank compression NO. Blanks (spaces) are translated to pauses during speech.

/DP/ Delivery in Proportional mode.

/VN/ Video echo NO. Speech is generated only by PRINT★ and PRINT!. (unless the "keyboard echo" is enabled with /KW/ or /KL), /KN/ Keyboard echo NO. Only PRINT★ and PRINT! will produce speech. (unless the "video echo" is enabled with /VW/ or /VL/).

Note 2: Some control codes are mutually exclusive, e.g. Spell YES (/SY/) and Spell NO (/SN/). In all cases, the last code entered remains in effect until a different control code changes it. All the controls with different first letters can be in effect simultaneously. Here are a few examples:

/4//KL//MY//PY/ This sequence of control codes sets the pitch level at 4, turns on the keyboard echo with the letter by letter option, math mode is set on, and all punctuation will be pronounced.

/KL//VW/ When these two options are used together, each letter is pronounced as you type it (Keyboard echo by letter), in addition, the word is pronounced when completed (Video echo by Word).

Note 3: Each control works independently. For example: Let's say /VW/ (Video by Word) and /KW/ (Keyboard by Word) are in effect. If you type a word, in BASIC or DOS, it will be pronounced twice; once when TALKER echoes the keyboard, and then when it echoes the screen.

Note 4: You can add a voiced message, or set any controls when you load TALKER. Simply put the message and or controls after the word TALKER and after the buffer size (if any). Note that TALKER always says "OK" when it finishes loading. Examples: from DOS READY type

TALKER (buffer size of 1 and no extra message)

TALKER 8 (2K buffer and no extra message)

TALKER /4/Hello (Buffer size of 1, pitch set to pitch 4, and "Hello" is said.)

TALKER 4 /VW/ (1 K buffer and Video echo by Word enabled)

A space is required after "TALKER". The rest of the line can include any number of spaces or punctuation between the control codes and the message. Remember that the command line in most DOSes is limited to 64 or 80 characters.

Note 5: If you wish to use the USR command to produce speech, the following DEFUSR must be executed:

DEFUSR0 = PEEK(16843)—5 + PEEK(16844)*256 +
65536*(PEEK(16844)>127)