WINH19/WINH19K/WINKER

Windows Emulation of the Heath/Zenith H/Z-19 Terminal with the Kermit File Transfer Protocol

INTRODUCTION

Winh19, Winh19k, and Winker are communications programs running under Microsoft Windows and emulating the Heath/Zenith H/Z-19 terminal. The programs provide both Heath and ANSI modes. Consequently, the main features of the Heath/Zenith series terminals are available in a windowing environment.

The emulation features a very precise rendering of the H/Z-19 keypad as well as a font containing the original graphics characters. Certain improvements have also been made over the original design. In particular, some function keys can be programmed, default settings can be made from the program, and the number of lines and columns the terminal has can be preset in win.ini.

Winh19 contains only the terminal emulator. At run time, it uses the least memory of the three programs. Winh19K and WinKer, which are functionally equivalent, additionally contain the Kermit file transfer protocol. Having most of the basic features of any Kermit implementation, the user can transfer files and control the host whenever the latter is also capable of running Kermit. The programs are distributed as the pair Winh19/Winh19k or as WinKer alone.

Since these are Windows programs, the user can have up to two transactions in progress simultaneously, and file transfers can take place while other programs are active.

Only a few features of the H19 are not available. Later we discuss the precise differences, but briefly they are: no key click control, no Enter key emulation on the keypad, and the special keys on the top row have been relabeled.

The current version also lacks a few standard features of Windows. There is no selective copy-to-clipboard function. However the entire

contents of the video buffer can be slapped to the clipboard, and text can be pasted into the program. In addition some keys do not have a corresponding menu item.

UNIX users can specify h19 in their TERM variable to obtain Heath mode. If this is not available, vt52 may suffice. Alternatively, one can specify h29, h19a, or ansi. Depending on how well the termcap file has been written, such features as cursor key operation may or may not be recognized.

The programs will run only in Windows 2.03, Windows/286, or Windows/386. Windows 1.x is already obsolete. If you are a registered owner of Windows, you can receive an upgrade from Microsoft at a nominal price.

To appreciate fully the features of this terminal, the user should refer to the original Heathkit documentation. We have summarized most of the commands in the Appendix, but the information is abbreviated. The best source of detailed information is in "Heathkit Manual for the Video Terminal, Model H19, Operation", Part Number 595-2284-01, Heath Company, Benton Harbor, MI 49022.

TERMINAL OPERATION

The programs are very easy to use. Simply copy the one you wish to your Windows directory and run it from Windows. You will be asked if the default entries should be added to win.ini. If you agree, win.ini will be modified and you can then continue. Otherwise the program will exit. If you agree to continue, the program will try to open the first communications port specified in win.ini. If this is not successful, it will try the next one listed. If this one also fails, the program will exit with a message. We assume only COM1: or COM2: are available.

If a port is opened and successfully initialized, you will find yourself already on-line and ready to communicate. Most communications and terminal parameters may be changed during the session using the Set Communications and Set Terminal options in the Emulator menu. The values you choose can be saved permanently by checking the 'Save' item in the dialog box. In addition, by checking the 'Default' item in the communications dialog box, the selected port becomes the first port tried when the program is run.

With the Edit menu you can paste text from the clipboard when on line, and you can copy the contents of the entire screen to the clipboard. No attempt has been made to preserve such attributes as inverse video or the graphics characters during copy.

If you plan to use the graphics character set of the H19 terminal, you should install the font Winh19.fon with the Windows control panel. Otherwise, you will get the normal Windows terminal (OEM) font, and if you go into the terminal's graphics mode, you will see the usual characters supplied by IBM for the control characters. You may want to install the extra font at any rate if you are using a display such as the EGA where the OEM font is really too small, or the AT&T/Olivetti 640x400 screen where the OEM font is a bit too large to get the entire H19 window on the screen. Winh19.fon is in the Windows 2.0 format.

Basic Functions

It is typical of terminals of this type that every operation is performed by its receiving commands as sequences of characters prefixed by the escape character. Such sequences can originate from the keyboard if the terminal is off-line or can be received from the host. Only one setting cannot be changed in this way, and that is whether the terminal is to run in local-echo mode or not. All other operations are obtained via escape sequences. Typically, however, a user needs to know very few of these commands; the host computer, if it knows the terminal type, will perform them as required. However, most basic operations can be carried out using the keyboard and/or the menu.

The Off Line/On Line menu item can be used to toggle between line and local modes. Either click on the item with the mouse or simply type ALT-O (Letter O). In local mode you can explore the entire range of the terminal's capabilities including the behavior of the keypad, which has been organized exactly as the keypad on the H19. In particular you can make mode and setting changes by typing the corresponding escape sequence. For example, to set the terminal to 300 baud when in Heath mode, go off-line and type ESC r C. To set a block cursor,use the sequence ESC x 4. See the Appendix for a complete list of such commands.

Function keys F1 to F5 and F7 to F9 can be programmed to send user-defined strings if the control, shift, or control-shift keys are also actuated. Function key F10, and if you have an extended keyboard, F11 and F12, can additionally be programmed in unshifted mode as well. F6, which plays the role of the ERASE key on the H19 cannot be altered. The ability to program the function keys is NOT part of the standard capabilities of the H19 but is an added feature. See the win.ini discussion below for more information on setting these keys.

Some terminal commands are available directly from the keypad or the function keys. For example, F6 will clear the screen from the present cursor position to the end of the screen, and Shift F6 will clear the entire screen. If the terminal has shifted keypad mode set, then the keypad performs editing functions. In particular, cursor movement is as expected. However, the rest of the keypad may function differently from the actions marked on the keys as its action in this program is exactly as on a real H19. A complete account of keyboard usage is in the Appendix.

Note that all of the editing key actions such as clear screen, delete a line, move cursor, etc., will not be transmitted to a host when on-line if the control key is held down.

You can send a break with CTRL-BREAK, or on an extended keyboard, CTRL-SCROLLOCK. The break signal is active as long as the key remains depressed. CTRL-BACKSPACE will send a DELETE character to the host. CTRL-J will send a line feed character. The emulator can be reset to its 'power-up' values by using the Reset selection in the Emulator menu.

Finally, you can use the Scroll Lock key to conveniently stop and start the display. Hold screen is also available. Simply go off line and type the appropriate escape sequence for the emulation mode in use. Then when a carriage return is received on the bottom line, the display will pause until you strike the Scroll Lock. This will allow the next line to display. You can see the entire screen by holding down the shift key when depressing Scroll Lock. To end the hold screen mode, again go off line and type the correct sequence.

Win.ini Settings

Here is a typical set of values in win.ini along with some comments.

```
[Winh19]
    port1=COM2:          ; try to open this port first.
    port2=COM1:          ; if unsuccessful, try this one next.
    COM1:=1200,e,7,1     ; set this port to 1200 baud
                         ;    even parity, 7 bit word, and 1 stop bit.
    COM2:=9600,n,8,1     ; set this one at 9600 baud, no parity,
                         ;    8 bit word, and 1 stop bit.
    BlockCursor=0            ; set to 1 for a block, 0 for an underline cursor.
    Keyclick=1           ; currently not implemented.
    WrapAround=1             ; terminal does line wrap if set to 1
    AutoLFonCR=0             ; does LF on receipt of CR if set to 1
    AutoCRonLF=0             ; does CR on receipt of LF if set to 1
    ANSIMode=0           ; Heath mode if 0, ANSI mode if 1
    ShiftedKeypad=1      ; Keypad is shifted if set to 1 (keypad sends
                         ;    edit function commands).
    FullDuplex=1         ; Terminal is in local echo if 0.
    lines=24             ; 24 lines plus status.  Maximum is 40
    columns=80           ; 80 columns.  Maximum is 95
    SF1=ATDT 5555555     ; Shift F1 sends this Hayes-like dialing sequence.
    SF2=whall            ; Shift F2 sends my login command.
    CF1=ats16=5d            ; Control F1 turns on analog loopback on a
                         ;    U. S. Robotics Courier modem.
    CSF2=atz             ; Control Shift F2 resets U. S. Robotics Courier modem.
    CSF3=dir\M           ; Command to display a directory on host.
```

The settings for COM1 and COM2 are in the format for an MS-DOS mode command. In particular, you cannot currently select either a default parity of mark or space. However, you can configure the terminal for such a parity type from within the program.

The number of lines or columns can only be set by editing win.ini. You may want to experiment with suitable values. On an EGA display, 27 lines works well. 30 lines gives a nice display on the AT&T/Olivetti 640x400 monitor. Finally, you can use up to 36 lines comfortably on the VGA. 95 is the upper limit for the number of columns. Don't forget to inform the host computer about the number of lines on your terminal if you select non-standard values.

The keystrings and corresponding keys are as follows:

F10 to F12   F10 to F12
SF1 to SF12   Shift F1 to Shift F12
CF1 to CF12   Control F1 to Control F12
CSF1 to CSF12   Control-shift F1 to Control-shift F12

You can change these strings from the program with the Edit Function Keys menu. Note that when editing, you have complete capability to cut, copy, and paste the strings from one selection to the next. However, you must use the keyboard interface at present, although the mouse can be used to select text for a copy or cut operation, as well as to save the string to win.ini.

Control characters ^@ (NUL) thru ^[(ESC) can be used in strings by prefixing with a backslash '\'. Thus '\M' means a carriage return and '\[' is an escape. To place a backslash in a string use '\\'.

Note that key F6 is currently always grayed and cannot be selected in the dialog box.

Winh19.fon

WINH19.FON is an OEM type font using a Modern typeface. Characters are 7x9 with extenders in an 8x10 cell. Characters 32 to 126 are standard ASCII but are not faithful reproductions of the original H19 forms because they were found to be too light on a Windows screen to be easily readable. However, characters 0 to 31 and 127, the graphics characters, are exactly the same as the original in shape and composition. In order to see the graphics characters, the user must first set the terminal to graphics mode. Then the lower case letter keys plus some of the non-alphabetic keys will display the graphics set. See the Heath manual for complete information including bitmaps of each character.

Differences with an actual H19

The following differences exist between our emulation and the real machine:

1. Key click cannot be turned on or off. This is simply regarded as too hardware-dependent to implement.

2. An extension has been made to the H19 Scroll key. When in hold screen mode, the Scroll Lock key functions just as on the H19. However, in normal mode, Scroll Lock can also be used to stop and start the display if you have XON-XOFF enabled on the host. The Scroll Lock key prevents any further reading of the communications port when toggled on. The effect is that the Windows communications driver sends XOFF to the host, effectively stopping the display.

3. Since many IBM-style, 83-key keyboards lack a keypad Enter key, the corresponding H19 functionality is also absent. Unfortunately, even on an extended (101/102-key) keyboard, it is impossible to determine in Windows whether the keypad or main ENTER key has been depressed.

4. On an extended keyboard, the Break key does not send a key released signal. So Windows simply returns both a keydown and keyup message when the key has been depressed. So you must use Control Scrollock to set and reset the communications break state on an extended keyboard.

5. Keyboard correspondence with the 83-key keyboard is as follows:

	HEATH H-19	WinH19
Main keyboard:		
	Off Line	ALT-O (menu)
	f1	F1
	f2	F2
	f3	F3
	f4	F4
	f5	F5
	Erase	F6
	Shift Erase	Shift F6
	BLUE	F7
	RED	F8
	WHITE (GRAY)	F9
	Reset	ALT-E-R (menu)
	Break	Control Break (see note 4 above)
	Scroll	Scroll Lock (see note 2 above)
	Line Feed	Control J
	Delete	Control Backspace
Keypad:		
	Insert Character (7)	Home (7)
	Up (8)	Up (8)
	Delete Character (9)	PgUp (9)
	Left (4)	Left (4)
	Home (5)	(5)
	Right (6)	Right (6)
	Insert Line (1)	End (1)
	Down (2)	Down (2)
	Delete Line (3)	PgDn (3)
	Zero (0)	Ins (0)
	Decimal (.)	Del (.)
	Enter	None

All other keys are the same. For more information, see the Appendix.

KERMIT OPERATION

Winh19k and Winker contain an implementation of the Kermit protocol developed by the Center for Computing Activities at Columbia University. The following information will help you use Kermit.

The Kermit protocol provides a way to effect an error-free file transfer files between two, possibly very different computers provided each is capable of running Kermit. Kermit also has several other features which allow one machine to act as a server while the other computer controls the server's actions by means of remote commands.

If you are a regular user of Windows, then there are several advantages to using a version of Kermit directly from Windows. First, since Windows is multi-tasking, you can run simultaneous transfers from up to two communications ports (but, see the notes below) and continue to use other applications. Secondly, transfers continue even when the window is iconic. Finally, programs written for Windows can be controlled by its memory management system for optimum memory usage.

Windows Kermit currently can send and receive files and has a fairly complete set of commands for interacting with a remote server. The program cannot act as a server (yet), and, at this time, only a few file-transfer parameters can be set. In addition, file compression is not yet implemented, and only the standard 1-byte checksum is available. The eight-bit quoting mechanism is in place so that binary file transfer over 7-bit connections is possible. But, there is no turn-around character mechanism, so certain IBM mainframe Kermits may not work with this implementation.

File transfers frequently fail because of incompatible parity and word size settings between the host and the local Kermit. You will usually get your best results by using eight bits and no parity. Any other settings will cause binary file transfers to be slower since the eight-bit quoting mechanism will go into action.

There are currently no DOS services allowing you to change directory, display a directory, type a file, etc. However, you can easily use the MSDOS.EXE window to show the files in a given directory. And, you can take advantage of some features in the SEND dialog box to help you move to another directory and to list the files there. After you

are done, just push the CANCEL button. For information on using SEND, see below.

Basic Usage

    Kermit is very easy to use. To send a file from the local computer (usually the controlling microcomputer) to the remote (host) machine the steps are:

    1. Run Kermit on the remote. At the prompt type 'receive'

    2. Choose the 'send' option on the local machine, select the files, and confirm the selection. The transfer will begin and continue until all files selected have arrived at the host or the transfer is canceled.

    To receive a file:

    1. Run Kermit on the remote machine and select 'send' along with the target file names.

    2. Select 'receive' on the local machine. The files will continue to arrive until the transfer is complete or is canceled.

    To use remote operations:

    1. Run Kermit on the remote machine and select 'server'.

    2. Issue all commands from the local machine. Typical commands are send files, get files, get help, type or delete a file on the host, etc..

    3. Terminate remote operations by issuing the 'finish', 'bye', or 'logout' command.

    During a transaction, the terminal display window is removed from the screen and various items reflecting the state of the Kermit process are shown. If a file is being transferred, statistics about the transfer are displayed. If a remote command is being executed, any information is written to the bottom line of the window and is scrolled upward. At the end of the transaction, a dialog box appears announcing success or failure. The display may be obscured by the box, but it can always be moved out of the way. However, once 'OK' has been pressed, the terminal display window is restored and the old

information is lost.

Once a file transfer has begun, you can minimize the window and have the transfer continue 'in the background'. The icon will display the packet types and numbers for both the outbound and inbound packets.

Warnings

File transfers usually fail because of differences between the local and remote parity and word size. If you cannot get even the first packet across, then try changing these parameters.

Windows is multitasking and you can run several instances of the same program. If you are running two copies, watch out for possible file conflicts such as trying to transfer files to the SAME file from two different remote Kermits. The DOS file system does not always protect you from this kind of error. It can happen that while you are receiving a file from one source, the same file can be successfully opened for sending. It usually means disaster for the incoming file.

Similarly, if the target file is a floppy disk, do not remove the disk until the transfer has completed.

Detailed Information about the Commands

Three menu selections control the operation of the Kermit protocol. They are: Protocol, Files, and Remote. Files handles send, receive, get, and the various means for canceling the transfer. Remote contains all commands which can be issued to a remote server. Finally, Protocol allows a limited selection of options.

All menus are provided with accelerators as indicated by the underlined character. Please see your Windows reference manual for instructions on using accelerator keystrokes. We have listed below when applicable the accelerator keystroke sequence for each command.

The Protocol Menu

Currently you may select the following items:

Timeout on: If this item is checked, then after a certain number of seconds have elapsed with no packets received, a retry will be attempted. If the number of retries exceed a given threshold, then the protocol will exit. Currently the elapsed time is fixed at 5 seconds and the number of retries is also 5. Neither can be varied in the current implementation.

Duplicate filename warning: If checked, then an incoming file with the same name as an existing one will be renamed so as not to overwrite the old one. The new name appends or replaces part of the old name with a unique generation number.

Discard incomplete file: If checked, then any file being received in which the protocol has been interrupted by an error condition or by the user will be deleted.

Bell at end of transfer: If checked, the bell will sound when a transaction is complete or canceled.

If the 'Save settings' box is also checked, then win.ini will be modified to reflect the new defaults under [Kermit].

The Files Menu

Using these selections, you can send or receive a file or a group of files. Before a transaction begins, you can select either of Send, Receive, or Get. After the transfer has begun, you may interrupt the transfer in various ways. Of course, you should have already set the remote Kermit as described above. Thus, before you attempt to receive files you should have already set the remote Kermit to send them. Similarly, if you are sending files to the remote Kermit, you must first command it to receive or place it into server mode. Finally, server mode on the remote host must be active in order to use the Get command.

Send (ALT-F-S): When Send is selected, a dialog box with several windows appears. Two are list boxes; the one on the left contains all files matching the search specification shown in the edit box at the top. The right list box contains a list of paths and disk drives which can be accessed. The push buttons allow you to select a list of files based on the search specification, send the selected files, or cancel the operation.

The basic principle is that a file name must be highlighted in the list box to be sent. Thus, you may send several files. However, you may select only those files from the current directory. Hence files from more than one directory cannot be sent during one transaction.

A file can be selected in several ways. The usual method is to click on the name with the mouse. More than one file can be highlighted by holding down the shift key during selection. You can also deselect one of several highlighted items in the list by again depressing the shift-key and clicking on the item. If you do not hold down the shift key, all items will be deselected except the one to which the mouse is pointing.

You can also use the keyboard to select files. Use the tab key to move to the files list box. There you can select files by using the arrow keys to move through the list and the shift key to highlight contiguous items in the list.

The default search specification is "*.*". You can choose another string to limit the files displayed in the list box. For example, if you tab to the edit box (or point to it with the mouse), type "*.txt", and press the RETURN key or the SELECT pushbutton, then not only will all files having that extension be displayed, but all will be selected as well. You can then send the entire list by activating the SEND button.

At any time, you can deselect all files but one in the list box by simply pointing and clicking on that file. You can deselect all file in the box by deleting the contents of the edit box and pressing RETURN.

You can move to another drive or directory by typing it into the edit box and pressing RETURN or double-clicking on the corresponding item in the right list box. The current search specification is appended to the path, the program moves to the new directory, and all matching files are displayed. A second RETURN or a push of the SELECT button then highlights the files list.

Finally, at any time, a single file located anywhere in the system can be selected and sent. Simply type its name, along with any required drive and/or path information and hit RETURN. If the file can be found, then a switch is made to that directory and the file name is displayed and selected. If the file cannot be found, no action takes place.

You may cancel the operation at any time by pressing the CANCEL button. Pushing SEND with no files highlighted has the same effect.

Receive (ALT-F-R): If the Receive menu item is chosen, then the program enters a state where it is awaiting the arrival of an initialization packet from the sender.

Get (ALT-F-G): If the remote host is in server mode, you may use this command to receive files. A dialog box appears in which you can type a single file name or a search specification. After the OK button has been pressed, the program sends a initialization packet and then waits for the file(s).

Exclude file (ALT-F-X): Once a transaction has begun, you can exclude the file currently being transferred by selecting this menu item. The rest of the files will not be affected. Note that if you have elected to keep partial files, the current file will NOT be deleted if you are receiving.

Zap batch (ALT-F-Z): Once the transaction has begun, you can kill the remaining transaction by making this selection. Only the current and remaining files in the list are affected. The comment about partial files in the paragraph above also applies.

Error abort (ALT-F-E): When this item is selected, an error packet is sent to the remote Kermit and the transaction is terminated. Incomplete files are discarded (or saved, if the partial file flag is set). Use this method to cleanly terminate a transaction once the exchange has started.

Cancel protocol (ALT-F-C): This selection is most useful as a way to cancel a transaction before any exchange of packets has begun. Note that the remote Kermit will attempt to continue if it has already started and you will may have to use other means to terminate the remote program, or, you can simply let it time out.

The Remote Menu

You may control many actions of the remote host by placing its Kermit into server mode and then using the various commands available under this menu item. Many of the commands will present you with a dialog box, although in some cases, the information is optional. In any case, you confirm your selection by pushing the OK button. A touch of the space bar is sufficient. Press the CANCEL button and the transaction will not start.

Some remote commands may return many lines of information. You can stop the display (for a limited time, as the remote Kermit may time out) and restart it by toggling the Scroll Lock key. Finally, during the transaction, you can use the various cancel options to abort the operation.

Help (ALT-R-H): This item will request the host to display the remote commands available. Some remote Kermits may be able to respond to help about a particular subject, and a dialog box is available to add the additional string.

Host Command (ALT-R-O): Use this item to have the host perform a shell command. An entry in the dialog box is mandatory.

Finish (ALT-R-F): Select this item to cause the remote Kermit to exit the server mode.

Logout (ALT-R-L): As the name implies, this item is used to log off the remote machine from server mode.

Bye (ALT-R-B): This selection logs off the user from the remote and exits the local Kermit.

Chdir (ALT-R-C): You can change directory on the remote server with this command. A dialog box will appear allowing you to enter the new directory, and, if needed, password. The password will not be displayed. On some remote Kermits, if you make no entries, the current directory is returned.

Directory (ALT-R-D): This item lists the files located on the remote host. An optional search specification can be given.

Type (ALT-R-T); You can type a file or group of files located on the remote host with this command.

Delete (ALT-R-D): This menu item allows you to delete files on the remote host.

Who (ALT-R-W): You can list the users on the system with this command. An optional string can also be sent.

Space (ALT-R-U): This command lists the disk space available on the host. An optional string can be sent.

APPENDIX

SUMMARY OF WINDOWS MAIN KEYBOARD USAGE FOR THE WINDOWS H19 EMULATOR

Note: Keyboards covered include IBM-style 83 and 101-key (extended) models. F10, F11, F12 are available only on 101 key keyboards. Other differences are noted below.

Key	**Heath**	**ANSI**	**Terminal action if received**
Scroll Lock			Display pauses
Ctrl Break (Ctrl Scrollock on ext kbd)			Sends break until released
Ctrl Backspace			Sends delete character (127)
F1	ESC S	ESC O S	None
F2	ESC T	ESC O T	None
F3	ESC U	ESC O U	None
F4	ESC V	ESC O V	None
F5	ESC W	ESC O W	None
F6	ESC J	ESC [J	Clear to end of display
F7	ESC P	ESC O P	None
F8	ESC Q	ESC O Q	None
F9	ESC R	ESC [R	None
F10 to F12	none	none	User defined (1)
Shift F1 to F5			User defined (2)
Shift F6	ESC E	ESC [2 J	Clear display and home cursor
Shift F7 to F12			User defined (2)
Ctrl F1 to F5			User defined (3)
Ctrl F6			Same as F6 but not transmitted
Ctrl F7 to F12			User defined (3)
Ctrl Shift F1 to F5			User defined (4)
Ctrl Shift F6			Same as shift F6
Ctrl Shift F7 to F12			User defined (4)

(1) Set Fn=string (max of 80 chars) in win.ini under [Winh19].
(1) Set SFn=string (max of 80 chars) in win.ini under [Winh19].
(2) Set CFn=string (max of 80 chars) in win.ini under [Winh19].
(3) Set CSFn=string (max of 80 chars) in win.ini under [Winh19].

18

Shifted keypad mode OFF, Alternate keypad OFF

Key	Heath	ANSI	Terminal action if received
Ins	0	0	Displays 0
End	1	1	Displays 1
Down Arrow	2	2	Displays 2
PgDn	3	3	Displays 3
Left Arrow	4	4	Displays 4
5 key	5	5	Displays 5
Right Arrow	6	6	Displays 6
Home	7	7	Displays 7
Up Arrow	8	8	Displays 8
PgUp	9	9	Displays 9
Del	.	.	Displays .
Shift Ins	0	0	Displays 0
Shift End	ESC L	ESC [L	Insert Line
Shift Down Arrow	ESC B	ESC [B	Cursor down 1 line
Shift PgDn	ESC M	ESC [B	Delete Line
Shift Left Arrow	ESC D	ESC [D	Cursor left 1 character
Shift 5 key	ESC H	ESC [H	Home cursor
Shift Right Arrow	ESC D	ESC [D	Cursor right 1 character
Shift Home	ESC @	ESC [4 h	Enter insert mode (1)
	ESC O	ESC [4 l	Exit insert mode (1)
Shift Up Arrow	ESC A	ESC [A	Cursor up 1 line
Shift PgUp	ESC N	ESC [P	Delete character
Shift Del	.	.	Displays .

Shifted keypad mode ON, Alternate keypad OFF

Key	Heath	ANSI	Terminal action if received
Ins	0	0	Displays 0
End	ESC L	ESC [L	Insert Line
Down Arrow	ESC B	ESC [B	Cursor down 1 line
PgDn	ESC M	ESC [B	Delete Line
Left Arrow	ESC D	ESC [D	Cursor left 1 character
5 key	ESC H	ESC [H	Home cursor
Right Arrow	ESC D	ESC [D	Cursor right 1 character
Home	ESC @	ESC [4 h	Enter insert mode (1)
	ESC O	ESC [4 l	Exit insert mode (1)
Up Arrow	ESC A	ESC [A	Cursor up 1 line
PgUp	ESC N	ESC [P	Delete character
Del	.	.	Displays .
Shift Ins	0	0	Displays 0
Shift End	1	1	Displays 1
Shift Down Arrow	2	2	Displays 2
Shift PgDn	3	3	Displays 3
Shift Left Arrow	4	4	Displays 4
Shift 5 key	5	5	Displays 5
Shift Right Arrow	6	6	Displays 6
Shift Home	7	7	Displays 7
Shift Up Arrow	8	8	Displays 8
Shift PgUp	9	9	Displays 9
Shift Del	.	.	Displays .

Shifted keypad mode OFF, Alternate keypad mode ON

Key	Heath	ANSI	Terminal action if received
Ins	ESC ? p	ESC O p	Displays p
End	ESC ? q	ESC O q	Displays q
Down Arrow	ESC ? r	ESC O r	Displays r
PgDn	ESC ? s	ESC O s	Displays s
Left Arrow	ESC ? t	ESC O t	Displays t
5 key	ESC ? u	ESC O u	Displays u
Right Arrow	ESC ? v	ESC O v	Displays v
Home	ESC ? w	ESC O w	Displays w
Up Arrow	ESC ? x	ESC O x	Displays x
PgUp	ESC ? y	ESC O y	Displays y
Del	ESC ? n	ESC O n	Displays n
Shift Ins	ESC ? p	ESC O p	Displays p
Shift End	ESC L	ESC [L	Insert Line
Shift Down Arrow	ESC B	ESC [B	Cursor down 1 line
Shift PgDn	ESC M	ESC [B	Delete Line
Shift Left Arrow	ESC D	ESC [D	Cursor left 1 character
Shift 5 key	ESC H	ESC [H	Home cursor
Shift Right Arrow	ESC D	ESC [D	Cursor right 1 character
Shift Home	ESC @	ESC [4 h	Enter insert mode (1)
	ESC O	ESC [4 l	Exit insert mode (1)
Shift Up Arrow	ESC A	ESC [A	Cursor up 1 line
Shift PgUp	ESC N	ESC [P	Delete character
Shift Del	ESC ? n	ESC O n	none

Shifted keypad mode ON, Alternate keypad mode ON

Key	Heath	ANSI	Terminal action if received
Ins	ESC ? p	ESC O p	Displays p
End	ESC L	ESC [L	Insert Line
Down Arrow	ESC B	ESC [B	Cursor down 1 line
PgDn	ESC M	ESC [B	Delete Line
Left Arrow	ESC D	ESC [D	Cursor left 1 character
5 key	ESC H	ESC [H	Home cursor
Right Arrow	ESC D	ESC [D	Cursor right 1 character
Home	ESC @	ESC [4 h	Enter insert mode (1)
	ESC O	ESC [4 l	Exit insert mode (1)
Up Arrow	ESC A	ESC [A	Cursor up 1 line
PgUp	ESC N	ESC [P	Delete character
Del	ESC ? n	ESC O n	Displays n
Shift Ins	ESC ? p	ESC O p	Displays p
Shift End	ESC ? q	ESC O q	Displays q
Shift Down Arrow	ESC ? r	ESC O r	Displays r
Shift PgDn	ESC ? s	ESC O s	Displays s
Shift Left Arrow	ESC ? t	ESC O t	Displays t
Shift 5 key	ESC ? u	ESC O u	Displays u
Shift Right Arrow	ESC ? v	ESC O v	Displays v
Shift Home	ESC ? w	ESC O w	Displays w
Shift Up Arrow	ESC ? x	ESC O x	Displays x
Shift PgUp	ESC ? y	ESC O y	Displays y
Shift Del	ESC ? n	ESC O n	Displays z

(1) Sends first sequence shown if not in insert character mode. Sends second sequence if in insert mode.

SUMMARY OF ESCAPE SEQUENCES RECOGNIZED BY THE WINDOWS H19 EMULATOR

Spaces are shown between characters in each sequence only for clarity. ESC means the character sent by the escape key (control []).

CURSOR FUNCTIONS	HEATH		ANSI
Cursor home	ESC H		ESC [H
			ESC [0;0 H
			ESC [1;1 H
			ESC [f
			ESC [0;0 f
			ESC [1;1 f
Cursor forward 1 character	ESC C		ESC [C
			ESC [1 C
Cursor forward n characters	none		ESC [n C
Cursor backward 1 character	ESC D		ESC [D
			ESC [1 D
Cursor backward n characters	none		ESC [n D
Cursor down one line	ESC B		ESC [B
			ESC [1 B
Cursor down n lines	none		ESC [n B
Cursor up 1 line		ESC A	ESC [A
			ESC [1 A
Cursor up n lines	none		ESC [n A
Reverse line feed	ESC I		ESC M
Cursor position report (1)	ESC n		ESC [6 n
Save cursor position	ESC j		ESC [s
Set to saved postion	ESC k		ESC [u
Address cursor (2)	ESC Y a b	ESC [r;c H	
			ESC [r;c f

(1) Heath mode replies with ESC Y a b, where a and b are characters whose ASCII values are, respectively, the row and column position of the cursor plus 31. ANSI mode replies with ESC [r;c R where r and c are character strings of digits for the row and column position of the cursor.

(2) In the Heath mode, a and b are ASCII characters whose values are the respective row and column postions offset by 31 (if the first row or column is regarded as 1). In ANSI mode, r and c are character strings of digits for the row and column position.

ERASING AND EDITING	HEATH		ANSI
Clear display and home cursor	ESC E		ESC [2 J
Erase to top of display	ESC b		ESC [1 J
Erase to end of display	ESC J		ESC [J
			ESC [0 J
Erase entire line	ESC l		ESC [2 K
Erase to beginning of line	ESC o		ESC [1 K
Erase to end of line	ESC K		ESC [K
			ESC [0 K
Insert one line		ESC L	ESC [L
			ESC [1 L
Insert n lines		none	ESC [n L
Delete line	ESC M		ESC [M
			ESC [1 M
Delete n lines		none	ESC [n M
Delete character		ESC M	ESC [P
			ESC [1 P
Delete n characters	none		ESC [n P
Enter insert character mode	ESC @		ESC [4 h
Exit insert character mode	ESC O		ESC [4 l

CONFIGURATION and MODES

	HEATH		ANSI	
Reset to Power-up state	ESC z		ESC [z	
Set baud rate (3)	ESC r char		ESC [r integer	
Enable status line	ESC x 1			ESC [> 1 h
Disable status line	ESC y 1			ESC [> 1 l
No key click (4)		ESC x 2		ESC [> 2 h
Key click (4)		ESC y 2		ESC [> 2 l
Hold screen mode		ESC x 3		ESC [> 3 h
		ESC [		
Exit hold screen mode	ESC y 3			ESC [> 3 l
		ESC \		
Block cursor		ESC x 4		ESC [> 4 h
Underscore cursor	ESC y 4			ESC [> 4 l
Cursor off	ESC x 5			ESC [> 5 h
Cursor on	ESC y 5			ESC [> 5 l
Shifted keypad		ESC x 6		ESC [> 6 h
		ESC t		
Unshifted keypad		ESC y 6		ESC [> 6 l
		ESC u		
Alternate keypad mode	ESC x 7		ESC [> 7 h	
		ESC =		ESC =
Exit alternate keypad mode	ESC y 7		ESC [> 7 l	
		ESC >		ESC >
Auto LF on CR		ESC x 8		ESC [> 8 h
Auto CR on LF		ESC x 9		ESC [> 9 h
ESC [20 m				
Reverse video mode	ESC p		ESC [7 m	
Exit reverse video mode		ESC q		ESC [m
				ESC [0 m
Graphics mode		ESC F		ESC [10 m
Exit graphics mode	ESC G		ESC [11 m	
Keyboard disabled	ESC }		ESC [2 h	
Keyboard enabled		ESC {		ESC [2 l
Wrap at end of line	ESC v		ESC [? 7 h	
Discard at end of line	ESC w		ESC [? 7 l	
Enter ANSI mode		ESC <		none
Enter Heath mode		none		ESC [? 2 h

(3)	Baud Rate	Heath mode	Ansi Mode
	110	A	1
	150	B	2
	300	C	3
	600	D	4
	1200	E	5
	1800	F	6
	2000	G	7
	2400	H	8
	3600	I	9
	4800	J	10
	7200	K	11
	9600	L	12
	19200	M	13

(4) Not yet implemented.

MISCELLANEOUS FUNCTIONS

	HEATH		ANSI
Identify as VT52 (5)	ESC Z		none
Transmit status line	ESC]		ESC [q
Transmit screen		ESC #	ESC [p

(5) Replies with ESC \ Z.