PIE 1.6
Full Screen Editor
CP/M®/HDOS

595-2758-02

HEATH COMPANY
BENTON HARBOR, MICHIGAN 49022

Printed in the
United States of America

PIE 1.6 Full Screen Text Editor
H-8/H-89/Z-89 Version 1.6

Thomas Crosley
Walter Bilofsky
September 23, 1981

INTRODUCTION

PIE 1.6 is a two-dimensional, cursor-based screen editor designed specifically for use with CRT systems. In PIE 1.6, the terminal screen acts as a "window" displaying a portion of the file being edited. PIE 1.6 uses function keys, performing simple operations which are immediately reflected in the file as displayed on the screen. What you see on the screen is what is in the file.

The screen window may be positioned to display any part of the file. Cursor motion keys allow you to position the cursor anywhere on the screen, and you can type changes to the file right on the screen at the cursor position. Other features include:

- Character and line insert and delete anywhere on the screen.
- String search forward and backward.
- Macro facility for search/replace and other functions.
- Move and copy single and multiple lines.
- Insert mode for inserting text into a line.
- Scrolling of text in the window.
- Append and clear to end of a line.
- Continuous typing mode ("hot zone") for rapid text entry.

PIE 1.6 generates standard text files, with tab characters expanded to blank spaces for disk storage. (You can suppress this feature to conserve disk space; see "Miscellaneous Ideas and Suggestions".)

PIE 1.6 may be used to prepare files for input to the TEXT formatting program. PIE 1.6 is also handy for entering and revising program and data files.

PIE 1.6 runs on the H-8, H-89 and Z-89 Computers. PIE 1.6 requires at least 32K of memory to run. If more memory is available, PIE 1.6 will be able to edit larger files. The amount of memory available for editing varies with the operating system and the machine configuration, but typically 20K to 24K characters of memory are occupied by PIE 1.6 and the operating system, with the remainder available to process the file being edited.

> **NOTE:** PIE 1.6 will not run on the H-8 Computer under CP/M unless the H-19 Terminal is connected through an H-8-4 serial interface (or a compatible interface using the type 8250 UART at port address 350 octal, 0E8 hexadecimal).

RUNNING PIE 1.6

The PIE 1.6 disk contains two versions of the software: PIEZ80, which runs only on the H-89 Computer, and PIE8080, which runs on the H-8 Computer with the H-19 Terminal. Be sure to use the correct version for your computer. Copy the program to a working diskette and place the original distribution diskette in a safe place. For your convenience, rename the working version to "PIE":

In HDOS: RENAME PIE.ABS=PIE*cpu*.ABS
In CP/M: REN PIE.COM=PIE*cpu*.COM

where '*cpu*' is Z80 or 8080 (the correct version for your computer).

PIE 1.6 is run when you type the command PIE, followed by the name of the file to be edited, followed by a RETURN (abbr. ㊏). Thus, to edit a file named DOCUMENT.TXT, type the command

PIE DOCUMENT.TXT㊏

If no file name is provided, PIE 1.6 will request one. PIE 1.6 will clear the screen, read the file in and scroll the first 24 lines onto the screen. If the file does not exist, PIE 1.6 will create it.

To edit, position the cursor anywhere on the screen, using the cursor motion keys, and type text just the way it is to appear in the file. Function keys f1 through f5 are used to move the file so that different portions of it appear on the screen. The following sections explain these operations in more detail, and describe the editing commands. "PIE 1.6 Function Key Location" (See page 17) provides a handy chart showing the location of all the command keys.

In the lower right-hand corner of the screen, PIE 1.6 displays the line number of the file on which the cursor is positioned, and the number of free bytes of memory remaining. When this number drops below 2000, PIE 1.6 will highlight it and give a warning message. If this occurs, **EXIT IMMEDIATELY; DO NOT CONTINUE EDITING.** See "Free Space," Page 15, for more information about free space.

There are several ways to end a PIE 1.6 editing session. The simplest is to type control-E (hold down CTRL and type "e"). This will write the file, with the changes made during the editing session, back onto the disk. The other options for ending the session are described below.

ENTERING TEXT

Type text into the file by positioning the cursor at the point where the text is to be entered, and typing the text. How to position the cursor is explained below.

Typing past column 72 causes the bell to sound. Attempting to type past column 80 causes the typed character to be ignored and the bell to sound. (This feature differs slightly when the Word Wrap option is enabled. See "Moving Text Between Lines," on Page 11).

You can use the cursor motion keys (the four keys with the arrows) to position the cursor at any position on the screen. To change the file at the cursor position, simply type the change onto the screen. Whatever is typed on the screen will be inserted in the file just as it appears on the screen.

Other keys you can use to position the cursor are:

HOME	Moves the cursor to the upper left-hand corner of the screen.
HOME	(SHIFTED): Moves the cursor to the lower left-hand corner of the screen.
TAB	Moves the cursor right to the next tab stop (multiples of 8 columns). (Note that this positions the cursor, but does not necessarily insert a tab in the file. PIE 1.6 can be patched to insert tabs automatically whenever they will save room in the file. See "Miscellaneous Ideas and Suggestions.")
ESC	Moves the cursor left to the next tab stop.
→	(shifted): Moves the cursor, alternately, to the beginning of the current line and to the end of the text on the current line.

To move the cursor a long distance, use the TAB and ESC functions, or the cursor motion keys with the REPEAT key.

Two keys have a dual function: they move the cursor and also perform another action.

BACKSPACE	Moves the cursor left one position, and erases the character at that position.
RETURN	Moves the cursor to the left margin, and down one line. If the cursor was on the bottom line of the screen, the file scrolls up one line in the window. (This is especially useful when you are typing text at the end of a file, since the window keeps repositioning itself as RETURN is typed.)

INSERT MODE

Normally, changes are simply typed over any existing text on the screen, and replace the existing text. In Insert Mode, a typed letter is inserted at the cursor position, and a space is opened up for it by moving all the characters on the line, starting at the cursor, one position to the right. In Insert Mode, the BACKSPACE key not only deletes the character to the left of the cursor, but also pulls the rest of the line, starting at the cursor, one position to the left to close up the space.

Insert Mode is turned on and off when you press the IC function key. When Insert Mode is on, a special message appears at the bottom of the screen.

If an attempt is made to insert a character in a full line (80 characters), the character is ignored, the bell sounds, and the message "Line full" appears at the bottom of the screen. In this situation, the DIVIDE function (ctrl-D; see Page 11) may be used to split up the full line into two lines, so the typing may continue.

Word Wrap should be turned off while you are in the Insert Mode. (See "Ctrl-W" under "Moving Text Between Lines.")

ARGUMENTS TO COMMANDS

Some of the function key commands may have their range specified by an **argument**, which is specified by typing the ENTER key, a numeric or string argument, and the function key. For example, the shifted DL key usually deletes one line, but the sequence ENTER 3 SHIFT DL deletes three lines. Arguments may be used with many of the commands described below.

SCREEN POSITIONING

Several function keys are used to position the screen at any desired point in the file.

f1 — +PAGE. Moves the screen 24 lines (one full screen) down in the file (but never past the end of the file).
ENTER n f1 moves n screenfulls down in the file.

f2 — +LINE. Moves the screen one line down in the file. Other ways to move the file one line down are (shift) up-arrow, and, if the cursor is on the bottom line of the screen, RETURN or down-arrow.
ENTER n f2 moves n lines down in the file.
ENTER f2 moves the screen down in the file so that the current cursor line becomes the first line on the screen.

f3 — GOTO. Moves the screen to the beginning of the file.
ENTER f3. Moves the screen so that the last line of the file appears on the screen.
ENTER n f3. Moves the screen so that line n of the file is at the top of the screen.

f4	−LINE. Moves the screen one line up in the file (but never past the top of the file). Other ways to move the file one line up are (shift) down-arrow, and, if the cursor is on the top line of the screen, up-arrow. **ENTER n f4** moves n lines up in the file.
f5	−PAGE: Moves the screen 24 lines (one screenful) up in the file (but never past the top of the file). **ENTER n f5** moves n screenfulls up in the file.
ENTER s 0	(the zero on the numeric pad): +SEARCH. Searches for the next occurrence of the string s in the file following the position of the cursor, and moves the line containing s to the top of the screen. If s is not found, beeps and does not change the screen.
0	(the zero on the numeric pad): A search forward is made for the last string given to a search command.
ENTER s .	(the period on the numeric pad): −SEARCH. Searches for the first occurrence of the string s in the file above the first line on the screen, and moves the line containing s to the top of the screen. If s is not found, beeps and does not change the screen.
.	(the period on the numeric pad). A search backward is made for the last string given to a search command.

You can add lines to the end of the text file by moving to the last line of text and typing on one of the lines below it. When the file is written out, PIE 1.6 will ignore any blank lines which follow the last line containing text.

If a command attempts to move far past the last text line (such as moving 100 pages down a short file), PIE 1.6 will move to a point where the last line is near the top of the screen. It is still possible to move further down, but not faster than one page at a time.

EDITING FUNCTIONS

The full power of PIE 1.6 lies in the special editing functions, which allow you to insert, delete and move lines of text, and other special capabilities. Two of these functions use a text buffer called the **temporary lines buffer**. You can place lines in this buffer using the PICK function. The contents of the buffer can then be put back into the file at any location with the PUT function.

The editing functions are:

DC	Deletes the character at the current cursor position. The characters to the right of the current cursor position are moved left to fill the space.
IL	Inserts a blank line at the current cursor line, moving the lines below the cursor down. **ENTER n IL** inserts n blank lines at the current cursor line.
DL	(shifted): Deletes the current cursor line. The contents of the current line are lost. **ENTER n DL (shifted)** deletes n lines (255 maximum) beginning with the current cursor line. The contents of the deleted lines are lost.

PICK — (red-square key): Places the current cursor line into the temporary lines buffer. The line is not deleted or altered.
ENTER n PICK places n lines (24 maximum), beginning with the current cursor line, into the temporary lines buffer. The lines are not deleted or altered.

PUT — (white-square key): Inserts the line(s) in the temporary lines buffer into the file at the current cursor line. Moves the current line and those below it down to make room for the inserted lines. The contents of the temporary lines buffer are not changed, and may be inserted again anywhere in the file.

ERASE — Erases all characters, beginning with the current cursor position, up to the end of the line.

Ctrl-K — QUOTE. Used to insert control characters in the file. The next character typed is entered on the screen as a control character. If it is a text character, the corresponding control character is entered instead. Control characters display on the screen in inverse video, but are otherwise treated as printing characters for editing purposes. Certain control characters, such as NUL (ctrl-@), tab, return and line feed may not be entered in this way.

MOVING TEXT BETWEEN LINES

PIE 1.6 is a text editor, and not a word processor. It is not intended to prepare text in a neat format for output to a printer. Thus, there are not many commands for moving around pieces of text lines.

PIE 1.6 is a convenient way to prepare text for input to a text formatter. In order to make PIE 1.6's share of this task easier, two features are included to help in entering and editing text, assuming the formatting is left to another program.

Ctrl-D — DIVIDE. Divides the current line at the cursor position. The characters at and to the right of the cursor are moved to the beginning of a new line inserted below the current line.

Ctrl-W — WRAP. Turns the **Word Wrap** mode on, and sets a right margin at the current cursor column (or column 76 if the cursor is in column 1). When the Word Wrap mode is on, typing past the right margin will cause a new line to be inserted, and the word(s) extending into the right margin will be moved down to the beginning of that line. The effect is to allow you to type words continuously, without ever worrying about hitting RETURN at the end of the line.

Pressing WRAP (ctrl-W) again turns Word Wrap mode off. Word Wrap mode is normally off when PIE 1.6 starts up, but PIE 1.6 can be patched to start up with Word Wrap mode on; see "Tabs and Other Patchables," Page 16.

Do not use Word Wrap while you are in the Insert Mode.

EXITING, SAVING, AND CHANGING DISKS

Ctrl-R

HDOS DISK REPLACE. Ctrl-R allows you to dismount and mount HDOS disks during an HDOS PIE 1.6 editing session. For example, the sequence "ENTER sy1: ctrl-R" will dismount the disk mounted in SY1:, request a new disk, mount it, and continue with the editing session.

Do not enter a REPLACE for a non-existent disk drive. Do not REPLACE Disks in SY0: unless Stand-Alone is in effect.

This function is necessary for HDOS versions of PIE 1.6 due to HDOS' disk mount/dismount format. It is not implemented in CP/M versions of PIE 1.6.

Ctrl-E

EXIT. Writes the edited file out, and returns to system command level.
ENTER s Ctrl-E does not change the original file, but writes the changed text out on the file named s instead, and then returns to the command level.
ENTER Ctrl-E aborts the editing session. PIE 1.6 will request confirmation, and, if it is received, will return to the command level. No files will be written out. All files will remain as they were at the start of the editing session, and all editing done during the session will be lost.

Ctrl-V

SAVE. Writes the edited file out, and then resumes editing at the top of the file.
ENTER s Ctrl-V does not change the original file, but writes the changed text out on the file named s instead. Then editing resumes at the top of the file.

If there is a file system error in writing out the file when an EXIT or SAVE function is invoked, PIE 1.6 gives the error message, does not exit, and returns to the editing state just before the EXIT command, without changing anything. Normally, PIE 1.6 writes the new version out without deleting the old version, so that some version of the file exists on the disk at all times.

If there is not enough room on the disk for both versions, PIE 1.6 asks if you want to delete the old version first. This can be dangerous if enough text has been added to the file so that the disk can not hold the new version even with the old version deleted. If you delete the old version and then find yourself in this situation, you can either: (1) delete enough lines from the version you are editing to make it fit; (2) save it as another file which already exists on the disk, deleting the old version of that file first; (3) exchange the disk for another with suitable free space (In CP/M — just swap disks, in HDOS swap disks by using the REPLACE (Ctrl-R) function); or (4) abort the session, losing both versions of the file. To minimize the chance of losing the file, PIE 1.6 will not exit until you do one of these four things.

MACROS: THE DO KEY

The DO key (the key marked with a blue square) is a user-definable macro key which can be easily programmed to perform many functions, including search and replace.

The DO key is used to record a macro, or sequence of function and text keys, as they are typed to perform an editing operation. Then the DO key is used again to repeat the macro operation once or many times.

The DO key commands are:

ENTER DO	(DO is the blue square key.) Begins recording the typed keystrokes. All typed commands and text appear normally on the screen, but in addition the keystrokes are recorded. This recording mode is terminated by striking the DO key again.
DO	Performs the recorded macro.
ENTER n DO	Performs the recorded macro n times. N must be a number.

Ordinarily, macro execution continues until the macro has been performed the specified number of times. However, if any error occurs the macro execution will terminate, and the error message will be displayed. Macro execution may also be halted by typing any key; the macro will stop and the typed key will be executed.

There is a limit on the number of keystrokes that can be recorded in a macro. The exact limit depends on the version of PIE 1.6 in use, and the mix of text and function keys in the macro. At least 250 text keys, or 83 function keys, may be used. When the limit is exceeded, an error message appears and the macro recording is aborted.

The operations which the DO key may be used to perform are limited chiefly by the imagination of the user. Several illustrations will be given here.

The most common DO operation is the search and replace function, which replaces all occurrences of one text string by another string. For example, to replace all occurrences of "man" by "person", type the following sequence:

ENTER DO	Begin recording.
ENTER man +SEARCH	Search to next "man".
per	Type "per" over "man".
IC	Enter insert mode.
son	Insert "son" after "per".
IC	Leave insert mode
DO	Terminate recording
ENTER 999 DO	Execute the macro 999 times.

If there are fewer than 999 instances of "man" in the file, the search will fail after the last instance has been replaced, and this error will halt the macro execution.

Sometimes you may want to inspect each instance of a string in a file, and decide for each one whether to replace it with another string. This is easily done by programming the DO key to perform the replacement, but not the search. Then the +SEARCH key may be pressed repeatedly to move to each occurrence of the string, and the DO key can be pressed to perform the replacement whenever that is desired.

Another useful operation is to indent a number of lines of text. Suppose the cursor is positioned at the first line of 50 lines of text, and you want to indent all 50 lines by placing ten blank spaces at the beginning of each line. You can do this by using:

IC	Enter insert mode.
ENTER DO	Begin recording.
" "	Type 10 blank spaces.
RETURN	Move to next line.
DO	Terminate recording
ENTER 49 DO	Insert spaces in next 49 lines.

MISCELLANEOUS IDEAS AND SUGGESTIONS

Error Conditions

Error conditions are indicated in one of two ways. If PIE 1.6 is asked to do something it can't do, like search for something that isn't in the file, or if an illegal function key is pressed, PIE 1.6 will sound the bell. If something special happens, like running out of memory for editing, a warning message will appear on the bottom line of the screen.

Free Space

In the lower right hand corner of the screen, PIE 1.6 displays the line number of the file on which the cursor is currently positioned, and the number of free characters remaining in the computer memory. As more text is entered, the number of free characters will decrease.

When this number drops below 2000, PIE will highlight it and display a warning message. If this happens, **you must exit almost immediately**. The characters on the screen, which may number as many as 1944, need to be stored in free space in order to exit, so when the free count goes under 2000 PIE is almost completely out of space.

If this happens, it is a good idea to break the file up into smaller files if possible. If you continue editing, PIE may abort and any editing done will be lost.

Exiting Your Terminal

Sometimes, the PIE 1.6 startup command is given, and then you decide not to edit a file (or can't remember the name of the file to edit). You may abort PIE 1.6 in this situation by hitting Ctrl-C in CP/M versions, or Ctrl-Z twice in HDOS versions. Once a file is displayed on the screen, neither of these conventions should be used to exit, as this may leave the terminal, keypad, and console driver in an unpredictable mode. Instead, you may abort the editing session by using ENTER Ctrl-E (See Page 12, "Exiting, Saving and Changing Disks").

Tabs and Other Patchables

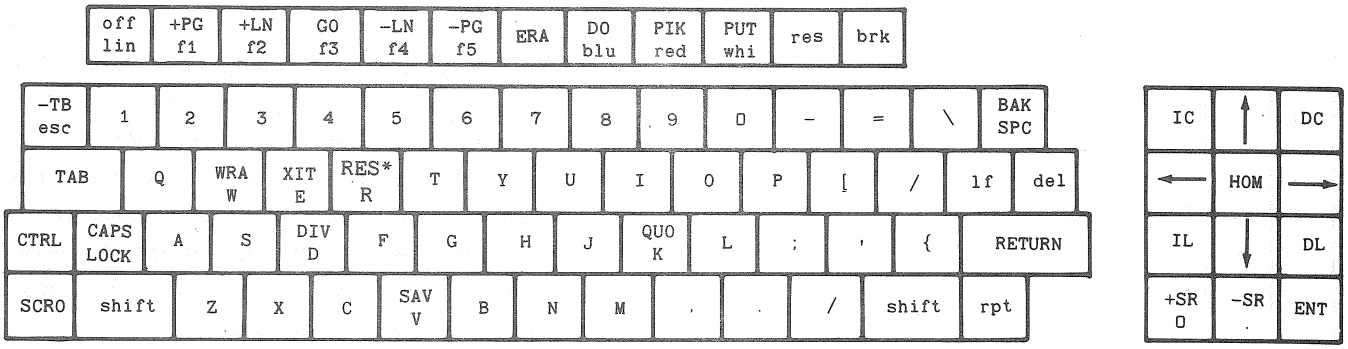
PIE 1.6 normally expands all tab characters to multiple spaces as part of the file save process. You may, however, patch PIE 1.6 so it conserves space on the disk by using tabs to represent two or more blanks ending on a tab stop. See Appendix A.

PIE 1.6 uses timing to detect the difference between the ESC key and the function keys. If you operate PIE 1.6 at terminal speeds below 1200 baud, the delay between characters may interfere with the proper reading of the function keys. Although we do not recommend operation of PIE 1.6 at such low speeds, you can do it if you patch the timing location shown in Appendix A. That location currently contains 0366 Octal (0F6 Hex). Try replacing it with a smaller number; 0350 Octal (0E8 Hex) should suffice for 300 baud operation.

Word Wrap is normally off when PIE 1.6 starts up. To patch it to be on, enter the column number at which to wrap in the memory location indicated in Appendix A.

In HDOS, patching is done using the PATCH program on the HDOS distribution disk. To patch under CP/M, use DDT and the SAVE command. See the appropriate manuals.

PIE 1.6 FUNCTION KEY LOCATION



*HDOS version only

FUNCTION KEY LABEL

Because the function key commands are so simple, labeling the keys may provide enough of a reference for normal use of PIE 1.6. You may find it helpful to paste labels with the function name on the front of each function key. Alternatively, this label strip may be cut out and placed above the top row of the keyboard to provide a quick reference to most PIE 1.6 function key commands.

+PAGE	+LINE	GOTO	-LINE	-PAGE		DO	PICK	PUT

Keypad:

0 +SRCH	. -SRCH

APPENDIX A

There are three options you may select by patching certain locations. They are:

1. TABS. As distributed, PIE 1.6 stores every character just as it appears on the screen, with TABs expanded to spaces. You may modify PIE 1.6 so groups of contiguous blank spaces in the text are represented as single TAB charaters at the time of file write. This "compression" feature conserves disk space. (Some printers do not recognize TAB characters and require blank spaces for "TAB Stops."

2. ESC. Certain function keys issue an ESCape character followed by one or more control code characters. PIE 1.6 is instructed to "wait" a short time after seeing an ESCape character to see if a control character comes along. The "time out" is set for baud rates above 1200. Should your terminal operate at a slower rate, you must increase the "time out" by lowering the ESCape timing byte.

3. Word Wrap. PIE 1.6, as distributed, signs on with Word Wrap off. You can modify PIE 1.6 to come up with Word Wrap on. To enable Word Wrap, the number of the column where the wrap will occur is patched into the memory location given below.

To modify a version of PIE 1.6, copy the desired version of PIE 1.6 from the distribution diskette. NEVER MODIFY A DISTRIBUTION DISKETTE, use it for creating programs on working diskettes only.

Use PIP to copy. That is, PIP *dev*:PIE.*ext*=*dev*:PIE*cpu*.*ext* ㊞, where *dev*: is the proper drive specifications for your system, *ext*. is the proper extension (.ABS for HDOS or .COM for CP/M), and *cpu* is the type of CPU in your computer (Z80 for H/Z-89 or 8080 for the H-8). ㊞ means press the RETURN key. For example, in CP/M, PIP A:PIE.COM=B:PIEZ80.COM ㊞ will create the working program PIE.COM on disk A: from Distribution program PIEZ80.COM, which is on disk B:. The same operation in HDOS; PIP SY0:PIE.ABS=SY1:PIEZ80.ABS ㊞, for disk SY0: and SY1: respectively.

The file DDT.COM must be on the system diskette in a CP/M system and the file PATCH.ABS must be on the HDOS system diskette in order to perform the patching.

The memory locations that you may modify for PIE 1.6 CP/M version (9/23/81) are:

PIEZ80.COM

Function	Memory Location	Current Value	New Value
TAB	0C80	018 (Save tabs as spaces on disk)	020 (Save tabs as tab characters on disk)
ESC	032D	0F6	0E0
Wrap	01FC	0FF	050 (Column 80)

PIE8080.COM

Function	Memory Location	Current Value	New Value
TAB	0F19	0C3 (Save tabs as spaces on disk)	0C2 (Save tabs as tab characters on disk)
ESC	039F	0F6	0E0
Wrap	0257	0FF	050 (Column 80)

To modify either or all of these functions, boot up a CP/M disk which contains the file PIP.COM and DDT.COM. (CR) will be used to indicate that you are to press the RETURN key.

1. Use PIP as mentioned previously to make a working copy of the proper PIE 1.6 distribution program for your system. Make PIE.COM the working copy's filename.

2. At the CP/M prompt >, type DDT PIE.COM (CR)

3. At the DDT prompt −, type R (CR)
   Type Snnnn mmm (CR), where nnnn is the memory location and mmm is the new value to be placed within. For instance, to make our working copy of PIEZ80.COM wrap at column 80, type S01FC 050 (CR). NOTE: 050 Hex is equivalent to 80 decimal.

4. Press the space bar.

5. To verify the change, type Dnnnn ⓒ, where nnnn is again the memory location being modified. The new value you had just changed in this location should appear in the screen memory dump display. In our example, type D01FC ⓒ. On the very next line, the location number 01FC will appear. To its right, 50, the new value of that location, should appear. The other numbers on that line and the following lines represent other locations and values, and can be ignored.

6. If you are satisfied, type Control C. (Hold the CTRL key and C down simultaneously.) At the system prompt ">", type SAVE nn PIE.COM (where nn=24 if you are patching the Z80 version or nn=27 if you are patching the 8080 version).

The memory locations that you may modify for PIE 1.6 HDOS version (9/23/81) are:

PIEZ80.ABS

Function	Memory Location	Current Value	New Value
TAB	055365	030 (Save tabs as spaces on disk)	040 (Save tabs as tab characters on disk)
ESC	044255	366	340
Wrap	043174	377	120 (Column 80)

PIE8080.ABS

Function	Memory Location	Current Value	New Value
TAB	060215	303 (Save tabs as spaces on disk)	302 (Save tabs as tab characters on disk)
ESC	045037	366	340
Wrap	043327	377	120 (Column 80)

To modify either or all of these functions, boot up an HDOS disk that contains the files PIP.SYS and PATCH.ABS. ⓒⓡ will be used to indicate that you are to press the RETURN key.

1. Use PIP as mentioned previously to make a working copy of the proper PIE 1.6 distribution program for your system. Make PIE.ABS the working copy's filename.

2. At the HDOS prompt >, type PATCH ⓒⓡ. PATCH will sign on and ask for the file name.

3. Type PIE.ABS. PATCH will respond... ADDRESS?

4. At this point, type in the appropriate address of the value that you want changed. For instance, suppose you want to turn Word Wrap on in the 8080 version of PIE. When you are asked for the address, enter 43327 ⓒⓡ. PATCH will display the current value, which should be 377. Type 120 ⓒⓡ. Now type Control D. (Hold the CTRL key and D down simultaneously.) NOTE: 120 Octal is equivalent to 80 Decimal (Column 80).

5. Make any further modifications as necessary following the same procedure. When you have completed, type Control D again. This will save your modifications to the PIE.ABS program.

PIE 1.6 is a two-dimensional, cursor-based screen editor designed specifically for use with CRT systems. In PIE 1.6, the terminal screen acts as a "window" displaying a portion of the file being edited. PIE 1.6 uses function keys, performing simple operations which are immediately reflected in the file as displayed on the screen. What you see on the screen is what is in the file.

The screen window may be positioned to display any part of the file. Cursor motion keys allow you to position the cursor anywhere on the screen, and you can type changes to the file right on the screen at the cursor position. Other features include:

- Character and line insert and delete anywhere on the screen.
- String search forward and backward.
- Macro facility for search/replace and other functions.
- Move and copy single and multiple lines.
- Insert mode for inserting text into a line.
- Scrolling of text in the window.
- Append and clear to end of a line.
- Continuous typing mode ("hot zone") for rapid text entry.

*CP/M is a registered trademark of Digital Research Inc.