NEVADA
EDIT ™

ELLIS COMPUTING ™
SOFTWARE TECHNOLOGY

NEVADA EDIT(tm)

Users' Reference Manual

Copyright (C) 1981 by Ellis Computing, Inc.

Contributing Authors for the CP/M(r) version:

John A. Starkweather, Ph.D.
Chuck Ellis

Ellis Computing, Inc.
3917 Noriega Street
San Francisco, CA 94122

COPYRIGHT

TRADEMARKS

NEVADA COBOL(tm), NEVADA FORTRAN(tm), NEVADA PILOT(tm), NEVADA EDIT(tm), NEVADA BASIC(tm) and Ellis Computing(tm) are trademarks of Ellis Computing, Inc.　CP/M is a registered trademark of Digital Research, Inc.　KAYPRO is a trademark of Non-linear Systems.　Xerox 820 is a trademark of Xerox Corp.　TRS-80 is a registered trademark of Tandy Corp.　Apple II is a trademark of Apple Computer, Inc. Osborne is a registered trademark of Osborne Computer Corp. Heath/Zenith is a trademark of Heath Corp.　IBM is a registered trademark of International Business Machines, Corp.

DISCLAIMER

NEVADA EDIT

A TEXT EDITOR FOR CP/M

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

SECTION 1

INTRODUCTION

The EDIT command initiates execution of a text editor program intended for use with the CP/M operating system. When the EDIT command is given, the named file is read into memory and the first screenful of lines are displayed. The user may then either edit the file using single-letter control character commands, or, in the case of a new file, simply begin entering text as though on a typewriter, with all typed material appearing on the screen and all editor commands available for making corrections.

The single-letter control character commands make it possible to move the cursor anywhere on the screen, or to move through the file so that any desired screenful of lines are visible. Characters or entire lines may be inserted, deleted, or moved from one part of the file to another. Text may be scrolled up and down, i.e., it is possible to move either backwards or forwards in a file and to stop this process when the desired part of the file is visible. There are also provisions for searching a file for particular strings of characters, replacing every occurrence of a given string with another string, and inserting another whole file into the text being edited.

At the conclusion of editing, the file may be written back to itself as an update, or it may be written to another named file, if several slightly different versions of one file are needed, or if a back-up of the same file is desired. Editor commands may also be used to divide a file into two or more files, if the original file becomes too long. The entire edited file is in memory at once, so it is even possible to choose NOT to save the file after editing. (This feature might be used if you made extensive corrections and suddenly realized they were not appropriate.)

This editor accepts lines containing no more than the screen width as specialized for your machine. Lines longer than the specialized line length are broken into two or more lines as required. Each continuation line begins with a (+) plus sign (user changeable). When the file is written back to the disk the (+) signs are removed and the lines concatenated. There is no minimum line length.

SECTION 2

OPERATING PROCEDURES

HARDWARE REQUIREMENTS

1. 8080/Z80/8085 microprocessor.
2. A minimum of 32K RAM.
3. Any disk drive.
4. CRT or Video display and keyboard.

SOFTWARE REQUIREMENTS

Digital Research Corp's CP/M operating system Version 1.4 or 2.2.
CP/M is a registered trademark of Digital Research Inc.

FILES ON THE DISTRIBUTION DISKETTE

NVEDIT.COM   is the text editor program.

NVEDIT.ERR   is the error message file and must be on the default drive during configuring.

NVEDIT.PRN   is the assembly language source code that can be used for special default modifications.

READ-ME.TXT contains new information not in the manual. *** Important read this file by typing:
                     TYPE READ-ME.TXT

EDTKEY.COM   is used to reassign key functions.

FILES NOT ON THE DISKETTE

EDIT.COM   this file will be created for your CRT.

GETTING STARTED

If the master disk is not write protected, do it now!

1. NEVADA EDIT is distributed on a DATA DISK without the CP/M operating system. There is no information on the system tracks, so don't try to "boot it up", it won't work!

2. On computer systems with the ability to read several disk formats, such as the KayPro computer, the master diskette must be used in disk drive B. This is the procedure, whenever the distribution disk is single density

and the normal machine format is double density. In general double density machines can read single density diskettes, provided the proper software disk drivers are in place. These software disk drivers may be in the CP/M bios or may be automatically inserted in the bios by software packages such as SETUP or UNIFORM.

3. Do not try to copy the master diskette with a COPY program! On most systems it won't work. You must use the CP/M PIP command to copy the files.

4. You must first, prepare a CP/M system's diskette for use as your NEVADA EDIT operations diskette. On 5 1/4 inch diskettes you may have to remove (use the CP/M ERA command) most of the files in order to make room for the EDIT files. None of the CP/M files are needed for NEVADA EDIT, but PIP.COM and STAT.COM are useful if you have the space. You may want to do a CP/M STAT command on the distribution disk so you will know how much space you need on your operational diskette. For more information read the CP/M manuals about the STAT command.

5. Then insert the newly created CP/M diskette in disk drive A, and insert the NEVADA EDIT diskette in drive B and type (ctl-c) to initialize CP/M. Now copy all the files from the EDIT diskette onto the CP/M diskette:

         PIP A:=B:*.*[VO]

If you get a BDOS WRITE ERROR message from CP/M during the PIP operation it usually means the disk is full and you should erase more files from the operational diskette.

On single drive systems, use the single drive copy program that comes with your hardware system to copy the files. Consult your hardware dealer for more information.

At this point, put the NEVADA EDIT diskette in a safe place. You will not need it unless something happens to your operations diskette. By the way, back up your operations diskette with a copy each week! If your system malfunctions you can then pat yourself on the back for having a safe back up copy of your work.

Now, boot up the newly created NEVADA EDIT operations diskette. Notice that CP/M displays the amount of memory for which this version of CP/M has been specialized.

Editing should be done on the same CP/M version or one of larger memory.

Next, configure NEVADA EDIT for your CRT or Video Display by
issuing the Command:

A>NVEDIT<CR>

The program will present you with a choice of Terminals and
prompt your next steps. EDIT.COM will be created on the
default drive by the process, so make sure their is 16K disk
space available. This is the EDIT program configured to
your equipment.

If your terminal is listed, you need do no more than press
one key in order to proceed. If your terminal is not
listed, press "W" for unknown, and you will be asked a
series of questions about its characteristics. See Section
5 for an example of the interaction involved in the
configuring process.

NVEDIT Version 3.0 configuring
Copyright (C) 1981, 1982 Ellis Computing, Inc.
@ ANSI MODE TERMINAL
A ADVANTAGE
B APPLE COMPUTER, 40 COLUMN DISPLAY   (sends soroc i/q 120)
C APPLE COMPUTER + VIDEX 80 COLUMN BOARD   (soroc i/q 120)
D BEEHIVE 150 or CROMENCO 3100
E COMMODORE 64
F FREEDOM 100
G HAZELTINE 1400 SERIES
H HAZELTINE 1500 SERIES
I HEATH H19/H89 OR ZENITH Z19/Z89
J HEWLETT-PACKARD 2621

Type a single letter to select terminal.
<Carriage Return> for more terminals

K IBM PERSONAL COMPUTER+ BABY BLUE CARD
L INFOTON I-100
M LEAR-SEIGLER ADM-3A
N LEAR-SEIGLER ADM-31
O MICROTERM ACT-IV
P OSBORNE I
Q PERKIN-ELMER 550 (BANTOM)
R PROCESSOR TECHNOLOGY SOL OR VDM
S SOROC IQ-120/140
T SUPERBRAIN   (control-w reassigned)
U TELEVIDEO 950
V TRS-80, MOD II (P. & T. CP/M)
W NONE OF THE ABOVE
Type a single letter to select terminal.
<Carriage Return> for more terminals R

ENTER FILE TYPE TO BE USED
AS DEFAULT (3 CHARACTERS)   CBL   (for COBOL)

DEFAULT TAB STOPS ARE SET FOR COBOL.
DO YOU WISH TO CHANGE THEM? (Y/N)   N

CONFIGURING OF EDIT.COM IS COMPLETE.

EDIT.COM SAVED ON THE DEFAULT DRIVE.

If your computer is an Apple, you must run the Microsoft
program to configure the Softcard (TM of Microsoft): MBASIC
CONFIGIO Select the Soroc IQ-120 as your software terminal.
Also, using the CONFIGIO, delete the reasignment of all
control keys.

If you are using the Videx 80 column board on your Apple,
the Control-A key is used as an upper-lower case switch.

The file NVEDIT.PRN contains information for further
configuring to meet special needs or to change keys used for
special functions. For example, some models of Televideo
terminals that do not respond to codes for insert line or
delete line require that you use this method to obtain these
functions. Select the Televideo choice when you run NVEDIT
and if these functions do not work, then use DDT to place a
zero in the IL and DL locations. This will force a
rewriting of the screen for these functions.

The NVEDIT.COM file is no longer needed on the disk and can
be removed if you need the disk space.

When you have a version working on your computer and
terminal, you may run the additional program EDTKEY if you
wish to change the keyboard keys used for specific edit
functions. The program will prompt you how to proceed. See
a sample run of this program in Section 6 of this manual.

EXECUTING EDIT

This section provides a general overview of the operation of
EDIT.

FORMAT-1:

                EDIT from-file<CR>

FORMAT-2:

                EDIT from-file [to-file]<CR>

DESCRIPTION:

 where:

        file = [unit:]file-name[.file-type]

        [ ] = optional parameters

        unit: = disk drive unit letter. If this parameter is
              not included, the default drive is used.

        from-file = the file that contains the text to be
                  edited. This file will be loaded into
                  memory when the EDIT command is entered.

        to-file = the file to which the edited file will
                  be written at the conclusion of
                  editing. If no [to file] is specified,
                  the file is written back to the
                  from-file.

        file-type = the CP/M file type. If not present the
                  default file type as specified during
                  configuring will be used.

The editor uses all the memory up to the bottom of CP/M.
Warning, if you are using several different machines each
with the same amount of memory, sometimes the bottom of CP/M
will be lower on one machine because of blocking routines in
CP/M.

STARTUP

A>EDIT from-file<CR>

The from-file is read from the disk into memory for editing.
If the named file does not exist, the editor asks whether a
new file by that name should be created:

        FILE IS NON-EXISTENT. CREATE?

If the character 'Y' is typed, the file will be created with
the specified type or the default type e.g.(.CBL), and the
new (empty) file will be loaded into memory. (This
procedure is one of the ways of creating a new file in
CP/M). If a character other than 'Y' is typed, control will
return to CP/M.

When a file is loaded successfully, the following
information is presented:

BYTES FREE 12345 DECIMAL
C/R TO CONTINUE

As soon as the carriage return (C/R) is entered, the first
screenful of the file is displayed, with the cursor at line
1 and character 1 (column 1). If the file contains just a
few lines, the remainder of the screen will be filled with
blank characters. The file is now ready for editing.

MEMORY USAGE

The EDIT program is read into memory starting at location
100H. The from-file is then read into a file area beginning
just after the EDIT program code, and ending at the
top-of-memory as indicated by CP/M.

The entire from-file must fit into available contiguous
memory. If an edit operation would increase the size of a
file so that the text would no longer fit in the text area,
a message appears on the cursor line:

FULL - TYPE CONTROL Q

The offending operation will not be performed. After typing
the Control and Q keys simultaneously, the user should
either shorten the file by removing text, or leave the
editor by typing the Control and K keys. If the second
alternative is chosen, the file will be written back onto
the disk; the editor can then be used to create a
continuation file, or to split the large file into two
smaller files, so that further additions can be made to
each. (The discussion of EN and ST, later, contains a
step-by-step procedure for spliting a file.)

TERMINATION

There are two ways to terminate the editing of a file:

1. Control and K keys terminate EDIT and update the file.

If a to-file was specified in the EDIT command, the contents
of the text area can be written to that file or to some
other file. If the to-file parameter was omitted from the
EDIT command, the from-file can be updated with the edited
material. (Remember that when you write to a file, you
write over the previous contents of that file.) In either
case, the user is given the option of what file is to be
written.

2. Control and B keys followed by AB <CR> aborts the EDIT program.

If you do not wish to save the file in memory by writing it to the to-file, type Control B and then enter "AB" followed by a carriage return.

CREATING YOUR FIRST FILE A walk through.

On your system type:

        EDIT SAMPLE <CR>

the following message will apear:

NEVADA EDIT (C) COPYRIGHT 1981,1982
ELLIS COMPUTING, INC.
REV 3.0 (0)
terminal name

A:SAMPLE CBL IS NON-EXISTENT. CREATE (Y/N)? y type y <cr>

BYTES FREE 33196 DECIMAL   (space available on your system)
C/R TO CONTINUE   type <cr>

at this point the entire screen is blank

so type your name or something.
and type a carriage return for next line.

THE GOOD GUYS AT ELLIS COMPUTING<cr>   an example.

now type Control-K
the following message appears:

*** EDITOR EXIT 00028 DECIMAL FILE COUNT

OK TO WRITE TO "A:SAMPLE CBL"? Y you type y

OK TO QUIT (Y/N)? Y you type y

and we are back to CP/M. the file can be displayed by typing:

TYPE SAMPLE.CBL<CR>

SECTION 3

EDITOR COMMANDS

In the introduction to this manual, it was explained that the editor can be used for two purposes. By entering the EDIT command and the name of a file that does not yet exist, one can create that file and use the editor to enter text (e.g., a program, a letter) for the first time. In this context the control character commands permit corrections to be made conveniently; it is possible to retype words, restructure paragraphs or other segments of text, without having to retype the surrounding material. The other use of the editor is to make alterations in existing text files (probably created in the same editor on an earlier occasion).

Below is a list of the control keys used by the editor. A more complete description of each command is given after the list. To enter a control character, type the Control key and the named letter key at the same time.

CONTROL KEYS:

Control E - move the cursor up one line
Control X - move the cursor down one line
Control S - move the cursor left one character
Control D - move the cursor right one character
Control I - move cursor to next tab stop
Control Z - scroll file up one line
Control W - scroll file down one line
Control C - scroll file up one half screen
Control R - scroll file down one half screen
Control Q - home cursor to line 1, column 1
             also escape from FULL
Control A - cursor to mid screen, column 1
Control V - toggle insert character mode; ON/OFF
Control ̄   - repeat next command n times (underline)
Control G - delete character under cursor
Control N - insert line above cursor
Control Y - delete line
Control \ - split line
Control T - delete from cursor to end of line
Control F - initiate string search mode
Control L - continue search for string
Control B - set TAB/BLOCK MOVE/NULL/LL/FL/ST/EN/PA/IF/TS/AB
Control U - execute TAB / BLOCK MOVE
Control K - exit editor
Control M - same as RETURN (below)

OTHER KEYS:

TAB   - move cursor to next tab stop (Control I)
RETURN   - insert line below cursor (Control M)

CURSOR POSITIONING   CURSOR POSITIONING

FUNCTION: To position the cursor on the screen.

COMMANDS:

                    Control E
                    Control X
                    Control S
                    Control D

DESCRIPTION:

The cursor positioning commands do not affect the contents of the file in memory; their purpose is to move the cursor to the desired position on the screen, so that insertions or changes may be made there.

        Control E move cursor up one line
        Control X move cursor down one line
        Control S move cursor left one character
        Control D move cursor right one character

Note that the keys S, D, E, X form a diamond on the keyboard.

Some keyboards also contain special cursor control keys such as the left arrow, right arrow, up arrow and down arrow. These keys are used without the Control key to move the cursor left, right, up or down, as above, but their use may require a configuring change to EDIT. See EDTKEY in the back of this manual.

The cursor marks the position at which the next typed character will appear on the screen (and so, also, in the file in memory). In general, any character entered from the keyboard, except a control character, will replace the character at the cursor position. (There is a special command that makes it possible to insert text at the cursor position without "typing over" any other characters. That command, the INSERT CHARACTER command, will be described later.)

CURSOR HOME   **CURSOR HOME**

FUNCTION: To move the cursor to the home position.

COMMAND:

                    Control Q

DESCRIPTION:

This command causes the cursor to be moved to line 1 and character position 1.

This command has another unrelated function described earlier in the discussion of how the editor uses memory. It is used as an escape from the memory FULL condition.

Because these functions are useful in two completely different contexts, there is never any ambiguity as to the user's intentions.

CURSOR MID SCREEN   **CURSOR MID SCREEN**

FUNCTION: To move the cursor to mid screen.

COMMAND:

                    Control A

DESCRIPTION:

This command causes the cursor to be moved to the mid screen line and character position 1.

SCREEN SCROLLING

FUNCTION: To scroll the screen up or down.

COMMANDS:

              Control Z
              Control W
              Control C
              Control R

DESCRIPTION:

Screen scroll commands are provided to allow the file to be "rolled" through the screen area until the desired file line is reached. Scrolling UP moves the file upward across the screen, so that a LATER part of the file becomes visible; scrolling DOWN moves the file downward across the screen, so that an EARLIER part of the file becomes visible.

        Control Z   scroll up one line
        Control W   scroll down one line
        Control C   scroll up one half screen
        Control R   scroll down one half screen

NOTE:

The scrolling of one half screen can be changed to a full screen by changing one byte in the common table. Run the EDTKEY.COM program to make the change.

TAB   TAB

FUNCTION: To move the cursor to the next tab stop.

COMMAND:

                    TAB or Control I

DESCRIPTION:

This command moves the cursor to the first tab stop to the right of the present cursor position, or to the first character position in the line, if there are no tab stops to the right of the cursor. Tab stops are set and cleared by means of the TS command (one of the "extended commands" described later).

NOTE:

During the initial configuration procedure the default tab stops are established by the user. If you will be writing NEVADA COBOL programs you may want to set the tab stops at col 6, 10, 12, 14, 16, etc., or if you prefer the ANSI-74 COBOL tab stops, they are 8, 12, 14, 16, etc.

FUNCTION: To search the file for occurences of a string
          of characters.

COMMAND:

                    Control F

DESCRIPTION:

When this command is given, the screen is cleared and a
colon (:) appears. At this point the editor expects to
receive an input line consisting of one to thirty-nine
characters and a carriage return. The input line becomes
the search string. Any occurrence of the string, regardless
what characters precede or follow it, will be recognized as
a match. Therefore, it is unnecessary to enter more
characters than are required to define the text uniquely,
i.e., "the qu" can be used to locate a line containing "the
quick brown fox.".

After accepting the input string, the editor searches the
file, beginning at the first line of the file and ending at
a matching string or the end of the file, whichever is
encountered first. Upon finding a matching string, the
editor positions the line containing the match at the first
line on the screen. If no match is found before the end of
the file, the first screenful of lines of the file are
displayed. If a match is found, the search may be continued
to the next match of the same string; see the Control L
command on the next page.

To search the entire file, type Control F, and enter the
search string. If the Control L command is used to search
from each string match to the next, all occurrences of the
input string will eventually be found.

NOTE: This command is very useful for correcting errors in
COBOL and other programs. If the compiler indicates an
error at line number 1234, you could use the search command
to locate and display that line number, and then make the
correction.

FUNCTION: To continue a string search.

COMMAND:

                    Control L

DESCRIPTION:

After a string has been located with the Control F command,
the Control L command may be used to search for the next
occurrence of the same string. This command causes the
editor to start searching with the first line below the
current cursor line and continue until a matching string is
found or until the end of file is reached. The Control L
command may be given as often as is desired.

INSERT CHARACTER   **INSERT CHARACTER**

FUNCTION: To toggle the insert character mode switch
            (on-off-on...).

COMMAND:

        Control V

DESCRIPTION:

Any input from the keyboard, other than an editor command,
is placed in the file in one of two modes. These modes,
normal and insert, are determined by the Control V command.
Each entry of Control V changes the input mode, i.e., if
insert mode is on when command is given, the command will
turn it off.

When insert mode is OFF, as it is when the editor is loaded
and whenever the cursor is moved to a new line, each input
character is placed at the cursor location, and the cursor
moves to the right one place. When insert mode is on,
however, each character is inserted into the file at the
cursor position, moving the character at that location, and
any characters to the right of the cursor, one position to
the right. If the line resulting from such an insertion is
more than the number of columns on a screen line, all
characters exceeding the line length are lost.

DELETE CHARACTER   **DELETE CHARACTER**

FUNCTION: To delete the character under the cursor.

COMMAND:

        Control G

DESCRIPTION:

The delete character command removes the character at the
current cursor position and moves each character to the
right of the cursor one position left.

INSERT LINE ABOVE   **INSERT LINE ABOVE**

FUNCTION: To insert a line above the current cursor line.

COMMAND:

        Control N

DESCRIPTION:

The line insert control moves the file down one line from the cursor, and inserts a new blank line at the old cursor position.   The cursor is moved to the first character position of the new line.   Use this command to insert a new line 'above' the current line.

INSERT LINE BELOW   **INSERT LINE BELOW**

FUNCTION: To insert a line below the current cursor line.

COMMAND:

        RETURN or Control M

DESCRIPTION:

Carriage return scrolls up one line and inserts a blank line in the file.   The cursor is moved to the first character position of the new line.   Use RETURN to insert a line 'below' the current cursor position.   No characters on the current line are deleted.

This command is used most frequently, for example, when entering a NEVADA COBOL program.   At the end of each statement you normally "hit" the return key (enter on some machines) to create the next line and position the cursor to the first position of the new line.

SPLIT LINE   **SPLIT LINE**

FUNCTION: To move the remainder of a line to a new line.

COMMAND:

        Control \

DESCRIPTION:

This control moves the character at the cursor position and
all characters in that line to the right of the cursor to a
new line. After the characters have been moved, the cursor
is placed at the first character position of the new line.

DELETE LINE   **DELETE LINE**

FUNCTION: To delete the current cursor line.

COMMAND:

        Control Y

DESCRIPTION:

This control removes the current cursor line from the file.

NOTE:

Position the cursor anywhere on the line to be deleted by
using the cursor control keys. Then type the Control Y keys
and "behold" the line is gone and the text is scrolled up.

BLANK REMAINING LINE   **BLANK REMAINING LINE**

FUNCTION: To blank the remaining part of a line.

COMMAND:

      Control T

DESCRIPTION:

This control deletes the character at the cursor position
and all characters in that line to the right of the cursor
(right truncation). After the characters have been deleted,
the cursor is placed at the first character position of the
following line.

REPEAT   **REPEAT**

FUNCTION: To repeat a command.

COMMAND:

      Control _ (underline)

DESCRIPTION:

This command has three parts: 1) the Control and _ keys
typed together, 2) the command to be repeated (a combination
of Control and some other key), and 3) a character
representing the number of times the command should be
repeated. The three parts of the command should be entered
in order, without blanks between them. Enter a carriage
return only after the third item in the command.

During the execution of a Control _ command, the cursor
disappears from the screen. If you are editing a file and
the cursor unexpectedly disapears, assume that you have
given the command by accident, and enter a repeat command
that will not affect the contents of your file, e.g., repeat
the Control S command once.

The editor determines how many times a command will be
repeated by considering the third item of the command with
an ASCII bias of 30 Hexadecimal. Thus, the characters 1
through 9 are taken to represent the corresponding numerical
values; to designate a numerical value greater than 9, type
the character whose hexadecimal ASCII value is 30 greater
than the number you want to represent.

EXAMPLE:

To delete 9 characters from the current cursor position,
type the following line (without the intervening blanks):

Control _ Control H 9 <CR>

FUNCTION: To exit from the editor.

COMMAND:

                        Control K

DESCRIPTION:

This command is the normal means of exit from the editor.

When the Control and K keys are typed simultaneously, the
following information appears on the screen:

12345 DECIMAL COUNT   ...byte count of file

OK TO WRITE TO "FILE-NAME"?

Type 'Y' to write to the to-file. If the command was
entered accidentally or if the user wishes to write to
another file the user can type 'N' and be asked:

ENTER NEW FILE NAME OR CR TO CONTINUE?

If a carriage return is typed the editor will return to the
text area. The editor will re-display the file with the
cursor and file position just as it was before the exit
command was given.

If a to-file was specified, the contents of the text area is
written to that file. If the named to-file does not exist
on the disk, it is created with the specified type or .CBL,
the default type. If the to-file parameter was omitted from
the EDIT command, the from-file is updated with the edited
material. (Remember that when you write to a file, you
write over the previous contents of that file.)

After a file has been written, the question is asked:

OK TO QUIT (Y/N)?

Type 'Y' to return to CP/M. Typing 'N' will cause return to
editor operation with re-display or the file.

FUNCTION: To make additional commands available.

COMMAND:

                        Control B

DESCRIPTION:

The function of the Control B command is really to make
eleven other commands available. Three of these commands
determine the function to be associated with Control U
(until that function is altered by another Control B), one
command determines the tab stops to be considered by Control
I, and the other seven are direct commands. When the
Control and V keys have been entered, the screen is cleared
and a colon (:) appears. The user is expected to type the
code for the desired function immediately after the colon.

EXIT - ABORT   **EXIT - ABORT**

FUNCTION: To abort an EDIT and return to CP/M.

COMMAND:

       Control B

    :AB <CR>

DESCRIPTION:

If the user does not wish to save the file in memory by writing it to the to-file, the Control B keys should be typed.   After the colon (:) prompt is displayed on the screen, type AB and a carriage return.   When this command is given, the question

     ABORT?

appears on the screen. If the character 'Y' is typed in answer, control is passed back to CP/M and no file update occurs.   The from-file still resides on disk, just as it did before the edit.

If the command was entered accidentally, the user can type another letter and return to the text area. The editor will re-display the file with the cursor and file position just as it was before the abort command was given.

VARIABLE TAB   **VARIABLE TAB**

FUNCTION: To set Control B for the TAB function.

COMMAND:

       Control B

    :TA <CR>

DESCRIPTION:

When Control U is set for the TAB function, the Control U command will expect one more keystroke to be typed in by the user.   This value will be used to tab the cursor to the corresponding character position in the cursor line.   To indicate a given character position, type the character whose ASCII value is 30 Hex greater than the value you want to represent.   (The numbering of character positions starts at 1, so type 1 to place the cursor over the leftmost character in a line, or A = col 17,   Z = col 42,   d = col 52,   p = col 64,   z = col 74).

When the EDIT program is loaded, the Control U command is set to perform the tab function.   If either block move or null mode has been set since the program was loaded, type TA or TAB (and an carriage return) after the colon.   The file area will immediately be displayed again, with file and cursor positions just as they were when the Control and B keys were typed.

NOTE:

There are actually two TAB functions within the editor.   The one above, and one that uses the TAB key or Control-I and tabs according to a table that is user changeable.

BLOCK COPY   **BLOCK COPY**

FUNCTION: To set <u>Control</u> U for the block copy function.

COMMAND:

      <u>Control</u> B

    :BC <CR>

DESCRIPTION:

After <u>Control</u> B is typed and the colon (:) prompt is
displayed on the screen, type BC and a carriage return to
set the <u>Control</u> U command to copy blocks of text from one
position to another within the file.   The procedure for
copying blocks is as follows:

1) If you have not turned on the block copy <u>Control</u> U
option, do so now (as above).

2) Insert a new blank line above the first of text to be
copied, and type the letter 'F' (first) in the FIRST
character position of the new line.

3) Next insert a new blank line below the last line of text
to be copied, and type the letter 'L' (last text to be
moved), in the FIRST character position of the new line.

4) Now, go to the line location of the file where the block
of text lines is to be copied, and insert a new blank line
there. Type the letter 'I' (insert) in the FIRST character
position of the new line.

5) Now type <u>Control</u> U

The block of lines inclosed within the 'F' and 'L' will be
copied from its position in the file, and duplicated in the
file at the text as the first (top) line of the screen.   The
'F', 'L', and 'I' lines will be removed from the file.

BLOCK MOVE   **BLOCK MOVE**

FUNCTION: To set <u>Control</u> U for the block move function.

COMMAND:

      <u>Control</u> B

    :BM <CR>

DESCRIPTION:

After <u>Control</u> B is typed and the colon (:) prompt is
displayed on the screen, type BM and a carriage return to
set the <u>Control</u> U command to move blocks of text from one
position to another within the file.   The procedure for
moving blocks is as follows:

1) If you have not turned on the block move <u>Control</u> U
option, do so now (as above).

2) Insert a new blank line above the first of text to be
moved, and type the letter 'F' (first) in the FIRST
character position of the new line.

3) Next insert a new blank line below the last line of text
to be moved, and type the letter 'L' (last text to be
moved), in the FIRST character position of the new line.

4) Now, go to the line location of the file where the block
of text lines is to be moved, and insert a new blank line
there. Type the letter 'I' (insert) in the FIRST character
position of the new line.

5) Now type <u>Control</u> U

The block of lines inclosed within the 'F' and 'L' will be
removed from its old position in the file, and inserted into
the file at the text as the first (top) line of the screen.
The 'F', 'L', and 'I' lines will be removed from the file.

NO FUNCTION   **NO FUNCTION**

FUNCTION: To set <u>Control</u> U for no function.

COMMAND:

        <u>Control</u> B

    :NU   <CR>

DESCRIPTION:

Typing this command in response to the <u>Control</u> B prompt will
turn the <u>Control</u> U function off, i.e., <u>Control</u> U will have
no effect if typed.

FIRST LINE   **FIRST LINE**

FUNCTION: To set the cursor to the first line of the file.

COMMAND:

        <u>Control</u> B

    :FL <CR>

DESCRIPTION:

The command causes the file to be displayed on the screen
with the first line of the file at the top of the screen.

FUNCTION: To set the cursor to the last line of the file.

COMMAND:

Control B

:LL <CR>

DESCRIPTION:

This command causes the file to be displayed on the screen
with the last line of the file at the bottom of the screen.

FUNCTION: To set the beginning of the file.

COMMAND:

Control B

:ST <CR>

DESCRIPTION:

This command deletes all text above the cursor line, so that
the cursor line becomes the first line of the file. Follow
these instructions carefully to use ST:

1) Position the cursor to the desired line.

2) Type Control B (colon prompt will appear).

3) Type ST and a carriage return. The file will be
displayed with the new first line at the top of the screen.

All information above the cursor line will be lost if the
file is written back onto the disk after this operation! If
you have already given the ST command and decide you really
do not want to have lost everything above the cursor, type
the Control B keys and AB to abort the edit operation, and
re-load the file with a new EDIT command. (If you abort EDIT
in this manner, you will lose ALL alterations you have made
in the file since you last loaded the EDIT program.)

END FILE   END FILE

FUNCTION: To set the end of the file.

COMMAND:

        Control B

    :EN <CR>

DESCRIPTION:

This command deletes the cursor line and all text below it.

1) Position the cursor to a line ONE LINE BELOW the line
that you want at the end of the file.

2) Type Control B (colon prompt will appear).

3) Type EN and a carriage return. The last lines of the
file will be displayed on the screen.

Just as with the ST command, the material that you delete
with EN will no longer exist in the file when you write it
back out to the disk.

NOTE: how to divide a larger file into smaller files

1) Enter the EDIT command, specifying the large file as the
from-file and creating a new to-file.

2) Set the cursor in the middle of the file, enter the
Control B command, and use EN so that only the first of the
previous file remains in the file area.

3) Enter the Control K command and write the contents of
the file area to the new to-file.

4) Enter the EDIT command, specifying the large file as the
from-file and creating a second new to-file.

5) Set the cursor to the same position as in step 2, enter
the Control B command, and use ST so that only the second
half of the previous file remains in the file area.

6) Enter the Control K command and write the contents of
the file area to the second new to-file.

7) Erase the large file using the CP/M ERA command.

REPLACE PATTERN   REPLACE PATTERN

FUNCTION: To search for and replace or delete text patterns
within a file.

COMMAND:

        Control B

    :PA R[S] <CR>
    :PA D[S] <CR>

DESCRIPTION:

This command allows the user to search for and replace or
delete text patterns within a file.

There are variants of the pattern replacement command:

    PA RS - Selective Pattern replacement
    PA R - Pattern replacement
    PA DS - Selective Pattern Deletion
    PA D - Pattern Deletion

After either of the first two variants is typed, the pattern
program will display:

PATTERN:

Type the pattern for which to search the file, and then type
the RETURN key. The program will display:

REPLACE:

Type the replacement pattern, and then type the RETURN key.

The DELete key may be used to correct an error in the typing
of a pattern.

Upon finding a pattern match in the file, the first form of
the pattern command will display the line in which the match
was made and position the cursor directly below the first
character of the matched text. The user now has the option
of allowing the replacement to occur on this line (type any
character other than 'N') or leaving the line as it is (type
the letter 'N'). Use this form if you are not sure whether
to replace all occurrences of the pattern in the file. If
you were to replace all occurrences of 'POP' with 'PUSH',
and a line of the text contained the label 'POP1', the
replacement of the character 'POP' would produce 'PUSH1'.

If the form PA R is used, all replacements will be made with no display of matched lines.

It is possible to replace a text string with a new string of a different character count. In this case, the file will be expanded or contracted as needed. This may take some time, for example; replacing all of the single spaces on this page with a single # sign would only take a second. But replacing all of the spaces with say, four # signs (####) would take somewhat longer, since each replacement would require moving the following file down three spaces for each replacement.

It is possible to overflow a line of text by replacing text patterns with new patterns of greater length. In this case, the pattern program displays the offending line and a message to the effect that overflow would occur if the replacement were made. The replacement is then not made. Type any key except the Control Q keys to continue the pattern replacement procedure. Type the Control Q keys to abort the pattern replacement procedure; then enter a carriage return to return to the text area.

At the completion of pattern replacement a message will be displayed giving the number of times the pattern was replaced. Enter a carriage return to return to the text area.

The other two forms of the command result in the deletion of the indicated pattern. PA D and PA DS are similar in operation to PA R and PA RS, respectively, except that the user is not asked to supply a replacement pattern, because no patterns are to be replaced.

At any time during the pattern replacement procedure the Control Q keys may be typed to abort the operation. All replacements made up to this point will remain, but no further replacements will be made.

INSERT FILE   **INSERT FILE**

FUNCTION: To insert an external file into the current file.

COMMAND:

        Control B

        :IF <U:file-name.ext> <CR>

DESCRIPTION:

If this command is typed after the colon prompt (:), the named file (an existing CP/M file with a legal filename where U: is the disk drive and (.ext) is the file extension) will be inserted into the file in memory. The insertion begins just BEFORE the current cursor position. The first line to be inserted pushes all characters on the right side of the cursor RIGHTWARD in the text area (as though that line had been inserted with the Control V command). All subsequent lines are inserted as a block, pushing the existing text downward; the file will expand to accommodate new lines (just as though they were entered from the keyboard), unless the insert operation would overflow the file area.

When the insertion is complete, the cursor is positioned on a blank line following the last line that was inserted.

File insertion can be stopped at any time by typing Control Q.

TAB STOPS   TAB STOPS

FUNCTION: To set or clear tab stops.

COMMAND:

                    Control B

              :TS <CR>

DESCRIPTION:

If this command is typed after the colon prompt (:), a row
of digits appears on the next line, with the cursor flashing
in the first character position, and inverse video or (.)
periods indicating current tab stops. "Inverse video" means
that on a screen that represents characters as black on a
white background, a character at a tab stop will be white on
a black background (and vice versa for the white-on-black
setting of the screen).

To set or clear a tab, use cursor control keys or the
corresponding control characters to move the cursor to the
character position whose status is to be changed; then
strike Control I or the TAB key. If that character position
position was previously a tab stop, it will no longer be; if
it was not a tab stop, it will be set. Control Y will clear
all tabs. Use a carriage return to exit the tab setting
mode.

To move the cursor to the next tab stop, give the Control I
or TAB command in normal editing mode. If there are no tab
stops set, or if no tab stops are set to the right of the
cursor, the cursor will be moved to the first column of the
current line.

NOTE:

If you find that you frequently use tab settings that are
different from the default tab settings, you can make a
second version of the EDITor and then rename it. Have one
for COBOL, one for FORTRAN, or the Assembler, etc.

SECTION 4

ERROR CODES AND MESSAGES

All of the editor messages are contained on a file named
NVEDIT.ERR and can be changed by the user. For example, you
may want to have your messages displayed in German, French
or some other language.

0001	IS NON-EXISTENT. CREATE (Y/N)?
0002	MEMORY ERROR IN FILE-AREA
	ADDRESS:
0003	EDIT D:SOURCE.TYP D:OUTFILE.TYP<CR>
	PLEASE CHECK AND TRY AGAIN
0004	EDIT ABORT
	LOAD EXCEEDS BUFFER SPACE
0005	EDIT ABORT...FILE ERROR
	AT ADDRESS
0006	ABORT?
0007	BYTES FREE
0008	FULL--TYPE CTRL Q
0009	OK TO WRITE TO
0010	*** EDITOR EXIT
0011	FILE COUNT
0012	FILE WILL OVERFLOW BUFFER
0013	PATTERNS CHANGED.
0014	LINE OVERFLOW OF
0015	BYTE(S) IN THIS LINE
0016	LINE NOT FOUND
0017	INPUT BUFFER OVERFLOW
0018	PATTERN
0019	REPLACE
0020	WHAT?
0021	MULTIPLE
0022	NO
0023	I BETWEEN F AND L
0024	L BEFORE F
0025	91 ERROR IN EXTENDING THE FILE
0026	92 END OF DISK DATA - DISK IS FULL
0027	93 FILE NOT OPEN
0028	94 NO MORE DIRECTORY SPACE - DISK IS FULL
0029	95 FILE CANNOT BE FOUND
0030	96 FILE ALREADY OPEN
0031	97 READING UNWRITTEN DATA
0032	C/R TO CONTINUE
0033	DECIMAL

SECTION 5

CONFIGURING AN UNKNOWN TERMINAL

In many cases you will find the terminal you are using
listed for a simple choice. Also, most of the newer
machines emulate one of the listed terminals. For example
the KAYPRO II emulates the Lear-Seigler ADM-3A. Other
frequently emulated terminals are the Lear-Seigler ADM-31
and the SOROC IQ-120. So if your machine is not listed try
these emulations first.

Where the machine emulates a standard terminal but the
screen size is different (24X80), you will have to patch the
line and column size with DDT. For example, the
Shuffleboard III (a board for the TRS-80) emulates the
ADM-3A but expects a 16x64 screen. The user would select
the ADM-3A terminal thus creating the program file EDIT.COM.
This file would then be patched as follows:

```
A> DDT EDIT.COM        (go ahead and try it)
        (here you will see 3 lines of information,
         and then -)
   -S131                (press RETURN after each line)
   0131 18 10           (you type the 10)
   0132 50 40           (you type the 40)
   0133 00 .            (you type a period)
   -                    (here type a ctrl-c)
A> SAVE 45 EDIT.COM
```

If the above procedures do apply in your case, you will be
asked some questions about your terminal and this may be
sufficient to get EDIT up and running. This is an example
of one such interaction. In this case the responses are
those for a Zenith Z-19 terminal and this method would
normally not be required. For further customizing of
special cases, the file NVEDIT.PRN provides additional
information.

```
A>NVEDIT

NVEDIT VERSION 3.0 (0) CONFIGURING
COPYRIGHT (C) 1981, 1982 ELLIS COMPUTING, INC.
@ ANSI MODE TERMINAL
A ADVANTAGE
B APPLE COMPUTER, 40 COLUMN DISPLAY
C APPLE COMPUTER + VIDEX 80 COLUMN BOARD
D BEEHIVE 150 OR CROMENCO 3100
E COMMODORE 64
F FREEDOM 100
```

```
G HAZELTINE 1400 SERIES
H HAZELTINE 1500 SERIES
I HEATH H19/H89 OR ZENITH Z19/Z89
J HEWLETT-PACKARD 2621


Type a single letter to select terminal.
<Carriage Return> for more terminals


K IBM PERSONAL COMPUTER+ BABY BLUE CARD
L INFOTON I-100
M LEAR-SEIGLER ADM-3A
N LEAR-SEIGLER ADM-31
O MICROTERM ACT-IV
P OSBORNE I
Q PERKIN-ELMER 550 (BANTOM)
R PROCESSOR TECHNOLOGY SOL OR VDM
S SOROC IQ-120/140
T SUPERBRAIN
U TELEVIDEO 950
V TRS-80, MOD II (P. & T. CP/M)
W NONE OF THE ABOVE
Type a single letter to select terminal.
<Carriage Return> for more terminals W

Enter 2 digits for number of lines
in the display. 24

Enter 2 digits for number of characters
per line. ('U' will restart entries.) 80

Enter M for memory-mapped display (bank 0 only), T for
a serial-connected terminal.  (M or T)  T

Most terminals position the cursor from
a sequence of characters as follows:
   Lead characters, differing for
     different terminals.
   The line number, sometimes offset
   Sometimes separator characters
   The column number, sometimes offset
   Sometimes ending characters
On some terminals column is before line.
Does your terminal follow
this general pattern? (Y/N) Y

Enter no. of lead characters
for cursor positioning. 2

Enter the first lead character
in hex, e.g. '1B'. 1B
Enter the next lead character. 59
```

Enter the no. of line/col separator
characters. 0

Enter the no. of ending characters. 0

Enter offset to be added to line value.
   Enter 2 hex characters, e.g. 20 20

Enter offset added to column value. 20

Is column entered before line? (Y/N) N

Following controls will speed editing.
If control is not available, enter zero.

Enter no of characters to clear screen
and home the cursor to upper left. 2
Enter two hex characters for each. 1B45

Enter no. of characters to insert line
above cursor position. 2
Enter two hex characters for each. 1B4C

Enter the no. of characters to delete
the cursor line. 2
Enter two hex characters for each. 1B4D

Enter file type to be used
as default (3 characters) CBL

Default tab stops are set for COBOL.
Do you wish to change them? (Y/N) N

Configuring of EDIT.COM is complete.

EDIT.COM saved on the default drive.

USER FEEDBACK ON CONFIGURING

Over the last several years, some customers have gotten back
to us telling how they have configured their unique systems.
We have not tested these procedures, but they may be of help
to you. If you have additional information, please get back
to us and it will be included in the READ-ME.TXT file and
later in the next revision of this manual.

VT101

The VT101 emulates the ANSI terminal except it does not have
the insert line function. Thus, this feature must be

deleted by patching 00 in the insert line area. See
NVEDIT.PRN for the exact location.

CDC110

The CDC110 emulates the ADM-3A except that scrolling a line
at a time overprints the last line. This is because a time
delay is required after sending a line feed. See NVEDIT.PRN
for the location of the time delay after line feed and put
some number in there until the proper time delay is reached.

EPSON

The epson emulates the ADM-3A, however control-P many be
intercepted by the hardware. Just reassign control-P to one
of the free control keys.

TRS-80

In order to enter a control key on the model 1, the customer
must hold down the shift-key and the down-arrow key. The
configuring process is as follows:

First, select the unknown terminal. The number of lines is
16. The number of characters per line is 64. The Memory
Mapped display = M. The memory map address = 3C00. The
scrolling port = 00. Then complete the other questions per
your needs. At this point the file EDIT.COM will be
created.

Next, use CP/M's DDT to patch the EDIT.COM file as follows:

```
 A>DDT EDIT.COM <CR>
    S432E <CR>
         00 FF <CR>    (you type the FF for upper case only)
         . <CR>        (you type a period)
Then type control-C    (return to CP/M)
A> SAVE 45 EDIT.COM <CR>
```

TRS-80 WITH MAPPER

The TRS-80 with the MAPPER emulate the SOROC IQ-120 except
for the screen size which is 16x64 and the line feed at end
of line must be activated. See NVEDIT.PRN for locations to
patch.

AMPEX DIALOG 80

The Ampex Dialog 80 emulates the ADM-3A.

LOBO MAX 80

The LOBO MAX 80 emulates the ADM-31.

APPLI-CARD

The Appli-card emulates the SOCOC IQ-120, except it has a 70 column screen that needs to be patched.

WIZARD-80

The Wizard-80 emulates the videx board.

OSBORNE 1

The Osborne 1 is on the menu, but you must use the setup program that comes with the osborne to set the line length to 128 and also set the arrow key to Word Star.

XEROX 820

The XEROX machines emulate either the ADM-3A or the ADM-31.

KAYPRO

The KAYPRO II emulates the ADM-3A. The arrow keys send the following codes. Left arrow = control-H, Down arrow = control-J, Up arrow = control-K, and the right arrow = control-L.

SECTION 6

CHANGING EDIT KEY FUNCTIONS

The following is a sample run of EDTKEY.COM, changing many of the keys used for EDIT functions. The first five functions using labeled keys were not changed.

```
A>EDTKEY
EDTKEY VERSION 1.3

FOR EACH EDITOR FUNCTION, YOU MAY CHANGE THE KEY WHICH
CAUSES IT TO OCCUR.  IF YOU WISH NO CHANGE, PRESS "RETURN".
THERE ARE TWO METHODS: CTRL-KEY OR ESC FOLLOWED BY KEY.  TO
CAUSE A FUNCTION TO USE THE CONTROL KEY ALONG WITH KEY "B",
PRESS ONLY "B".  TO CAUSE A FUNCTION TO USE "ESC" FOLLOWED
BY "B", PRESS THAT SEQUENCE.  CODES 7FH AND ABOVE (8-BIT
CODES) MAY BE USED ALSO.

DO NOT USE THE CTRL KEY UNLESS TO ABORT THE PROCESS.  FOLLOW
THE DIRECTIONS ABOVE FOR EACH FUNCTION LISTED.

INSERT NEXT LINE. NORMALLY M ("RETURN" OR "ENTER" KEY)
MOVE CURSOR DOWN. NORMALLY J ("LINE FEED" KEY)
MOVE CURSOR TO NEXT TAB STOP. NORMALLY I ("TAB" KEY)
BACKSPACE ONE CHARACTER. NORMALLY H ("BACKSPACE" KEY)
BACKSPACE AND ERASE A CHARACTER. NORMALLY "DELETE" KEY
CURSOR UP ONE LINE. E
CURSOR DOWN ONE LINE. X
CURSOR LEFT ONE CHARACTER. S
CURSOR RIGHT ONE CHARACTER. D
INSERT CHARACTER MODE ON. V
INSERT CHARACTER MODE OFF. V
DELETE CHARACTER UNDER CURSOR. G
DELETE LINE. Y
DELETE FROM CURSOR TO END OF LINE. T
CURSOR TO UPPER LEFT (HOME). Q
SCROLL UP ONE LINE. Z
SCROLL DOWN ONE LINE. W
SCROLL UP HALF OR FULL SCREEN. C
SCROLL DOWN HALF OR FULL SCREEN. R
CURSOR TO MID SCREEN, COLUMN ONE. A
INITIATE STRING SEARCH MODE. F
CONTINUE SEARCH FOR STRING. L
EXIT FROM EDITOR OR SEARCH MODE. K
SCROLLING IS NORMALLY 1/2 SCREEN.
DO YOU WANT IT TO BE FULL SCREEN? (Y/N) Y

CHANGES ARE NOW COMPLETE.
```

TABLE OF ASCII CODES (Zero Parity)

Upper Octal	Octal	Decimal	Hex	Character	
0000	000	0	00	ctrl @	NUL
0004	001	1	01	ctrl A SOH	Start of Heading
0010	002	2	02	ctrl B STX	Start of Text
0014	003	3	03	ctrl C ETX	End of Text
0020	004	4	04	ctrl D EOT	End of Xmit
0024	005	5	05	ctrl E ENQ	Enquiry
0030	006	6	06	ctrl F ACK	Acknowledge
0034	007	7	07	ctrl G BEL	Audible Signal
0040	010	8	08	ctrl H BS	Back Space
0044	011	9	09	ctrl I HT	Horizontal Tab
0050	012	10	0A	ctrl J LF	Line Feed
0054	013	11	0B	ctrl K VT	Vertical Tab
0060	014	12	0C	ctrl L FF	Form Feed
0064	015	13	0D	ctrl M CR	Carriage Return
0070	016	14	0E	ctrl N SO	Shift Out
0074	017	15	0F	ctrl O SI	Shift In
0100	020	16	10	ctrl P DLE	Data Line Escape
0104	021	17	11	ctrl Q DC1	X On
0110	022	18	12	ctrl R DC2	Aux On
0114	023	19	13	ctrl S DC3	X Off
0120	024	20	14	ctrl T DC4	Aux Off
0124	025	21	15	ctrl U NAK	Negative Acknowledge
0130	026	22	16	ctrl V SYN	Synchronous File
0134	027	23	17	ctrl W ETB	End of Xmit Block
0140	030	24	18	ctrl X CAN	Cancel
0144	031	25	19	ctrl Y EM	End of Medium
0150	032	26	1A	ctrl Z SUB	Substitute
0154	033	27	1B	ctrl [ESC	Escape
0160	034	28	1C	ctrl \ FS	File Separator
0164	035	29	1D	ctrl] GS	Group Separator
0170	036	30	1E	ctrl ^ RS	Record Separator
0174	037	31	1F	ctrl _ US	Unit Separator
0200	040	32	20	Space	
0204	041	33	21	!	
0210	042	34	22	"	
0214	043	35	23	#	
0220	044	36	24	$	
0224	045	37	25	%	
0230	046	38	26	&	
0234	047	39	27	'	
0240	050	40	28	(	
0244	051	41	29	)	
0250	052	42	2A	*	
0254	053	43	2B	+	
0260	054	44	2C	,	
0264	055	45	2D	-	
0270	056	46	2E	.	
0274	057	47	2F	/	
0300	060	48	30	0	
0304	061	49	31	1	
0310	062	50	32	2	
0314	063	51	33	3	
0320	064	52	34	4	
0324	065	53	35	5	
0330	066	54	36	6	
0334	067	55	37	7	
0340	070	56	38	8	
0344	071	57	39	9	
0350	072	58	3A	:	
0354	073	59	3B	;	
0360	074	60	3C	<	
0364	075	61	3D	=	
0370	076	62	3E	>	
0374	077	63	3F	?	

TABLE OF ASCII CODES (Cont'd) (Zero Parity)

Upper Octal	Octal	Decimal	Hex	Character	
0400	100	64	40	@	
0404	101	65	41	A	
0410	102	66	42	B	
0414	103	67	43	C	
0420	104	68	44	D	
0424	105	69	45	E	
0430	106	70	46	F	
0434	107	71	47	G	
0440	110	72	48	H	
0444	111	73	49	I	
0450	112	74	4A	J	
0454	113	75	4B	K	
0460	114	76	4C	L	
0464	115	77	4D	M	
0470	116	78	4E	N	
0474	117	79	4F	O	
0500	120	80	50	P	
0504	121	81	51	Q	
0510	122	82	52	R	
0514	123	83	53	S	
0520	124	84	54	T	
0524	125	85	55	U	
0530	126	86	56	V	
0534	127	87	57	W	
0540	130	88	58	X	
0544	131	89	59	Y	
0550	132	90	5A	Z	
0554	133	91	5B	[	
0560	134	92	5C	\	
0564	135	93	5D	]	
0570	136	94	5E	^	
0574	137	95	5F	_	
0600	140	96	60	`	
0604	141	97	61	a	
0610	142	98	62	b	
0614	143	99	63	c	
0620	144	100	64	d	
0624	145	101	65	e	
0630	146	102	66	f	
0634	147	103	67	g	
0640	150	104	68	h	
0644	151	105	69	i	
0650	152	106	6A	j	
0654	153	107	6B	k	
0660	154	108	6C	l	
0664	155	109	6D	m	
0670	156	110	6E	n	
0674	157	111	6F	o	
0700	160	112	70	p	
0704	161	113	71	q	
0710	162	114	72	r	
0714	163	115	73	s	
0720	164	116	74	t	
0724	165	117	75	u	
0730	166	118	76	v	
0734	167	119	77	w	
0740	170	120	78	x	
0744	171	121	79	y	
0750	172	122	7A	z	
0754	173	123	7B	{	
0760	174	124	7C			
0764	175	125	7D	}	
0770	176	126	7E	~	Prefix
0774	177	127	7F	DEL	Rubout

Note: Both tables also include a "Paper tape (123.4567P)" column at left, shown as punched-hole dot patterns.

Ellis Computing, Inc.
3917 Noriega Street, San Francisco, CA 94122

SOFTWARE LICENSE AGREEMENT

IMPORTANT: All Ellis Computing programs are sold only on the condition that the purchaser agrees to the following License.

ELLIS COMPUTING agrees to grant and the Customer agrees to accept on the following terms and conditions nontransferable and nonexclusive Licenses to use the software program(s) (Licensed Programs) herein delivered with this Agreement.

TERM:

This Agreement is effective from the date of receipt of the above-referenced program(s) and shall remain in force until terminated by the Customer upon one month's prior written notice, or by Ellis Computing as provided below.

Any License under this Agreement may be discontinued by the Customer at any time upon one month's prior written notice. Ellis Computing may discontinue any License or terminate this Agreement if the Customer fails to comply with any of the terms and conditions of this Agreement.

LICENSE:

Each program License granted under this Agreement authorizes the Customer to use the Licensed Program in any machine readable form on any single computer system (referred to as System). A separate license is required for each System on which the Licensed Program will be used.

This Agreement and any of the Licenses, programs or materials to which it applies may not be assigned, sublicensed or otherwise transferred by the Customer without prior written consent from Ellis Computing. No right to print or copy, in whole or in part, the Licensed Programs is granted except as hereinafter expressly provided.

PERMISSION TO COPY OR MODIFY LICENSED PROGRAMS:

The customer shall not copy, in whole or in part, any Licensed Programs which are provided by Ellis Computing in printed form under this Agreement. Additional copies of printed materials may be acquired from Ellis Computing.

The NEVADA EDIT Licensed Programs which are provided by Ellis Computing in machine readable form may be copied, in whole or in part, in machine readable form in sufficient

number for use by the Customer with the designated System, for back-up purposes, or for archive purposes. The original, and any copies of the Licensed Programs, in whole or in part, which are made by the Customer shall be the property of Ellis Computing. This does not imply that Ellis Computing owns the media on which the Licensed Programs are recorded.

The Customer agrees to reproduce and include the copyright notice of Ellis Computing on all copies, in whole or in part, in any form, including partial copies of modifications, of Licensed Programs made hereunder.

PROTECTION AND SECURITY:

The Customer agrees not to provide or otherwise make available the NEVADA EDIT Program including but not limited to program listings, object code and source code, in any form, to any person other than Customer or Ellis Computing employees, without prior written consent from Ellis Computing, except with the Customer's permission for purposes specifically related to the Customer's use of the Licensed Program.

DISCLAIMER OF WARRANTY:

Ellis Computing makes no warranties with respect to the Licensed Programs.

LIMITATION OF LIABILITY:

THE FOREGOING WARRANTY IS IN LIEU OF ALL OTHER WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL ELLIS COMPUTING BE LIABLE FOR CONSEQUENTIAL DAMAGES EVEN IF ELLIS COMPUTING HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

GENERAL:

If any of the provisions, or portions thereof, of this Agreement are invalid under any applicable statute or rule of law, they are to that extent to be deemed omitted. This is the complete and exclusive statement of the Agreement between the parties which supercedes all proposals, oral or written, and all other communications between the parties relating to the subject matter of this Agreement. This Agreement will be governed by the laws of the State of California.

CORRECTIONS AND SUGGESTIONS

All suggestions and problems must be reported in writing. Please include samples if possible.

EDIT VERSION _____ ( ) SERIAL #_____

Operating system and version _____

Hardware configuration _____

ERRORS IN MANUAL:

SUGGESTIONS FOR IMPROVEMENTS TO MANUAL:

ERRORS IN EDITOR:

SUGGESTIONS FOR IMPROVEMENT TO EDITOR:

MAIL TO: Ellis Computing, Inc.
               3917 Noriega Street
               San Francisco,CA 94122

FROM: NAME_____ DATE_____

            ADDRESS_____

            CITY,STATE,ZIP_____

            PHONE NUMBER_____

If you wish a reply include a self-addressed postage-paid envelope. Thank you.

CONTROL KEYS: PAGE

OTHER KEYS: