



Guide to the Ultimate Editors

The Ultimate Corp.
East Hanover, NJ

Version 1.1

Guide to the Ultimate® Editors

Document No. 6939

Version 1.2

© 1991, The Ultimate Corp., East Hanover, NJ.

All Rights Reserved.

Printed in the United States of America.

How to Obtain This Guide

The *Guide to the Ultimate Editors* is included with the system documentation set.

To obtain additional copies, consult The Ultimate Corp.'s *Documentation Update* brochure or call the Ultimate Administration department (201-887-9222).

Proprietary Information

This work is the property of and embodies trade secrets and confidential information proprietary to Ultimate, and may not be reproduced, copied, used, disclosed, transferred, adopted, or modified without the express written approval of The Ultimate Corp.

UltiWriter

© 1989 by The Ultimate Corp and KEO/SOFT International Corporation,
Tustin, CA.

Operating System Release 10, Revision 210

© 1989, The Ultimate Corp.

Ultimate® and Ultimate UPDATE® are registered trademarks of The Ultimate Corp. UltiWriter™ is a trademark of The Ultimate Corp.

Contents

Preface	vii
How the Guide Is Organized.....	vii
Conventions.....	viii
1 Introduction	1-1
Features of the Editors	1-1
Compatibility.....	1-2
Choosing an Editor.....	1-2
2 Introduction to the Screen Editor	2-1
Invoking the Screen Editor.....	2-2
Editing Keys.....	2-4
Editor Modes.....	2-5
Using the Screen Editor	2-6
Exiting an Item or Command	2-6
Moving Around in the Item	2-6
The Ruler.....	2-8
Selecting Text.....	2-8
The Clipboard.....	2-9
Inserting Text	2-9
Deleting Text.....	2-10
Recovering Deleted Text.....	2-10
Copying and Moving Text	2-10
Changing Case.....	2-11
Split Screen.....	2-11
Merging Items.....	2-11
Searching and Replacing Text.....	2-11
Prestoring Commands.....	2-11
Breaking Lines	2-12
Creating Blank Lines.....	2-12
3 Screen Editor Keys	3-1
Arrow Keys.....	3-4
<BACKSPACE>	3-5
BOL.....	3-6
COMMAND GLOSSARY	3-7
.....	3-10
<DEL CHAR/LINE>.....	3-11

DOCUMENT	3-12
EOL.....	3-13
EXIT.....	3-14
Exiting a Command.....	3-14
Exiting an Item.....	3-14
<HOME>.....	3-16
<INS CHAR/LINE>.....	3-17
INSERT	3-18
MERGE DOCUMENT	3-20
PAGE.....	3-21
<PAGE NEXT>	3-22
PARAGRAPH FWD/BACK.....	3-23
REFRESH SCREEN.....	3-24
RETURN	3-25
RULER	3-26
SEARCH.....	3-28
SEARCH & REPLACE	3-30
SELECT.....	3-33
SENTENCE FWD/BACK	3-36
TAB FWD/BACK.....	3-37
UNDO.....	3-38
WORD FWD/BACK	3-39
4 Introduction to the Line Editor	4-1
Invoking the Line Editor.....	4-2
How the Line Editor Uses Buffers.....	4-5
Using the Line Editor.....	4-7
Line Editor Commands.....	4-7
Exiting an Item.....	4-7
Moving Around in the Item	4-8
Strings.....	4-8
The Wildcard Feature.....	4-9
Inserting Text	4-9
Merging and Cutting Text.....	4-10
Searching and Replacing Text.....	4-10
Deleting Text.....	4-10
Cancelling Commands.....	4-11
Prestoring Commands.....	4-11
Suppressing Output Display.....	4-11
Advanced Techniques.....	4-12
Creating Blank Lines.....	4-12
Breaking Lines	4-12

Trimming Lines.....	4-14
Using Wildcards to Replace Text.....	4-15
Non-Printable Characters	4-15
5 Line Editor Commands	5-1
↓ Command - RETURN	5-4
? Command - Display Item Name and Current Line.....	5-5
^ Command - Turn Wildcard Feature Off and On.....	5-6
A Command - Locate Again.....	5-8
AL Command - Append Line.....	5-9
AS Command - Turn Assembler Formatting On and Off	5-10
B Command - Bottom.....	5-12
BL Command - Break Line.....	5-13
C Command - Display Column Numbers.....	5-15
CL Command - Cut Line.....	5-16
DE Command - Delete Line.....	5-18
EX Command - Exit Item without Saving.....	5-20
F Command - Flip Buffers.....	5-22
FD Command - Delete Item	5-23
FI Command - File Item.....	5-24
FS Command - File Save.....	5-26
G Command - Go To Line	5-28
HX Command - Turn Hexadecimal Display On and Off	5-29
I Command - Insert Lines	5-30
JL Command - Join Lines	5-34
L Command - List Lines.....	5-35
L Command - Locate String.....	5-36
M Command - Turn Macro Expansion On and Off.....	5-39
ME Command - Merge Lines.....	5-41
N Command - Next.....	5-43
P Command - Prestore	5-44
PD Command - Prestore Display.....	5-48
Q Command - Go to Address.....	5-49
R Command - Replace.....	5-50
S Command - Turn Suppress Display On and Off	5-53
S? Command - Display Item Size	5-55
T Command - Top.....	5-56
TB Command - Set Tabs.....	5-57
U Command - Up.....	5-58
X Command - Cancel Last Change	5-59
XF Command - Cancel All Changes.....	5-61
Z Command - Set Display Zone.....	5-62

Appendices

A. Screen Editor Messages.....	A-1
B. Screen Editor Files.....	B-1
C. Line Editor Messages.....	C-1
D. RUNOFF Commands.....	D-1

Index.....	index-1
------------	---------

Figures

Figure 1-1. Attribute Definition Item (Screen Editor).....	1-4
Figure 1-2. Attribute Definition Item (Line Editor).....	1-4
Figure 2-1. Sample Screen Editor Display.....	2-3
Figure 2-2. Function Key Layout.....	2-4
Figure 3-1. Text Before Material is Inserted.....	3-19
Figure 3-2. Text After Material is Inserted.....	3-19
Figure 4-1. Examples of EDIT Commands.....	4-4
Figure 4-2. Buffer Contents before the F Command.....	4-6
Figure 4-3. Buffer Contents after the F Command.....	4-6
Figure 4-4. Trimming the Right End of a Line.....	4-14
Figure 4-5. Trimming the Left End of a Line.....	4-15
Figure 4-6. Using Wildcards to Replace Text.....	4-15
Figure 4-7. Removing a Control Character.....	4-16

Tables

Table 3-1. Editing Key Definitions.....	3-1
Table 5-1. Line Editor Commands.....	5-1
Table D-1. RUNOFF Embedded Commands.....	D-2
Table D-2. RUNOFF Line Commands.....	D-3

Preface

This manual is intended as a guide for users of Ultimate's screen editor and line editor. It covers all aspects of using the two editors with the Ultimate Operating System and file structure.

How the Guide Is Organized

Chapter 1 compares the screen and line editors.

Chapter 2 gives an introduction to the screen editor and the system level Terminal Control Language (TCL) command associated with it. It also provides an overview of the screen editor's features.

Chapter 3 describes each of the screen editor keys. They are presented in alphabetical order by name.

Chapter 4 gives an introduction to the line editor and the system level (TCL) commands associated with it. It also provides an overview of the line editor's features.

Chapter 5 describes each of the line editor commands. The commands are presented in alphabetical order.

Four appendices are included: Appendix A explains the messages generated by the screen editor; Appendix B describes two SYSPROG files used by the screen editor; Appendix C explains the messages generated by the line editor; Appendix D is a summary of RUNOFF commands.

An index is also provided.

Conventions

This guide employs a variety of conventions to describe the features and commands of the Ultimate editors. The conventions are explained below.

Convention	Description
UPPER CASE	Characters printed in upper case are required and must appear exactly as shown. Names of keys (such as RETURN) and commands (such as DELETE) are also shown in upper case.
lower case	Characters or words printed in lower case are parameters to be supplied by the user (for example, line number, data, filename, etc.).
{ }	Braces surrounding a parameter indicate that the parameter is optional and may be included or omitted at the user's discretion. The braces themselves are not to be included.
RETURN	The word RETURN indicates a physical carriage return pressed at the keyboard. It is used in text. See ↵.
↵	This symbol has the same meaning as RETURN. It is used in examples.
enter / type	The word <i>enter</i> means that after the specified keys are pressed, a carriage return is needed to complete the operation. The word <i>type</i> means that the specified keys are to be pressed, but a carriage return is not to be included.
<KEY>	Named or designated key other than letters or numbers (for example, <SHIFT>).
<KEY1-KEY2>	Key1 must be pressed and held down while key2 is pressed.

Courier	Courier typeface is used for messages or prompts displayed by the system and, when boldfaced, for user input.
bold	In Courier font, boldface type is used to indicate user input.
filename	<p>The word filename can indicate any of the following, depending on the section of the file to be edited:</p> <ul style="list-style-type: none">• dataname. Specifies the DATA section of a file with the same name as its DICT.• dictname,dataname. Specifies a DATA section when the dictname has multiple DATA sections.• DICT dictname. Specifies the DICT section of a file.• DICT dictname,dataname. Specifies the DICT section of a file. Same as DICT dictname above.• DATA dataname. Specifies the DATA section of a file with the same name as its DICT. Same as dataname above.• DATA dictname,dataname. Specifies the DATA section when the dictname has multiple DATA sections. Same as dictname,dataname above.

Notes

1 Introduction

The Ultimate Operating System includes two text editors: a full screen editor and a line editor. The editors permit on-line modification of such items as BASIC programs, data files, and file dictionaries. The following paragraphs summarize features of the editors, discuss their compatibility, and describe their strengths.

Features of the Editors

The editors have several features in common:

- ability to insert and delete text
- ability to locate and replace text
- ability to retrieve deleted text
- ability to merge text from the same or different item in the same file
- ability to store frequently used commands or text for later recall

The screen editor and the line editor also have features not shared by the other editor.

The screen editor has the following additional features:

- screen orientation
- sophisticated cursor movement throughout the entire item
- editing keys
- insert and replace editing modes
- split screen in which two items can be viewed at once
- a clipboard for copying and moving text, either between items or within a single item
- optional prompting for itemname and filename
- ability to change the case of text (upper and lower case)

The line editor has the following additional features:

- command orientation
- ability to merge text from an item in another file
- ability to save current item to an item in another file
- ability to edit a select-list
- ability to edit items of any size
- ability to save text in compressed format
- assembly formatting
- hexadecimal display
- line number prompting on input

Compatibility

The output of the screen and line editors is identical, and an item created with one editor can be retrieved and edited with the other.

However, there are differences in using the editors. The screen editor edits by moving the cursor anywhere in the item and making changes directly in the text. The line editor uses typed commands to change specified lines; text is not changed directly.

The default tab settings also differ, but when text is tabbed in either editor, spaces are inserted that remain when the item is retrieved by the other editor. The screen editor has margins that can be changed; margins in the line editor cannot be changed. In both editors, the margins control only the way the data is displayed; reaching the right margin does not force the start of a new attribute.

Choosing an Editor

Terminal type, the size of attributes and items, and the use to which the editor will be put are the most important criteria in choosing one of the editors. The factors are discussed below.

Terminal Type. The screen editor supports the following terminals (and many others that can emulate them): ADDS 2020 and 2025, and WYSE 50 and 60. The line editor can be used on all standard terminals.

Size. The screen editor limits attributes and items to 32,000 bytes. The line editor limits attributes to 465 characters, but has no limit on item size.

Editor Use. The line editor has the ability for hexadecimal display and assembly formatting. The line editor can be used to search lengthy programs for a word or variable and list all the occurrences at once. The line editor can also save the current item to another file. Also, only the line editor can edit select-lists and compressed items.

The screen editor allows you to edit text using function key commands similar to word processing. The screen editor's search and replace function contains flexible options concerning capitalization and whether the string can be embedded in other text. The screen editor also has a split screen feature that enables two items in the same file to be viewed at the same time. Creating a blank line (null attribute) is slightly easier with the screen editor.

Figures 1-1 and 1-2 on the next page illustrate the keystrokes necessary to create an attribute definition item, first using the screen editor and then the line editor. Although the keystrokes are slightly different, the finished results are identical.

```
COL 019 LINE 0004 PAGE 0001
Enter E)xit no-save U)pdate S)ave & exit
....+█...1....T....2....T....3....T....4....T....5....T....6....T....7....R..

0001 A ↓
0002 1 ↓
0003 Comments ↓
0004 ↓
0005 ↓
0006 ↓
0007 ↓
0008 ↓
0009 T ↓
0010 50 ↓
0011 █ <F15> s
```

Figure 1-1. Attribute Definition Item (Screen Editor)

```
New item
Top
.I ↓
001+A ↓
002+1 ↓
003+Comments ↓
004+<CTRL-^> ↓
005+<CTRL-^> ↓
006+<CTRL-^> ↓
007+<CTRL-^> ↓
008+<CTRL-^> ↓
009+T ↓
010+50 ↓
011+ ↓
.FI ↓
```

Figure 1-2. Attribute Definition Item (Line Editor)

2 Introduction to the Screen Editor

The screen editor can create, edit, and delete PROCs, BASIC programs, and items in data files and file dictionaries. It can be used to write assembly language programs, but cannot display assembly formatting. It cannot edit items larger than 32,000 bytes.

Note: The screen editor should not be used to edit Ultimate UPDATE[®] definition items. To edit these items, use the UPD-DEF system command to ensure that the items will be validated.

The screen editor includes the following features:

- cursor movement throughout the item with arrow and function keys
- insert and replace editing modes
- ability to insert and delete text
- ability to locate and replace text
- ability to retrieve deleted text
- ability to merge text from the same or different item in the same file
- ability to store frequently used commands or text for later recall
- split screen in which two items can be viewed at once
- a clipboard for copying between items, and for cutting and pasting

The screen editor can be used with the following terminals:

Terminal Name	TERM Type	Terminal Setup
ADDS 2020, 2025	A	factory defaults
WYSE 50	V or W	ADDS Viewpoint enhanced
WYSE 50	X	Wyse 50 in native mode
WYSE 60	S	Wyse 60 in native mode

Other terminals that can emulate one of these should also work; contact your dealer representative or Ultimate TAC.

Invoking the Screen Editor

The screen editor is invoked at TCL by the SE command.

Syntax

SE {filename} {itemlist} {(L)}

filename is the name of the file that contains items to be edited. If no filename is specified, the screen editor prompts for it.

itemlist may contain one or more explicit item-IDs or an asterisk (*) to indicate all items in the file. If no item-ID is specified, the screen editor prompts for one.

(L) loads all 16 function keys before the editor is entered. If the L option is not specified, the current values of the function keys are used. If the value of a key does not match the screen editor's requirements, that key's editing function is not available. If the EXIT key (<F15>) is not loaded, <CTRL-X> can be used to exit the item.

Description

If more than one item is specified to be edited, a prompt similar to the following is displayed before each item is retrieved:

```
edit < itemname > (y/n/q) ?
```

To edit the item, enter **Y**. To skip the item and go on to the next, enter **N**. To exit the editor, enter **Q**. Either upper or lower case response is accepted by the screen editor.

When an item is retrieved, the screen editor displays it in full screen format with attribute numbers placed at the left margin. Figure 2-1 shows the basic elements of the screen editor display.

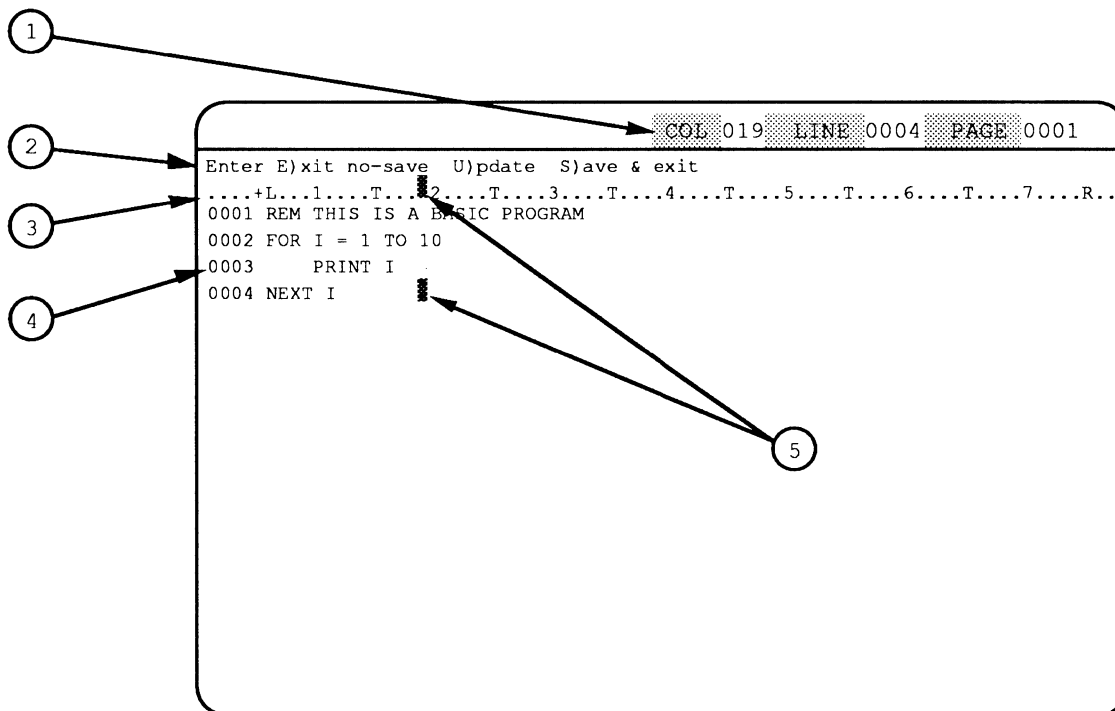


Figure 2-1. Sample Screen Editor Display

The following elements are displayed in Figure 2-1:

1. Status line, including dynamically updated column, line, and page indicators.
2. Message line, which is used to prompt for additional information. The message shown is for the EXIT command. When two options are shown within parentheses, such as (Y/N), the first option is the default and can be selected with a carriage return. Responses can be input in either upper or lower case.
3. Ruler containing left (L) and right (R) margin markers, tab stops (T), and a column counter.
4. Attribute numbers.
5. Dual cursors; the cursor on the ruler mirrors the actual cursor.

Editing Keys

The screen editor makes use of function keys, arrow keys, control key sequences, and other terminal keys such as <INS CHAR/LINE> and . Each key is described in alphabetical order in Chapter 3.

Figure 2-2 displays the layout of the function keys on a typical keyboard. The lower row is the unshifted function key command. The upper row is the shifted function key command. Only the function keys illustrated below are used by the screen editor.

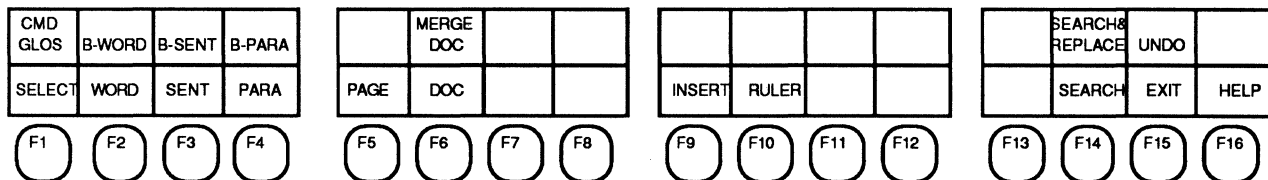


Figure 2-2. Function Key Layout

The rest of this chapter provides an overview of the screen editor's features and indicates which editing keys are required for each editing operation. In the descriptions that follow, the name of the key and its function key number or control key sequence are given. For example, to exit, you are directed to use the EXIT key (<F15> or <CTRL-X>).

Editor Modes

The screen editor has two modes: *replace* (overtyping) and *insert*. In replace mode, anything you type over existing text replaces it. When you reach the right margin, one of two things happens:

- If it is the bottom line of the item, the cursor wraps to the left margin of the next line and text entry can continue.
- If it is not the last line in the item, the terminal beeps and you cannot insert more text until you either enter insert mode or use the insert line (<INS LINE>) key. The RETURN key does not produce a new line in replace mode unless you are on the last line of the item.

In insert mode, characters are added at the cursor, and the existing text is moved to the right. When you get to the right margin, a new line is inserted, and the cursor drops to the left margin of that line so that text entry can continue. This occurs from any place in the item. The RETURN key can also be used to insert a new line from any place in the item. When you are in insert mode, the letters INS are displayed on the status line.

When you retrieve an item, you are always in replace mode. To toggle between replace and insert modes, press INSERT (<F9>).

Using the Screen Editor

This section presents an overview of the screen editor's features. The editing keys used to perform the functions introduced here are described in detail in Chapter 3. The following topics are covered:

Exiting a Command or Item	Copying and Moving Text
Moving Around in the Item	Changing Case
The Ruler	Split Screen
Selecting Text	Merging Items
The Clipboard	Searching and Replacing Text
Inserting Text	Prestoring Commands
Deleting Text	Breaking Lines
Recovering Deleted Text	Creating Blank Lines

Exiting an Item or Command

The EXIT key (<F15> or <CTRL-X>) is used to exit either from the item or from an unfinished editing command. If no command is requesting more information prior to execution, you can exit the item after choosing whether to save changes made. A third option on the EXIT menu allows you to update the item and continue editing. If the editor is waiting for a response prior to executing a command, EXIT cancels the command and returns to the text.

Moving Around in the Item

The screen editor allows you to use cursor control keys to move anywhere within the margins of the editor. On any line, you can go past the last character of text until you reach the right margin. Unless noted, cursor control keys do not repeat when held down.

The following cursor control keys can be used to move around the item:

arrow keys	The arrow keys move the cursor in the direction of the arrow. If the key is held down, the action repeats.
<BACKSPACE>	<BACKSPACE> moves the cursor left one space. Depending on the keyboard, it may also delete the character. If the key is held down, the action repeats.

BOL	BOL (<CTRL-L>) moves the cursor to the beginning of the line.
EOL	EOL (<CTRL-E>) moves the cursor to the end of the line.
<HOME>	<HOME> moves the cursor to the top of the current screen. If it is held down, the action repeats.
PAGE	PAGE (<F5>) gives you the choice of going to the beginning or end of the entire item or to a specified line number.
<PAGE NEXT>	<PAGE NEXT> moves the cursor to the bottom line of the current screen.
PARAGRAPH FWD/BACK	PARAGRAPH FWD (<F4>) moves the cursor to the next paragraph. PARAGRAPH BACK (<SHIFT-F4>) moves the cursor back a paragraph.
RETURN	RETURN moves the cursor to column 1 of the current line or the next line. In insert mode, a new line is also created. If the key is held down, the action repeats.
SENTENCE FWD/BACK	SENTENCE FWD (<F3>) moves the cursor to the next sentence. SENTENCE BACK (<SHIFT-F3>) moves the cursor back a sentence.
TAB FWD/BACK	TAB FWD (<TAB>) moves the cursor to the next tab stop. TAB BACK (<SHIFT-TAB>) moves the cursor left one tab stop. If the key is held down, the action repeats.
WORD FWD/BACK	WORD FWD (<F2>) moves the cursor to the beginning of the next word. WORD BACK (<SHIFT-F2>) moves the cursor back a word.

The Ruler

The ruler is displayed under the message line at the top of the screen (see Figure 2-1), and is used to control format. You can enter the ruler to make margin and tab changes and to set a zone where word-wrap applies. Changes made to the ruler govern the entire item but are not saved when you exit the item. Each time you retrieve an item, the default ruler is displayed, and the text returns to the margins governed by the default settings. However, tabbed text remains in the format in which it was input.

In addition to the cursor in your text, there is also a cursor on the ruler. The two cursors occupy the same column position, allowing you to determine which column in the text you are in. When you work in the ruler, the cursor in the text disappears temporarily, leaving only the ruler cursor. When you exit the ruler, the dual cursors return.

Each dot on the ruler represents one column on the screen. There are 77 columns visible on the ruler and 240 columns available altogether. The L indicates the left margin; the R indicates the right margin. Each T indicates a tab stop.

The ruler may also contain plus signs (+). They are located at ten-space intervals in the fives position to make the ruler easier to read. Plus signs are not displayed when another character fills that position on the ruler.

To enter and exit the ruler, use the RULER key (<F10>) when the cursor is on line 0001.

Selecting Text

An important feature of the screen editor is the ability to select text and then delete it, copy it, move it, or change its case.

You select text by first positioning the cursor at the beginning (or end) of the intended block and pressing the SELECT key (<F1>). A menu with four options is displayed on the message line above the ruler. To select text, use the cursor control keys to highlight the text before responding to the options. Once the text is selected, you can perform any of the four operations by responding to the prompt. See the topics The Clipboard, Deleting Text, Copying and Moving Text, and Changing Case in this chapter.

The Clipboard The clipboard is a 32K storage buffer used in copying and moving text. If there is a CLP indicator on the status line above the ruler, there is material on the clipboard. You can view and edit the clipboard by temporarily moving it to a blank area in the item.

To use the clipboard, select the text and choose the copy or move option. This transfers the text to the clipboard. Then move the cursor to a new location in the same item or another item in the same file and transfer the contents of the clipboard to the new location. See Copying and Moving Text.

Each time you copy or move text to the clipboard, the text is appended to the end of whatever is currently on the clipboard. This feature allows you to rearrange material efficiently by copying several pieces of text to the clipboard in their proper order, and then transferring the entire material to its intended location.

Inserting Text There are three keys that govern the insertion of text:

- INSERT (<F9>)
- <INS CHAR> (insert character)
- <INS LINE> (insert line)

INSERT toggles between replace and insert modes; <INS CHAR> and <INS LINE> allow you to insert a limited amount of text while staying in replace mode.

Attribute marks, value marks, and subvalue marks are called *system delimiters*. They can be inserted into the text by typing the following key sequences in either replace or insert mode:

Delimiter	Keys	Display	ASCII Code
attribute mark (^)	RETURN	none	CHAR(254)
value mark (])	<CTRL-]>]	CHAR(253)
subvalue mark (\)	<CTRL-\>	\	CHAR(252)

Neither the <CTRL> key nor the attribute mark (^) is displayed, but the] and \ are.

Deleting Text You can delete text by selecting it with the SELECT key and choosing the D)delete option from the menu that is displayed. The following terminal keys can also be used to delete text:

- <DEL CHAR> (delete character)
- <DEL LINE> (delete line)
- and, if your terminal supports it, the destructive backspace (frequently labelled)

Recovering Deleted Text You can recover just-deleted material with the UNDO key (<SHIFT-F15>), but you must press it before any operation except cursor movement is *completed*. (If a second operation is begun and cancelled, UNDO will still recover the deletion.)

Copying and Moving Text You can copy and move text from one area of your item to another or from one item to another item in the same file. You accomplish this by selecting the text with the SELECT key and choosing the C)copy or M)ove option from the select menu.

When you *copy* selected text, the text remains in its original location, and a copy of it is placed on the clipboard. When you *move* selected text, the text is removed from its original location and placed on the clipboard.

Once a new location has been chosen for the text, you can either copy or move the text from the clipboard. If you copy the text from the clipboard, a copy remains on the clipboard. If you move text from the clipboard, the text is removed from the clipboard.

Note that copying text from the clipboard may leave unwanted material there. You can avoid this situation in any of the following ways:

- delete what is on the clipboard using the D option of the SELECT key.
- transfer text from the clipboard to your item, by moving it (which deletes the text on the clipboard), rather than by copying it.
- move the contents of the clipboard to a blank area in your item, delete portions you no longer need, and move the edited text back to the clipboard.

Changing Case

You can change the capitalization of a portion of text by selecting it with the SELECT key and choosing the !)case option from the select menu. You can choose from all upper case, all lower case, or capitalizing the first letter in each sentence.

Split Screen

The split screen feature allows you to have two items open at a time, with the first item in the top half of the screen and the second item in the bottom half. You open the second item by pressing the DOCUMENT key (<F6>) from within the first item.

This feature can be used to copy or move text between two items. Material can be copied or moved from one item to the clipboard. Once control is passed to the other item, the text can be copied or moved from the clipboard to that item. You can also use the split screen to view a second item without editing it.

Merging Items

You can merge all or part of another item into the item you are editing by using the MERGE DOCUMENT key (<SHIFT-F6>). The text that you merge is inserted on the lines above the cursor. Only a copy of the text is merged; the text in the original item is not affected. Both items must be in the same file.

Searching and Replacing Text

The screen editor has both SEARCH and SEARCH & REPLACE features. SEARCH (<F14>) locates specified text. SEARCH & REPLACE (<SHIFT-F14>) allows you to search for a string and then change or delete it. You can specify parameters to control the way the search is performed.

Prestoring Commands

The COMMAND GLOSSARY key (<SHIFT-F1>), allows you to store keystrokes (text and/or editing commands) in the screen editor glossary for later execution. For example, programmers might choose to store standard comments describing their programs and insert them at appropriate points. Or, firms could store memo headings and frequently used text.

Breaking Lines

In insert mode, a line can be broken into two separate lines by pressing RETURN. The break occurs to the left of the cursor; the character the cursor is on and all text to the right move to the next line. Lines below move down to make room. If RETURN is pressed at the beginning or end of a line, the break also creates a blank line (see *Creating Blank Lines*, below).

In replace mode, lines cannot be broken; pressing RETURN moves the cursor to the left margin of the next line without disturbing the text.

Creating Blank Lines

Two keys can be used to create a blank line (null attribute): the RETURN key or the <INS LINE> key. There are several ways to break a line:

- In insert mode, press RETURN at the end of any line. A blank line is inserted below, and the cursor moves to the left margin of the new line. Any text below the new line moves down.
- In insert mode, press RETURN when the cursor is at the left margin of a line. The text on that line moves down a line, creating a blank line. The cursor remains at the left margin of the blank line.
- In insert mode, press RETURN twice in the middle of a line of text. The first RETURN breaks the line at that point; the second RETURN inserts a blank line at the break. The cursor moves to the left margin of the blank line.
- In either replace or insert mode, press RETURN when the cursor is at the end of the document. This creates a blank line.
- In either replace or insert mode, press <INS LINE> from any place on the line. A blank line is inserted above that line. The cursor moves to the left margin of the blank line.

3 Screen Editor Keys

Table 3-1 lists the name, keystrokes involved, and function performed by the screen editor keys. Except where noted, the keys are described later in the text in the same order as they are listed in the table below.

Table 3-1. Editing Key Definitions (1 of 3)

Name	Key	Description
Arrow Keys	← → ↑ ↓	moves cursor in direction of arrow.
BACKSPACE	<BACKSPACE>	moves cursor one space to the left.
BOL	<CTRL-L>	moves cursor to beginning of line.
COMMAND GLOSSARY	<SHIFT-F1>	stores keystrokes or text or executes stored command.
DEL		deletes character to left of cursor.
DEL CHAR/LINE	<DEL CHAR/LINE>	deletes character cursor is on / deletes line cursor is on.
DOCUMENT	<F6>	opens another item or displays item-ID.
EOL	<CTRL-E>	moves cursor to end of line.
EXIT	<F15> or <CTRL-X>	exits command or item.

Table 3-1. Editing Key Definitions (2 of 3)

Name	Key	Description
HOME	<HOME>	if cursor is on first line of screen, moves cursor to previous screen; from anywhere else, moves cursor to top of screen.
INS CHAR/LINE	<INS CHAR/LINE>	inserts space at cursor / inserts line above cursor.
INSERT	<F9>	toggles between insert and replace (overtyp) modes.
MERGE DOCUMENT	<SHIFT-F6>	merges text.
PAGE	<F5>	displays page options.
PAGE NEXT	<PAGE NEXT>	from last line of screen, moves cursor to next screen; from anywhere else, moves cursor to bottom of screen.
PARAGRAPH FWD/BACK	<F4> / <SHIFT-F4>	moves cursor to beginning of next/previous paragraph.
REFRESH SCREEN	<CTRL-R>	refreshes screen.
RETURN	↵	in replace or insert mode, moves cursor to column 1 of next line; in insert mode, also inserts new line.
RULER	<F10>	enters or exits ruler.

Table 3-1. Editing Key Definitions (3 of 3)

Name	Key	Description
SEARCH	<F14>	searches for specified text.
SEARCH & REPLACE	<SHIFT-F14>	searches for and replaces specified text.
SELECT	<F1>	selects text.
SENTENCE FWD / BACK	<F3> / <SHIFT-F3>	moves cursor to beginning of next/previous sentence.
TAB FWD/BACK	<TAB> / <SHIFT-TAB>	moves cursor to next/previous tab stop indicated on ruler.
UNDO	<SHIFT-F15>	recovers last deletion.
WORD FWD/BACK	<F2> / <SHIFT-F2>	moves cursor to beginning of next/previous word.

Arrow Keys (← → ↑ ↓)

The arrow keys move the cursor one column or line at a time. When held down, the arrow keys repeat. The arrow keys can be used to move the cursor into areas where there is no text.

- ← (left arrow) moves the cursor one column to the left. If the cursor is at the left margin of any line except the first line, the left arrow moves the cursor to the right margin of the previous line. If the cursor is at the left margin of the first line, the left arrow key has no effect.
- (right arrow) moves the cursor one column to the right, up to the right margin. If the cursor is at the right margin of any line except the last line, the right arrow moves the cursor to the left margin of the next line. If the cursor is at the right margin of the last line, the right arrow key has no effect.
- ↑ (up arrow) moves the cursor up one line, while keeping it in the same column. If the cursor is on the first line of the item, the up arrow key has no effect.
- ↓ (down arrow) moves the cursor down one line, while keeping it in the same column. If the cursor is on the last line of the item, the down arrow key has no effect.

<BACKSPACE>

The <BACKSPACE> key moves the cursor back one space. On some keyboards, this key works the same as the left arrow key, moving the cursor without changing text. However, on other keyboards, the key labeled <BACKSPACE> is a backspace delete key (also known as a destructive backspace). In this case, as it backs up one space, it replaces the character to the left of the cursor with a space. Text is not moved.

When the <BACKSPACE> key is held down, the action repeats.

BOL (<CTRL-L>)

The BOL (Beginning-Of-Line) key (<CTRL-L>) moves the cursor to the left margin from anywhere on the line.

In ruler mode, the BOL key scrolls the ruler one screen width to the right each time it is pressed.

COMMAND GLOSSARY (<SHIFT-F1>)

The COMMAND GLOSSARY key (<SHIFT-F1>) stores and executes keystrokes (text and editing commands). For example, you could store a paragraph or a series of screen editor commands, and then execute the material when needed. If the text is lengthy and had already been entered elsewhere in the same file, you can enter the keystrokes necessary to open the item containing the text and merge the appropriate lines into the item you are editing. Each time you execute that glossary, the identified text is inserted at the cursor location.

You can also use the COMMAND GLOSSARY key to delete previously stored text and commands.

Storing a New Command Glossary

To create and store a new command glossary, press the COMMAND GLOSSARY key. The following prompt is displayed on the message line:

```
Enter personal glossary 0-9, or glossary name:
```

To create a new glossary, press COMMAND GLOSSARY again. The following prompt is displayed:

```
Enter glossary name:
```

Enter either a *number* from 0 to 9 or a *name* of up to 10 characters. The first character of a name must be alphabetic, but the rest can be letters, numbers, spaces, or any other characters.

If you enter a *number* that has already been stored, the new material automatically overwrites the old; the screen editor does *not* check for existing command numbers. Since all glossaries are stored in an UltraWriter file called KFJOB, which is accessible from all user accounts, a numbered glossary could easily be inadvertently changed by someone else.

If you enter a *name* that already exists, the following prompt is displayed:

```
Duplicate glossary name - Replace? (N/Y):
```

If you want to replace the glossary with a different group of keystrokes, enter **Y**.

If you do not want to change the contents of the old glossary, enter **N**. The prompt to enter a name is redisplayed:

Enter glossary name:

To create a glossary, enter a new name. To return to the text, press EXIT (<F15> or <CTRL-X>).

After you select a number or a name, the characters GLO appear on the status line. The cursor returns to the text so you can type the text and/or screen editor commands to be stored.

When you are finished, press COMMAND GLOSSARY again. The command is stored, and GLO is no longer displayed. You may now return to editing.

Note: Do not press EXIT when you are finished unless you want the EXIT command to become a part of the glossary item.

Executing a Command Glossary

To recall a previously stored command glossary, place the cursor at the point where the command is to be executed and press the COMMAND GLOSSARY key. The following prompt is displayed on the message line:

Enter personal glossary 0-9, or glossary name:

To execute the glossary, type its number (do *not* press RETURN) or enter its name (press RETURN). If the item exists, the stored keystrokes (text and/or editing commands) take effect at the cursor location.

If the item does not exist, the following prompt is displayed:

Undefined glossary, <RET> to continue:

To return to the text, press RETURN. Now you can continue editing, or you can try COMMAND GLOSSARY again with a different name.

**Deleting a
Command
Glossary**

To delete a named or numbered command glossary, press the COMMAND GLOSSARY key. The following prompt is displayed on the message line:

Enter personal glossary 0-9, or glossary name:

Press COMMAND GLOSSARY again. The glossary name prompt is displayed:

Enter glossary name:

Enter the name or number of the item to delete. If you enter a name, the following prompt is displayed. If you enter a number, this message is skipped.

Duplicate glossary name - Replace? (N/Y):

To continue with the deletion, enter **Y**.

The letters GLO are displayed on the status line, and the cursor returns to the text. Press COMMAND GLOSSARY again. The item is deleted, and the letters GLO are no longer displayed.

**Cancelling the
Command**

The process of storing, recalling, or deleting a glossary can be cancelled by pressing EXIT (<F15> or <CTRL-X>) in response to any prompt on the message line.

The (delete) key functions as a destructive backspace; it moves the cursor one column to the left and replaces any character that was there with a space. Text to the right does not move. When it is held down, the action repeats.

Deleted text can be recovered by using the UNDO key (<SHIFT-F15>) before completing any other editing operation. (Cursor movement is not considered an editing operation and does not interfere with the effectiveness of the UNDO key.)

Note: Not all keyboards have a key.

<DEL CHAR/LINE>

The <DEL CHAR/LINE> key has two functions. Unshifted, <DEL CHAR> is activated; shifted, <DEL LINE> is activated.

<DEL CHAR> deletes the character the cursor is on each time it is pressed. All text to the right of the cursor moves one space to the left.

<DEL LINE> deletes the line the cursor is on. It does not matter where on the line the cursor is located; the entire line is deleted, and the lines below the cursor move up one line.

Deleted text can be recovered by using the UNDO key (<SHIFT-F15>) before completing any other editing operation. (Cursor movement is not considered an editing operation and does not interfere with the effectiveness of the UNDO key.)

DOCUMENT (<F6>)

The DOCUMENT key (<F6>) opens a second item or displays the name of the current item.

The screen editor employs a split screen concept. Two items from the same file can be displayed at the same time: the first item in the top half of the screen, the second in the bottom half. When the second item is opened, control passes to it; the first item cannot be edited until the second is closed.

Press DOCUMENT whenever you are editing an item and want to view, change, or copy to or from a second item. After you press the key, the following prompt is displayed on the message line:

Enter document name:

Enter the name of the second item; it can be a name that did not previously exist. To cancel the command, press EXIT (<F15> or <CTRL-X>).

Once a name is entered, the new item is opened and control is passed to it. You have full editing capabilities in the second item. You cannot move back and forth between the items; to copy material from the first item to the second, select the text and put it on the clipboard before opening the second item. To copy from the second item to the first, put the text on the clipboard before you close the second item. (The clipboard and copying text are described in Chapter 2.)

When you close the second item, the first item fills the screen. The cursor returns to the character it was on when you pressed DOCUMENT.

The DOCUMENT key can also be used to display the name of the item you are editing. At the prompt for the document name, press EXIT (<F15> or <CTRL-X>). A message similar to the following is displayed:

Document name - *ITEM-ID* - VERSION 9.2

ITEM-ID is the name of the item being edited; 9.2 is the version of UltiWriter™ that supports the screen editor.

EOL (<CTRL-E>)

The EOL (End-Of-Line) key (<CTRL-E>) moves the cursor to the position immediately past the last character of text on the current line. Blank spaces at the end of a line are considered characters.

In ruler mode, the EOL key scrolls the ruler one screen width to the left each time it is pressed.

EXIT (<F15> or <CTRL-X>)

The EXIT (<F15> or <CTRL-X>) key cancels a partially completed editing command or exits the current item.

Exiting a Command

The following commands all present prompts on the message line. Any of these commands can be cancelled by pressing EXIT whenever a prompt is displayed:

- COMMAND GLOSSARY
- DOCUMENT
- EXIT (from item)
- MERGE DOCUMENT
- PAGE
- RULER
- SEARCH
- SEARCH & REPLACE
- SELECT

When EXIT is pressed, the command is cancelled and the cursor returns to the character it was on when the command was initiated.

Exiting an Item

To exit an item, press EXIT whenever the cursor is in the text of the item. The following prompt is displayed:

```
Enter E)xit no-save U)pdate S)ave & exit
```

To cancel the exit command, press EXIT again. Otherwise, type the first letter of the desired option (do *not* press RETURN). The options are described below.

E)xit no-save

The E option exits the item without saving it. If the item has never been saved, it is lost.

After you enter E, the following prompt is displayed:

Are you sure? (Y/N) :

To confirm the exit, enter **Y** or just press RETURN. If the SE command entered at TCL specified only one item or if this is the last item to be edited, the system returns to TCL. If the SE command specified multiple items, the current item is exited; control returns to the screen editor to edit the next item, if any, specified in the SE command. If there are no additional items, control returns to TCL.

If you do not want to exit without saving your changes, enter **N**. The cursor returns to the character it was on when EXIT was pressed.

U)update

The U option saves your editing and returns to the same place in your item, so you can continue your work.

Hint: Use the U option often to save your work. Also, use this option just before any operation you are unsure of, such as copying or moving text. If the results are not what you expected, use the E option to exit from the item. When you retrieve the item again, you will have the last saved version.

S)ave & exit

The S option saves the current version of your item before exiting. If the SE command entered at TCL specified only one item or if this is the last item to be edited, the system returns to TCL. If the SE command specified multiple items, the current item is exited; control returns to the screen editor to edit the next item, if any, specified in the SE command. If there are no additional items, control returns to TCL. Use this option to exit an item and save all the changes you made.

<HOME>

The <HOME> key moves the cursor to the top of the current screen, or to the bottom of the next screen. When it is held down, the action repeats.

If the cursor is anywhere on the screen except the first line, <HOME> moves the cursor to the top of the screen, in the same column.

If the cursor is on the first line of the screen, <HOME> moves the cursor to the last line of the previous screen, in the same column. If the cursor is on the first line of the first screen, <HOME> refreshes the screen.

Note: On some keyboards, <HOME> is labeled <PREV SCREEN>.

<INS CHAR/LINE>

The <INS CHAR/LINE> key has two functions. Unshifted, <INS CHAR> is activated; shifted, <INS LINE> is activated. These keys operate in both replace and insert modes.

<INS CHAR> inserts a blank space at the cursor each time it is pressed. Text to the right of the cursor moves right one space.

<INS LINE> inserts a blank line directly above the line the cursor is on and moves the cursor to the left margin of the new line. All lines below the cursor move down one line.

INSERT (<F9>)

The INSERT key (<F9>) toggles between replace (overtyping) and insert modes. In replace mode (the default each time an item is retrieved), new text is typed over existing text, replacing it. In insert mode, new text is inserted to the left of the cursor, and all text to the right of the cursor moves to the right. New lines are created as necessary.

When the screen editor is in insert mode, the letters INS are displayed on the status line.

On the next page, Figure 3-1 shows the text before new material is inserted; Figure 3-2 shows the results after the insertion.

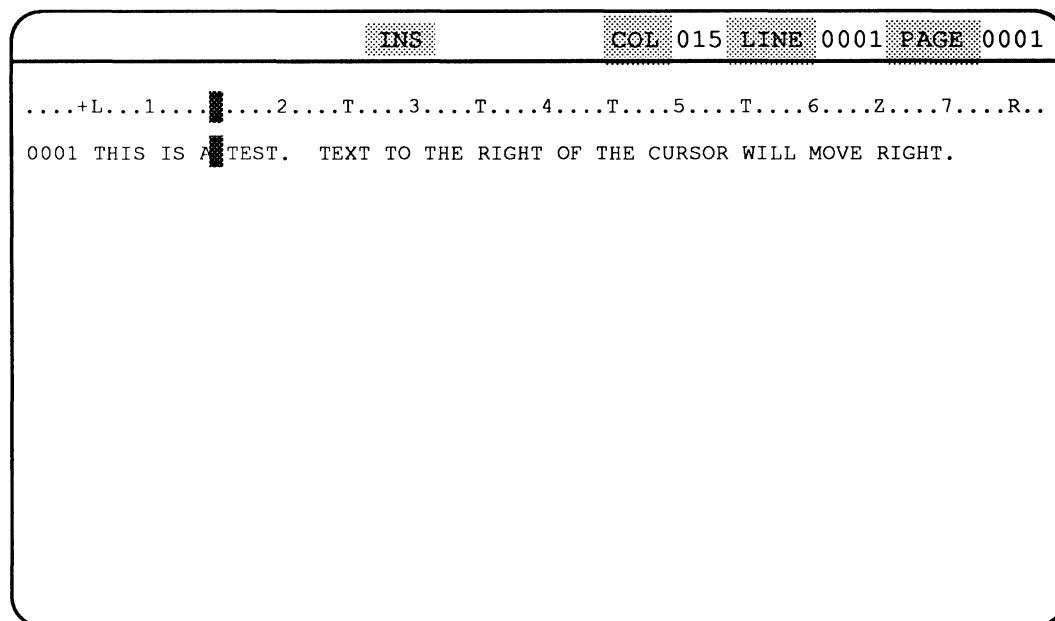


Figure 3-1. Text Before Material Is Inserted

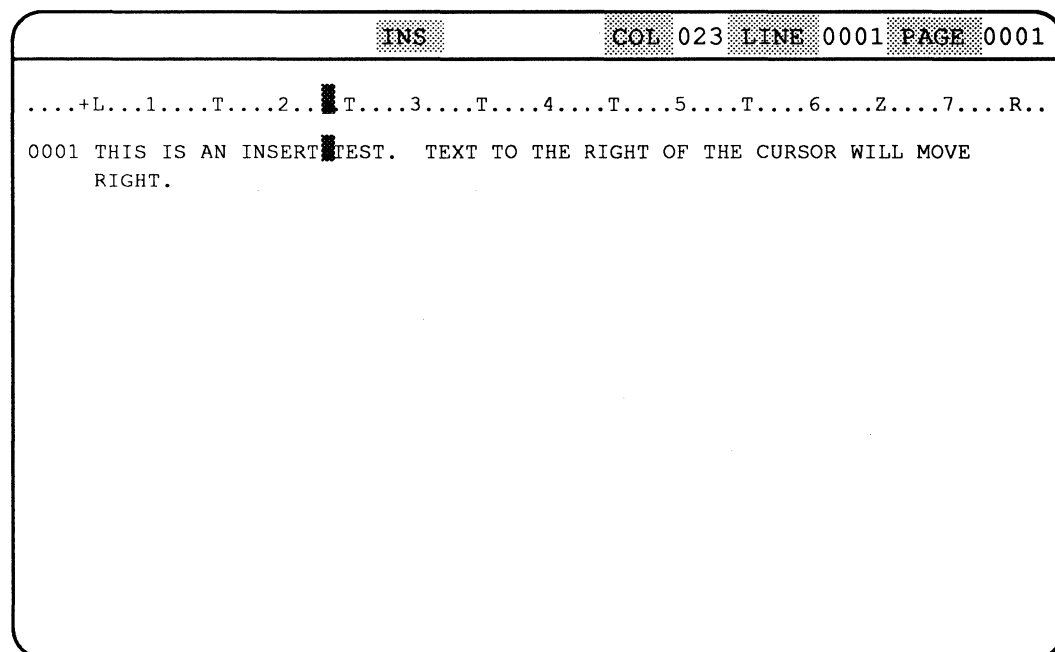


Figure 3-2. Text After Material Is Inserted

MERGE DOCUMENT (<SHIFT-F6>)

The MERGE DOCUMENT key (<SHIFT-F6>) inserts all or part of a second item into the item being edited. Both items must be in the same file. Only a copy of the text is inserted; the original text is unaffected. The text is inserted on lines above the line the cursor is on.

To merge lines, first determine which lines from the source item you want to insert. Then edit the item that is to receive the inserted material and position the cursor where the insertion is to take place.

When you press MERGE DOCUMENT, you are prompted for the name of the item that contains the material to be merged:

Enter merge document name:

Then you are prompted for the starting and ending line numbers:

Enter start line number:

Enter end line number:

When you reply, the specified text is inserted above the line the cursor is on, and all lines below move down.

If the file you are working in does not contain an item with the name you specified, the following message appears:

Document does not exist, <RET> to continue:

When you press RETURN, you are returned to the text.

You can cancel the MERGE DOCUMENT command at any of the prompts by pressing EXIT (<F15> or <CTRL-X>).

PAGE (<F5>)

The PAGE key (<F5>) moves the cursor to a specific line or to the top or bottom of an item. When you press PAGE, the following prompt is displayed on the message line:

Enter T)op, B)ottom or Line number

To move to the top (first line) of the item, press **T**. Do *not* press RETURN.

To move to the bottom (last line) of the item, press **B**. Do *not* press RETURN.

To move to a specific line number, type the line number and press RETURN. The screen is repainted with that line at the top.

In each case, the cursor remains in the same column.

To cancel the PAGE command, press EXIT (<F15> or <CTRL-X>) rather than selecting an option.

Note: This is the editing key PAGE, *not* the terminal key <PAGE NEXT>.

<PAGE NEXT>

The <PAGE NEXT> key moves the cursor to the bottom of the current screen, or to the top of the next screen.

If the cursor is anywhere on the screen except the last line, <PAGE NEXT> moves the cursor to the bottom of the screen, in the same column.

If the cursor is on the last line of the screen, <PAGE NEXT> scrolls the screen up and moves the cursor to the first line of the next screen, in the same column.

Note: This is the standard terminal key, not the editing key PAGE (<F5>). On some terminals it is labelled <NEXT SCREEN>.

PARAGRAPH FWD/BACK (<F4>/<SHIFT-F4>)

From anywhere in a paragraph, the PARAGRAPH FWD key (<F4>) moves the cursor to the first character of the next paragraph.

From the first character of a paragraph, PARAGRAPH BACK (<SHIFT-F4>) moves the cursor to the first character of the preceding paragraph. From the middle of a paragraph, PARAGRAPH BACK moves the cursor to the first character of that paragraph.

Note: Except for the first and last paragraphs in an item, the screen editor defines a paragraph as characters preceded and followed by a carriage return. From the first paragraph, PARAGRAPH BACK moves the cursor to the beginning of the item; from the last paragraph, PARAGRAPH FWD moves the cursor two columns past the last character in the item.

A paragraph is the same as a non-null attribute. A null attribute is not considered a paragraph.

REFRESH SCREEN (<CTRL-R>)

The REFRESH SCREEN key (<CTRL-R>) momentarily clears the screen and then repaints it with the same display. This is useful in instances when the screen display becomes garbled.

RETURN (↵)

In replace mode, RETURN moves the cursor to the left margin of the next line without disturbing the text.

In insert mode, the action depends on where the cursor is on the line:

- If the cursor is at the left margin, RETURN creates a null attribute on that line, and all text on that line and lower lines moves down. The cursor remains at the left margin of the newly created null attribute.
- From the middle of a line, RETURN creates a new line and moves the cursor and all text from the cursor to the right end of the line to the new line. Text on lines below moves down.
- If the cursor is beyond the last character on the line, RETURN moves the cursor to the left margin of the next line and creates a null attribute there. Text on lines below moves down.

In either mode, if RETURN is held down, the action repeats.

RULER (<F10>)

The RULER key (<F10>) enters and exits ruler mode to permit you to make format changes to your item. Changes made to the ruler are *not* saved with the item. Each time you retrieve an item, the default ruler, including margins and tab settings, is displayed. Text is formatted according to the default margins, but is tabbed according to the tab settings in effect when it was input. The first segment (77 columns) of a sample ruler is shown below.

....+L...1....T....2....+....3....+....4....+....5....+....6..Z..+....7....R..

Entering and Exiting Ruler Mode

To enter ruler mode, the cursor must be on line 0001. Press RULER. The characters RUL are displayed on the status line. In ruler mode, the only cursor you see is on the ruler; no cursor is displayed in the text.

When you are editing the ruler, you can only move the cursor to the right and left on the ruler. The following cursor keys are available:

← (left arrow)	BACKSPACE
→ (right arrow)	TAB FWD
BOL	TAB BACK
EOL	

To exit the ruler and retain any changes, press RULER again. To exit the ruler without retaining any changes, press EXIT (<F15> or <CTRL-X>). The dual cursors are redisplayed; the cursor in the text returns to the character it was on when the command was initiated.

Changing Margins

To change the left or right margin, enter the ruler, move the cursor to the desired location, and type **L** for the left margin or **R** for the right margin. The new L or R is inserted, and the old L or R disappears. The L cannot be placed left of column 6, because attribute numbers are displayed on the left side of the screen.

To work with an area wider than your screen, the right margin can be set to a maximum of 240. Press EOL (<CTRL-E>) to move to the next section of the ruler. The ruler scrolls to the left as an additional section is displayed. The text does not scroll while you are in the ruler. To set your right margin, position your cursor and type **R**.

Press EOL and BOL (or any of the other cursor keys) as many times as you need to move forward and backward along the ruler. When you exit the ruler with the RULER key, existing text is reformatted according to the new margins.

Changing Tab Settings

To insert a tab setting, enter the ruler, move the cursor to the desired location, and press **T**. A T is displayed on the ruler.

To delete a tab stop, move the cursor to the location containing the tab stop, and press **T**. The T is removed from the ruler.

Existing tabbed text is not reformatted, but any new lines that are entered are formatted according to the new settings.

SEARCH (<F14>)

The SEARCH key (<F14>) allows you to search for character strings. When one of the strings is found, the editor stops there, with the cursor positioned at the first character of the string.

The search begins at the cursor location where you pressed SEARCH. To search the entire item, move to the beginning of the item before initiating the search.

When you press SEARCH, the following prompt is displayed:

```
n - search string:
```

where *n* is the number of the search string being entered.

Enter the string you want to search for. It can contain up to 20 characters. When you are finished, another prompt appears so that you can enter another search string. You can enter up to nine strings.

When you have entered all the desired search strings, you can begin the search immediately with the default parameters by pressing SEARCH at the next prompt, or you can review and change the default parameters by pressing RETURN. The default parameters are:

- Case dependent? (N)
- Partial or full? (P)

If you press RETURN, the prompt for the first parameter is displayed:

```
Case dependent? (N/Y):
```

To search for all possible capitalizations (the default), enter **N** or press RETURN. To search only for text with the same capitalization as the search string, enter **Y**.

The prompt for the second search parameter is displayed:

```
Partial or full? (P/F):
```

Partial specifies that the search string can be embedded in other text; full specifies that the search string must stand alone.

To cancel the command before it executes, press EXIT (<F15> or <CTRL-X>) at any of the prompts, rather than selecting an option.

The search begins. The screen editor stops at the first character of each occurrence of the search string, so that it can be edited. When you are ready to continue the search, press SEARCH *twice*. The process continues until the last occurrence is reached or you decide to quit by simply not pressing SEARCH again. If you press SEARCH twice at the last occurrence, the terminal beeps, and the cursor remains on the first character of the last occurrence.

If no occurrence of the string is found, the terminal beeps, and the cursor returns to the point where the SEARCH key was originally pressed.

You may search the item as many times as you wish by returning to the beginning of the text and pushing SEARCH twice. The search strings remain until you overwrite them or exit the item.

SEARCH & REPLACE (<SHIFT-F14>)

The SEARCH & REPLACE key (<SHIFT-F14>) searches for character strings, and replaces them with specified replacement strings. You can designate this replacement to be done automatically or only after you have evaluated each case individually.

The search begins at the cursor location where you pressed SEARCH & REPLACE. To search the entire item, move to the beginning of the item before initiating the search.

When you press SEARCH & REPLACE, the following prompt is displayed:

```
n - search string:          replace string:
```

where *n* is the number of the search string being entered.

Enter the string you want to search for. It can contain up to 20 characters. When you press RETURN, the cursor moves to the replace string prompt. Enter the replacement string, also with a maximum of 20 characters. A null string is accepted; you can use it to delete occurrences of the search string. When you are finished, another prompt appears so that you can enter another search string and replacement string. You can enter up to nine search strings and replacement strings.

When you have entered all the desired search and replacement strings, you can begin the search immediately with the default parameters by pressing SEARCH (<F14>, *not* SEARCH & REPLACE) at the next prompt, or you can review and change the default parameters by pressing RETURN. The default parameters are:

- Verify changes? (N)
- Case dependent? (N)
- Replace exact? (N)
- Partial or full? (P)

If you press RETURN, the prompt for the first parameter is displayed:

```
Verify changes? (N/Y):
```


To replace all strings without stopping at each one (the default), enter **N** or press RETURN. To have the screen editor stop at each occurrence so that you can decide whether to replace it, enter **Y**.

The prompt for the second parameter is displayed:

Case dependent? (N/Y) :

To search for all possible capitalizations (the default), enter **N** or press RETURN. To search only for text with the same capitalization as the search string, enter **Y**.

The prompt for the third parameter is displayed:

Replace exact? (N/Y) :

To have the replacement capitalized as it is in the text it is replacing (the default), enter **N** or press RETURN. To have the replacement capitalized as specified in the replacement string, enter **Y**.

The prompt for the fourth parameter is displayed:

Partial or full? (P/F) :

Partial specifies that the search string can be embedded in other text; full specifies that the search string must stand alone.

If you entered **Y** at the prompt to verify changes, the next prompt is skipped, and the search begins. If you entered **N** (instructing the editor not to stop to verify each change), the following prompt is displayed:

List changes to printer? N)o D)efault printer
A)ux port:

This list shows the changes that were made. It shows the page number, attribute number, and the entire line of old and new text. An example of such a list, in which the word saving is replaced by spending, is shown below.

```
P0001 L0001      ...many people enjoy saving.
P0001 L0001      ...many people enjoy spending.

P0001 L0016      Saving helps the economy grow....
P0001 L0016      Spending helps the economy grow....
```

If you do not want to print a list of the items replaced, enter **N**. If you want the list directed to your assigned printer, enter **D**. If you want the list printed at the printer attached to your auxiliary port, enter **A**.

To cancel the command before it executes, press EXIT (<F15> or <CTRL-X>) at any of the prompts, rather than selecting an option.

When the prompts have been answered, the search and replace begins. If you entered **N** to verify changes, the editor makes all the replacements at once. If you selected **Y**, the editor stops at each occurrence and asks you whether to replace or not:

Replace? (N/Y) :

To leave the text as it is, enter **N** or just press RETURN. To replace the text, enter **Y**. The editor moves to the next occurrence.

Whether the replacement was automatic or verified by you each time, the terminal beeps at the last occurrence, and the cursor is positioned at the first character of that string.

If no occurrence of the string is found, the terminal beeps, and the cursor returns to the point where the SEARCH & REPLACE key was originally pressed.

SELECT (<F1>)

The SELECT key (<F1>) selects text so that it can be deleted, copied, moved, or have its case changed. When you press SELECT, the following select prompt is displayed:

```
Select D)elete C)opy M)ove !)case
```

To work with text in the item, select the text before responding to the prompt; to work with material on the clipboard, respond to the prompt immediately. Both alternatives are described below.

If an option is selected before text is selected and nothing is on the clipboard, the screen editor cancels the command and returns to the item.

Editing Text in Your Item

To delete, copy, move, or change the case of text in your item, position the cursor at the beginning or end of the intended text and press SELECT. The select prompt is displayed. Before responding, select the text using the appropriate cursor control keys; the message remains on the screen. The text you select is displayed in low intensity. (If it is not, set your terminal protect mode to dim.)

After you have selected the text, respond to the prompt by typing the first letter of the desired option. The options are described below.

When the operation is complete, the cursor returns to the position where the command was initiated.

D)elete

The selected text is deleted. (Deleted text can be recovered by pressing the UNDO key (<SHIFT-F15>) immediately.)

C)opy

A copy of the text is transferred to the clipboard, but also remains in its original location. Text transferred to the clipboard is appended to whatever else is already there.

M)ove

The text is transferred to the clipboard and deleted from its original location. Text transferred to the clipboard is appended to whatever else is already there.

!)case

The selected text is changed to all upper case, all lower case, or the first letter of each sentence capitalized. The following prompt allows you to make this choice:

```
Enter U)pper L)ower F)irst:
```

Editing Text on the Clipboard

If the letters CLP are displayed on the status line, there is text on the clipboard. To delete, copy, or move text on the clipboard, press SELECT and respond to the select prompt when it is first displayed, without selecting text. The options are described below.

When the operation is complete, the cursor returns to the character it was on when the command was initiated.

D)elete

The text on the clipboard is deleted.

C)opy

A copy of the text on the clipboard is transferred to the item, beginning just to the right of the cursor. The text also remains on the clipboard.

M)ove

The text is transferred to the item, beginning just to the right of the cursor. The text is deleted from the clipboard.

!)case

This option cannot be used with material on the clipboard.

**Cancelling the
Command**

To cancel the SELECT command before it is completed, press EXIT (<F15> or <CTRL-X>) at one of the prompts, rather than selecting an option.

SENTENCE FWD/BACK (<F3>/<SHIFT-F3>)

From anywhere in a sentence, the SENTENCE FWD key (<F3>) moves the cursor to the first character of the next sentence. If the cursor is in the last sentence of an item, the SENTENCE FWD key moves the cursor two columns to the right of the last character.

From the first character of a sentence, SENTENCE BACK (<SHIFT-F3>) moves the cursor to the first character of the preceding sentence. From the middle of a sentence, SENTENCE BACK moves the cursor to the first character of that sentence.

Notes: The screen editor defines the following as sentences:

- *Except for the last sentence in an attribute, a sentence is defined as characters, a period, and two or more spaces.*
- *The last sentence in an attribute is defined as characters, an optional period, optional spaces, and a carriage return or end-of-item marker.*

The SENTENCE FWD/BACK keys treat question and exclamation marks as periods.

A null attribute is not considered a sentence.

TAB FWD/BACK (<TAB>/<SHIFT-TAB>)

The TAB FWD key (<TAB>) moves the cursor to the next tab setting indicated on the ruler. Any text or spaces between the cursor and the next tab setting are not disturbed; if there is no more text on the line, spaces are inserted up to the next tab. If there are no more tabs on the ruler, the cursor does not move. If the <TAB> key is held down, the action repeats.

To move to the previous tab setting on the ruler, use TAB BACK (<SHIFT-TAB>). If the key is held down, the action does not repeat.

For information on setting tabs, see the RULER section. If the tab settings are changed, text formatted according to previous settings is not affected.

Note: Tab stops in the screen editor cannot be set with the TABS system command.

UNDO (<SHIFT-F15>)

The UNDO key (<SHIFT-F15>) recovers text deleted with the D option of the SELECT command, the <DEL CHAR/LINE> key, or the key. To recover the text, you must press UNDO before any operation, except cursor movement, is *completed*. (If a second operation is begun and cancelled, UNDO is still effective.)

WORD FWD/BACK (<F2>/<SHIFT-F2>)

From anywhere in a word, the WORD FWD key (<F2>) moves the cursor to the first character of the next word.

From the first character of a word, WORD BACK (<SHIFT-F2>) moves the cursor to the first character of the preceding word. From the middle of a word, WORD BACK moves the cursor to the first character of that word.

Note: The screen editor defines a word as any characters that are preceded by spaces or a beginning of line mark and followed by spaces, any punctuation mark, or an end of line mark.

Notes

4 Introduction to the Line Editor

The line editor can create, edit, and delete PROCs, BASIC programs, assembly language programs, and items in data files and file dictionaries.

Note: The line editor should not be used to edit Ultimate UPDATE definition items. To edit these items, use the UPD-DEF system command to ensure that the items will be validated.

The line editor has the following features:

- ability to edit select-lists
- ability to edit items of any size
- ability to insert and delete text
- ability to locate and replace text
- ability to recover deleted text
- ability to merge text from an item in another file
- ability to save current item to an item in another file
- ability to save text in compressed format
- ability to store frequently used commands or text for later recall
- assembly formatting
- hexadecimal display
- line number prompting on input
- two variable-length temporary buffers

Invoking the Line Editor

The line editor is invoked at TCL by the ED{IT}, EEDIT, or EDIT-LIST command. ED{IT} is the normal edit command; EEDIT operates on items stored in compressed format; EDIT-LIST edits select-lists.

Syntax

ED{IT} filename itemlist {(options)}

EEDIT filename itemlist {(options)}

EDIT-LIST select-list

filename is the name of the file that contains items to be edited.

itemlist may contain one or more explicit item-IDs or an asterisk to indicate all items in the file. The itemlist can be omitted if there is an active select-list.

select-list is the name of a previously saved list. When this command form is used, the list itself is edited.

(options

- A** turns on assembly code formatting. See the AS command in Chapter 5.
- M** turns on the macro expansion flag. See the M command in Chapter 5.
- P** routes all output to the print spooler.
- S** turns on the option to suppress line numbers or object code. See the S command in Chapter 5.
- X** turns on the hexadecimal output toggle before entering the line editor. See the HX command in Chapter 5.

Description

The editor signals that it is ready to start editing the item by displaying the following:

Top
.

The prompt character, a period, indicates that a line editor command is to be entered.

If the specified item does not already exist on file, the following message is displayed above the Top message:

New item

If the EDIT command specified editing multiple items, the item-ID of the current item to be edited is also displayed. When the editing of an item is terminated via a FI (file item), FD (file delete), or EX (exit) command, if there are other items in the itemlist, the line editor is automatically re-entered and the next item is displayed.

The EEDIT form of the command performs the same function as ED{IT}, but compresses spaces and asterisks when the item is filed. When an item created by EEDIT is later edited by EEDIT, it is expanded so that it can be displayed and edited in the normal (uncompressed) manner. (When edited with ED{IT}, however, the item is still displayed in compressed format; if you want ED{IT} to be able to retrieve it in normal format, copy it at TCL with the ECOPY command. See the *Ultimate System Commands Guide* for details about using ECOPY.)

The EDIT-LIST form is used to edit a select-list that has been previously saved by the SAVE-LIST command (see the *Ultimate System Commands Guide* for details about using SAVE-LIST). If the select-list is unspecified, a null item-ID is assumed.

Examples of the ED{IT}, EEDIT, and EDIT-LIST commands are shown in Figure 4-1 on the next page.

Note: In the examples on the next page, the ↵ symbol represents the RETURN key.

<pre>:</pre>	TCL prompt.
<pre>:ED XYZ ITEM1 ↵</pre>	EDIT command.
New item	Specifies that ITEM1 is new item.
Top	Top message from line editor.
.	Editor prompt.
<pre>:EDIT DICT XYZ PRO ↵</pre>	EDIT command (specifies dictionary section).
Top	
<pre>.EX ↵</pre>	EXIT command (exits editor)
'PRO' exited	
<pre>:ED XYZ NO1 NO2 (S) ↵</pre>	EDIT command with multiple item-IDs and the S option. Item NO1 is edited first.
NO1	
Top	
<pre>.EX ↵</pre>	
'NO1' exited	
NO2	Editor automatically re-entered to edit NO2.
Top	
<pre>.EX ↵</pre>	
'NO2' exited	
<pre>:EEDIT XYZ NO1</pre>	EEDIT command (item displayed in same format as ED or EDIT, but is filed in compressed format).
Top	
<pre>.EX ↵</pre>	
'NO1' exited	
<pre>:EDIT-LIST CARS ↵</pre>	EDIT-LIST command
Top	
<pre>.EX ↵</pre>	
'CARS' exited	

Figure 4-1. Examples of EDIT Commands

How the Line Editor Uses Buffers

The line editor uses two variable-length temporary buffers (the current buffer and the update buffer) to edit an item. When the editor is entered, the item to be edited is copied from the file to the current buffer (which initially is buffer 1). Each line (attribute) of the item is numbered in ascending order from Top to EOI (End Of Item).

The line editor uses the current line concept, where the current line is the only line on which editing can take place. A current line pointer points to the current line of the item, and an EOI pointer points to the last line of the item.

As editing is performed, the modified line and all previous lines are copied to the update buffer (which initially is buffer 2).

The editing process continues working on the text in the current buffer. As lines in the item are inserted, changed, or deleted, the line editor builds a new, updated version of the item in the update buffer. Updating must continue in an ascending line number sequence until an F (flip buffers) command is entered; lines preceding the last line changed cannot be edited until the buffers are flipped. The F command merges the updates with the previously existing item, automatically resequences the line numbers, and reverses the roles of the buffers. Now buffer 1 becomes the update buffer, and buffer 2 is the current buffer.

The line editor applies the revisions in the update buffer to the text in the current buffer only when the F command is executed. The F command does not file the text to the disk; it compiles the latest version in the current buffer and repositions the current line pointer to the top of the item. Editing then occurs in the other buffer. You may flip the buffers as often as needed before filing or exiting the item.

This double buffer process is shown in Figures 4-2 and 4-3. Figure 4-2 shows a four-line item in buffer 1 (the current buffer) with the current line pointer positioned at line 2. Two lines (1234 and 567) are then inserted after line 2, as can be seen in buffer 2 (the update buffer). When an F command is issued, the buffers are flipped as shown in Figure 4-3. Here buffer 2 has become the current buffer. Further modifications made to the item are assembled in buffer 1, which has become the update buffer.

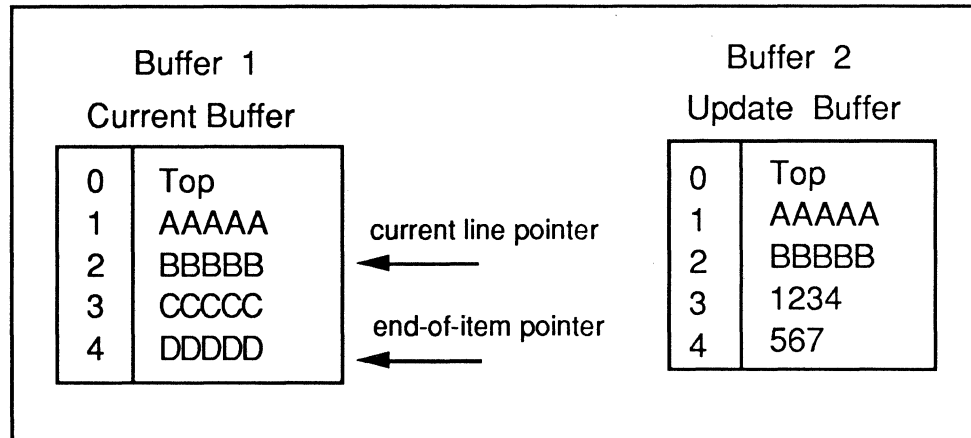


Figure 4-2. Buffer Contents before the F Command

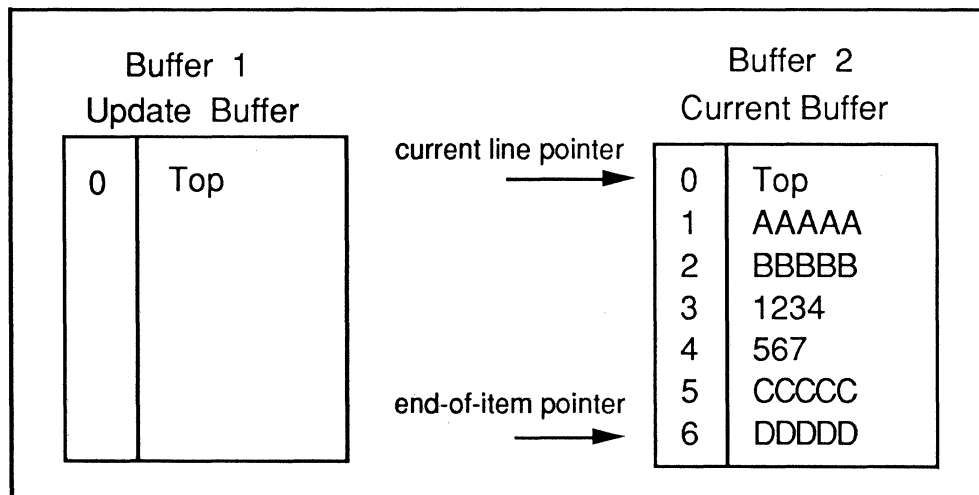


Figure 4-3. Buffer Contents after the F Command

Using the Line Editor

This section presents an overview of the line editor's features. The editor commands involved with the features introduced here are described in detail in Chapter 5. The following topics are covered in this section:

Line Editor Commands	Merging and Cutting Text
Exiting an Item	Searching and Replacing Text
Moving Around in the Item	Deleting Text
Strings	Cancelling Commands
The Wildcard Feature	Prestoring Commands
Inserting Text	Suppressing Output Display

Line Editor Commands

Line editor commands contain from one to three characters and are generally mnemonic. A command can only be entered when the line editor prompt (.) is displayed. The command must appear as the first non-blank input character on a line. It may be entered in either upper or lower case; the following are equivalent:

DE de De dE

Parameters may follow some commands. Blanks may be inserted between parameters for clarity, but blanks embedded within a parameter are not permitted. All commands must end with a carriage return.

Exiting an Item

You can exit an item with or without saving the changes that have been made.

The EX{T} command exits without saving. If changes have been made, the editor first queries to make sure you do not want to save the changes. The EXT form of the exit command returns control to TCL; if the T is omitted, the editor retrieves the next item in the itemlist. If there are no more items, control returns to TCL.

The FI command saves the item and automatically retrieves the next item in the itemlist. If there are no more items, control returns to TCL. Options exist that permit saving to another item or an item in another file, and to overwrite an item with the same name.

Moving Around in the Item

You can move the current line pointer to any line in the item. When the line pointer is repositioned, the new current line is displayed. The line pointer can be moved:

- to the next line by pressing RETURN.
- forward or back to a specific line number, using the G command.
- forward or backward a specified number of lines, using the N or U command.
- to the top or bottom of the item, using the T or B command.
- to an assembly code address (if in assembly format), using the Q command.

All the commands except T and B display the new line.

Strings

A string is defined as a series of characters that is surrounded by a pair of identical, non-numeric delimiters that do not appear within the string itself. See the examples below.

Valid String	Delimiter
/123 AB/	/
.EFG.	.
;. For example, this is a valid string;;	;
PThis is also a valid stringP	P

Strings can be used in the AL, BL, DE, L, and R commands to match with an identical string in the lines being searched. If the string is found, the action specified in the command is executed. In an R (replace) command, a second string can be included; if the first string is found, it is replaced by the second.

The colon (:) is a reserved delimiter; it indicates that the string must begin in column 1 for a match to occur. For example, the following string requires that LOOP be found in columns 1 through 4:

```
:LOOP:
```

However, the following creates a match if LOOP is found anywhere on the line:

```
/LOOP/
```

The ending delimiter of the string is necessary only if further parameters follow the string specification, or if trailing blanks are to be included as part of the string.

The Wildcard Feature

The caret (^) can be used as a wildcard to indicate that any character in that position in a search string is acceptable as a match. For example, the following search string creates a match if the specified line contains AB, followed by any character, and then CD:

```
/AB^CD/
```

The wildcard function may be turned off and on by using the ^ (caret) command.

Inserting Text

You can insert one or more lines of text. The insertion begins on the line following the current line. You can also insert system delimiters.

To insert a single line of text, use the I command, followed by a space and the text to be inserted. The carriage return at the end exits insert mode.

To insert multiple lines use the I command, followed immediately by a carriage return. Type the text and end each line with a carriage return. When all the text has been entered, press RETURN at a new line, rather than entering text. This exits insert mode.

Attribute marks, value marks, and subvalue marks are called *system delimiters*. They can be inserted into the text by typing the following control <CTRL> key sequences when you are in insert mode:

Delimiter	Keys	Display	ASCII Code
attribute mark	<CTRL-^>	^	CHAR(254)
value mark	<CTRL-]>]	CHAR(253)
subvalue mark	<CTRL-\>	\	CHAR(252)

The control key is not displayed, but the ^,], and \ are.

Merging and Cutting Text

Merging text copies it in a new location without deleting it from its original location; cutting text moves it to a new location and deletes it from its original location.

To merge text, use the ME command. You can specify the lines to be merged and the item and file to merge from.

To cut text, use the CL command. You can specify the lines to cut. You cannot cut lines from another item.

Searching and Replacing Text

You can replace entire lines, or you can search for a specified string and replace it with another string. In either case, the action begins at the current line and continues for the specified number of lines.

To replace entire lines, use the R command followed by the number of lines to replace. Insert mode is entered automatically, and you are prompted to begin entering text by the display of the first line number to be replaced.

To replace one string with another, use the R command followed by the number of lines to be searched. Include the search string and the replacement string. If the search string is not included, the replacement string is placed at the beginning of every line. You can specify whether all occurrences of the search string are to be replaced, or just the first occurrence.

Deleting Text

There are two ways to delete text:

- One or more lines, beginning with the current line, can be deleted using the DE command.
- Specified text can be replaced using the R command; if the replacement string is null, the original text is deleted.

Note: In some cases, the line editor permits the cancellation of commands. If a DE or R command was issued by mistake, cancelling the command can recover the deleted text. See the section *Cancelling Commands*.

Cancelling Commands

You can cancel the last command that changed the content of your item, but you must do it before entering any other command that alters the contents of the update buffer. (Actions such as moving around the item, changing the nature of the display, and locating strings do not alter the contents of the update buffer, so they can be issued before the cancel command without interfering with it.)

You can also cancel all the changes made since the last time you executed an F (flip buffers) or an FS (file save) command.

To cancel the last change, use the X command; to cancel all changes since you last flipped the buffers or filed the item, use the XF command.

Prestoring Commands

Up to ten strings of frequently used line editor commands can be prestored and executed later in the same editing session. Multiple commands can be stored in each prestored command. Execution begins at the current line.

When the editor is entered, the command L22 is automatically loaded as P0. When executed, it causes 22 lines to be displayed.

Prestored commands are not saved when the editor is exited.

Suppressing Output Display

Most commands may be optionally preceded by a period, which suppresses text output (but not editor messages) without affecting the operation of the command. Suppressing display may be useful in cases such as editing over a modem.

There are two exceptions:

- A period before a P (prestore) command does not suppress output display. (For example, an L (list lines) command within a prestore command displays text even when the P is preceded by a period.)
- A period before the ^ (wildcard), AS (assembly formatting), HX (hexadecimal display), M (macro expansion), and S (suppress output) commands suppresses the function entirely.

Advanced Techniques

The line editor's basic editing features were described in the previous section. This section will present some sophisticated techniques using the basic features, particularly the searching and replacing function and the use of strings and wildcards.

Creating Blank Lines

A blank line (that is, a null attribute) can be created in several ways:

- If the line contains some text, the text can be replaced with a null:

```
.R/text on line//↵
```

- The single line I (insert) command can be used without entering data. Type I at the prompt, followed by a single blank and RETURN. This inserts a null attribute and returns to the line editor prompt:

```
.I ↵
```

- The I command can be used to enter several null attributes. For each null attribute desired, simply enter the control key sequence <CTRL-^> (control-caret) in the line. This has the effect of inserting an attribute mark in the text. The following example creates three blank lines:

```
.I <CTRL-^> <CTRL-^> <CTRL-^> ↵
```

Breaking Lines

In the editor, each line is a separate attribute that is terminated by an attribute mark placed there by the system. Within a string, the attribute mark is the control key sequence <CTRL-^>. When <CTRL-^> is placed in a string, the line editor breaks the string at that point and starts a new line.

The caret prints when you type it, but the <CTRL> does not. After the buffers are flipped and the lines are listed, the caret is not displayed.

The R (replace) command can be used to insert the attribute mark. For example, the following command inserts an attribute mark between characters C and D:

```
012 ABCDEF
.R/ABCDEF/ABC^DEF/ ↵
```

After flipping the buffers, the line is broken into two shorter lines:

```
012 ABC
013 DEF
```

Similarly, the following replacement inserts an attribute mark at each occurrence of the letter A on the current line (the U option causes the replacement at every occurrence of A):

```
RU/A/A^/ ↵
```

For example, the previous command converts the line

```
023 A MACHINE THAT CAN DO ANYTHING
```

into the following six lines:

```
023 A
023+ MA
023+CHINE THA
023+T CA
023+N DO A
023+NYTHING
```

While inserting text, you may enter <CTRL-^> to create as many lines as desired, one line per <CTRL-^> sequence. For example, entering the following in insert mode:

```
002+ABC^DEF ↵
```

results, when the buffers are flipped, in two lines being created:

```
002 ABC
003 DEF
```

Note: The BL command can also be used to break lines. See Chapter 5 for details.

Trimming Lines

Trimming involves specialized deletion techniques. To trim a line at a specified place, include the key sequence <CTRL-underscore> (control-underscore) in the replacement string of an R command. For example, if a line contains XYZABC, the following command truncates the line after XYZ and discards ABC:

```
R/XYZ/XYZ_/_
```

When blanks are used as the first string in a R command, the line editor searches for a string of blanks of the same or lesser length in the line. It is therefore possible to trim all trailing blanks off an item using one of the following forms:

```
R/          //
R/          /<CTRL-underscore>
```

These forms assume that the blank string is longer than any string of blanks within the lines being trimmed.

The example in Figure 4-4 uses the attribute mark (^) to show the end of the text; the mark is not actually displayed.

```
.L2 _      L (list lines) command.
001 Line1  ^      Line1 with trailing blanks.
002 Line2  ^      Line2 with trailing blanks.
EOI 002

.T _
Top
.R2/          // _      R command has more blanks than
                    Line1 or Line2.

.F _      F (flip buffers) command.
Top
.L2 _
001 Line1^      Line1 trimmed.
002 Line2^      Line2 trimmed.
EOI 002
.
```

Figure 4-4. Trimming the Right End of a Line

It is also possible to trim the left end of a line by replacing a string of carets (wildcards) with a null, as illustrated in Figure 4-5.

032 This is a dog. That is a cat.	The original line.
.R/^^^^^^^^^^^^^^^^^^^^^^//	Replace command.
032 That is a cat.	Edited line.

Figure 4-5. Trimming the Left End of a Line

Using Wildcards to Replace Text

Because the R command ignores spaces preceding the first string delimiter, text on a line can be easily replaced using wildcards. Go to the line to be changed. At the prompt, type the R command and space over to the column before the portion you want to change. Type the first delimiter and enough of the string to make it unique, and fill the rest with wildcards. Then type another delimiter and the replacement string. Press RETURN. The results are shown in Figure 4-6.

.G7 ↵	
007 An instruction, statement, or subroutine	
.R /, st^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^/ or command ↵	
007 An instruction or command	

Figure 4-6. Using Wildcards to Replace Text

Non-Printable Characters

The ASCII character set includes both printable and non-printable characters. Characters that are not printable include the control characters, which have an ASCII hexadecimal value of 00 through 1F.

The line editor displays most control key sequences as a period (.). (The system delimiters <CTRL-^>, <CTRL-]>, and <CTRL-\> are exceptions.) In normal display, the period does not indicate what the character is; however, in hexadecimal display, the actual hexadecimal value is shown.

If you do not know what the control key sequence is, it can be removed by replacing a unique string that includes the control sequence with the string of your choice. The control sequence can be marked with a caret (wildcard) in the first string of the R command. Figure 4-7 replaces the control character and everything that follows it with other text.

```
005 The period .indicates a control key sequence.
.R          /^in^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^/is gone.␣
005 The period is gone.
.
```

Figure 4-7. Removing a Control Character

5 Line Editor Commands

Table 5-1 lists the line editor commands in the same order as they are described in the chapter. The commands are presented in upper case, but lower case is also accepted by the line editor.

All the commands used in the examples are described in this chapter. They are listed alphabetically.

Table 5-1. Line Editor Commands (1 of 3)

Command Syntax	Description
↵ (RETURN)	advances line pointer.
?	displays item name and current line.
^	turns wildcard feature off and on.
A	locates search string again.
AL{n}/string/	appends string to end of line.
AS	turns assembler formatting on and off.
B	goes to bottom of item.
BL{n}/{string} [{p{-q}}]	breaks lines.
C	displays column numbers.
CLm{-n}	cuts lines.

Table 5-1. Line Editor Commands (2 of 3)

Command Syntax	Description
DE{n}/{string}/{p{-q}}	deletes line.
EX{T}	exits item without filing.
F	flips buffers.
FD	deletes item.
FI{{O}}{(new-filename) {new-item-ID}}	files and exits item.
FS{{O}}{(new-filename) {new-item-ID}}	files item without exiting.
{G}{n}	goes to line number.
HX	turns hexadecimal display on and off.
I {text}	inserts lines.
JL	joins lines.
L{n}	lists lines.
L{n}/string{/p{-q}}	locates string.
M	turns macro expansion on and off.
ME{n}/{item-ID} {/m}	merges lines.
ME{n}(filename item-ID){m}	
N{n}	advances line pointer.
P{n} {cmd} {subvalue-mark cmd...}	creates or executes prestored command.
PD{n}	displays prestored commands.

Table 5-1. Line Editor Commands (3 of 3)

Command Syntax	Description
Q/address/	goes to assembly code address.
R{U}{n}{/{string1}/{string2}}{/p{-q}}	replaces string.
S	turns suppress display on and off.
S?	displays item size.
T	goes to top of item.
TB {nn nn nn nn nn...nn}	sets or clears tabs.
U{n}	moves line pointer up.
X	Cancels last change.
XF	Cancels all changes.
Z{p{-q}}	sets display zone.

↵ Command - RETURN

The ↵ (RETURN) command advances the line pointer one line and lists that line.

Syntax ↵ (RETURN)

Description In line editor command mode, RETURN advances the current line pointer one line and lists that line.

```
Top
.↵
001 This is line 1
.
```

RETURN has two other functions:

- All line editor commands are terminated with a RETURN.
- From multiple line insertion, RETURN terminates the current line and presents a new line; if the line is null, RETURN puts the editor back into command mode. From single line insertion, RETURN terminates the line and puts the editor back into command mode.

```
Top
.I ↵
000+ ↵
.
```

? Command - Display Item Name and Current Line

The ? command displays the item-ID of the item being edited and the position of the line pointer within the item.

Syntax ?

Description When ? is entered, the line editor displays the message:

item-ID L *nnn*

Item-ID is the name of the item and *nnn* is the line number where the current line pointer is positioned.

```
.? ↵  
LETTER L 035       Line pointer is at line 35 of item LETTER.
```

^ Command - Turn Wildcard Feature Off and On

The ^ command turns the wildcard feature off and on.

Syntax ^

Description The wildcard feature causes a caret (^) used in a locate or replacement string to match any character. (See the discussion of wildcard use in Chapter 4.)

The ^ command toggles between treating the caret as a specific ASCII character (^ on), and as a wildcard matching any character (^ off).

If the caret is currently treated as a wildcard (the default), the ^ command turns ASCII representation on so that the caret can be treated as a specific ASCII character. The following message is displayed:

```
/^\ on
```

If the caret is currently treated as a specific ASCII character, the ^ command turns ASCII representation off so that the caret can be treated as a wildcard. The following message is displayed:

```
/^\ off
```


Top	
.^ ↓	^ command.
/^ \ on	Caret now treated as an ASCII character.
.P ↓	L22 command stored in prestore 0.
001 7*9	
002 7+9	
003 7^9	
EOI 003	End of item.
.T ↓	T (top) command.
Top	
.R3/^/* ↓	R (replace) command.
003 7*9	Line where replacement occurred.
EOI 003	
.F ↓	F (flip buffers) command.
.P ↓	
001 7*9	
002 7+9	
003 7*9	
EOI 003	
.^ ↓	
/^ \ off	Returns to default.

A Command - Locate Again

The A (locate again) command repeats execution of the last L (locate) command.

Syntax A

Description The line editor begins the search at the current line, using the most recently executed L (locate) command. If no locate has been performed during the current editing session, the editor displays the message:

Cmdnd?

If the locate string cannot be found, the editor stops the search at the end of the item and displays a message similar to the following:

EOI 056

The number indicates the last line in the item.

If the current line pointer is at EOI, the terminal beeps, and the line editor executes the L (locate) command from line 001.

<pre>: ED MYFILE ABC ↵ Top . L/ABC/ ↵ 001 ABC123 . A ↵ 002 123ABC . A ↵ EOI 009</pre>	<p>Original L (locate) command.</p> <p>A command finds next occurrence.</p> <p>A command finds no other occurrences.</p> <p>End of item.</p>
---	--

AL Command - Append Line

The AL (append line) command appends a string to the end of one or more lines.

Syntax

AL{n}/string/

n is the number of lines to which the string is to be appended, beginning with the current line. If **n** is not specified, only the current line is affected.

/ represents a string delimiter. It may be any non-numeric character not included in the **string** itself.

string contains the characters to be appended to the end of each line.

Description

The editor displays each line that is modified. The current line pointer is positioned at the last modified line.

Note: To cancel the AL command, use the X or XF command.

.G6 ↵	G (go) command.
006 Hello	
.AL3/ ABC/ ↵	AL command; appends ABC to three lines.
006 Hello ABC	
007 Goodbye ABC	
008 That's all ABC	

AS Command - Turn Assembler Formatting On and Off

The AS (assembler formatting) command toggles the assembly code switch on and off. This feature is used by assembly programmers to format assembly code source programs.

Syntax A S

Description When the assembler switch is off, the line editor displays the text exactly as it exists in the item. When the assembly switch is on, the file is assumed to be an assembly program, and output from the assembly code source program lines is in assembly listing format.

If the assembler feature is currently off (the default), the AS command enables it, and the following message is displayed:

```
AS on
```

If the assembler feature is currently on, the AS command disables it, and the editor issues the message:

```
AS off
```

This mode may also be turned on when entering the line editor by using the A option on the EDIT command. See Chapter 4.

Some assembly code source programs contain the assembled object code and the original source text separated by value and subvalue marks. When AS is on, these programs look similar to the following:

```
007 01B A00499      LOOP STORE D1      SAVE ACCUMULATOR  
                  object code      source code      comment field
```

With AS off, the line shown above is displayed as:

```
007 LOOP STORE D1 SAVE ACCUMULATOR\01B A00499
```

Turning on assembly formatting does not affect the search columns in the L (locate), DE (delete), or R (replace) commands.

When the AS mode is on, the following commands are affected:

- The S (suppress) command suppresses code in the object field, rather than line numbers.
- The M (macro expansion) command causes macros to be expanded.
- The Q (go to address) command positions the line pointer at a specified assembly code address location.

```

:ED MODES UTIL ↓
Top
.L3 ↓                               L (list lines) command.
001  FRAME 006] FRM: 006\001 7FF00006] ORG 1\001
002  *SYSTEM*UTILITY
003  *1MAY90
.AS ↓                               Turns on assembly formatting mode.
AS on
.T ↓                               T (top) command.
Top
.L3 ↓
001  0001 7FF00006          FRAME 006
      0001
002          *SYSTEM*UTILITY
003          *1MAY90
.S ↓                               Suppresses object code.
Suppress on
.T ↓
Top
.L3 ↓
001          FRAME 006
002 *SYSTEM*UTILITY
003 *1MAY90
.S ↓                               Clears suppress mode.
Suppress off
.AS ↓                               Clears assembler formatting mode.
AS off
.

```

B Command - Bottom

The B (bottom) command positions the current line pointer at the end of the item.

Syntax B

Description The line editor positions the current line pointer at the End Of Item (EOI) and displays a message similar to the following:

```
EOI 166
```

The number indicates the last line of the item.

<pre>.B ↵ EOI 047</pre>	Indicates that line 047 is the last line in the item and that the current line pointer is positioned there.
-----------------------------	---

BL Command - Break Line

The BL (break line) command divides a line into two lines using a specified string as the break point.

Syntax

BL{**n**}/**{string}**/**{p{-q}}**}

- n** indicates the number of lines, beginning with the current line, that are to be searched and in which the break is to be made if the string is found. If **n** is omitted, only the current line is checked.
- /** is a string delimiter. It can be any non-numeric character not present in the **string** itself. The first delimiter is required, even if no **string** is included.
- string** contains the string of characters at the end of which the break is to be made; if it is not specified, the line is broken before the first character.
- p** is the first column to be searched for the **string**. If **p** is not specified, the search begins in column 1. If **q** is specified, **p** must also be specified.
- q** is the last column to be searched for the string. If **q** is not specified, the search continues to the end of the line. If **q** is less than **p**, it is assumed to be the same as **p** (a one-column search).

Description

Each line of **n** lines, beginning at the current line, is searched for the **string**; if it is found, the line is divided. The search for the specified **string** is not column-dependent unless the optional column parameters **p** and **q** are present, or unless the string delimiter is a colon.

Only the first occurrence of the search string in each line causes a break; later occurrences are ignored.

Note: To cancel the BL command, use the X or XF command.

:ED TEST ITEM1 ↵	
Top	
.P ↵	L22 command stored in prestore 0.
001 123ABC456	
002 ABCDEFABCXYZ	
003 ABCXYZ	
004 ABC123	
EOI 004	
.T ↵	T (top) command.
Top	
.BL2/ABC/ ↵	BL command; searches next two lines for ABC.
001 123ABC	First line affected by BL command.
001+456	Rest of first line.
002 ABC	Second line affected.
002+DEFABCXYZ	Rest of second line; second ABC caused no break.
.G3 ↵	
003 ABCXYZ	
.BL// ↵	BL command with no string specified.
003	Line broken before first character.
003+ABCXYZ	Rest of line.
.G4 ↵	
004 ABC123	
.BL/ABC/2-4 ↵	Break line if ABC is in columns 2-4.
EOI 004	
.F ↵	F (flip buffers) command.
Top	
.P ↵	
001 123ABC	Previous line 001.
002 456	Previous line 001+
003 ABC	Previous line 002
004 DEFABCXYZ	Previous line 002+
005	Previous line 003
006 ABCXYZ	Previous line 003+
007 ABC123	Previous line 004. Not broken by BL because
EOI 007	ABC is not in columns 2-4.
.	

C Command - Display Column Numbers

The C (column number display) command prints a list of column numbers across the screen so that the user can easily determine the column position of data on a line.

Syntax C

Description The line editor responds by printing column numbers across the screen. The numbers are printed in a two-row display: the first row shows the tens position; the second row shows single digits.

This can be most helpful when editing fixed-field data, or when formatting display screens.

On most screens, 75 columns are displayed.

```
.C ↵  
      1      2      3      4      5  
123456789012345678901234567890123456789012345678901234...
```

CL Command - Cut Line

The CL (cut line) command cuts the specified lines and inserts them following the current line. The lines are *deleted* from their original location.

Syntax

CLn{-m}

n is the line number to be cut. If **m** is also specified, **n** is the beginning line number of the range of lines to be cut.

m indicates the last line number to be cut.

Description

CL *cuts* the lines from their old location and inserts them following the current line. The lines are not displayed in their new location until the buffers are flipped and the lines listed.

After completion of the CL command, the current line pointer is positioned at the last line number involved in the cut operation.

For information on transferring text without deleting the original, see the ME (merge) command.

Note: To cancel the CL command, use the X or XF command.

```
:ED FILE1 ITEM1 ↵
Top
.P ↵           L22 command stored in prestore 0.
001 Line1
002 Line2
003 Line3
004 Line4
EOI 004
.CL2-3 ↵      CL command (moves lines 2 and 3).
EOI 004
.F ↵         F (flip buffers) command.
Top
.P ↵
001 Line1    }   Lines 2 and 3 moved from original location; only
002 Line4    }   lines 1 and 4 (now numbered 1 and 2) are left.
003 Line2    }   Lines 2 and 3 (now numbered 3 and 4) moved to
004 Line3    }   end of item.
EOI 004
```

DE Command - Delete Line

The DE (delete) command causes one or more lines to be deleted from the item.

Syntax

DE{**n**}/{**string**}/{**p**{-**q**}}

n is the number of lines to be deleted. If **n** is omitted, only the current line is deleted. If a **string** is specified, **n** is the number of lines to be searched for the **string**.

/ is a string delimiter. It can be any non-numeric character not present in the **string** itself.

string contains the characters to be searched for; if it is null, a match is assumed and the line is deleted.

p is the first column to be searched for the string. If **p** is not specified, the search begins in column 1. If **q** is specified, **p** must also be specified.

q is the last column to be searched for the string. If **q** is not specified, the search continues to the end of the line. If **q** is less than **p**, it is assumed to be the same as **p** (a one-column search).

Description

In the simple DE command, the specified lines are deleted. The current line pointer remains at the line where the last deletion occurred, allowing other operations to be performed there.

When a **string** is specified, the number of lines specified by **n** are searched; any lines containing the string are deleted. Lines that are deleted are listed, and the current line pointer is advanced the number of lines specified by **n**.

Note: To cancel the DE command, use the X or XF command.

```
:ED TEST ITEM ↵
Top
.P ↵                L22 command stored in prestore 0.
001 123XYZ
002 AAAAAAA
003 XYZ123
004 ABABABAB
EOI 004
.T ↵                T (top) command.
Top
.DE2 ↵              DE command (deletes 2 lines).
.F ↵                F (flip buffers) command.
Top
.P ↵
001 XYZ123
002 ABABABAB
EOI 002              Original lines 1 and 2 deleted.
.T ↵
Top
.DE99/123/ ↵        DE command (deletes lines containing 123).
001 XYZ123          Deleted line is listed.
EOI 002
.F ↵
Top
.P ↵
001 ABABABAB        The item after deletions.
EOI 001
.
```

EX Command - Exit Item without Filing

The EX (exit) command exits the item without saving it and retrieves the next item. If there are no more items, control returns to TCL. The EXT form of the command exits both the item and the line editor, returning control unconditionally to TCL.

Syntax

EX{T}

T cancels the EDIT command and returns control unconditionally to TCL.

Description

The EX format exits from the item. If the EDIT command entered at TCL specified only one item or if this is the last item to be edited, the system returns to TCL.

If the EDIT command specified multiple items, the current item is exited; control returns to the line editor to edit the next item specified in the EDIT command; if there are no additional items, control returns to TCL. The EXT form cancels the EDIT command and returns control unconditionally to TCL.

If changes were made, the following prompt is displayed before the exit:

 Sure (Y/N=CR) ?

To exit the item without saving changes, enter **Y**. To cancel the command and remain in the editor, enter **N** or press RETURN. To exit and save changes, use the FI (file item) command.

When the item is exited, the editor displays a message similar to the following:

 '*item-ID*' exited

When you exit, the current version is not filed. If the item has never been filed, it is lost.

```
:ED FILE ITEM1 ↵
.Top
.EX ↵      Exits editor and returns to TCL (no other items to edit).
'ITEM1' exited.
:

.ED FILE ITEM1 ITEM2 ITEM3 ↵
ITEM1
Top
.EX ↵      Exits current item; re-enters editor for next item.
'ITEM1' exited.
ITEM2
Top
.EXT ↵     Exits editor and returns to TCL.
'ITEM2' exited
:
```

F Command - Flip Buffers

The F (flip buffers) command toggles the line editor's buffers and merges updated lines into the item.

Syntax F

Description Updated lines are merged with the previously existing item by flipping the current buffers. This merges the editor's buffers so that the buffer being edited becomes the new current buffer. The current line pointer is set to Top (line 000).

If desired, the item stored in the current buffer may be edited further. Because editing operations must continue in ascending line number order, flipping the buffers is necessary before a line earlier than the current line can be modified. Flipping the buffers is also necessary before recent changes in the item can be listed.

Note: Flipping the buffers does not save (file) the item.

For an explanation of how the line editor uses the double buffer concept, see Chapter 4.

. F ↵	F command.
Top	Line pointer at line 000.
.	

FD Command - Delete Item

The FD (file delete) command deletes the current item from the file.

Syntax

FD

Description

No warning prompt is given unless changes have been made to the item. In that case, the following message is displayed:

```
Sure (Y/N=CR) ?
```

To delete the item, enter **Y**. The item is deleted and the next item in the itemlist is entered. If there are no more items in the itemlist, control returns to TCL. To cancel the command and remain in the editor, enter **N**.

Multiple deletes can be accomplished by the use of a P (prestore) command, or by the DELETE system command at TCL.

Note: A deleted item is erased from the system and is no longer accessible to any Ultimate software. However, if you have deleted an item in error, you may be able to recover it via the TCL command RECOVER-FD. For more information on the RECOVER-FD command, see the Ultimate System Commands Guide.

```
. ED TEST ITEM1 ↵
Top
. FD ↵
'ITEM1' deleted
:
```

FI Command - File Item

The FI (file item) command saves the item to the disk. If other items remain to be edited, control is passed to the next item in the itemlist; if there are no more items, control returns to TCL. The extended form of the command permits filing the item with a different item-ID in the current file or to a different file with the same or a different item-ID.

Syntax

FI{O}{(new-filename) {new-item-ID}}

O indicates that the filing process can overwrite an item with the same name.

new-filename saves the item into a different file. The file must already exist. The filename must be preceded by a left parenthesis, with no space between the **FI**{O} and the parenthesis.

new-item-ID saves the item with a new item-ID.

Description

The buffers are flipped automatically before the save begins. After the item is filed, the editor displays the following message:

```
'item-ID' filed
```

If the itemlist in the EDIT command contained more than one item-ID, control returns to the line editor so that the next item can be edited. If only one item was specified or if the current item was the last in the list, control is transferred to TCL, or to the command that called the editor.

The extended format of the FI command allows you to file the item currently being edited either with a different item-ID in the current file, or with the same or a different item-ID in a different file.

Note: Filing the item under a different item-ID or to a different file leaves the original item unchanged.

If the new-filename does not exist, or if the O option is not included in the extended format and the line editor finds an existing item with the

specified filename and item-ID, it displays the following message and returns control to the edit command prompt:

Cmnd?

:ED AFILE ABC ↵	Applies to all examples below.
Top	
.FI ↵	Saves with same name.
'ABC' filed	
:	
Top	
.FI XYZ ↵	Saves as XYZ; ABC is unchanged.
'XYZ' filed	
:	
Top	
.FI (BFILE ↵	Saves as ABC in file BFILE; ABC in current file is unchanged.
'ABC' filed	
:	
Top	
FIO DEF ↵	Saves as DEF, overwriting existing DEF; ABC is unchanged.
'DEF' filed	
:	
Top	
.FI (DICT BFILE DEF ↵	Saves as DEF in dictionary of BFILE; ABC is unchanged.
'DEF' filed	
:	

FS Command - File Save

The FS (file save) command updates an item to the disk and returns control to the line editor for further editing of the item. Extended forms of the command allow saving the item either under a different item-ID in the current file or in a different file with the same or a different item-ID.

Syntax

FS{**O**}{**(new-filename)** **{new-item-ID}**}

O indicates that the filing process can overwrite an item with the same name.

new-filename saves the item into a different file. The file must already exist. The filename must be preceded by a left parenthesis, with no space between the **FS**{**O**} and the parenthesis.

new-item-ID saves the item with a new item-ID.

Description

The buffers are flipped automatically before the save begins. When the file save is completed, the line editor displays the message:

Top

The line pointer is set to 000 (top of the item). Editing may resume at that point.

The extended format of the FS command allows you to file the item currently being edited either with a different item-ID in the current file, or with the same or a different item-ID in a different file.

***Note:** Filing the item under a different item-ID or to a different file leaves the original item unchanged.*

If you save an item to another file or item-ID, the original item-ID and filename remain as the default. For example, assume you edit item1, save it as item2, and then edit some more. If you execute a simple FS command without specifying an item-ID or filename, the text, with all the editing changes, is saved as item1. If you want to continue saving

the item as item2, you must specify that at each save. After the initial save as item2, the overwrite option must be included.

If the new-filename does not exist, or if the O option is not included in the extended format, and the line editor finds an existing item with the specified filename and item-ID, it displays the message and returns control to the edit command prompt:

Cmd?

Hints: Use the FS command often, especially before any operation you are unsure of. If the results are not what you expected, use the EX command to exit the item. When you retrieve the item again, you will have the last saved version.

:ED AFILE ABC ↵	Applies to all examples below.
Top	
.FS ↵	Saves with same name.
Top	
:	
Top	
.FS XYZ ↵	Saves as XYZ; ABC on disk is unchanged.
Top	
:	
Top	
.FS (BFILE ↵	Saves as ABC in file BFILE; ABC in current file is unchanged.
Top	
:	
Top	
.FSO DEF ↵	Saves as DEF, overwriting existing DEF; ABC on disk is unchanged.
Top	
:	

G Command - Go To Line

The G (go to line) command moves the current line pointer to the specified line number and lists the line.

Syntax {G}{n}

n indicates the line number to which the current line pointer is to be moved. The pointer can go forward or backward in the file, but no editing can take place on lines prior to the last one edited until the buffers are flipped.

Description

Either the G or the n is optional, but one must be specified. If just a number is entered at the line editor prompt, a G command is assumed. If the number is omitted or is 0, the line pointer is positioned at Top. The pointer can go forward or backward in the file, but no editing can take place on lines prior to the last one edited until the buffers are flipped.

If n is greater than the last line number of the item, the line pointer is positioned at the EOI.

```
004 Data
.G6 ↵           Goes to line 6.
006 Data
.45 ↵          G assumed; goes to line 45.
045 Data
.G ↵           G0 assumed; goes to top.
Top
.G99 ↵         Number exceeds item size; goes to end of item.
EOI 061
.
```

HX Command - Turn Hexadecimal Display On and Off

The HX (hexadecimal display) command toggles the hexadecimal output switch on and off.

Syntax **HX**

Description The hexadecimal output feature displays output in ASCII hexadecimal (base-16: 0-9 and A through F). When the switch is on, all output is in hexadecimal format. When the switch is off, output is in ASCII character representation.

If hexadecimal display is off (the default), the HX command enables it, and the following message is displayed:

```
Hex on
```

If hexadecimal display is on, the HX command disables it, and the following message is displayed:

```
Hex off
```

The hexadecimal output switch can also be turned on by using the X option in the EDIT command issued at TCL. See Chapter 4 for details.

```
Top
.L ↵          L (list lines) command.
001 THIS LINE IS DISPLAYED IN HEXADECIMAL BELOW.
.T ↵
Top
.HX ↵
Hex on
.L ↵
001 54484953204C494E45204953...
.
```

I Command - Insert Lines

The I (insert) command is used for data entry. Either a single line or multiple lines can be inserted.

Syntax

I {text}

text is the data to be inserted on a single line. If specified, a space must separate the I and the text. The carriage return at the end of the text terminates insert mode. If **text** is omitted, multiple line insertion is assumed.

Description

Whether one line or multiple lines are inserted depends on whether text is included after the I command. If a RETURN immediately follows the I, with no **text** included, multiple line insertion mode is entered, and each RETURN creates a new line. If **text** follows the I command, single line insertion mode is entered, and the terminating RETURN exits insert mode. In single line insertion, a single space must separate the I and the **text**; additional spaces are construed as part of the **text**.

Single Line Insertion

If text is included after the I command, insert mode is entered for that line only; the terminating RETURN exits insert mode and returns control to the line editor command mode. The line is inserted immediately following the line at which the current line pointer is positioned.

Only one blank space is permitted between the I command and the text on the line; additional blanks are considered part of the inserted text. A single space followed immediately by a carriage return inserts a null line.

With single line insertion, the buffers are not automatically flipped when editing a new item; an F command must be issued before the inserted line can be listed.

Multiple Line Insertion

The I command followed immediately by RETURN causes the line editor to enter insert mode. The editor displays the current line number and a plus (+) sign; this indicates that the next keyboard input will be assigned to the current line plus one.

The cursor is positioned immediately to the right of the plus sign, awaiting keyboard input. Until insert mode is exited, each RETURN terminates a line and begins a new one. To exit multiple line insert mode, enter a null input line (a RETURN only).

In a *new* item, the lines begin at 001, and the numbers are incremented as each line is entered. When insert mode is exited in a new item, the buffers are flipped automatically.

```

:ED MYFILE GOODATA ↵
New item
Top
.I ↵
001+New data ↵
002+More data ↵
003+↵           Null line exits insert mode.
Top             Buffers are flipped automatically in new item.
.

```

In an *existing* item, the new lines are inserted after the current line. The line number in the prompt is the one pointed to by the current line pointer. The number does not change as long as insert mode is on. If insert mode is entered when the item is first opened (or if the T (top) command has just been executed), the current line pointer is positioned at 000, and the new lines are inserted before line 001. With an existing item, the buffers are not flipped.

Note: To cancel the I command, use the X or XF command. (However, with multiple line insertion in a new item, initial data entry cannot be cancelled, because the buffers are automatically flipped when insert mode is exited.)

```
:ED MYFILE OLDITEM ↵
Top
.I ↵
000+Data ↵           At Top, insertion begins following line 000.
000+↵               Null line exits insert mode.
.                   F command not automatic.
```

```
:EDIT TEST ITEM4 ↵
New item
Top
.I data ↵           Single line insertion (new item).
EOI 000
.F ↵               F command not automatic.
Top
.L5 ↵              L (list lines) command.
001 data
EOI 001
```

```
:EDIT ABC ITEM5 ↵
Top
.P ↵               L22 command stored in prestore 0.
001 ABCDEFG
002 HIJK
EOI 002
.G1 ↵
001 ABCDEFG
.I 12345 ↵         Single line insertion (old item).
.F ↵               F command not automatic.
Top
.P ↵
001 ABCDEFG
002 12345           Location of inserted line.
003 HIJK
EOI 003
.
```

```

:EDIT TFILE TEST ↵
Top
.P ↵                               L22 command stored in prestore 0.
001 LINE 1
002 LINE 2
003 LINE 3
EOI 003
.T ↵
Top
.I ↵                               Multiple line insertion (old item).
000+New line A ↵                   New lines input.
000+New line B ↵
000+↵                               Insert mode terminated.
.G2 ↵                               G (go to) command.
002 LINE 2
.I ↵
002+New line C ↵                   New line input.
002+↵                               Insert mode terminated.
.F ↵                               F command not automatic.
Top
.P ↵
001 New line A
002 New line B
003 LINE 1
004 LINE 2
005 New line C
006 LINE 3
EOI 006
.

```

JL Command - Join Lines

The JL (join lines) command appends the current line to the end of the preceding line.

Syntax JL

Description Only two lines may be joined with each command. The current line is deleted after it has been appended to the end of the preceding line.

Note: To cancel the JL command, use the X or XF command.

```
: ED TEST ITEM1 ↵
Top
.P ↵           L22 command stored in prestore 0.
001 Line1
002 Line2
003 Line3
EOI 003
.JL ↵         JL command.
002 Line2Line3  Current line is appended to preceding line.
EOI 003        Buffers are not yet flipped; EOI is still 003.
.F ↵         F (flip buffers) command.
Top
.P ↵
001 Line1
002 Line2Line3
EOI 002        EOI is now 002.
```

L Command - List Lines

The L (list lines) command lists the specified number of lines in the current buffer, starting from the current line plus one.

Syntax

L{n}

n indicates the number of lines to list. If it is omitted, only the next line is listed.

Description

If **n** is greater than the number of lines from the current position to the end of the item, all the lines to EOI are listed.

If the line pointer is at EOI, the terminal beeps when the command is issued, and the line editor lists the next **n** lines starting from line 001.

The L command positions the current line pointer at the last line listed.

Note: To list recent changes, the buffers must be flipped.

Top	
.L ↵	L command with no <i>n</i> parameter.
001 data	Only one line listed.
.L99 ↵	L command where <i>n</i> exceeds item length.
002 data	
.	
.	
048 data	
EOI 048	
.L ↵	L command from EOI.
Top	
001 data	
.	

L Command - Locate String

The L (locate) command searches for characters that match a specified string, starting at the current line.

Syntax

L{n}/string{/p{-q}}

n indicates the number of lines to be searched for the designated **string**. If it is omitted, the next occurrence of the **string** is searched for, and the search ends when it is found or the end of the item is reached.

/ is a string delimiter. It can be any non-numeric character not included in the **string** itself.

string represents the group of characters to be searched for. It must be specified and cannot be null.

p is the first column to be searched for the **string**. If **p** is not specified, the search begins in column 1. If **q** is specified, **p** must also be specified.

q is the last column to be searched for the **string**. If **q** is not specified, the search continues to the end of the line. If **q** is less than **p**, it is assumed to be the same as **p** (a one-column search).

Description

The search begins on the line following the current line; to search the entire item, you must be at Top. For example, if the line pointer is positioned at line 3, the following command searches lines 4 and 5 for the **string** XYZ:

```
L2/XYZ/
```

The parameters **p** and **q** restrict the search to specified columns. If **q** is less than **p**, it is assumed to be equal to **p** (a single column search).

For commands without the **n** parameter, if the **string** is located, the line pointer advances to that line and the line is listed. For commands with the **n** parameter, all occurrences of the **string** within **n** lines are found and listed, and the line pointer is incremented by **n** lines from its position at the start of the command. Therefore, the line pointer might not be set at the last line listed. For example, the following L instruction positions the line pointer at line 9, not line 8.

```
005 data
.L4/PRINT/ ↵
006 I PRINT
008 YOU PRINT
```

If the current line pointer is at EOI, the terminal beeps, and the line editor searches **n** lines beginning with line 001.

<pre>:ED MYFILE TEST ↵ Top .L8 ↵ 001 FIRST ITEM 002 SECOND ITEM 003 ITEM THREE 004 ITEM FOUR 005 NO MORE ITEMS EOI 005 .T ↵ Top .L/ITE/ ↵ 001 FIRST ITEM .T ↵ Top .L5/ITEMS/ ↵ 005 NO MORE ITEMS EOI 005 .T ↵ Top .L9/IT/1-2 ↵ 003 ITEM THREE 004 ITEM FOUR EOI 005 .L:ITEM: ↵ Top 003 ITEM THREE EOI 005 .T ↵ Top .L/N/1 ↵ 005 NO MORE ITEMS EOI 005 .</pre>	<p>L (list lines) command.</p> <p>Locate first occurrence only of ITE.</p> <p>T (top) command.</p> <p>Five lines checked for ITEMS.</p> <p>Locate IT in columns 1 through 2.</p> <p>Locate next occurrence of ITEM, beginning in column 1.</p> <p>From EOI, editor goes to top.</p> <p>Locate next occurrence of N in column 1.</p>
---	---

M Command - Turn Macro Expansion On and Off

The M (macro expansion) command is used by assembly programmers to expand macros in assembly code source program listings.

Syntax M

Description When the AS (assembly formatting) mode is in effect, the M command expands the display of macros. You can turn the M command on with AS off, but it has no effect.

The M command toggles between displaying expanded macros and suppressing the display.

If macro expansion is off (the default), the M command enables it, and the following message is displayed:

```
Macro on
```

If expansion is on, the M command disables it, and the following is displayed:

```
Macro off
```

This mode may also be turned on when entering the line editor by using the M option of the EDIT command. See Chapter 4.

```
.AS ↵      Turn AS on; Macro still off.
AS on
.74 ↵      Display line 74 with Macro off.
074 0045 4F330C3C          BCH R15,X'33',CHGO25 ; BRIF 3040
.M ↵       Turn Macro on.
Macro on
.74 ↵      Display line 74 with Macro on.
074                      BCH R15,X'33',CHGO25 ; BRIF 3040
                          +BCL X'33',R15,CHGO25
                          0045 4F330C3C          +BC: X'33',R15,CHGO25,6
```

ME Command - Merge Lines

The ME (merge lines) command is used to merge one or more lines into the current item from either the current item or another item. The item may be in the current file or another file.

Syntax

ME{n}/{item-ID}{/m}

ME{n}(filename item-ID){m}

n indicates the number of lines to merge. If it is omitted, one line is merged.

/ delimits the **item-ID** when both items are in the same file. It may be any character (except parentheses) not present in the **item-ID**. The first delimiter is required in all circumstances; both are required when **m** is specified.

item-ID is the name of the item containing the lines. If it is omitted, the current item is assumed. If a **filename** is given, an **item-ID** must also be specified.

() enclose the **filename** and **item-ID**; they are required when the items are in different files.

filename is the name of the file containing the item. If it is omitted, the current file is assumed. The **filename** must be preceded by a left parenthesis and followed by at least one blank.

m indicates the beginning line number to copy. If omitted, the copying begins from line 001.

Description

The lines are inserted after the current line. The ME command with no parameters specified merges the first line of the item currently being edited. The merged lines are not displayed until the buffers are flipped and the affected lines are listed.

For information on transferring text and also deleting the original, see the CL (copy lines) command.

Note: To cancel the ME command, use the X or XF command.

```
:EDIT FILE1 ITEM1 ↵
Top
.P ↵                               Displays contents of ITEM1.
001 11111
002 22222
EOI 002
.EX ↵
'ITEM1' exited
:EDIT FILE1 ITEM2 ↵
Top
.P ↵                               Displays contents of ITEM2.
001 AAAAA
002 BBBBB
003 CCCCC
EOI 003
.G2 ↵                               Current line pointer is at line 002.
002 BBBBB
.ME2/ITEM1/1 ↵                     ME command; merges two lines
.F ↵                               from ITEM1, starting with line 1.
Top
.P ↵                               Displays merged item.
001 AAAAA
002 BBBBB
003 11111 }
004 22222 }                       Two lines from ITEM1
005 CCCCC                           inserted after line 2.
EOI 005                               Rest of original ITEM2.
.
```

N Command - Next

The N (next) command advances the line pointer by the specified number of lines and lists the new current line.

Syntax N { n }

n indicates the number of lines to advance.

Description If **n** is omitted or is zero, the current line is listed, and the line pointer is not changed.

If **n** is greater than the number of lines remaining from the current line pointer to the end of the item, the current line pointer is positioned at EOI, but the line is not listed.

Note: *N is the opposite of the U (up) command.*

For information on going to a specific line number, see the G (go to) command.

005 Data	
.N ↵	N command with no parameter.
005 Data	Same line listed.
.N4 ↵	N command with <i>n</i> parameter.
009 Data	Advances four lines.
.N6 ↵	N command with <i>n</i> exceeding item length.
EOI 011	

P Command - Prestore

The P (prestore) command has two functions: it is used to create and store a string of line editor commands, and it is used to invoke a previously stored string of commands.

Syntax

P{n} {command} {subvalue-mark command ...}

- n** is the prestore command number (0 through 9) to be created or executed. If the number is omitted, P0 is assumed.
- command** can be any legitimate line editor command. There must be a space between the P instruction and the first **command**.
- subvalue-mark** is the key sequence <CTRL-**>**. It is used to separate multiple editor commands.

Description

A maximum of 10 prestore commands (P0 through P9) can be created. To list the commands currently available, use the PD command, which is described alphabetically in this chapter.

Any prestore can be changed by creating another prestore with the same name. Once a prestore is created, it is retained until the editor is exited and control is returned to TCL; prestores are not saved when the editing session is finished. However, prestores are retained from item to item during a session.

When the editor is entered, the following prestore command is automatically created:

```
P0 L22
```

This command lists 22 lines.

Operation of the P command depends on whether you are creating a new command or executing one that was previously created.

Creating a Prestore Command

To create a prestore instruction, enter P, a number from 0-9, and one or more line editor commands. When two or more commands are placed in one prestore instruction, separate the commands with subvalue marks (use the <CTRL-\> keystrokes to enter each mark). All valid line editor commands are usable, including other prestore instructions.

Each prestore command buffer is allocated 100 bytes; if the command exceeds 100 characters, the excess is put into the next buffer but is considered part of the same command. To create a prestore in P1 that is 150 characters long, do not put anything in P2.

The following are examples of prestore commands:

```
P1 DE5 ␣  
P2 T\RU100/DOG/CAT/\F\RU100/dog/cat/\FI ␣
```

P1 deletes five lines. P2 replaces all occurrences of DOG or dog with CAT or cat, respectively, in the first hundred lines of text and files the item. The forward slashes are string delimiters; the backslashes indicate the subvalue marks (<CTRL-\>) that separate the commands.

The following limitations exist on use of the prestore command:

- Attribute marks (<CTRL-^>) and segment marks (<CTRL-_->) cannot be stored in a prestore.
- A carriage return cannot be stored in a prestore, because a carriage return terminates all line editor commands, including the prestore. This means that the single line insert command can be used in a prestore, but the multiple line insert command cannot, because the carriage return after the first line terminates the prestore command.

Executing a Prestore Command

To execute a previously created prestore, move to the line where execution is to begin and enter Pn , where n is the number of the prestore command you want to invoke.

If an undefined prestore is called, the editor displays the message:

Cmd?

Using Prestore Commands for Repetitive Tasks

To use a prestore command for a repetitive task, the command may either be activated each time it is to be used, or it may call itself. A prestore command that calls itself terminates only when it runs out of items to process. This means that a prestore that calls itself must have an EX, FI, or FD in the command string. If it does not have such an item exit command in the string, it loops indefinitely in the current item. The only exit from this condition is to press the <BREAK> key, which calls the system debugger, and then enter END to exit the debugger.

The primary use of a prestore calling itself is to manipulate many items with a single command string initiated once. It is particularly useful for searching for specified strings in text files and replacing them as necessary.

Note: To cancel the results of executing a P command, use the X or XF command.

The following example searches the first 500 lines of each item in a BP (BASIC program) file for the name GENERAL.LEDGER and replaces it with GL. Then it files the item, retrieves the next one, and repeats the process.

```
: ED BP * ↵
ITEM-ID                               First item.
Top
.P1 RU500/GENERAL.LEDGER/GL/\FI\P1 ↵  Creates the prestore.
.P1 ↵                                  Calls the prestore.
```

When the replacement is complete in each item, the following messages are displayed:

```
EOI nnn
'ITEM-ID' filed.
NEXT-ITEM-ID
Top
.
.
.
```

The process is repeated with each item in the itemlist. When the last item has been processed, control returns to TCL.

PD Command - Prestore Display

The PD (prestore display) command displays one or all current P (prestore) instructions.

Syntax

PD{n}

n specifies the number (0-9) of the prestore instruction to be displayed. If it is omitted, all stored instructions are displayed.

Description

The line editor does not save prestore instructions when the editor is exited, so only those prestores created during the current editing session can be displayed. When multiple commands are stored in a prestore, each command is displayed on a separate line.

```
:ED BP ITEM1 ↵
Top
.P1 RU500/GENERAL.LEDGER/GL/\FI\P1 ↵
.PD ↵
P0 L22                               Default prestore.
P1 RU500/GENERAL.LEDGER/           Second prestore.
  FI
  P1
```

Q Command - Go to Address

The Q (go to assembly address) command is used by assembly programmers to position the line pointer at a specified address location in the assembly object code and display its contents.

Syntax

Q/address/

/ is the **address** delimiter. It may be any non-numeric character not present in the **address** itself.

address is the hexadecimal address of the object code that is to be located and its contents displayed.

Description

When the AS (assembly formatting) mode is in effect, the Q command allows the line editor to access and display the contents of the specified **address** location.

If the Q command is given when AS is off, the following message is displayed:

```
Cmdnd?
```

If the exact address cannot be found, the command finds the closest preceding **address** and displays its contents.

```
.AS ↵
AS on
.Q/FF/ ↵   Locate hexadecimal address FF in the assembly object
             code and display its contents.
131 00FF A06BD1  DIVX SCALE
```

R Command - Replace

The R (replace) command is used to replace a number of lines, or to replace one character string with another character string.

Syntax

R{U}{n}{/}{string1}/{string2}{/p{-q}}

- U** replaces all the occurrences of **string1** in a line. If it is omitted, only the first occurrence is replaced.
- n** in line replacement, **n** indicates the number of lines (starting from the current line) to be replaced. If it is omitted, only the current line is replaced. In string replacement, **n** indicates the number of lines to be searched for **string1**. If it is omitted, only the current line is searched.
- /** is the string delimiter. It may be any non-numeric character not included in the **string** itself.
- string1** is the group of characters searched for. If it is null, **string2** is inserted at the beginning of each line.
- string2** is the replacement string. If it is null, **string1** is deleted.
- p** is the first column to be searched for **string1**. If **p** is omitted, the search begins in column 1. If **q** is specified, **p** must also be specified.
- q** is the last column to be searched for **string1**. If **q** is not specified, the search continues to the end of the line. If **q** is smaller than **p**, it is assumed to be equal to **p** (a one-column search).

Description

There are two replacement modes: line replacement, and string replacement.

Line Replacement

If only the R (and an optional **n**) is entered, the editor enters line replacement mode. The first line number to be replaced is displayed as a prompt to begin input. As each line is typed and RETURN is pressed, the next line number to be replaced is displayed. When the number of lines specified by the **n** parameter have been entered, the editor command prompt is displayed. At this point the buffers can be flipped and the revised item can be listed using the L command.

While typing in a replacement line, the editor is in insert mode. This mode can be exited and the replacement command terminated by pressing RETURN without entering text. In this case, the original line is not changed.

String Replacement

In the string replacement form of the command, delimiters enclose **string1** and **string2**. The delimiter after **string2** may be omitted if the column specification **{p-q}** is not used.

A null line can be created by having an empty replacement string (/) replace an entire line. A similar result can be obtained by having the replacement string contain only an attribute mark (<CTRL-^>), but this also creates an additional line.

The R command is case sensitive; it searches for **string1** capitalized exactly as it is entered.

To make several corrections within a single line, you may change them one at a time, using the R string command for each, without using the F command in between. On each use of the R command in this case, the command operates on the result of the last command, rather than on the original line. When you are finished, flip the buffers once.

Note: To cancel the R command, use the X or XF command.

:ED F1 ABC ↵	
Top	
.L10 ↵	
001 ABCDEF	
002 ABCDEF	
EOI 002	
.T ↵	
Top	
.R2 ↵	Line replacement command.
001 123ABC ↵	} Replacement lines being input.
002 XXXXXAB ↵	
.F ↵	F (flip buffers) command.
Top	
.R2/AB/HHH/ ↵	String replacement command.
001 123HHHC	} Lines where replacement took place.
002 XXXXXHHH	
EOI 002	
.F ↵	
Top	
.R2/HHH/S/4-6 ↵	String replacement command.
001 123SC	Line where replacement took place
EOI 002	(line 002 did not fit criteria).
.F ↵	
Top	
.R2/HHH// ↵	String replacement command.
002 XXXXX	Line where replacement took place.
EOI 002	
.F ↵	
Top	
.R/***/ ↵	String replacement command.
001 ***123SC	Line where replacement took place.
.	

S Command - Turn Suppress Display On and Off

The S (suppress) command toggles between printing and suppressing line numbers during line editor displays and listings. When assembly code is being output, the object code, rather than line numbers, is suppressed.

Syntax S

Description Each entry of a suppress command toggles the switch. The action taken depends on the current status of the AS (assembly formatting) switch, as follows:

AS	Suppress	Result
Off	Off	Default condition. Line numbers are shown when the item is displayed.
Off	On	Line numbers are suppressed when the item is displayed.
On	Off	Item is assumed to be in assembly language. Object code is included when the item is displayed.
On	On	Item is assumed to be in assembly language. Object code is suppressed when the item is displayed.

When the S command is issued, the line editor displays the new status of the SUPPRESS switch. If the suppress feature is off (the default), the S command enables it, and the following message is displayed:

```
Suppress on
```

If the suppress feature is on, the S command disables it, and the following is displayed:

```
Suppress off
```

This mode may also be turned on when entering the line editor by using the S option of the EDIT command. See Chapter 4.

```
:ED FILE ITEM1 ITEM2 ↵      AS and S both off (default).
Top
.S ↵
Suppress on
.L ↵
This is line 1.              Line number not displayed.
.S ↵
Suppress off
.L ↵
002 This is line 2
.EX ↵
'ITEM1' exited
ITEM2                        AS and S both off.
Top
.G49 ↵
049 MOV SRO,IR ITEM-START\0010 E264E6
.S ↵
Suppress on
.N ↵                          N (next) command.
    MOV SRO,IR ITEM-START\0010 E264E6
.AS ↵
AS on
.S ↵
Suppress off
.N ↵
049 0010 E264E6             MOV  SRO,IR             ITEM-START
.S ↵
Suppress on
.N ↵
049      MOV  SRO,IR             ITEM-START
.
```


S? Command - Display Item Size

The S? (item size) command displays the number of bytes in the item being edited.

Syntax S?

Description In response to the S? command, the line editor displays a message similar to the following:

```
Item length is 1836 bytes
```

The length includes the text of the item and characters associated with the item's creation and storage. The line editor places no limit on item length.

```
:ED TEST ITEM1 ↵
Top
.S? ↵
Item length is 619 bytes
```

T Command - Top

The T (top) command positions the current line pointer at the beginning of the item.

Syntax T

Description After the command is entered, the line editor displays the message and sets the line pointer to line 000:

Top

Note: Although the line pointer can be moved to any line in the item, editing cannot be done on lines above the latest change until the buffers are flipped.

```
010 Text
.T ↵
Top
.
```

TB Command - Set Tabs

The TB (set tabs) command sets new tab positions and clears the old ones.

Syntax

TB {n n n n n...n}

n is a column number where a tab is to be set. The series can contain up to 15 numbers. The numbers must be in ascending order, with each number separated by a blank or a delimiter. If no numbers are included, existing settings are cleared.

Description

When the command is executed, the line editor replaces whatever tab stops were in effect with new stops at the columns indicated. The following command, for example, clears the old tabs and sets four new tab stops.

```
TB 15 20 31 40
```

The current tab stops, if any, are used whenever the editor is in input mode and <TAB> is pressed. Pressing the <TAB> key causes a series of spaces to be inserted, moving the cursor to the next specified tab stop. To backspace over a tab, use the <BACKSPACE> for each blank that was inserted.

Note: The key sequence <CTRL-I> is equivalent to the <TAB> key.

If no tabs are set, pressing the <TAB> key causes a <CTRL-I> character (hexadecimal 09) to be inserted in the item.

When the line editor is exited, the last tab settings are saved. They are retained until they are changed or you log off the system (logging to another account does not affect the settings).

Tabs set by the line editor are identical to those set by the TABS command executed at TCL. Tabs set by either method replace previous tabs set by the other method. For information on the TABS system command, see the *Ultimate System Commands Guide*.

U Command - Up

The U (up) command decrements (moves up) the current line pointer by the specified number of lines and lists the new current line.

Syntax U{n}

n indicates the number of lines to move up.

Description If **n** is omitted or is zero, the current line is listed, and the line pointer is not changed. If **n** is equal to or greater than the number of lines from the current line to the top of the item, the line pointer is reset at line 000.

Note: U is the opposite of the N (NEXT) command.

009 Data	
.U ↵	U command with no parameter.
009 Data	Same line listed.
.U2 ↵	U command with <i>n</i> parameter specified.
007 Data	Moves up two lines.
.U10 ↵	<i>N</i> parameter exceeds number of lines to top of item.
Top	
.	

X Command - Cancel Last Change

The X (cancel last change) command cancels the effect of the last command that physically altered a line.

Syntax X

Description The most recent command that altered a line (changed the update buffer) is cancelled. Actions such as moving around in the item, changing the display, and locating strings do not alter the contents of the update buffer.

The following commands can be cancelled:

AL (append line)	DE (delete line)	ME (merge lines)
BL (break line)	I (insert)	P (prestore)
CL (cut line)	JL (join lines)	R (replace string)

The X command is effective only if executed before another command that alters a line, flips the buffers, or saves the item. Actions that do not alter the contents of the update buffer can be issued before the cancel command without interfering with it.

Note: If multiple string replacements have been made on a single line, the X command returns the line to its original state.

When the command has been cancelled, the editor displays a message similar to the following:

```
L 020
```

The number specifies the line where editing can begin without flipping the buffers (the line of the last uncanceled change, or 000 if no change was made except the one that was cancelled). The current line pointer is still at the line where the X command was specified.

```
.G2 ↵
002 Data
.DE ↵
.G4 ↵
004 Data
.DE ↵
.X ↵      Deletion of line 004 cancelled.
L 002      Location of last uncancelled editing.
.
```

XF Command - Cancel All Changes

The XF (cancel all changes) command cancels the effect of all commands that have physically altered lines since the last F (flip buffers) or FS (file save) command.

Syntax XF

Description All commands that have altered a line (changed the update buffer) since the last flip of the buffers are cancelled. Actions such as moving around in the item, changing the display, and locating strings do not alter the contents of the update buffer.

The following commands can be cancelled:

AL (append line)	DE (delete line)	ME (merge lines)
BL (break line)	I (insert)	P (prestore)
CL (cut line)	JL (join lines)	R (replace string)

When the commands have been cancelled, the line editor displays the message:

```
L 000
```

The 000 indicates that editing can begin from the top of the item. This is possible because the XF command cancelled all editing since the last flip of the buffers. The line pointer is still positioned at the line where the XF command was called, not at line 000. To move to the top, use the T (top) command.

Z Command - Set Display Zone

The Z (zone) command sets column limits for the display of lines.

Syntax `Z{p{-q}}`

p is the first column of the zone. If it is omitted, the zone begins in column 1. If **q** is specified, **p** must also be specified.

q is the last column of the zone. If it is omitted, the zone continues to the end of the line. If **q** is less than **p**, it is assumed to be the same as **p** (a one-column zone).

Description

If both **p** and **q** are omitted, the zone is reset to display all columns. If **q** is less than **p**, then **q** is assumed to be equal to **p** (a one-column zone).

The zone affects all commands that display a line:

↵ (carriage return)	L (list lines)
AL (append line)	L (locate)
BL (break line)	N (next)
DE (delete line)	Q (go to assembly address)
G (go to line)	R (replace)
JL (join lines)	U (up)

In addition, the zone affects the P (prestore) command if it contains a line editor command that causes a display of edited material.

Setting a zone does not affect the search for a string in the BL, DE, L(locate), or R commands; but when the affected lines are displayed, only the columns in the zone can be seen. Setting a zone does not affect line editor messages.

Use of the Z command is illustrated below.


```
:ED FILE ITEM ↵
Top
.L2 ↵
001 This is line 1.
002 Now is the time for all people to come running.
.Z2-9 ↵
.T ↵
Top
.L2 ↵
001 his is l
002 ow is th
.
```

Notes

A Screen Editor Messages

The following prompts, status messages, and error messages are generated by the screen editor.

1 - search string:

Displayed during SEARCH function to request the string to search for. The number in the message increments until all strings have been entered (maximum of nine). Enter the string. When all strings have been entered, press RETURN at the next prompt.

1 - search string: replace string:

Displayed during SEARCH & REPLACE to request search and replacement strings. The number in the message increments until all strings have been entered (maximum of nine). Enter the search string; then enter the replacement string. When all desired strings have been entered, press RETURN at the next prompt.

Are you sure? (Y/N):

Displayed when the E)xit no-save option is selected from the EXIT prompt. To exit without saving, enter Y; to return to the item, enter N.

Case dependent? (N/Y):

Displayed during SEARCH and SEARCH & REPLACE. To search for all possible capitalizations of the search string, enter N; to search for the string exactly as input, enter Y.

CLP

Indicates there is text on the clipboard.

Document does not exist, <RET> to continue:

Displayed during MERGE DOCUMENT function if the item-ID entered does not exist or is in another file. Press RETURN to continue. If desired, enter a different item-ID.

Screen Editor Messages

Duplicate glossary name - Replace? (N/Y)

Displayed when creating a new COMMAND GLOSSARY and the name chosen is already in use. To avoid replacing the existing glossary, enter N; to replace it with new text and editing commands, enter Y.

edit <item-ID> (y/n/q)

Displayed as each item in an itemlist is retrieved. To edit the item press Y; to go to the next item in the list, press N; to quit the editor, press Q. Do not include RETURN in any response to these options.

Enter document name:

Displayed when DOCUMENT key is pressed. Enter an item-ID of an item in the same file or of a new item, or press RETURN to have the current item-ID displayed.

Enter end line number:

Displayed during MERGE DOCUMENT function. Enter the ending line number of the text you want to merge.

Enter E)xit no-save U)pdate S)ave & exit

Displayed when the EXIT key is pressed and no unfinished command is pending. To exit without saving, enter E; to save the item and continue editing, enter U; to save and exit, enter S.

Enter glossary name:

Prompt to select a name for a new COMMAND GLOSSARY. Enter a number (0-9) or a name (maximum of 10 characters, the first being alphabetic).

Enter merge document name:

Displayed during MERGE DOCUMENT function. Enter the item-ID of the item to be merged into current item.

Enter personal glossary 0-9 or glossary name:

First COMMAND GLOSSARY prompt. To execute a previously stored glossary, enter its number or name; to create a new glossary, press COMMAND GLOSSARY again.

Enter start line number:

Displayed during MERGE DOCUMENT function. Enter the beginning line number of the text you want to merge.

Enter T)op B)ottom or Line number

Displayed when the PAGE key is pressed. To go to the top of the item, press T (do *not* press RETURN); to go to the bottom, press B (do *not* press RETURN); to go to a line number, enter the number (press RETURN).

Enter U)pper L)ower F)irst:

Displayed when the !)case option is chosen from the SELECT prompt. To change the selected text to all capitals enter U; to change to all lower case, enter L; to have only the first letter of each sentence capitalized, enter F.

Filename?

Displayed when the SE command is issued without a filename and item-ID. Enter the filename and, at the next prompt, enter the item-ID.

GLO

Indicates the COMMAND GLOSSARY function is in operation.

INS

Indicates the screen editor is in insert mode.

Item?

Displayed when item is retrieved with no item-ID specified. Enter item-ID at the question mark.

KF.MSG.xx file error

Displayed when the KF.MSG file or the Q-pointer to it is missing. The screen editor cannot be used until the file can be accessed.

KF.JOB file not defined, <RET> to continue:

Displayed when the KFJOB file or the Q-pointer to it is missing. The COMMAND GLOSSARY function cannot be used until the file can be accessed.

List Changes to printer? N)o D)efault printer A)ux port:

Only displayed when N has been entered at the prompt to verify changes during SEARCH & REPLACE. To omit printing changes, enter N; to print the changes on the default printer, enter D; to send them to an auxiliary port, enter A.

Maximum item size exceeded, <RET> to continue:

Displayed upon exiting a document that has grown larger than 32,000 characters during the editing session. The item can be exited without saving it. To save it, the item must be reduced to less than 32,000 characters by deletion or by moving a portion to another document.

Partial or full? (P/F):

Displayed during SEARCH and SEARCH & REPLACE. To search for the search string either as stand-alone words or embedded in other text, enter P; to search for the string only as stand-alone words, enter F.

Replace? (N/Y):

Only displayed when Y has been entered at the prompt to verify changes during SEARCH & REPLACE. At each occurrence of the search string, the screen editor offers the choice of leaving the text as it is (N) or replacing it (Y). Enter the appropriate letter.

Replace exact? (N/Y):

Displayed during SEARCH & REPLACE. To have the replacement string capitalized as the text it is replacing rather than exactly as you input it, enter N. To have the replacement string capitalized as you input it, enter Y.

RUL

Indicates the ruler has been entered. To return to the text, press the RULER key again.

SEL

Indicates the SELECT function is in progress.

Select D)elete C)opy M)ove !)case

Displayed when the SELECT key is pressed. To edit text in the item, select the text before responding to the prompt; to edit text on the clipboard, respond immediately. To delete the material, enter D. To copy the material to or from the clipboard without disturbing the original, enter C. To move material to or from the clipboard *and delete the original*, enter M. To change the capitalization of text in the item, enter !. (The capitalization of text on the clipboard cannot be changed.)

Selected text exceeds 32000 characters, <RET> to continue:
The item being retrieved is larger than 32,000 bytes. When RETURN is pressed, the first 32,000 characters are displayed. You can move a portion to a second document to reduce the item's size, or you can exit the item and edit it with the line editor.

Undefined glossary, <RET> to continue:
Displayed when an attempt is made to execute a COMMAND GLOSSARY that does not exist. Press RETURN to continue. If desired, choose a different glossary name.

Verify changes? (N/Y):
Displayed during SEARCH & REPLACE. To make all replacements automatically, enter N; to have the screen editor stop at each occurrence of the search string so you can decide whether to replace it, enter Y.

Notes

B Screen Editor Files

The screen editor uses two files on the SYSPROG account. Q-pointers to these files are placed in each user's account by the upgrade procedures. The two files are KF.JOB and KF.MSG.EN. They are described below.

KF.JOB File

KF.JOB holds all users' prestored commands (see "COMMAND GLOSSARY" in Chapter 3). Glossary names can be either numeric (0-9) or alphanumeric. Since there are only ten numbers available and all glossaries are stored in the same file, those with unique alphanumeric names are far less likely to be inadvertently overwritten by someone else.

KF.MSG.EN File

This file holds the screen editor messages, including status indicators, messages requesting additional information, and error messages. The messages are described in Appendix A. The status and message lines, where these messages are displayed, are shown on the terminal illustrated in Figure 2-1 (page 2-3).

Notes

C Line Editor Messages

This appendix presents a list of the messages output by the line editor.

Can not prestore control-^ or control-__

An attempt was made to include <CTRL-^> or <CTRL-__> in a prestore command.

Cmnd?

Illegal editor command was entered.

Col#?

One of the following column errors has occurred:

- Illegal characters follow the recognized end of the command.
- Illegal format for a column-number limit specification.
- Non-numeric characters used for p and q parameters in locate, replace, or merge commands.

EOI *nnn*

End-of-item reached at line *nnn*.

L *nnn*

Line number *nnn*. Depending on the circumstances, the message refers to the current line number, the line number at which editing can begin, or the line number where editing had just occurred.

Not on file

Item specified in merge command is not in the current or indicated file.

Seqn?

Out-of-sequence update; editing must be done in ascending line number sequence until an F (flip buffers) command is entered.

String?

Illegal string specification, or missing string. This message may also occur as a result of an illegal numeric parameter specification which causes a part of the numeric parameter to appear as if it were a string.

Sure (Y/N=CR) ?

Verification prompt when exiting an item when unsaved changes have been made.

Top

Top-of-item (line 000) reached.

'xxx' deleted

Item with name xxx has been deleted from the disk.

'xxx' exited

Item with name xxx has been exited.

'xxx' filed

Item with name xxx has been saved to the disk.

D RUNOFF Commands

RUNOFF is used to produce formatted output, controlling such aspects as document and chapter organization, page formatting and justification, indexing, and text appearance. RUNOFF commands can be inserted into a text file using either editor. When the RUNOFF verb and the file name and item name containing the text is entered at TCL, the text of the item, in RUNOFF format, is directed to either the screen or the printer. It can be corrected with one of the editors and reprinted with RUNOFF.

There are two kinds of RUNOFF commands: embedded commands and line commands.

Embedded Commands

Some of the commands are embedded into lines of text. Examples are commands that call for the capitalization, underlining, or boldfacing of one or more characters. These commands are usually one or two characters, such as the ampersand (&) (which causes the next character to be printed underlined). When these characters should be part of the printed text, preceding them with the underscore command (_) causes RUNOFF to ignore them as commands. Embedded commands are shown in Table D-1.

Line Commands

The rest of the RUNOFF commands are line commands. There may be more than one command on a line, but there can be no text included on that line. Each line of text is processed in text mode; each line of commands is processed in command mode. Each line command is a word or abbreviation preceded by a period. Line commands are shown in Table D-2.

The tables present the command format, abbreviation when it exists, and meaning for each RUNOFF command. The commands are listed here in alphabetical order. For more detail, consult the *Ultimate Editor and RUNOFF Reference Guide* (document number 6838).

Table D-1. RUNOFF Embedded Commands

Command	Meaning
&	underlines next character.
&^	begins underline mode.
&\	ends underline mode.
@	outputs next character in boldface.
@^	begins boldface mode.
@\	ends boldface mode.
>text	begins text at next tab.
<text	ends text at next tab.
^	outputs next character in upper case.
^^	outputs following text in upper case.
\	outputs next character in lower case.
\%	outputs following text in lower case.
_c	overrides a special character.

Table D-2. RUNOFF Line Commands (1 of 4)

Command	Abbr	Meaning
.*		indicates the line is a comment.
.BEGIN PAGE	.BP	begins new page.
.BOX {OFF} {l,r}		encloses text in box.
.BREAK	.B	causes partially filled line to be output.
.CAPITALIZE SENTENCES	.CS	capitalizes first word of sentences.
.CENTER	.C	centers text on line.
.CHAIN {filename} item-ID		chains document to another item.
.CHAPTER {n} name		formats and numbers chapter headings automatically.
.CONTENTS		prints table of contents.
.CRT		directs output to terminal.
.ENDCASE	.EC	ends upper case or lower case command.
.FILL	.F	fills each output line.
.FOOTING		stores next line as a footing.
.HEADING		stores next line as a heading.
.HILITE {c} {OFF}		highlights text.

Table D-2. RUNOFF Line Commands (2 of 4)

Command	Abbr	Meaning
.INDENT {n}		indents next line the indicated number of spaces.
.INDENT MARGIN {n}	.IM {n}	indents margin the indicated number of spaces.
.INDEX text		creates index entry.
.INPUT		reads next line of text from terminal.
.JUSTIFY	.J	fills and justifies text.
.LEFT MARGIN n		sets left margin.
.LINE LENGTH n		sets line length.
.LOWER CASE	.LC	outputs all text except first letter of each sentence in lower case.
.LPTR		directs output to printer.
.NOCAPITALIZE SENTENCES	.NCS	turns off CAPITALIZE SENTENCES.
.NOFILL	.NF	turns off both JUSTIFY and FILL.
.NOJUSTIFY	.NJ	turns off JUSTIFY.
.NOPARAGRAPH		produces continuous, filled text.
.PAGE NUMBER {n}		sets page number.

Table D-2. RUNOFF Line Commands (3 of 4)

Command	Abbr	Meaning
.PAPER LENGTH {n}		sets number of lines per page.
.PARAGRAPH {n}		determines when paragraph begins and amount of indentation.
.PFILE n		directs output to print file.
.PRINT		displays next line on terminal.
.PRINT INDEX		prints index entries.
.READ {filename} item-ID		reads specified item.
.READNEXT		reads next item in select-list.
.SAVE INDEX filename		saves index to file.
.SECTION n text		indicates section within a chapter.
.SET TABS {n...n}		clears old tab stops, and sets new tab stops.
.SKIP {n}	.SK {n}	outputs indicated number of blank lines.
.SPACE {n}	.SP {n}	outputs indicated number of blank spaces.
.SPACING {n}		sets line spacing (single, double, etc.).

Table D-2. RUNOFF Line Commands (4 of 4)

Command	Abbr	Meaning
.STANDARD		sets defaults.
.TEST PAGE		sets rules for beginning new page.
.UPPER CASE	.UC	outputs text in upper case.

Index

␣ command 5-4

? command 5-5

^ command 5-6

- A -

A command 5-8

again command 5-8

AL command 5-9

append line command 5-9

arrow keys 2-6, 3-4

AS command 5-10 thru 5-11

assembler formatting 5-10 thru 5-11

attribute definition items 1-4

attribute length 1-3

attribute marks

line editor 4-9, 4-12 thru 4-13, 4-15 thru 4-16, 5-46

screen editor 2-9

- B -

B command 5-12

<BACKSPACE> key 2-6, 3-5

BEGINNING-OF-LINE key 2-7, 3-6

BL command 5-13

blank lines (*see null attributes*)

BOL key 2-7, 3-6

bottom command 5-12

break line command 5-13

buffers 4-5 thru 4-6

- C -

C command 5-15

cancel all changes command 5-61

cancel change command 5-59 thru 5-60

cancelling a command

line editor 4-11, 5-59 thru 5-60, 5-61

screen editor 2-6, 3-14

caret (^) command 5-6

carriage return 2-5, 2-7, 3-25, 5-4, 5-45

changing case of text 2-11, 3-34

CL command 5-16 thru 5-17

clipboard 2-9, 3-34

column command 5-15

COMMAND GLOSSARY Key 2-11, 3-7 thru 3-9

conventions used in manual viii-ix

cut line command 5-16 thru 5-17

copying text

screen editor 2-10, 3-33

<CTRL-^> 4-9, 4-12 thru 4-13, 5-45

<CTRL-] > 2-9, 4-9

<CTRL-\> 2-9, 4-9

<CTRL-_> 4-14, 5-45

<CTRL-E> 2-7, 3-13

<CTRL-L> 2-7, 3-6

<CTRL-R> 3-24

<CTRL-X> 2-6, 3-14 thru 3-15

current line pointer 4-5 thru 4-6, 4-8

cursor movement

arrow keys 3-4

<BACKSPACE> key 3-5

BOL key 3-6

EOL key 3-13

<HOME> key 3-16

PAGE key 3-21

<PAGE NEXT> key 3-22

PARAGRAPH FWD/BACK keys 3-23

RETURN key 3-25

SENTENCE FWD/BACK keys 3-36

summary 2-6 thru 2-7

TAB FWD/BACK keys 3-37

WORD FWD/BACK keys 3-39

cutting text

line editor 4-10, 5-16 thru 5-17

- D -

DE command 5-18
 key 3-10
<DEL CHAR/LINE> key 3-11
delete command 5-18
deleting control characters 4-15 thru 4-16
deleting text
 line editor 4-10, 5-18, 5-23
 screen editor 2-10, 3-10, 3-11, 3-33
delimiters
 string 4-8
 system (line editor) 4-9, 4-12 thru 4-13, 4-15
 thru 4-16
 system (screen editor) 2-9
DOCUMENT key 3-12
dual cursors 2-3, 2-8, 3-26

- E -

ECOPY command (TCL) 4-3
ED{IT} command (TCL) 4-2 thru 4-4
EDIT-LIST command (TCL) 4-2 thru 4-4
editor compatibility 1-2
editor modes 2-5
EEDIT command (TCL) 4-2 thru 4-4
END-OF-LINE key 2-7, 3-13
EOL key 2-7, 3-13
EX command 4-7, 5-20 thru 5-21
exit command 4-7, 5-20 thru 5-21
EXIT key 2-6, 3-14 thru 3-15
exiting an item
 line editor 4-7, 5-20 thru 5-21, 5-24
 screen editor 2-6, 3-14 thru 3-15

- F -

F command 4-5 thru 4-6, 5-22
FD command 5-23
features
 line editor 1-1 thru 1-3, 4-1
 screen editor 1-1 thru 1-3, 2-1
FI command 4-7, 5-24 thru 5-25

file delete command 5-23
file item command 5-24 thru 5-25
file save command 5-26 thru 5-27
filename
 line editor 4-2
 screen editor 2-2
flip buffers command 4-5 thru 4-6, 5-22
FS command 5-26 thru 5-27
function key layout 2-4

- G/H -

G command 5-28
go to address command 5-50
go to line command 5-28
hexadecimal output on/off command
 5-29 thru 5-30
<HOME> key 2-7, 3-16
HX command 5-29 thru 5-30

- I -

I command 5-30 thru 5-33
<INS CHAR/LINE> key 2-5, 3-17
insert command 5-30 thru 5-33
INSERT key 3-18 thru 3-19
insert mode 2-5, 3-18, 3-25
inserting text
 line editor 4-9, 5-30 thru 5-33
 screen editor 2-5, 2-9, 3-17, 3-18 thru 3-19
item size
 displaying 5-55
 limits 1-3
item-IDs
 displaying (line editor) 5-5
 displaying (screen editor) 3-12
itemlist
 cancelling (line editor) 4-7, 5-20 thru 5-21
 use of (line editor) 4-2
 use of (screen editor) 2-2

- J/K -

JL command 5-34
 join lines command 5-34
 KF.JOB file 3-7, A-1
 KF.MSG.EN file A-1

- L -

L (list lines) command 5-35
 L (locate) command 5-36 thru 5-38
 line editor
 advanced techniques 4-12 thru 4-16
 buffers 4-5 thru 4-6
 commands (*see Chapter 5*)
 invoking 4-2 thru 4-4
 messages (*see Appendix C*)
 using 4-7 thru 4-11
 line length 1-2
 lines
 breaking (line editor) 4-12 thru 4-13, 5-13
 breaking (screen editor) 2-12, 3-25
 copying (screen editor) 2-10, 3-33
 creating (line editor) 4-12 thru 4-13, 5-32
 creating (screen editor) 2-9, 2-12, 3-17
 cutting (line editor) 4-10, 5-16 thru 5-17
 deleting (line editor) 4-10, 5-18
 deleting (screen editor) 2-10, 3-11
 inserting (line editor) 4-9, 5-30 thru 5-33
 inserting (screen editor) 2-5, 2-9, 3-17,
 3-18 thru 3-19
 joining (line editor) 5-34
 merging (line editor) 4-10, 5-41 thru 5-42
 merging (screen editor) 2-11, 3-20
 null (line editor) 4-12, 5-52
 null (screen editor) 2-12, 3-23, 3-25, 3-35
 recovering (line editor) 4-10, 5-23,
 5-59 thru 5-60, 5-61
 recovering (screen editor) 2-10, 3-38
 replacing (line editor) 4-10, 4-12 thru 4-13,
 4-15 thru 4-16, 5-50 thru 5-52
 replacing (screen editor) 2-11, 3-30 thru 3-32
 trimming (line editor) 4-13 thru 4-15

list lines command 5-35
 locate command 5-36 thru 5-38

- M -

M command 5-39 thru 5-40
 macro expansion command 5-39 thru 5-40
 margins 2-8, 3-26
 ME command 5-41 thru 5-42
 MERGE DOCUMENT key 3-20
 merge lines command 5-41 thru 5-42
 merging text
 line editor 4-10, 5-41 thru 5-42
 screen editor 2-11, 3-20
 moving around the item
 line editor 4-8
 screen editor 2-6 thru 2-7
 moving text (screen editor) 2-10, 3-34 thru 3-35

- N/O -

N command 5-43
 next command 5-43
 <NEXT SCREEN> key (*see <PAGE NEXT>*)
 non-printable characters 4-15 thru 4-16
 null attributes
 line editor 4-12, 5-51
 screen editor 2-12, 3-23, 3-25, 3-36
 null command 5-4
 overwrite mode (*see replace mode*)

- P/Q -

P command 5-44 thru 5-47
 PAGE key 2-7, 3-21
 <PAGE NEXT> key 2-7, 3-22
 PARAGRAPH FWD/BACK keys 2-7, 3-23
 PD command 5-48
 prestore command
 line editor 4-11, 5-44 thru 5-47
 screen editor (*see COMMAND GLOSSARY*)
 prestore display command 5-48
 <PREV SCREEN> key (*see <HOME> key*)
 Q command 4-8, 5-11, 5-49

- R -

R command 5-50 thru 5-52
RECOVER-FD command (TCL) 5-23
recovering deleted text
 line editor 4-10, 5-23, 5-59 thru 5-60, 5-61
 screen editor 2-10, 3-38
REFRESH SCREEN key 3-24
replace command 5-51 thru 5-52
replace mode 2-5, 3-18, 3-25
RETURN key
 line editor 5-4
 screen editor 2-5, 2-7, 3-25
ruler 2-8, 3-26 thru 3-27
RULER key 3-26 thru 3-27
RUNOFF Commands (*see Appendix D*)

- S -

S command 5-53 thru 5-54
S? command 5-55
saving items
 line editor 5-24 thru 5-25, 5-26 thru 5-27
 screen editor 3-14 thru 3-15
screen editor
 editing keys (*see Chapter 3*)
 files (*see Appendix B*)
 function key layout 2-4
 invoking 2-2
 messages (*see Appendix A*)
 terminal display 2-3
 using 2-6 thru 2-12
SE command (TCL) 2-2
SEARCH key 3-28 thru 3-29
SEARCH & REPLACE key 3-30 thru 3-32
searching and replacing text
 line editor 4-10, 5-50 thru 5-52
 screen editor 2-11, 3-30 thru 3-32
SELECT key 3-33 thru 3-35
selecting text 2-8, 3-33 thru 3-35
SENTENCE FWD/BACK keys 2-7, 3-36
set tabs command 5-57

split screen 2-11, 3-12
strings 4-8 thru 4-9
subvalue marks
 line editor 4-9, 5-44
 screen editor 2-9
suppress on/off command 5-53 thru 5-54
suppressing output display 4-11
system delimiters
 line editor 4-9, 5-45, 5-45
 screen editor 2-9

- T -

T command 5-56
TAB FWD/BACK keys 2-7, 3-37
tab settings
 line editor 1-2, 5-57
 screen editor 1-2, 2-8, 3-27, 3-37
TABS command (TCL) 3-37, 5-57
TB command 5-57
terminals supported (screen editor) 2-1
text
 changing case (screen editor) 2-11
 copying (screen editor) 2-10, 3-33
 cutting (line editor) 4-10, 5-16 thru 5-17
 deleting (line editor) 4-10, 5-18, 5-23, 5-50
 deleting (screen editor) 2-10, 3-10, 3-11, 3-33
 inserting (line editor) 4-9, 5-9, 5-30 thru 5-33
 inserting (screen editor) 2-5, 2-9, 3-17, 3-18 thru 3-19
 merging (line editor) 4-10, 5-41 thru 5-42
 merging (screen editor) 2-11, 3-20
 moving (screen editor) 2-10, 3-34
 recovering (line editor) 4-10, 5-23, 5-59 thru 5-60, 5-61
 recovering (screen editor) 2-10, 3-38
 replacing (line editor) 4-10, 4-12 thru 4-13, 4-15 thru 4-16, 5-50 thru 5-52
 replacing (screen editor) 2-11, 3-30 thru 3-32
 selecting (screen editor) 2-8, 3-33 thru 3-35
top command 5-56

Trimming lines 4-14 thru 4-15
truncating lines 4-14

- U -

U command 5-58
UNDO key 2-10, 3-10, 3-11, 3-33, 3-38
unprintable characters 4-15 thru 4-16
up command 5-58
value marks
 line editor 4-9
 screen editor 2-9

- W -

wildcard characters 4-9, 4-15, 5-6
wildcard on/off command 5-6
WORD FWD/BACK keys 2-7, 3-39

- X/Z -

X command 5-59 thru 5-60
XF command 5-61
Z command 5-62
zone command 5-62

Index

Notes

FROM:

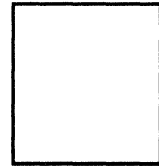
Name: _____ System Number: _____

Company _____

Address: _____

City: _____ State: _____ Zip: _____

Fold and tape. Please do not staple.



**The Ultimate Corp.
717 Ridgedale Avenue
East Hanover, NJ 07936
Attn: Technical Support**

Fold and tape. Please do not staple.

Ultimate

THE ULTIMATE CORP.



ID
Number _____

Suggestion Form

Ultimate welcomes your suggestions. If you have a suggestion or would like to recommend an enhancement, please complete this form. Please attach additional sheets, if necessary.

Name	Phone Number ()	System Number	Date
Suggestion:			

FROM:

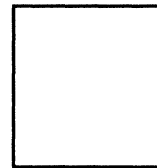
Name: _____ System Number: _____

Company _____

Address: _____

City: _____ State: _____ Zip: _____

Fold and tape. Please do not staple.



**The Ultimate Corp.
717 Ridgedale Avenue
East Hanover, NJ 07936
Attn: Technical Support**

Fold and tape. Please do not staple.

Ultimate

THE ULTIMATE CORP.



ID
Number _____

Reader Comment Form

Ultimate welcomes your comments. If you find a problem or error in this manual, or can suggest an improvement, please complete this form. Please attach additional sheets, if necessary.

Name	Phone Number ()	System Number
Name of Manual	Document Number	Date

Comments:

--

FROM:

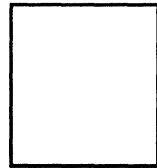
Name: _____ System Number: _____

Company _____

Address: _____

City: _____ State: _____ Zip: _____

Fold and tape. Please do not staple.

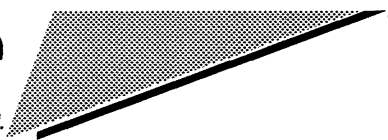


**The Ultimate Corp.
717 Ridgedale Avenue
East Hanover, NJ 07936
Attn: Technical Support**

Fold and tape. Please do not staple.

Ultimate

THE ULTIMATE CORP.



717 Ridgedale Avenue
East Hanover, NJ 07936
(201) 887-9222
FAX (201) 887-9546