

Roach

LEVEL 68
INTRODUCTION TO
EMACS TEXT EDITOR
ADDENDUM A

SUBJECT

Changes to the Manual

SPECIAL INSTRUCTIONS

This is the first addendum to CP31, Revision 0, dated March 1981. Refer to the Preface for "Significant Changes".

Insert the attached pages into the manual according to the collating instructions on the back of this cover. Throughout the manual, change bars in the margins indicate technical changes and additions; asterisks denote deletions. These changes will be incorporated in the next revision of this manual.

Note:

Insert this cover after the manual cover to indicate the updating of the document with Addendum A.

SOFTWARE SUPPORTED

Multics Software Release 10.1

ORDER NUMBER

CP31-00A

February 1983

36238
6C183
Printed in U.S.A.

Honeywell

COLLATING INSTRUCTIONS

To update the manual, remove old pages and insert new pages as follows:

Remove

iii through vii, blank
7-1 through 7-3, blank

Insert

iii through vii, blank
7-1 through 7-4

The information and specifications in this document are subject to change without notice. This document contains information about Honeywell products or services that may not be available outside the United States. Consult your Honeywell Marketing Representative.

Multics Emacs was modelled after the EMACS editor at the Massachusetts Institute of Technology (MIT) Artificial Intelligence Lab. EMACS was written, in TECO by staff members of the MIT AI Lab and the MIT Laboratory for Computer Science. without whose encouragement and support this project would not have been possible.

Significant Changes in CP31-00A

The ^S (string-search) request now accepts a numeric argument.

The ^R (reverse-string-search) request now accepts a numeric argument.

The ESC % (query-replace) request now accepts six new responses.

For purposes of clarity and ease of use, the MPM set has been reorganized. The six former MPM manuals, the Tools manual, and the RCP Users' Guide have been consolidated into a new set of three manuals:

Multics Programmer's Reference Manual (AG91)
contains all the reference material from the former eight manuals.

Multics Commands and Active Functions (AG92)
contains all the commands and active functions from the former eight manuals.

Multics Subroutines and Input/Output Modules (AG93)
contains all the subroutines and I/O modules from the former eight manuals.

The following manuals are obsolete:

<u>Name</u>	<u>Order No.</u>
MPM Peripheral Input/Output	AX49
MPM Subsystem Writers' Guide	AK92
Programming Tools	AZ03
MPM Communications I/O	CC92
Resource Control Users' Guide	CT38

References to the old MPM Commands manual still exist on pages not published with this addendum. When this manual is revised, the references in the text to the old manual will be changed to reflect the new organization.

CONTENTS

	Page
Section 1	Introduction 1-1
	The Terminal 1-2
	The Screen 1-5
	The Keyboard 1-5
	Control Key 1-5
	Escape Key 1-6
	Linefeed Key 1-6
	Delete Key 1-6
	Carriage Return Key 1-6
	The Modem 1-6
	Technical Requirements 1-7
	Logging In 1-7
	Invoking the emacs Command 1-9
	The Initial Display 1-10
	Summary of Terms 1-11
Section 2	Entry and Editing of Text 2-1
	Typing in Text 2-1
	Editing Text 2-2
	Self-Inserting Characters 2-2
	Cursor Moving Requests 2-2
	Deleting Requests 2-3
	Other Requests 2-3
	Summary of Terms 2-4
Section 3	Requests for Moving the Cursor 3-1
	Moving the Cursor Up or Down a Line 3-1
	^P and ^N 3-2
	Moving Forward or Backward a Character 3-2
	^B and ^F 3-2
	The Point 3-4
	Moving to the Beginning or End of a Line 3-4
	^A and ^E 3-4
	Words 3-5
	Moving Forward or Backward a Word 3-5
	ESC B and ESC F 3-5
	Moving to the Beginning or End of the Buffer 3-8
	ESC < and ESC > 3-8
	Moving Through a Buffer Screen by Screen 3-8

CONTENTS (cont)

		Page
	^V and ESC V	3-8
	Moving through a Buffer by Locating Character Strings	3-9
	Numeric Arguments	3-9
	Requests Accepting Numeric Arguments	3-10
	^X=	3-10
	Aborting a Request or Prompt Response ^G	3-11
	Exiting from the editor	3-11
	^X^C	3-11
	Logging Out	3-14
	Summary of Terms	3-14
	Summary of Requests	3-15
Section 4	Deletions	4-1
	Deleting a Character	4-1
	# and the Delete Key	4-1
	^D	4-3
	Deleting Lines	4-4
	@	4-4
	^K	4-4
	Deleting a Word	4-7
	ESC # and ESC Delete Key	4-7
	ESC D	4-7
	Deleting a Region	4-8
	^@	4-9
	Exchanging the Mark and the Point	4-9
	^X^X	4-9
	^W	4-10
	Retrieving Text	4-10
	^Y	4-11
	Popping the Mark	4-14
	Literal Character Entry	4-14
	^Q	4-14
	Summary of Terms	4-15
	Summary of Requests	4-16
Section 5	Files	5-1
	Writing a File Out	5-1
	^X^W	5-1
	^X^S	5-3
	Reading a File In	5-3
	^X^F	5-3
	^X^R	5-4
	Access Restrictions	5-5
	Inserting a File	5-5
	^XI	5-5
	Editing Multiple Buffers	5-6

CONTENTS (cont)

		Page
	Switching Buffers	5-6
	^XB	5-6
	Listing Buffers	5-7
	^X^B	5-7
	Restoring the Screen after a Local Display	5-7
	Executing a Multics Command from within Emacs	5-7
	^X Carriage Return and ^X^E	5-8
	Clearing and Redisplaying the Screen	5-8
	^L	5-8
	Summary of Terms	5-9
	Summary of Requests	5-9
Section 6	Spacing and Formatting	6-1
	Fill Mode	6-1
	ESC X fillon and ESC X filloff	6-2
	Margins	6-2
	^X.	6-3
	^XF	6-3
	Adjusted Right Margin	6-3
	ESC X opt paragraph-definition-type	6-4
	ESC Q	6-5
	Inserting Blank Lines	6-5
	^O	6-5
	Indentation	6-6
	ESC I	6-6
	Summary of Terms	6-8
	Summary of Requests	6-8
Section 7	Searches and Substitutions	7-1
	Searching for a Character String	7-1
	^S and ^R	7-1
	Substituting One Character String for Another	7-2
	ESC X replace and ESC %	7-3
	Summary of Requests	7-4
Section 8	Typing Shortcuts	8-1
	Changing the Case of Words	8-1
	ESC L, ESC U, ESC C	8-1
	Underlining	8-3
	ESC and ^Z	8-3
	Transposing Characters	8-4
	^T	8-4
	Summary of Requests	8-4
Section 9	Help	9-1

CONTENTS (cont)

	Page
Asking For a Request's Description . . .	9-1
ESC ?	9-1
^	9-2
Listing the Emacs Requests	9-3
ESC X make-wall-chart	9-3
Summary of Requests	9-5
Appendix A	
Alphabetized List of Fundamental Mode Requests	A-1
Extended Requests	A-3
Appendix B	
List of Fundamental Mode Requests by Function	B-1
Appendix C	
Glossary	C-1
Index	i-1

ILLUSTRATIONS

Figure 1-1.	A Screen Terminal	1-3
Figure 1-2.	A Terminal Keyboard	1-4
Figure 3-1.	Editor Entry and Exit	3-14
Figure 9-1.	Sample Column of the Wall Chart	9-4



SECTION 7

SEARCHES AND SUBSTITUTIONS

Searching is the term applied to the editor's method of locating a particular character sequence, or character string. You may want to do this simply to move the cursor to a portion of your text where some editing is required, or you may be seeking to replace a character string with something else. This latter replacement of one character string with another is termed substitution. Usually, the easiest way to make a substitution is to search for the string to be replaced, delete it when found, and type in the correction. However, occasionally you wish to replace a particular string throughout the buffer; the substitution requests serve that purpose.

SEARCHING FOR A CHARACTER STRING

Emacs provides two requests just for locating a character string, which you are prompted for. One of these searches forward from the current point to the end of the buffer. The other searches backward from the current point to the beginning of the buffer.

^S AND ^R

The ^S request is named string-search, and searches forward. When you type it, you are prompted:

String Search: _

Type in the characters you wish located, exactly as you expect them to appear, including any spaces where required, and terminate the prompt with CR. If the search succeeds, the point is left immediately after the first occurrence of the character string. If no such string is found, the cursor does not move and Emacs responds in the minibuffer:

```
Search fails.
```

Responding to a search request's prompt with only a carriage return reuses the last search string supplied (the default search string). Thus, you can locate several (or all) occurrences of the same string without continually retyping it.

With a numeric argument, $\wedge S$ searches forward for the specified occurrence of the string; e.g., ESC 5 $\wedge S$ searches forward for the 5th occurrence of the string. If there aren't 5 occurrences, it leaves the point immediately after the last occurrence, and gives you a message saying how many occurrences it found.

The $\wedge R$ request, reverse-string-search, searches backward through the buffer. It works just like $\wedge S$, except it leaves the point in front of the located string (so you do not keep locating the same string). Also, its prompt appears as:

```
Reverse String Search: _
```

With a numeric argument, $\wedge R$ searches backward for the specified occurrence of the string. If there aren't that many occurrences, it leaves the point in front of the last occurrence, and gives you a message saying how many occurrences it found.

SUBSTITUTING ONE CHARACTER STRING FOR ANOTHER

These next two requests search for a specified character string, which you are prompted for, and replace it with another, also prompted for.

ESC X replace and ESC %

The ESC X replace extended request substitutes one string for another at all occurrences of the first string between the current point and the buffer's end. You are prompted for the string to be replaced, and then for the string that is to replace it. If no occurrences of the first string are found, the cursor does not move. Otherwise, the point is left after the last substituted string.

The ESC % request, query-replace, allows you to substitute one string for another selectively. You are prompted for the search string and the replacement string individually; end each prompt with CR. This request searches forward for the first string, puts the point after it, and waits for one of the following responses (type the appropriate keys):

space

replaces this occurrence of the first string with the second. Then searches for the next occurrence of the first string and waits for a response again.

CR

leaves this occurrence of the first string unchanged and searches for the next occurrence of the first string, again waiting for a response after locating it.

. (period)

replaces this occurrence of the first string with the second and then terminates the query replace.

^G

leaves this occurrence of the first string unchanged and terminates the query replace.

ESC

same as ^G.

!

replaces all occurrences of the first string from the current point to the end, without querying again.

, (comma)

replaces this occurrence of the first string with the second, immediately updating the screen. Then searches for the next occurrence of the first string and waits for a response again.

^L

redisplay the screen.

^_

displays a description of the allowable responses (i.e., prints this list).

?

same as ^_.

SUMMARY OF REQUESTS

^S string-search

^R reverse-string-search

(ESC X) replace

ESC % query-replace

HONEYWELL INFORMATION SYSTEMS
Technical Publications Remarks Form

CUT ALONG LINE

TITLE

LEVEL 68
INTRODUCTION TO EMACS
TEXT EDITOR
ADDENDUM A

ORDER NO.

CP31-00A

DATED

FEBRUARY 1983

ERRORS IN PUBLICATION

Empty box for reporting errors in the publication.

SUGGESTIONS FOR IMPROVEMENT TO PUBLICATION

Empty box for providing suggestions for improvement to the publication.



Your comments will be investigated by appropriate technical personnel and action will be taken as required. Receipt of all forms will be acknowledged; however, if you require a detailed reply, check here.

FROM: NAME _____

DATE _____

TITLE _____

COMPANY _____

ADDRESS _____

PLEASE FOLD AND TAPE—
NOTE: U. S. Postal Service will not deliver stapled forms



NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 39531 WALTHAM, MA 02154

POSTAGE WILL BE PAID BY ADDRESSEE

HONEYWELL INFORMATION SYSTEMS
200 SMITH STREET
WALTHAM, MA 02154



ATTN: PUBLICATIONS, MS486

Honeywell

CUT ALONG LINE

FOLD ALONG LINE

FOLD ALONG LINE