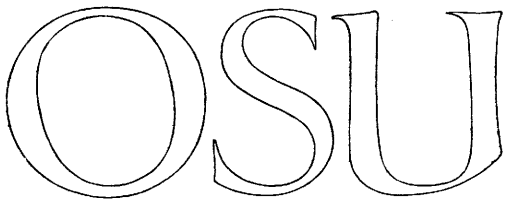
ccm-70-7 (R)

OS-3 EDITOR MANUAL

by
Fred Dayton

January, 1971



COMPUTER CENTER
Oregon State University
Corvallis, Oregon 97331

OS-3 EDITOR MANUAL

by

Fred Dayton

ccm-70-7(R)

Computer Center
Oregon State University
Corvallis, Oregon 97331

TABLE OF CONTENTS

EDIT COMMANDS (listed in alphabetical order)

EDIT

The OS-3 editor is designed to give a remote user (working from a Teletype or CDC 210 and 211 terminals) the facilities for generating and editing text material such as programs and data files. All editing is done using only the printing characters of the terminal (with exceptions noted later). Text is stored in a core scratch pad for editing. The scratch pad must be loaded and edited text must be placed back on a file before it can be used by other routines. By using a core scratch area editing time is significantly reduced, however, there is a limit to the amount of text that may be edited at one time.

The user should probably have a copy of the OS-3 Reference Manual (ccm-70-8).

NOTATION FOR THIS MANUAL

LINE

EDIT uses line numbers to reference text being edited. These line numbers may be assigned by the user, or indirectly, by EDIT. Where several line parameters are needed in commands, a modifying digit will be added (LINE3). A single line is the smallest block of information which can be handled by the Editor. Line numbers associated with the text being edited are printed at the left margin of the Teletype page and are separated from the regular text by a colon (:). If there is no line number associated with a line, a relative line number in parenthesis will be printed in place of the line number; the colon will still be printed. The relative line number is the line count since the last sequenced line.

Line number zero is permanently assigned to a fictitious line which is located immediately before the first line of text in the scratch pad. This line will always be present, but will never be copied by any form of output command.

Limited line number arithmetic of the form LINE+C (where C is a constant) is allowed wherever lines are referenced within EDIT. Addition is the only arithmetic operation allowed. This feature is useful when working with un-numbered lines interspersed among numbered lines. LINE 29+6 would refer to the sixth line past line number 29. No spaces may occur between the line number and the + nor between the + and the constant.

TELETYPE EDITING

The following operations may be used in either the command or the text modes. Command mode is indicated by the] printed at the left margin.

The at sign (@) (upper case letter P) is used to indicate that the user wishes to retype a line. When the @ is printed, the carriage will not return, but all information in that line up to and including the @ is ignored. The user may continue typing on the same line, re-entering the desired information; the line must eventually end with carriage return.

The backslash (\) (upper case letter L) may be used to delete individual characters within a line of typing. For example: ABCC\DEFF\\FG would be taken as ABCDEFG.

The carriage return is used at the end of every line to indicate that the user wishes EDIT to process the line of information just typed. The carriage will return, linefeed, and type the] if in command mode.

The tab key (equivalent to Control Shift I) is used to cause a tab in a way similar to that of an ordinary typewriter. Hitting the tab key will cause the editor to space to the next set stop, whether the Teletype moves or not. Tab stops are preset to 10, 20, 40, and 72.

TV EDITING

This manual is designed basically for the Teletype or other similar 8-bit ASCII character-oriented device. The user wishing to edit on a TV (CDC 210 or 211 terminal) is expected to be familiar with Teletype editing.

Because of the nature of a TV a different approach is needed to handle information in "screen fulls." The top line of a screen is the command entry area. A] will appear in the upper left-hand corner of the screen when in command mode. EDIT always returns to command mode at the end of a screen of input. The screen can be thought of as containing a series of lines, each line terminated by a return. The screen buffer is scanned and each "line" is handled separately. This allows stacking of commands as long as one does not generate any output to the terminal. If one of the commands puts EDIT in text mode then the rest of the lines on the screen are treated as text, and EDIT will return to command mode when the send mark (▲) is encountered.

The main difference between TV and Teletype editing is in typing text into the scratch pad. To enter text type INPUT and S or I parameters if desired followed by a return and the rest of the text until the screen is full. Hit send at the end of the last line of text. When the text is processed a] will be written in the upper left corner of the screen. The rest of the input text should be entered the same way only APPEND followed by a return should be used instead of INPUT. INSERTing and REPlacing text allows only a screen full of corrections to be sent in at once. Again as with INPUT and APPEND, the editor will return to command mode at the end of the screen.

DIRECTORY FILE

EDIT will update a directory file upon request, if one exists and the file is neither abnormal/unavailable nor file protected. Everytime a user changes a file with a COUT, FILE, or OUT command the user's directory file will be updated. Thus the directory file will reflect the last change date. The user should be famaliar with cc-69-1, DEFINE and DIRECT, by Gilbert A. Bachelor.

To request EDIT to update a directory file call EDIT with a parameter string of the form: EDIT,NAME where NAME is the name of the directory. If NAME is omitted and the comma is present EDIT will assume DIRECTRY for the name.

EDIT commands

APPEND,LUN

The APPEND command will add text to the end of the scratch pad from the logical unit specified. Information on this LUN may be in any form generated by EDIT. LUN will be rewound if it is in the range 50 to 59. If no LUN is specified, then the text is taken from the terminal with sequence numbers provided. The sequence numbers continue from the largest sequence number found in the text.

APPEND should be used to input text from a TV. APPEND should appear on the first line of the screen followed by a return and up to one screen full of text, each line terminated by a return. To transmit the screen to EDIT, press send after the last line is typed in.

A name may be substituted for the logical unit number. The name must refer to a currently saved file. This file will be rewound before use.

EXAMPLES:

```
APPEND
APPEND,5
APPEND,ZAP
```

BKSP,LUN,LUN,...,LUN

This command causes each of the logical units listed to back up one record. A logical unit may be listed more than once; the LUN will be backspaced once for each time it is listed.

EXAMPLES:

```
BKSP,65,31
BKSP,23
BKSP,34,34,34,34
```

BKUPC,C

This command is used to change the backup (\) character. This allows the user to type in text with \ as a valid character. C must be one of the 63 printing characters except comma (,).

EXAMPLES:

```
BKUPC,:
BKUPC,Z
BKUPC,*
```

CIN,LUN,DECKID

This command is used to enter information in COSY format into the scratch pad. The scratch pad is cleared before execution of this command. Leading BCD records are ignored; input begins with the first COSY binary record. Line sequence numbers starting with 1 and increasing in increments of 1 are stored with each line. CIN will rewind logical units numbered between 50 and 59, if possible, before execution. If no LUN is specified, then logical unit 54 is assumed. When the last COSY record has been processed, CIN will search forward past file mark. If DECKID is present the file is searched looking for the requested deck. The file must contain 1 or more COSY decks in COSY library format. (See ccm-70-6).

A name may be substituted for the logical unit number. The name must refer to a currently saved file.

EXAMPLES:

```
CIN,45
CIN
CIN,ZAP
CIN,ZILCH,PROGRAM2
```

CLEAR,LUN,LUN,...,LUN

The CLEAR command will reset certain indicators associated with each of the logical units listed. The indicators which are reset are:

BINARY RECORD PROCESSED
FILE MARK JUST READ

The following indicators are not changed:

ABNORMAL/UNAVAILABLE
ADDRESS ERROR
END OF DATA
FILE PROTECT
LOAD POINT
SAVED FILE

EXAMPLES:

```
CLEAR,33,99,71
CLEAR,26
```

COUT,LUN,DECKID,ED

This routine will write out a COSY compressed deck of the text in the scratch pad. If LUN is not specified, logical unit 54 is assumed. COUT will write a file mark,and if the LUN is backspaceable,will write a second file mark and backspace over it. If both DECKID and ED are omitted then no COSY/ card is generated. If either parameter is present then a BCD card of the form DECKID COSY/ ED will be generated. DECKID may be any string of up to 8 BCD characters terminated by a comma. ED may be any number between 1 and 99. If ED is omitted then 01 is used. If DECKID is omitted then blanks are used. (See ccm-70-6).

A name may be substituted for the logical unit number in this command. If a previously defined name is used, the file associated with it will be rewound before use, if the name has not been previously defined, then a new file is created and saved.

EXAMPLES:

```
COUT
COUT,ZAP
COUT,50,ZILCH,22
```

DELETE,NAME

The DELETE command is used to destroy a saved file. The saved file must not be file protected. If NAME was not equipped, then the contents of the file will be destroyed as soon as this command is executed.

EXAMPLES:

```
DELETE,PROGRAM
DELETE,DATA
```

DIRNAME,NAME

DIRNAME is used to change the name of the directory file. The file must already exist.

EXAMPLES:

```
DIRNAME,ZILCH
DIRNAME,MYDIR
```

EQUIP,LUN=<ELEMENT>

The EQUIP statement allows the user to associate logical unit numbers with particular pieces of hardware. These devices are subsequently referred to by LUN. The LUN must be an integer constant between 0 and 99. Normally, the following logical units are associated with the following devices:

LUN	DEVICE
60	STANDARD INPUT
61	STANDARD OUTPUT

EDIT uses logical units numbered between 57 and 59. Therefore, a user should not use these logical units.

Logical unit 100 is permanently assigned to the user's standard input unit. It may appear only on the right-hand side of the equal sign (=) in an EQUIP statement.

Standard input and standard output both refer to the user's terminal.

The <ELEMENT> to the right of the equal sign must be a previously defined logical unit number (60, 61, 100, or a LUN which appears to the left of the equal sign in an earlier EQUIP statement) or one of the following:

FILE — The word FILE is used to define a LUN as a sequential disk storage area. This storage area may be saved for future use by using a SAVE command. If the file is not saved, it will be destroyed when the user logs off, or when the LUN is unequipped.

LP — The characters LP equip a line printer as an output device. Any information sent to this LUN is printed on the high-speed line printer.

NAME — The (NAME) is used to equip a LUN equivalent to the file (NAME). (NAME) may be the name of a public file, or the name of one of the user's private files which he has previously saved under the currently used account number and user number.

NULL — The word NULL is used to define a device which will absorb any information sent to it.

PLOT — The word PLOT is used to define a LUN equivalent to an X-Y plotter.

PUN — The word PUN is used to define a LUN as a card punch. Information sent to this LUN will be punched on cards. (Punch units must be labeled to prevent loss of output.)

RAF — The characters RAF are used to define a random access disk storage area. This storage area may be saved for future use with a SAVE command. If it is not saved, the information will be lost when the user logs off or unequips the unit.

TASK — The word TASK is used to define a LUN to be the future input unit for a remote batch job.

Only a single LUN may be equipped in each EQUIP statement.

EXAMPLES:

```
EQUIP,1=ZAP
EQUIP,9=NULL
EQUIP,10=1
```

ERASE,LINE1,LINE2

ERASE is used to remove lines from the scratch pad. LINE1 specifies the number of the first line to be erased. LINE2 identifies the number of the last line to be erased. All lines between LINE1 and LINE2 are removed. LINE2 must be associated with a line which appears after LINE1 in the scratch pad.

If LINE2 is omitted then only LINE1 is erased. If either LINE1 or LINE2 is invalid, the scratch pad is not changed.

EXAMPLES:

```
ERASE,20,30
ERASE,39,71+3
ERASE,23
```

EXIT

EXIT should be used to return to control mode. This command will unequip the directory file.

EXAMPLE:

```
EXIT
```

FETCH,LUN

This command will transfer information into the scratch pad. This text must have been created by a FILE command. If no LUN is specified, then logical unit 55 is assumed.

The sequence numbers for each line, the tab stops and the sequence and increment parameters (S and I) are restored.

A name may be substituted for the logical unit number in this command. This name must refer to a currently saved file; this file will be rewound before use.

EXAMPLES:

```
FETCH,37
FETCH
FETCH,ZAP
```

FILE,LUN

This command will save the contents of the scratch pad on a file. The tab setting, sequence parameters (S and I), along with the sequence numbers for each line are saved. If LUN is omitted, then logical unit 55 is assumed. Since this command compresses information to save file space, this command should be used to save information from the scratch pad. The FORTRAN compiler will accept information stored with this command.

A name may be substituted for the logical unit number in this command. If a previously defined name is used, the file associated with it will be rewound before use, if the name has not been previously defined, then a new file is generated and saved.

EXAMPLES:

```
FILE,40
FILE
FILE,FORTPROG
```

FIN,LUN,RL

The FIN command (file input) is used to enter information to the scratch pad from the logical unit specified. Input from this unit continues until a file mark or end of data is encountered. Sequence numbers are provided, and the work area is cleared before the command is executed.

If no LUN is specified in the FIN command, then logical unit 54 is assumed. The FIN command will rewind logical units numbered between 50 and 59 before execution.

The RL parameter specifies the character count of the input record. It can be used only when BCD records are being read.

A saved file name may be substituted for LUN.

A FIN command will transfer any valid type of information to the EDIT work area from a file.

EXAMPLES:

```
FIN,34
FIN
FIN,DATA
FIN,ZAP,20
```

FP,LUN

The letters FP are used to indicate file protect. LUN must be either a saved file name or a previously equipped logical unit number which must refer to a file or RAF. File protection prevents the user from taking any action which would destroy or modify the contents of the file. Information may still be read from the file. A protected file which the user does not save, will be lost at LOGOFF.

EXAMPLES:

```
FP,33
FP,45
FP,PROGRAM
```

FWSP,LUN,LUN,...,LUN

This command causes each logical unit specified to be spaced forward one record.

Any number of logical units may be listed; all must have been previously defined. A logical unit may be listed more than once; the LUN will be forward-spaced once for each time it is listed.

EXAMPLES:

```
FWSP,2,67,23
FWSP,41
FWSP,67,67,67,67
```

INPUT,S,I

INPUT is used to enter text from the terminal. INPUT clears the scratch pad. A sequence number is provided for each line. The parameters S and I in the INPUT command specify the first sequence number and the sequence number increment, respectively. If both or either one is omitted, the value set by the last FETCH, INPUT or RESEQ command is assigned to the missing parameter (the parameters are set to 1 when EDIT is entered).

After EDIT types a line number and a colon, the user should enter the desired text. Each line of text must end with CARRIAGE RETURN. To terminate an INPUT operation and return from text to command mode, the user should depress ESCAPE, CONTROL SHIFT W, or ALT MODE.

From a TV, INPUT should be used to enter the first screen full of text. This clears the scratch pad of any old text. After the first screen full is entered the rest of the text should be entered with APPEND.

The INPUT command destroys any information which may have already been in the scratch pad.

EXAMPLES:

```
INPUT,100,100
INPUT
INPUT,,5
INPUT,100
```

INSERT,LINE

The INSERT command allows the user to insert information between existing lines in the scratch pad. After this ccmmand is given, the EDIT program enters the text mode, accepting information which it will insert following LINE. A relative line number followed by a colon (:) is printed for each line inserted.

On a Teletype, any number of lines may be entered (on a TV only one screen full may be inserted); each line must be followed by a CARRIAGE RETURN. No line sequence numbers are provided for the inserted lines. To terminate an insert operation and return from text to command mode, the user should depress ESCAPE, CONTROL SHIFT W, or ALT MODE.

If LINE is omitted, the text is inserted before the first line of the work area.

EXAMPLES:

```
INSERT,189
INSERT
INSERT,22+3
INSERT,0+2
```

KILLC,C

This command is used to change the cancel character (@). This allows the user to type in text with @ as a valid character. C must be one of the 63 printing characters except comma (,).

EXAMPLES:

```
KILLC,:
KILLC,Z
KILLC,*
```

LASTLINE

This command prints the line number of the last sequenced line in the scratch pad.

EXAMPLE:

```
LASTLINE
```

LIBRARYCALL,PARAMETERS

The Editor allows calls to system processors (such as FORTRAN, ALGOL,...etc.). EDIT will unequip the directory file if equipped. For information concerning these routines and their parameters consult the OS-3 Reference Manual (ccm-70-8).

The following is a list of valid library calls:

```
ALGOL
BASIC
COMPASS
COPY
COSY
DATE
FORTRAN
LABEL
LOAD
OSCAR
```

EXAMPLES:

```
ALGOL,L=5,X,I=7
BASIC
COMPASS,L=2,R,P,S,I=50
FORTRAN,L=7,I=ZING
LOAD,50
OSCAR
```

LIST,LINE1,LINE2

The LIST command will list the contents of the scratch pad on the user's terminal. LINE1 and LINE2 are, respectively, the first and last line number which the user wishes to have listed. If both parameters are omitted, then all the scratch pad is listed. If LINE1 is omitted then lines from the beginning of the scratch pad through LINE2 are listed. If LINE2 is omitted, then LINE1 is listed. If LINE2 is zero, LINE1 and all lines after it are listed.

To terminate an undesired listing operation, enter control mode and type MI to get back to the Editor without destroying the information in the scratch pad.

EXAMPLES:

```
LIST,1,8
LIST,8
LIST
LIST,10,0
LIST,13+5
LIST,12+2,13+5
LIST,0+2,0+5
LIST,,8
```

MAXLINE

This command will print the largest sequence number found in the scratch pad.

EXAMPLE:

```
MAXLINE
```

MOVE,LINE1,LINE2,LINE3

The MOVE command will erase LINE1 through LINE2 and insert them in the same order after LINE3. All three parameters must be specified for the command to be executed properly. The line numbers associated with LINE1 through LINE2 are carried with them. If LINE2 is omitted, only LINE1 is moved.

If a parameter error is detected, the move does not take place.

EXAMPLES:

```
MOVE,10,20,45
MOVE,45,45,3
MOVE,67,72,48
MOVE,10+2,17+1,160+3
MOVE,13, ,14
```

OUT,LUN,N,R=NUMBER1,S=NUMBER2

The OUT command is used to copy information from the scratch pad to the logical unit specified. This text is written in BCD. If no LUN is specified, then logical unit 54 is assumed.

NUMBER1 specifies the line length in characters to be written out. If no value is specified then variable length records are written. N is used to specify that sequence numbers are to be placed in the last 5 characters of the line. N will preset the line length to 80 if no R was specified. NUMBER2 is the number of characters to shift the line to the right. It is useful to line up (center) printed material on a line printer. If S is omitted then S is set to 1 for line printer or Teletype output; otherwise it is zero. For most cases N, R, and S may be omitted. EDIT will write a file mark at the end of the output (except on a TASK) and backspace over it, if possible. Later output to this unit would then destroy the file mark. If LUN is a logical unit between 50 and 59 it will be rewound before use if possible.

A name may be used in place of the logical unit number with this command. If this name refers to a currently saved file, then this file will be rewound and used. If the name has not been previously defined, then a new file will be generated and saved.

EXAMPLES:

```
OUT,23
OUT,41,N
OUT,50,N,R=24
OUT,TEST
OUT,,S=10
OUT,3,N,R=120,S=4
```

RELEASE,LUN,LUN,...,LUN

The RELEASE command destroys the information stored on each of the logical units listed and returns the file space that was used to OS-3. The logical units are still defined, however.

EXAMPLES:

```
RELEASE,33,67,21
RELEASE,32
```

REP,LINE1,LINE2

The REPlace command functions as a combination of the ERASE and INSERT commands. The REP command will erase LINE1 through LINE2 and then shift to text mode for the user to enter new lines of information just as with an INSERT command. Any number of new lines may be entered (on a TV, only one screen full may be inserted at one time). The new lines will not have sequence numbers associated with them. If LINE2 is omitted, then only LINE1 will be erased. Each line entered in text mode must end with CARRIAGE RETURN. To return to command mode, the user should depress ESCAPE, CONTROL SHIFT W, or ALT MODE.

EXAMPLES:

```
REP,5,7
REP,89
REP,15+2,17+1
```

RESEQ,S,I

The RESEQuence command causes line numbers to be assigned to each line in the user's EDIT scratch pad. These numbers will start with S and increase in increments of I. If the parameters S and I are omitted, then the last values of S and I given since one entered EDIT will be used. S and I are initialized to one. They may be changed by a FETCH, INPUT, or RESEQ command.

EXAMPLES:

```
RESEQ,100,100
RESEQ
```

RESET

The RESET command will unequip all logical units from 0 to 99 (except the directory LUN) and equip 60 and 61 to LUN 100 (user's terminal). This gives the same effect as logging off and on again.

EXAMPLE:

```
RESET
```

REWIND,LUN,LUN,...,LUN

The REWIND command is used to return the user to the start (load point) of the logical units specified. Any number of logical units may be rewound using a single command. Each of the logical units listed must have been previously equipped as a file or RAF.

EXAMPLES:

```
REWIND,45,37,02,99
REWIND,02
```

RFP,LUN

The RFP command will remove file protection from the LUN.

LUN may be the name of a saved file.

EXAMPLES:

```
RFP,1
RFP,ZAP
```

SAR,LINE1,LINE2,/STRING/,/REPLACEMENT/

This command will search the scratch pad looking for a string of characters anywhere in a line which identically matches STRING. If a match is found, the characters which matched STRING are replaced by REPLACEMENT. All occurrences of STRING in a line are replaced. If the replacement string is NULL (i.e.//) then the string is deleted. If the replacement parameter is omitted then the sequence number of each line containing STRING is printed. LINE1 and LINE2 are, respectively, the first and last line number which the user wishes to have searched. If both parameters are omitted, then all the scratch pad is searched. If LINE1 is omitted then lines from the beginning of the work area through LINE2 are searched. If LINE2 is omitted, then LINE1 is searched. If LINE2 is zero LINE1 and all lines after it are searched.

The slashes (//) are delimiting characters--any printing character except comma (,) may be used. However, the character must not occur within STRING or REPLACEMENT.

EXAMPLES:

```
SAR,1,5,/ZIP/,/ZAP/
SAR,7,9,'ZILCH',(GASP(
SAR,,,/BUFFER/
SAR,1,,/A=B/,/A=B(1)/
SAR,,3,/IDINT/,/IDENT/
```

SARL,LINE1,LINE2,/STRING/,/REPLACEMENT/

This command is the same as the SAR command except that when a match is found the line is listed. The line is listed after it has been altered.

EXAMPLES:

```
SARL,1,5,/ZIP/,/ZAP/
SARL,7,9,'ZILCH',(GASP(
SARL,,,/BUFFER/
SARL,1,,/A=B/,/A=B(1)/
SARL,,3,/IDINT/,/IDENT/
```

SARLM,M,LINE1,LINE2,/STRING/,/REPLACEMENT/

This command is a combination of the SARL and SARM commands (see SAR, SARL and SARM).

EXAMPLES:

```
SARLM,Q,10,30,/123ZQ/,/123/
SARLM,R,,,/ABRR9/,/ABR9R/
SARLM,X,10,,/A(10,X/,/A(20,IX/
```

SARM,M,LINE1,LINE2,/STRING/,/REPLACEMENT/

This command is the same as the SAR command except for the introduction of a mask character M. Wherever M appears in the STRING the character 'masked' always is a match. The masked character is saved and can be used in REPLACEMENT.

An illustration follows:

text is 123NRB39
mask character is Z (SARM,Z,.....)
STRING is /23ZZBZ/
REPLACEMENT is /234Z678Z/
resulting altered text is 1234N678R9

EXAMPLES:

SARM,Q,10,30,/123ZQ/,/123/
SARM,R,,,/ABRR9/,/ABR9R/
SARM,X,10,,/A(10,X/,/A(20,IX/

SEFB,LUN,LUN,...,LUN

This command will cause each logical unit specified to be spaced backward until a file mark is passed, or the beginning of the file is found. A logical unit may be listed more than once; the SEFB command will be executed as many times as the LUN is listed.

EXAMPLES:

```
SEFB,21,33,76
SEFB,37
SEFB,29,29,29
```

SEFF,LUN,LUN,...,LUN

This command will cause each logical unit specified to be spaced forward until a file mark is passed, or the end of the file is found. Each of the logical units specified must have been previously defined. A logical unit may be listed more than once; the SEFF command will be executed as many times as the LUN is listed.

EXAMPLES:

```
SEFF,39,73,21
SEFF,35
SEFF,26,26,26
```

STATUS,LUN,LUN,...,LUN

The STATUS command will type information about each of the logical units listed.

CHARACTER	MEANING
+	read only (file protected)
[	load point
]	end of data
F	file mark
B	binary record
A	abnormal/unavailable
E	address error
S	saved file

EXAMPLES:

```
STATUS,33,12
STATUS,47
STATUS,ZAP
```

TAB,T1,T2,...,T6

The TAB command sets up to 6 tab stops at the points specified. Whenever the user depresses CONTROL SHIFT I (tab) on a Teletype or ↓ on a TV, EDIT will tab to the next stop exactly as a typewriter would. This tab takes place whether the carriage on the user's terminal actually moves or not. The tab stops are initialized to 10, 20, 40, and 72.

Tab stops must be listed in ascending order.

EXAMPLES:

```
TAB,7,10,30,50
TAB,10,45
TAB,7
```

TABC,C

This command is used to change the tab character (horizontal tab on a Teletype or ↓ on a TV). This allows the user to use the ↓ on a TV or pick a substitute tab character on a terminal without a horizontal tab character (ASCII 227). C must be one of the 63 printing characters except comma (,).

EXAMPLES:

```
TABC,:
TABC,Z
TABC,*
```

TAPE

The TAPE command is used to indicate that one wishes to enter information into the scratch pad from a prepunched paper tape. The TAPE command clears the scratch pad and enters text mode waiting to read the tape. No sequence numbers are provided, and no colons are printed.

The prepunched paper tape should have both a leader and a trailer prepared by punching a series of 50 to 100 RUBOUT characters. This may be done quite easily by depressing REPT-RUBOUT keys for about 5 seconds. Individual lines of information on the tape must be separated by CARRIAGE RETURN, LINEFEED in that order. RUBOUTS within the text are ignored.

A CARRIAGE RETURN must follow the command TAPE. After reading the tape the user should depress ESCAPE, CONTROL SHIFT W, or ALT MODE to return to the EDIT command mode.

EXAMPLE:

```
TAPE
```

TTP

This command will generate a paper tape on the Teletype. The program will punch out 5 inches of RUBOUT characters, a CONTROL SHIFT R, the text, CONTROL SHIFT T, and 5 more inches of RUBOUTs. After entering this command and depressing CARRIAGE RETURN, the user should immediately activate the paper tape punch on his Teletype.

This command may be terminated by the user depressing BREAK. This action will return the Teletype to control mode and any paper tape generated will be useless. One must then type MI to return to EDIT.

EXAMPLE:

```
TTP
```

UNEQUIP,LUN

UNEQUIP causes the LUN specified to be deleted from the available logical unit numbers. If LUN is a file which has not been saved, its contents are lost.

If LUN is an output device, such as the printer or card punch, the information on LUN is sent to the proper output device at this time. Logical units 57 and 58 may not be unequipped.

EXAMPLES:

```
UNEQUIP,04
UNEQUIP,99
```

WFM,LUN,LUN,...,LUN

The WFM command writes a file mark on each of the logical units specified. Each of the logical units must have been previously defined.

EXAMPLES:

```
WFM,45,93,01
WFM,1
```

WIDTH,NUMBER

This command sets the line length for listing output on the user's terminal. It is used to determine the point at which EDIT should do a CARRIAGE RETURN, LINE FEED.

EXAMPLES:

```
WIDTH,40
WIDTH,80
```