

EXORSET CRT EDITOR  
REFERENCE MANUAL

The information in this document has been carefully checked and is believed to be entirely reliable. However, no responsibility is assumed for inaccuracies. Furthermore, such information does not convey to the purchaser of the product described any license under the patent rights of Motorola Inc. or others.

Motorola reserves the right to change specifications without notice.

EXORset, EXORdisk, and XDOS are trademarks of Motorola Inc.

First Edition December 1980  
Copyright 1980 by Motorola Inc.



## TABLE OF CONTENTS

	Page
CHAPTER 1 - GENERAL INFORMATION	
1.1 Introduction . . . . .	1-01
1.2 Minimum Configuration For Editor Operation . .	1-01
1.3 EXORset Console Features . . . . .	1-01
1.4 Loading The Disk Operating System (XDOS) . . .	1-01
1.5 Invoking The Editor . . . . .	1-02
1.6 Modes Of Operation . . . . .	1-03
1.6.1 Command Editing Mode . . . . .	1-03
1.6.2 Page Editing Mode . . . . .	1-04
CHAPTER 2 - EDIT COMMANDS	
2.1 Introduction . . . . .	2-01
2.2 Command Syntax . . . . .	2-01
2.3 1-65535 command . . . . .	2-02
2.4 CHANGE command . . . . .	2-02
2.5 DELETE command . . . . .	2-04
2.6 PRINT command . . . . .	2-04
2.7 SAVE command . . . . .	2-05
2.8 QUIT command . . . . .	2-06
2.9 DUPLICATE command . . . . .	2-06
2.10 BLOCK/LINE MERGE rules . . . . .	2-07
2.11 E command . . . . .	2-08
2.12 EXTEND command . . . . .	2-08
2.13 FIND command . . . . .	2-08
2.14 INSERT command . . . . .	2-10
2.15 LIST command . . . . .	2-10
2.16 MERGE command . . . . .	2-11
2.17 MOVE command . . . . .	2-11
2.18 NUMBER command . . . . .	2-12
2.19 RANGE command . . . . .	2-13
2.20 RESEQUENCE command . . . . .	2-13
2.21 SEARCH command . . . . .	2-14
2.22 TAB command . . . . .	2-14
2.23 VERIFY command . . . . .	2-15
2.24 XTRACT command . . . . .	2-16
2.25 EXAM command . . . . .	2-16
2.26 Editor Messages . . . . .	2-16

### EFFICIENT OPERATION OF THE CRT EDITOR

#### LIST OF ILLUSTRATIONS

FIGURE 1-1	Command Editing Mode . . . . .	1-05
1-2	Page Editing Mode . . . . .	1-08

#### LIST OF TABLES

TABLE 1-1	Command Editing Mode Commands . . . . .	1-06
1-2	Command Editing Mode Controls . . . . .	1-07
1-3	Page Editing Mode Commands . . . . .	1-09
1-4	Editor Messages . . . . .	2-18



## CHAPTER 1

### GENERAL INFORMATION

#### 1.1 INTRODUCTION

Motorola's XDOS CRT EDITOR provides the users of EXORset development systems with powerful text editing capabilities. The Editor simplifies both the initial text and source program entry, as well as subsequent program modification. As a disk based editor, text may be entered either from a diskette file or terminal keyboard. The Editor operates in conjunction with the EXORset system and allows the user to perform editing, employing specifically designed features of the EXORset (function keys, direct access to the screen memory).

The Editor provides a standard set of editing functions, as found in most microprocessor editors:

- . insert string/line
- . change string
- . list program
- . delete string/line

In addition, the Editor allows more powerful editing functions, such as:

- . multi-string replacement
- . block moves
- . verification of editing changes
- . CRT oriented text modifications
- . editing ranges
- . tabbing

#### 1.2 MINIMUM CONFIGURATION FOR EDITOR OPERATION

The Editor requires as a minimum configuration the following:

- . EXORset
- . 32K Random Access Memory
- . EXORset Mass Storage Unit  
(Minifloppy drives or EXORDisk III)
- . XDOS operating system

The use of additional memory or a line printer is optional.

#### 1.3 EXORSET CONSOLE FEATURES

The Editor takes advantage of the EXORset console features, namely the 16 function keys and the keyboard. The Editor recognizes both lower-case and upper-case characters. Also, the function keypad interface is set up such that holding down a function key allows to repeat the software function associated with it for as long as the key is held down.

#### 1.4 LOADING THE DISK OPERATING SYSTEM (XDOS)

Loading the XDOS operating system is initiated by typing the "XDOS" command while under control of the EXORbug monitor (the "." prompt character is displayed at that time). Once loaded, XDOS identifies itself and displays an equal sign as its prompt character. Any XDOS command may then be entered. The loading procedure is illustrated below :

```
depress restart or BREAK key
.XDOS (carriage return)
XDOS XX.YY   (XX.YY is the XDOS version/revision number)
=
```

### 1.5 INVOKING THE EDITOR

The XDOS command E will invoke the Editor to create or modify a named file. The syntax for the edit command is as follows. All elements bounded by brackets are optional; e.g., [;options]. All elements outside the brackets must be entered. All user input is terminated by a carriage return (RETURN).

SYNTAX : E FILENAME1 [, [FILENAME2] [, :DRIVE]] [; [TS] [FN]]

WHERE : FILENAME1 is the file to be edited. If FILENAME1 exists, it will be edited. If it does not exist, it will be created and made available for editing.

FILENAME2 is the file name of the edited result. FILENAME2 is only valid if FILENAME1 exists. If FILENAME2 is not specified, the original FILENAME1 will be deleted at the end of the edit process, and the edited result will be renamed FILENAME1. If FILENAME2 is specified, FILENAME1 is retained as an XDOS file without any alteration by the Editor.

DRIVE specifies the drive on which to put the second scratch file. If DRIVE is not specified, or if the diskette in DRIVE cannot be accessed (disk not ready or not DOSGENed), the Editor will select either of drive 0 or 1 which features the largest amount of free space to store the second scratch file. In effect, the Editor uses temporary files named SCRATCH1 and SCRATCH2. SCRATCH1 is used whenever FILENAME1 exists. When the edit is terminated via QUIT or a SAVE command, SCRATCH1 is renamed FILENAME2 (or FILENAME1 if FILENAME2 was not specified). SCRATCH1 is always built on the destination file drive. SCRATCH2 is used to support multiple edit passes in a single edit session.

NOTE : If during an editing session, the Editor does not come to its normal orderly exit stage due to unrecoverable disk errors, or to the BREAK (ABORT) key or RESTART button being activated, the files SCRATCH1 and SCRATCH2 may still be present on the disk(s). However, the user is not required to delete these files prior to initiating the next editing session as the scratch files are automatically deleted, if they exist, upon entry in the Editor.

TS specifies the initial tab stop settings for the Editor as

follows:

- A implies assembler tabs(8,15,24) with the space bar as a special tab character and whereby an "\*" in column 1 will disable the special tab character.
  - F implies FORTRAN tabs (7) with the space bar as a special tab character and whereby a "C" in column 1 will disable the special tab character.
  - C implies COBOL tabs (6,9,12) with the space bar as a special tab character.
  - P implies PASCAL tabs (3,5,7,9,...) with the forward tab key (F8) as the tab command
- FN is the file numbering option. An N implies edit with line numbers. A -N implies edit without line numbers. FN defaults to N when FILENAME1 does not exist, and is determined by the first line in FILENAME1 when it does exist. Whenever a line numbered file is edited without using the line numbers, the line numbers are treated as data.

The response to the E command is shown in the following example.

EXAMPLE:

```
=E LTR2 (carriage return)
XDOS CRT EDITOR  REV XX.XX (XX.XX is the version number)
COPYRIGHT BY MOTOROLA 1980
```

When FILENAME1 exists, the Editor will check to determine if there is sufficient disk space to hold the edited result and the work file on the drives specified. If there is not sufficient space to store an edited file the size of which is identical to the input file, the Editor will display "INSUFFICIENT DISK SPACE ON DRIVE X CONTINUE (Y/N)?" A "Y" response will cause the Editor to continue; an "N" response will cause the Editor to return control to XDOS.

In addition, the Editor checks for sufficient disk space prior to executing any command that might increase the size of the edited file.

Once the Editor has been invoked, it will display a message indicating that the file, FILENAME1, was opened or created new, and whether the file is line numbered (with line numbers).

## 1.6 MODES OF OPERATION

The Editor has two processing modes: COMMAND EDITING and PAGE EDITING.

### 1.6.1 COMMAND EDITING Mode

When the Editor is used to edit an existing file, it dynamically adapts to command editing (refer to Figure 1-1). With command editing, the user enters edit commands at the cursor position following the prompt (>) on line 20, while continually viewing a portion of the file on the CRT screen (lines 1 through 18). Thus, the screen becomes a window where the portion of the file currently acted upon is displayed. The Editor prompt in this window (column 1) serves as

a pointer to indicate the current file position. Table 1-1 provides a summary of the commands. Table 1-2 lists the special character commands that are controlled by the use of the EXORset's special-purpose keys when performing command editing.

#### 1.6.2 PAGE EDITING Mode

Page editing is initiated when the Editor is invoked to create a new file (refer to Figure 1-2), or when one of the following commands is entered:

```
CHANGE //  
INSERT (for text files without line numbers)  
NUMBER (for source files with line numbers)
```

Page editing is terminated via a Command mode change request (use of the F1 function key).

With page editing, the user is positioned on the window and uses the EXORset function keys to insert, change, or delete characters and/or lines within the file. Table 1-3 lists the special character commands that are controlled by use of the EXORset function keys when performing page editing.

The Editor will accept only valid printable ASCII characters. Any non-printable character received from the keyboard or file in a data stream will be ignored. The EXORset keyboard provides the means of inputting the full 128-character ASCII set, plus various function keys. The function keys F2 through F16 are used to select the special character commands in the page editing mode, as shown in Tables 1-1 and 1-2.

In page editing or command mode, the bottom two lines on the screen are used to display the mnemonic function name for each of the sixteen function keys.



```

>00010  NAM POLL
00020  OPT MEM
00030  PIA1AC EQU $4005
00040  PIA1BC EQU $4007
00050  PIA2AC EQU $4009
00060  PIA2BC EQU $400B
00070  ORG $100
00080  POLL LDA A PIA1AC
00090  BMI ROUT1
00100  ROL A
00110  BMI ROUT2
00120  LDA A PIA1BC
00130  BMI ROUT3
00140  ROL A
00150  BMI ROUT4
00160  LDA A PIA2AC
00170  BMI ROUT5
00180  ROL A
EDITING OLD FILE: LTR2      .SA:O WITH LINE NUMBERS
> cursor rests here
F1  F2  F3  F4  F5  F6  F7  F8  F9  F10 F11 F12 F13 F14 F15 F16
CMND DUPL PG ^ PG v LN ^ LN v  T <  T >  <  >  ^  v  I CH D CH I LN D LN

```

FIGURE 1-1. COMMAND EDITING MODE

TABLE 1-1. COMMAND EDITING Mode Commands

COMMAND	DESCRIPTION
C, CHANGE	. Change a string within a record (or a group of records).
DEL, DELETE	. Delete data from the file.
D, DUPLICATE	. Copy a group of records from one place within a file to another place within the file.
E	. Terminate the edit session. Begin editing another file.
EX, EXTEND	. Insert data at the end of records.
F, FIND	. Find a string or a line.
I, INSERT	. Insert data into the file. (For editing files without line numbers.)
L, LIST	. Display a portion of the file.
MERGE	. Retrieve records from another XDOS file and include them in the edited file.
MOVE	. Move a group of records from one place within the file to another place within the file.
N, NUMBER	. Insert data into the file and prompt with the next line number. (For editing files with line numbers.)
P, PRINT	. Print data at an associated line printer.
Q, QUIT	. Terminate the edit session and return control to XDOS.
R, RANGE	. Establish default values for the vertical and horizontal ranges. (For LIST, PRINT, CHANGE, FIND, and SAVE.)
RESEQUENCE	. Number, renumber, or unnumber the file.
SAVE	. Save the edited file into the named file.
S, SEARCH	. Find the next occurrence of a string.
TAB	. Alter the tab settings used for data input.
V, VERIFY	. Display on the screen, any time a change is made to the file, the values after the change.
X, XTRACT	. Insert the data in the buffer into the file.
1-65535	. Insert, modify, or delete a line from the file. (This command type is for editing files with line numbers.)

TABLE 1-2. COMMAND EDITING Mode Controls

COMMAND	DESCRIPTION
F1	. Stop processing; await a new command
F3	. Scroll the screen forward one page.
F4	. Scroll the screen back one page.
F5	. Scroll the screen forward one line.
F6	. Scroll the screen back one line.
F7	. Move the cursor to the left to the previous tab stop.
F8	. Move the cursor to the right to the next tab stop.
F9	. Move the cursor to the left one column with no character modification.
F10	. Move the cursor to the right one column with no character modification.
F11	. Move the display pointer up one line on the display, with page wraparound.
F12	. Move the display pointer down one line on the display, with page wraparound.
F13	. All columns in the line, starting with the cursor column, are moved right one column. The cursor is then set at the inserted space.
F14	. All columns in the line, starting with the cursor column, are moved left one column. The cursor does not change position.
CTRL-W	. Pause the processing of this list.
CTRL-P	. Abort the processing of this command.
CTRL-X	. Cancel this line.
DEL	. Delete the previous character.

>00010 \_ cursor rests here

EDITING NEW FILE: LTR3 .SA:O WITH LINE NUMBERS  
>  
F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14 F15 F16  
CMND DUPL PG ^ PG v LN ^ LN v T < T > < > ^ v I CH D CH I LN D LN

FIGURE 1-2. PAGE EDITING MODE

TABLE 1-3. PAGE EDITING Mode Commands

COMMAND	DESCRIPTION
F1	. Stop page editing; set the mode to the COMMAND mode.
F2	. Duplicate the data on the line above this one.
F3	. Scroll the screen forward one page.
F4	. Scroll the screen back one page.
F5	. Scroll the screen forward one line.
F6	. Scroll the screen back one line.
F7	. Move the cursor to the left to the previous tab stop.
F8	. Move the cursor to the right to the next tab stop.
F9	. Move the cursor to the left one column with no character modification.
F10	. Move the cursor to the right one column with no character modification.
F11	. Move the display pointer and the cursor up one line on the display, with page wraparound.
F12	. Move the display pointer and the cursor down one line on the display, with page wraparound.
F13	. All columns in the line, starting with the cursor column, are moved right one column. The cursor is then set at the inserted space.
F14	. All columns in the line, starting with the cursor column, are moved left one column. The cursor does not change position.
F15	. Insert a line at the current file position.
F16	. Delete the line at the current file position.
CTRL-X	. Cancel this line.
DEL	. Delete the previous character.



\_\_\_\_\_

## EDIT COMMANDS

## 2.1 INTRODUCTION

There are two levels of edit features available: a basic set, which the user may master in a relatively short period of time, and an advanced set, which gives the user much more flexibility in editing. The advanced set may be mastered as needed.

The basic command set includes the l-65535, CHANGE, DELETE, PRINT, SAVE, and QUIT commands, along with the elementary edit features.

The Editor prompts the user for the next command by displaying a greater-than sign(>). The command is initiated by typing a carriage return. Many of the commands have an abbreviated form, or alias, that the programmer may use if desired. A description of the Editor commands follows.

## 2.2 COMMAND SYNTAX

Editor commands may be entered in either upper or lower case and in either a short or long form. Only the first four characters of any command have significance, so that all commands may be abbreviated to the first four characters. In all of the commands, the character "\*" may be used to indicate the current line position when referring to a line number.

In defining the commands provided by the Editor, the most common syntactical elements, vertical and horizontal range, are defined as follows:

```
VERTICAL-RANGE = [[BEGINNING-LINE-NUMBER] [- ENDING-LINE-NUMBER]]
                  or
                  [[COUNT1 - ] COUNT2]
```

WHERE: BEGINNING-LINE-NUMBER / ENDING-LINE-NUMBER apply to line numbered files.

COUNT1 / COUNT2 apply to unlined files.

COUNT1 is the offset from the current line at which to begin the function performed. The offset may be negative.

COUNT2 is the number of lines upon which to perform the function.

The default for omitted values (where not overridden by the RANGE command) will result in the function being performed on the current line, unless otherwise stated.

HORIZONTAL-RANGE = [:COLUMN1 [ - COLUMN2]]

WHERE: COLUMN1 is the column on each line in which to begin the function to be performed.

COLUMN2 is the column on each line in which to end the function to be performed.

The horizontal range may extend from column 1 to 132; for line numbered files, column 1 of the horizontal range corresponds to physical column 7 of the display.

The default values where no horizontal range is specified will result in the function being performed on the entire line.

### 2.3 1-65535 COMMAND

**FUNCTION:** Insert, modify, or delete a line from the file. (This command applies to line-numbered files only.).

**SYNTAX:** LN DATA

**WHERE:** LN is the line number of the line to be inserted, modified, or deleted. The Editor supports 5-digit line numbers, with 65535 being the highest value.  
DATA is the new value of the line.

The entry of a line number with no data will imply a request to delete the line.

The line number inserted, modified, or the line immediately following the line deleted is at the current line position.

**EXAMPLES:** 1. To insert line number 200 into the file.  
>200 DATA ON LINE 200cr  
  
2. To change line number 200  
>200 NEW DATA ON LINE 200cr  
  
3. To delete line number 200 from the file.  
>200cr  
  
4. To insert a null line numbered 200 (line 200 does not currently exist in the file).  
>200cr

### 2.4 CHANGE COMMAND

**FUNCTION:** Change a string within a record (or a group of records).

**SYNTAX:** CHANGE [VERTICAL RANGE] [HORIZONTAL RANGE]  
[[.TRANSPARENT CHAR] [;COUNT] STRING1] [[STRING2][REPETITION]]

**ALIAS:** C,change,c

**WHERE:** STRING1 is the target string to be found and changed.  
STRING2 is the value to which STRING1 is to be changed.  
COUNT indicates that the COUNT<sup>th</sup> occurrence of STRING1 in a line is the target string.  
REPETITION indicates the number of times STRING1 is to be found and changed.  
TRANSPARENT-CHAR is a character within STRING1 that is to be ignored when testing for equality to STRING1.

An "A" in the COUNT or REPETITION field indicates all occurrences. The default value for COUNT is 1 (the first occurrence), and the default value



for REPETITION is all lines within the VERTICAL-RANGE. String data is delimited by the first non-blank, non-numeric character. The same delimiter must also terminate STRING1 and STRING2 (if present). Care should be used in picking a string delimiter as the use of "." or ";" when the TRANSPARENT-CHAR or COUNT is not specified will cause unpredictable results. When the change command is executed with no parameters, the next occurrence of the string data last specified by the user is modified. A null STRING1 will be interpreted to imply insert STRING2 at the beginning of the HORIZONTAL-RANGE. A null STRING2 will be interpreted to imply delete STRING1. If STRING2 does not exist, the user will begin page editing with the cursor resting on the first character of STRING1. When only COLUMN1 of the HORIZONTAL-RANGE is specified and no string is specified, the user will begin page editing with the cursor resting on the column position specified; A change command with only a null STRING1 initiates page editing at the current line position.

The last data line modified by the change is at the current position line.

- EXAMPLES:
1. Change the string SAM to BILL on the current position line.  
>C /SAM/BILL/cr
  2. Change the first occurrence of the string SAM to BILL on each line from line numbers 100 through 200, inclusive.  
>C 100-200 /SAM/BILL/cr
  3. Change the first occurrence of the string SAM (that occurs after column 30 and before column 60) to the string BILL on line number 100.  
>C 100 :30-60 /SAM/BILL/cr
  4. Change the third occurrence of the string SAM to BILL on the current position line.  
>C ;3/SAM/BILL/cr
  5. Change all occurrences of the string SAM to BILL on each of the lines from line number 100 to 200, inclusive.  
>C 100-200 ;A/SAM/BILL/cr
  6. Delete the string BOB from line number 120.  
>C 120 /BOB//cr
  7. Insert the string BOB at the beginning of the current line.  
>C //BOB/cr
  8. Change the strings LABELx (where x can be any character or digit) to the string 1.25 on the line numbers from 100 to 500, inclusive.  
>C 100-500 .X/LABELX/1.25/cr
  9. Change the first occurrence of the string SAM to BILL on the lines from line number 100 to 200. Then change the next occurrence of the string SAM to BILL from wherever it is to the end of the file.  
>C 100-200/SAM/BILL/1cr  
>Ccr
  10. Change the second occurrence of the string LABxL (where x can be any character or digit) to the string LABEL where it

occurs between columns 30 and 60 on line numbers 100 to 130, inclusive.

>C 100-300 :30-60 .X;2/LABXL/LABEL/cr

11. Change the string SAM to BILL on the next 5 lines (where the file being edited does not have line numbers).

>C \*-5 /SAM/BILL/cr

12. Begin page editing at the beginning of this line.

>C //cr

13. Begin page editing at the beginning of the string SAM on this line.

>C /SAM/cr

14. Begin page editing on column 12 of line number 100.

>C 100 :12cr

15. Begin page editing on the second occurrence of the string X on line 200.

>C 200 ;2/X/cr

## 2.5 DELETE COMMAND

FUNCTION: Delete data from the file.

SYNTAX: DELETE [VERTICAL-RANGE] [HORIZONTAL-RANGE]

ALIAS: DEL,delete,del

Specification of a HORIZONTAL-RANGE will cause the data within the range to be deleted from each line within the VERTICAL-RANGE.

The line immediately following the last line affected is at the current position line.

EXAMPLES: 1. Delete the line at the current file position.

>DELcr

2. Delete the lines between lines 100 and 200, inclusive.

>DEL 100-200cr

3. Delete the data between columns 10 and 20, inclusive, on line 210.

>DEL 210 :10-20cr

4. Delete line number 210.

>DEL 210cr

5. Delete the next 5 lines (when editing a file without line numbers).

>DEL \*-5cr

## 2.6 PRINT COMMAND

FUNCTION: Print data at an associated line printer.

SYNTAX: PRINT [VERTICAL-RANGE] [SPACING]

ALIAS: Print,P,p

WHERE: SPACING is "D" to indicate double spacing and "T" to indicate triple spacing.

The print command does not alter the display.

EXAMPLES: 1. Print the entire file.  
>PRINTcr

2. Print this line.  
>PRINT \*cr

3. Print the lines between line numbers 100 and 200, inclusive, with double spacing.  
>PRINT 100-200 Dcr

4. Print the next 10 lines (when editing a file without line numbers) with triple spacing.  
>PRINT \*-10 Tcr

## 2.7 SAVE COMMAND

FUNCTION: Save the edited file into the named file.

SYNTAX: SAVE [[VERTICAL-RANGE] FILENAME]

ALIAS: Save

WHERE: FILENAME implies that an extract function is to be performed. The referenced data (as modified by the edit) is written into the new file: FILENAME.

FILENAME must not already exist as an XDOS file. If no VERTICAL-RANGE is specified, the current modified version of the file is written into FILENAME. When FILENAME is not specified, the current modified version of the source file will replace the old version, and the old version is deleted from XDOS. The new version is then reopened and repositioned to the beginning of the file for further editing.

The save command with FILENAME specified does not alter the display and, without FILENAME specified, resets the display to the beginning of the file.

EXAMPLES: 1. Save the edited file to disk.  
>SAVEcr

2. Save the edited file into a new file named SAM.  
>SAVE SAMcr

3. Extract the lines between line numbers 100 and 500, inclusive, and put them into a new file named SAM.  
>SAVE 100-500 SAMcr

4. Extract the next 20 lines (when editing a file without

line numbers) and put them into a new file named SAM.  
>SAVE \*-20 SAMcr

## 2.8 QUIT COMMAND

FUNCTION: Terminate the edit and return control to XDOS.

SYNTAX: QUIT [A]

ALIAS: Quit,Q,q

WHERE: A implies abort the edit.

The quit command causes the results of the edit to be saved prior to termination unless the A option is specified.

The screen is cleared.

EXAMPLES: 1. Save the results and terminate the edit.  
>QUITcr

2. Abort the edit and do not modify the source file.  
>QUIT Acr

## 2.9 DUPLICATE COMMAND

FUNCTION: Copy a group of records from one place within a file to another place within the file.

SYNTAX: DUPLICATE [VERTICAL-RANGE] [/NEW-LINE-NUMBER [,NEW-INCREMENT]]

ALIAS: D,duplicate,d

WHERE: NEW-LINE-NUMBER is the destination line number for the first line copied (line numbered files only)  
NEW-INCREMENT is the line number increment to be applied to each subsequent line of the copy.

The duplicate command will copy the requested data into a buffer area. The data can then be moved from the buffer via the XTRACT command. The duplicate command, when used with line numbered files specifying the NEW-LINE-NUMBER, will automatically invoke the XTRACT command to copy the data from the buffer to the target area. See the BLOCK/LINE MERGE RULES on the following page for the treatment of line number conflicts.

If the NEW-LINE-NUMBER was not utilized, the screen is unchanged. If it was specified, the current position line contains the line following the last line copied at its new location.

EXAMPLES: 1. Duplicate this line (put this line into the XTRACT buffer).  
>Dcr

2. Duplicate lines 100 to 300, inclusive.  
>D 100-300cr

3. Duplicate the next 10 lines (when editing a file without line numbers).

>D \*-10cr

4. Duplicate this line and put it at line number 111.

>D /111cr

5. Duplicate line numbers 110 to 130, inclusive, and put them at line number 151, with an increment of 2 between lines.

>D 110-130/151,2

6. Duplicate line 130 and put it at 143.

>D 130/143cr

## 2.10 BLOCK/LINE MERGE RULES

When the establishment of new line numbers for a block insert generates a conflict (a potential duplication or overlap of line numbers), the editor will display:

"LINE NUMBER CONFLICT AT XXXXX RESEQUENCE (Y/N)?"

If the user responds with a Y, the balance of the file will be resequenced only until there is no further conflict. If the user responds with an N, the duplicated lines in the file will be overlayed with the new lines. Consider the following example:

User file prior to any editing:

```
010 1st line
020 2nd line
030 3rd line
040 4th line
050 5th line
060 6th line
070 7th line
080 8th line
090 9th line
100 10th line
```

User enters: DUPLICATE 10-50/74,4

The Editor will begin processing:

```
010 1st line
020 2nd line
030 3rd line
040 4th line
050 5th line
060 6th line
070 7th line
074 1st line
078 2nd line
```

LINE NUMBER CONFLICT AT 00080 RESEQUENCE (Y/N)?

If the user types Y, the rest of the file will be:

```
082 3rd line
```

086 4th line  
090 5th line  
091 8th line  
092 9th line  
100 10th line

If the user types N, the rest of the file will be:

080 8th line  
082 3rd line  
086 4th line  
090 5th line  
100 10th line

## 2.11 E COMMAND

**FUNCTION:** An Editor command that will terminate this edit session and initiate the editing of another named file. (For command specifics, refer to paragraph 1.5 ).

## 2.12 EXTEND COMMAND

**FUNCTION:** Insert data at the end of records.

**SYNTAX:** EXTEND [VERTICAL-RANGE] STRING

**ALIAS:** EX,extend

**WHERE:** STRING is the value to be appended to the end of the records in the VERTICAL-RANGE.

The STRING must be delimited by the first non-blank non-numeric character. The same delimiter must be used to terminate the string.

The current position line is the last line modified by the Extend command.

**EXAMPLES:**

1. Put a comma at the end of the line at the current file position.  
>EX /,/cr
2. Put a period at the end of line 100.  
>EX /./cr
3. Put a period at the end of all lines between line numbers 100 and 200, inclusive.  
>EX 100-200 /./cr

## 2.13 FIND COMMAND

**FUNCTION:** Find a string or a line.

**SYNTAX:** FIND [VERTICAL-RANGE] [HORIZONTAL-RANGE]  
[.TRANSPARENT-CHAR] [;COUNT] STRING [REPETITION]

ALIAS: F,find,f

WHERE: TRANSPARENT-CHAR is a character within the STRING that is to be ignored in determining search satisfaction.  
STRING is the string to be found.  
COUNT indicates that the COUNT'th occurrence of the STRING in a line is the target string.  
REPETITION indicates the number of times STRING is to be found.

The STRING must be delimited by the first non-blank, non-numeric character. The same delimiter must be used to terminate the string. When a Find command is executed with no parameters, it implies a request to find the next occurrence of the last STRING entered by the Find command. Used in this fashion, the Find command begins the string search at the current position line. When the Find command is executed with only a VERTICAL-RANGE specified, it implies a request to position the Editor at the start of the VERTICAL-RANGE. FIND 0 will imply a request to position the editor at the beginning of the file and, conversely, FIND 65535 will imply a request to position the Editor at the end of the file. When the VERTICAL-RANGE is not specified, the string search will begin at the beginning of the file. An A in the COUNT or REPETITION field will imply a request to find all the occurrences of the STRING. The default value for COUNT is 1, and the default value for REPETITION is 1. When the REPETITION is greater than 1, the screen will display only the lines with an occurrence of the STRING. If the screen gets full, the display will scroll up until the list is exhausted or the user types CTRL-W to pause the list or CTRL-P to abort the list. The screen may be returned to the normal display by the entering of a null command or any other valid Editor command.

The current position line will always be the line with the last occurrence of the STRING.

- EXAMPLES:
1. Position the Editor to the beginning of the file.  
>F 0cr
  2. Position the Editor to the end of the file.  
>F 65535cr
  3. Position the Editor to line number 140  
>F 140cr
  4. Position the Editor on the first occurrence of the string SAM in the file.  
>F /SAM/cr
  5. Position the Editor on the first line with two occurrences of the string SAM between line number 100 and 200.  
>F 100-200 ;2/SAM/cr
  6. List all occurrences of the string SAM.  
>F /SAM/Acr
  7. List all lines with 2 or more occurrences of the string SAM.  
>F ;2/SAM/Acr

## 2.14 INSERT COMMAND

FUNCTION: Insert data into the file. (For editing files without line numbers.)

SYNTAX: INSERT [OFFSET]

ALIAS: I,insert,i

? WHERE: OFFSET is the offset from the current position line at which to begin inserting.

The Insert command initiates the input process in which all data following the Insert command is inserted into the file until a mode-change is input (F1 key).

When the Insert command is received, the current line position is set to the point at which input is to begin. The current line is cleared, pushing down all lines on the screen to make room for the data to be inserted. The cursor is then positioned on the cleared line to accept input. When page editing is terminated (F1 key), the cursor is returned to the command line. The Insert command in the command mode initiates page editing.

EXAMPLE: 1. To insert data into the file at the current file position.  
>Icr

## 2.15 LIST COMMAND

FUNCTION: Display a portion of the file.

SYNTAX: LIST [VERTICAL-RANGE]

ALIAS: L,list,l

The list command will cause the data within the VERTICAL-RANGE to be displayed on the screen. If the list is too long for the screen, the display will be scrolled up until the list is exhausted or the user types CTRL-W to pause the list or CTRL-P to abort the list. When the listing is finished or aborted, the user can return the screen to the normal display by entering a null command or any valid Editor command.

EXAMPLES: 1. Display the entire file.  
>Lcr

2. Display only the current file position.  
>L \*cr

3. Display the lines between line numbers 100 and 200, inclusive.  
>L 100-200cr

4. Display the next 10 lines (when editing a file without line numbers).  
>L \*-10cr



## 2.16 MERGE COMMAND

**FUNCTION:** Retrieve records from another XDOS file and include them in the edited file.

**SYNTAX:** MERGE [VERTICAL-RANGE] FILENAME [NEW-LINE-NUMBER  
[,NEW-INCREMENT]]

**ALIAS:** Merge

**WHERE:** FILENAME specifies a valid XDOS ASCII file.  
VERTICAL-RANGE specifies the range in FILENAME to be copied into the edited file.  
NEW-LINE-NUMBER is the line number to be assigned to the first line from FILENAME when it is inserted into the file.  
NEW-INCREMENT is the increment to be added to each successive line number as the lines are inserted into the edited file.

If the VERTICAL-RANGE is not specified, it will imply the request to get the entire FILENAME. If the user is editing without line numbers, the data from FILENAME will be inserted at the current position line. The line numbers that may be in FILENAME are lost upon extract. The line numbers applied will be as specified via NEW-LINE-NUMBER and NEW-INCREMENT, if specified, or will default to the current position line plus 10 with an increment of 10. If the establishment of new line numbers causes a conflict, the BLOCK/LINE MERGE RULES outlined in section 2.10 will apply.

The current position line following a merge will be the line following the last line inserted by the merge.

**EXAMPLES:** 1. Insert all the lines in the file named SAM into the edited file at the current file location.  
>MERGE SAMcr

2. Insert all the lines in the file named SAM into the edited file beginning with line number 400 with an increment of 1 between line numbers.  
>MERGE SAM 400,1cr

3. Insert line numbers 100 to 300, inclusive, in the file named SAM into the edited file, beginning with line number 500 and with increments of 2 between lines.  
>MERGE 100-300 SAM 500,2cr

## 2.17 MOVE COMMAND

**FUNCTION:** Move a group of records from one place within the file to another place within the file.

**SYNTAX:** MOVE [VERTICAL-RANGE] [/NEW-LINE-NUMBER [,NEW-INCREMENT]]

**ALIAS:** Move

**WHERE:** NEW-LINE-NUMBER is the destination line number for the first line moved (line numbered files only).  
NEW-INCREMENT is the line number increment to be applied to

each subsequent line of the move.

The Move command generates a DUPLICATE command followed by a DELETE command. For additional information, see those command descriptions.

- EXAMPLES:
1. Move this line to the XTRACT buffer.  
>MOVEcr
  2. Move line number 100 to the XTRACT buffer.  
>MOVE 100cr
  3. Move the lines between line numbers 100 and 200, inclusive, into the XTRACT buffer.  
>MOVE 100-200cr
  4. Move the next 10 lines into the XTRACT buffer (when editing a file without line numbers).  
>MOVE \*-10cr
  5. Move the line at the current file position to line number 131.  
>MOVE /131cr
  6. Move line 120 to 123.  
>MOVE 120/123cr
  7. Move the lines between 150 and 200, inclusive, to line number 190, with an increment of 1 between lines.  
>MOVE 150-200/190,1cr

## 2.18 NUMBER COMMAND

FUNCTION: Insert data into the file and prompt user with the next line number for insert (for editing line numbered files only).

SYNTAX: NUMBER [NEW-LINE-NUMBER [,NEW-INCREMENT]]

ALIAS: N,number,n

WHERE: NEW-LINE-NUMBER is the first line number to be prompted.  
NEW-INCREMENT is the value to be added to the NEW-LINE-NUMBER to form each succeeding line number prompted.

The Number command initiates the input process in which all data following the command is inserted into the file. The Number command with no parameters will cause the Editor to form the line number prompt by adding the default increment of 10 to the last line number in the file. Prompting will continue after each carriage return until a mode-change is encountered (F1 key). At that time, the Editor will again accept edit commands.

When the Number command is received, the current position line is set to the point at which input is to begin. The current line is cleared, pushing down all lines on the screen to make room for the data to be inserted. The cursor is then positioned on the cleared line following the line number prompt. When page editing is terminated (F1 key), the cursor is returned to the

command line.

EXAMPLES: 1. Begin page editing and begin prompting line numbers at the end of the file in increments of 10.  
>Ncr

2. Begin page editing at line number 100.  
>N 100cr

3. Begin page editing at line number 200, with line number prompts in increments of 1.  
>N 200,1cr

## 2.19 RANGE COMMAND

FUNCTION: Establish a default value for the vertical and horizontal ranges.

SYNTAX: RANGE [VERTICAL-RANGE] [HORIZONTAL-RANGE]

ALIAS: R, range, r

The Range command with no parameters deletes the previously established ranges. The Range command provides the default ranges for LIST, PRINT, CHANGE, FIND, and SAVE.

EXAMPLES: 1. Set the default VERTICAL-RANGE to lines 100 to 300, inclusive.  
>R 100-300cr

2. Set the default HORIZONTAL-RANGE to edit a full 132 character line.  
>R :1-132cr

3. Set the VERTICAL-RANGE default to lines 100 to 200, inclusive, and set the HORIZONTAL-RANGE to columns 16 to 80, inclusive.  
>R 100-200 :16-80cr

## 2.20 RESEQUENCE COMMAND

FUNCTION: Number, renumber, or unnumber the file.

SYNTAX: RESEQUENCE [BEGINNING-LINE-NUMBER] [/NEW-LINE-NUMBER  
[,NEW-INCREMENT]] [N]

ALIAS: Resequene

WHERE: BEGINNING-LINE-NUMBER is the line number at which to begin resequencing.  
NEW-LINE-NUMBER is the line number to be assigned to the first line resequenced.  
NEW-INCREMENT is the increment to be applied to form each subsequent line number within the file.  
N implies a request to strip the line numbers from the file.

If BEGINNING-LINE-NUMBER is not specified, the entire file will be resequenced. If NEW-LINE-NUMBER is not specified, resequencing will begin with the BEGINNING-LINE-NUMBER plus NEW-INCREMENT. If NEW-INCREMENT is not specified, it defaults to 10. The N option is mutually exclusive to the other options on this command.

The current line position is not altered by this command; however, the display will be altered to show the effect of the resequence.

EXAMPLES: 1. Resequence the entire file. Begin the new line numbers with line number 10, with increments of 10 between line numbers.  
>RESEcr

2. Resequence the file from line number 150 to the end of the file. Use increments of 10 between line numbers.  
>RESE 150cr

3. Resequence the entire file. Begin the line numbers with line number 100, with an increment of 2 between line numbers.  
>RESE /100,2cr

## 2.21 SEARCH COMMAND

FUNCTION: Find the next occurrence of a string.

SYNTAX: SEARCH [[.TR-CHR] [;COUNT] STRING]

ALIAS: S,search,s

WHERE: STRING is the string to be found.

The STRING is delimited by the first non-blank character. The same delimiter must be used to terminate the string. The Search command will begin a string search at the line following the current line position. A Search command with no STRING will search for the next occurrence of the last string specified by a Search command or a Find command. Thus, the Search command is equivalent to a FIND 1-65535 /STRING/ in an unlined file, or FIND n-65535 / STRING/ in a numbered file (where n is the current line number + 1).

The Search command places the current line position at the next occurrence of the string.

EXAMPLES: 1. Find the next occurrence of the string SAM.  
>S /SAM/cr

2. Find the first occurrence of the string BILL and then find the next occurrence.  
>F /BILL/cr  
>Scr

## 2.22 TAB COMMAND

FUNCTION: Alter the tab settings used for data input.

SYNTAX: TAB [CHAR] [COUNTN]1-20

ALIAS: Tab

WHERE: CHAR is the non-numeric, non-space character to designate a tab request during page editing.  
COUNTN is the column position to which a tab is to be set.

When no CHAR is specified, the default tab key is the TAB key. The tabs can be reset by entering the Tab command with no parameters. The 1 to 20 COUNTN's specified must be separated by commas. The COUNTN values will add additional tab stops until the tabs are reset by a Tab command with no parameters.

The display is unchanged by the Tab command.

EXAMPLES: 1. Set tabs at columns 10 and 20 and use the character Z as a special tab key.

>TAB Z 10,20cr

2. Add an additional tab stop at column 15.

>TAB 15cr

3. Delete all tab stops.

>TABcr

## 2.23 VERIFY COMMAND

FUNCTION: Display on the screen, any time a change is made to the file, the value after the change.

SYNTAX: VERIFY [OFF]

ALIAS: V,verify,v

WHERE: OFF disables the verify.

The verify is enabled by entering the Verify command with no parameters. Certain multiple update commands will display a list of occurrences rather than a sequential portion of the file surrounding the changed record. Should the list become too long for a single screen, the display will scroll up until the list is exhausted or the user types CTRL-W to pause the list or CTRL-P to abort the list. The screen can be returned to the normal display by entering a null command or any other valid Editor command. Use of the CTRL-P control will cause the update command to stop processing as of the last line listed on the screen.

Execution of the Verify command does not alter the display.  
The Verify command causes other update commands to display as noted above.

EXAMPLES: 1. Enable verify.

>Vcr

2. Disable verify.

>V OFFcr

## 2.24 XTRACT COMMAND

FUNCTION: Insert the data in the buffer into the file.

SYNTAX: XTRACT [NEW-LINE-NUMBER [,NEW-INCREMENT]] [A]

ALIAS: X,xtract,x

WHERE: NEW-LINE-NUMBER is the point at which to insert the first line from the buffer.  
NEW-INCREMENT is the increment to be applied to each subsequent line number inserted into the file.  
A indicates that the xtract buffer is to be deleted.

In unlined files, the xtract will always copy the data in the buffer into the file at the current position line. If the establishment of new line numbers generates a conflict, the BLOCK/LINE MERGE RULES outlined in paragraph 2.10 will be applied to resolve the conflict.

EXAMPLES: 1. Insert the data in the XTRACT buffer into the file at the current file position.  
>Xcr  
  
2. Insert the data in the XTRACT buffer into the file beginning with line number 200, with an increment of 10 between lines.  
>X 100cr  
  
3. Insert the data in the XTRACT buffer into the file beginning with line number 200, with an increment of 1 between lines.  
>X 200,1cr

## 2.25 EXAM COMMAND

FUNCTION: Examine all or portion of an external file.

SYNTAX: EXAM [VERTICAL-RANGE] FILENAME

ALIAS: EXA

WHERE: VERTICAL-RANGE specifies the range in FILENAME to be listed to the console.

If the VERTICAL-RANGE is not specified, it will imply the request to list the entire FILENAME.

This command proves useful prior to invoking a MERGE command as it allows to determine the vertical range of an external file to be further merged to the file being edited.

The EXAM command can also be invoked to display a user-written file that could include the repertory of commands of the CRT Editor (MENU function).

## 2.26 EDITOR MESSAGES

Table 1-4 contains a list of the Editor messages that may be encountered when using the Editor.

TABLE 1-4. Editor Messages

MESSAGE	PROBABLE CAUSE
NAME REQUIRED	The Editor (or MERGE or EXAM commands) were invoked without specifying the file to be edited / merged / listed.
ILLEGAL FILE NAME	The name of the file specified is not a valid XDOS file name.
INVALID DEVICE	The device specified is not a valid XDOS device.
INVALID DRIVE	The drive number specified is not supported by XDOS.
XXXXXXXXXX.DUP DUPLICATE FILE NAME	The file name specified cannot be created as it already exists.
XXXXXXXXXX.DUP IS WRITE PROTECTED	The file name specified has been marked write protected and cannot be edited without specifying FILENAME2.
XXXXXXXXXX.DUP IS DELETE PROTECTED	The file name specified has been marked delete protected and cannot be edited without specifying FILENAME2.
XXXXXXXXXX.DUP HAS INVALID FILE TYPE	The file name specified is not an ASCII file and cannot be edited.
OPTION CONFLICT	The options specified are conflicting.
INSUFFICIENT DISK SPACE ON DRIVE X CONTINUE (Y/N)?	When invoking the Editor, the system compared the free space on drive X to the file size of the file to be edited (when editing an existing file). If drive X is the drive on which the edited result is to be stored, a DISK SPACE FULL message will appear and the edit will be aborted when the file is written to disk. If drive X is not the drive on which the edited result is to be stored, the edit may continue as long as the edit only makes one pass through the file. An attempt to reposition the Editor to the beginning of the file may result in the disk space full error. If DRIVE was specified in the Edit command as the same as the drive on which the edited result is to be stored, the first time this message appears, it is the caution to do a one pass edit. A response of Y will

allow the edit to continue. A response of N will return control to XDOS.

In the course of edition, the message indicates that any attempt to increase the file size may cause a disk space full error. Answering "Y" may cause unpredictable results.

SOURCE FILE SEQUENCE ERROR. CONTINUE (Y/N) ?

The source file contains non-numeric data in columns 1-5, or does not contain a space in column 6; or the line numbers in the source file are not in ascending order; or the line numbers in the source file are not unique. Answering "N" exits the Editor. Answering "Y" causes the edit session to continue; however, the file will be considered as unnumbered.

WHAT? An invalid edit command was entered.

COMMAND SYNTAX ERROR

A valid edit command was entered but the syntax of the command was not correct.

DEVICE NOT READY

A MERGE or PRINT command was received and the appropriate device is not ready.

XXXXXXXXXX.XX DOES NOT EXIST

A MERGE or EXAM command was received but the specified FILENAME does not exist.

COMMAND ABORTED

The CTRL-P control was entered during file access, and the command last entered has been aborted prior to completion.

STRING NOT FOUND

The string specified in the CHA?NGE, FIND, or SEARCH command cannot be found in the specified ranges.

LINE NUMBER OVERFLOW RESEQUENCE

An attempt to generate a line number resulted in a value greater than 65535, and wrapped around to or beyond 00000. The file must be resequenced or the command results are unpredictable.

XTRACT BUFFER OVERFLOW

An attempt to move or duplicate too large a block has occurred.

SOURCE LINES NOT FOUND

An attempt to MERGE, EXAM, MOVE, DUPLICATE, SAVE, or XTRACT occurred and the referenced data does not exist; or the xtract buffer is empty.

EOF ENCOUNTERED

An attempt to MOVE or DUPLICATE a block of data was terminated by the end of a file. Data up to that point has been processed.

BOF OR EOF ENCOUNTERED

An attempt to page or scroll beyond the beginning or the end of the file has occurred.

LINE NUMBER CONFLICT AT XXXXX RESEQUENCE (Y/N)?



Block editing has resulted in a line number conflict. The Editor will resolve the conflict according to the rules described in paragraph 1.6.9).

IS THE SOURCE FILE LINE NUMBERED (Y/N)?

A MERGE or EXAM command was entered and the Editor needs to know if the file to be merged or examined is numbered or not.



## EFFICIENT OPERATION OF THE CRT EDITOR

---

The user is recommended to observe the following simple rules to make an optimal use of the XDOS CRT EDITOR :

1-.Although the Editor can be operated in a 32K RAM environment, the larger the RAM capacity, the faster the overall operations since disk accesses will be less frequent. As the EXORset supports two memory maps, it is thus recommended to invoke XDOS in the map having the largest size in terms of contiguous RAM starting at address 0000.

2-.Using diskettes containing fragmented files will result in some speed reduction; the BACKUP command, and its "R" option, will help in reorganizing files.

3-.In order to avoid "DISK SPACE FULL" problems, it is suggested to edit source files on diskettes having the maximum amount of free space; this is especially true in a mini-floppy disk environment. Although the following is not mandatory, it is a good practice to insert the system disk with the Editor and some other useful commands (DIR, FREE, DEL, ...) in drive #0, while using a DOSGENed diskette in drive #1 to hold the files to be edited, and these files only.

4-.It is highly recommended to frequently save the edited buffer to the disk during an editing session and not only upon editor exit. The SAVE command, when invoked without arguments, serves that purpose.

