

**WRITER'S  
MASTER COPY**

The logo for Scientific Data Systems (SDS) is centered over the text. It consists of the letters 'S', 'D', and 'S' in a stylized, outlined blue font. The letters are interconnected, with the 'D' having a vertical line on its right side that extends to the top and bottom of the 'S' on either side.

SCIENTIFIC DATA SYSTEMS

Reference Manual

**DO NOT REMOVE**

SDS 940 QED

**QED**  
**REFERENCE MANUAL**  
for  
**SDS 940 TIME-SHARING COMPUTER SYSTEMS**

PRELIMINARY EDITION

January 1969

90 11 12B



SCIENTIFIC DATA SYSTEMS/701 South Aviation Boulevard/El Segundo, California 90245

## REVISION

This publication, SDS 90 11 12B, is a minor revision of the SDS 940 QED Reference Manual, SDS 90 11 12A (dated April, 1968). A change in the text from that of the previous manual is indicated by a vertical line in the margin of the page.

## RELATED PUBLICATIONS

<u>Title</u>	<u>Publication No.</u>
SDS 940 Computer Reference Manual	90 06 40
SDS 940 FORTRAN II Reference Manual	90 11 10
SDS 940 BASIC Reference Manual	90 11 11
SDS 940 DDT Reference Manual	90 11 13
SDS 940 CAL Reference Manual	90 11 14
SDS 940 Conversational FORTRAN Reference Manual	90 11 15
SDS 940 Time-Sharing System Technical Manual	90 11 16
SDS 940 TAP Reference Manual	90 11 17
SDS 940 Terminal User's Guide	90 11 18

### NOTICE

The specifications of the software system described in this publication are subject to change without notice. The availability or performance of some features may depend on a specific configuration of equipment such as additional tape units or larger memory. Customers should consult their SDS sales representative for details.

# CONTENTS

PREFACE	iv		
1. INTRODUCTION	1		
Conventions	1	O <sup>c</sup> x	7
Operating Procedures	2	S <sup>c</sup>	7
Log In	2	F <sup>c</sup>	7
Escape	2	X <sup>c</sup> x	7
Exit and Continue	2	P <sup>c</sup> x	7
Log-Out	2	K <sup>c</sup>	7
		E <sup>c</sup>	7
		H <sup>c</sup>	8
		U <sup>c</sup>	8
		N <sup>c</sup>	8
		R <sup>c</sup>	8
		T <sup>c</sup>	8
		Y <sup>c</sup>	8
		M <sup>c</sup>	8
		J <sup>c</sup>	8
2. ENTERING TEXT INTO QED	3		
Reading Paper Tape	3	4. STRING BUFFERS	9
Reading a File	3	bLOAD	9
Creating New Text	3	a <sub>1</sub> ,a <sub>2</sub> ;bLOAD	9
Text Addressing	3	bBUFFER	9
Addressing by Position	3	bKILL	9
Addressing by Dot	4	B <sup>c</sup>	9
Addressing by Dollar Sign	4	a <sub>1</sub> ,a <sub>2</sub> ;bGET	10
Addressing by Labels	4	SUBSTITUTE	10
Addressing by Arbitrary Text	4	V <sup>c</sup>	11
Miscellaneous Address Notations	4	L <sup>c</sup>	11
Correcting Typing Errors	5	String Buffer Diagnostics	11
A <sup>c</sup>	5	FULL	11
W <sup>c</sup>	5		
Q <sup>c</sup>	5	5. TEXT OUTPUT	12
Tabs	5	Output to a File	12
Main Text Buffer Diagnostics	5	Output to Teletype	12
Nearly Full	5	*a <sub>1</sub> , a <sub>2</sub> /	12
No Room	5	*a <sub>1</sub> , a <sub>2</sub> PRINT	12
Fatal Error	5	* <sup>(f)</sup>	12
Edit Terminated	6	Output to Paper Tape	13
3. TEXT MODIFICATION	6		
a <sub>1</sub> INSERT	6		
a <sub>1</sub> , a <sub>2</sub> DELETE	6		
a <sub>1</sub> , a <sub>2</sub> CHANGE	6		
a <sub>1</sub> , a <sub>2</sub> EDIT	6		
a <sub>1</sub> , a <sub>2</sub> MODIFY	7		
Control Characters	7		
C <sup>c</sup>	7		
D <sup>c</sup>	7		
Z <sup>c</sup> x	7		

## APPENDIX

QED SUMMARY	15
-------------	----

## PREFACE

QED is a generalized text editor that can be used to create or modify programs, data, or reports for any of the SDS 940 subsystems. It is particularly suited to time-sharing, since the power and flexibility of an on-line terminal makes it possible for the user to edit text more efficiently than he could with a card or magnetic tape system. Briefly, QED allows the user to

- Insert, create, or delete numeric or alphabetic characters and lines of text.
- Copy, modify, or delete all or portions of an existing line of text.
- Access any line of text by simply specifying a set of characters or digits contained in the line.
- Store commonly used phrases in special character string buffers for insertion into the main text buffer.
- Tabulate text in one to six columns, with the tabs set automatically or by the user.

QED can be used in many ways. The user may have his program or data typed off-line on paper tape or cards and later edit it on-line through QED. Or, he may prepare it on-line in QED and then output his results to the appropriate file. This is particularly convenient when a program is lengthy or the text is difficult to handle. In addition to creating new text with QED, the user may also edit his existing files by simply calling a file into QED, modifying or updating it as required, and then writing it back onto the disc file storage.

In summary, the SDS 940 user who learns QED will find that he has an efficient and time-saving editing tool that can be used continually to modify or create any textual information.

# 1. INTRODUCTION

Many 940 computer users require a means of entering and editing text strings written in some symbolic language. The online time-sharing editor, QED, provides a simple command language allowing the user to think in terms of text structure rather than in some arbitrary framework. QED is basically line-oriented, but the generality of text referencing is preserved by the use of a content-addressing feature. Although QED treats the user's text as a single string of characters in a "main text buffer", line structure is maintained by regarding stored carriage return characters as line delimiters. Thus, lines can be referred to by line number, although line numbers are subject to change if carriage returns are added or deleted within the string during the course of editing.

QED has three modes of operation:

1. The command mode.
2. The text mode.
3. The line edit mode.

The primary state of QED is the command mode. A carriage return is executed and an \* printed as an indication that the command mode has been entered. The commands INSERT, DELETE, and PRINT are sufficient to perform any editing operation, although additional commands are provided for convenience.

When an INSERT command is given followed by a carriage return, QED enters the text mode. In this mode, one or more lines of text may be inserted into the main text buffer at a specified location. Termination of the text mode is accomplished by means of the control D character (denoted in this manual by D<sup>c</sup>). A control character is generated by first depressing and holding the control key (CTRL) and then striking the key for the specific control character. Usually control characters do not print on the Teletype, to avoid cluttering lines of printed text with non-text characters. A convenient feature of the text mode is the capability of setting and using tab stops.

The DELETE command may be used to delete one or more specified lines of text from the main text buffer. The text mode is not entered when a DELETE command is given, since no new text is to be entered into the buffer as a result of the command.

The PRINT command may be used to print one or more specified lines (in the main text buffer) on the Teletype. As with DELETE, the text mode is not entered. Non-printing characters (i. e., control characters that are normally not printed) are preceded by an ampersand (&).

QED has the capability of transferring text between the main text buffer and disc storage or paper tape. The READ and WRITE commands can be used for these functions.

QED also provides means of making minor changes in lines of text in the main text buffer, where the user may want to

insert or delete a few characters. Such changes may be accomplished most readily in the line edit mode, entered by use of an EDIT command specifying the line in which changes are to be made. In the line edit mode, QED will respond to various special control characters in addition to those normally recognized in the text mode. Line edit functions allow text characters to be copied selectively from the old line into the new one, and allow additional text characters to be inserted as specified. A line edit is terminated either by a D<sup>c</sup>, causing the remainder of the old line to be copied into the new one, or by a carriage return, causing the remainder of the old line to be discarded.

The SUBSTITUTE command may be used to substitute a specified string of characters for another specified string of characters in a specified line or lines.

QED is not a programming language. However, using QED, it is possible to "program" simple editing operations. This can be accomplished by placing control commands in one of ten auxiliary string buffers via a LOAD command and then calling that buffer via a BUFFER command. Since a buffer may call itself, program looping is possible. Also, since command strings may be stored as text, they can themselves be edited before being executed by QED.

## CONVENTIONS

For clarity, several conventions have been used throughout this manual:

1. Underscored copy in an example represents copy produced by QED. Copy that is not underscored in an example represents copy typed by the user.
2. Nonprinting control characters are represented by an alphabetic character and a superscript c (e. g., D<sup>c</sup>). To obtain a nonprinting character, the user simultaneously depresses the specified alphabetic key and the Control (CTRL) key. For editing purposes some control characters will cause a symbol to be printed, but this symbol does not appear in the final version of an edited line.
3. The Ⓔ notation appearing after some lines in the examples represents the Carriage Return key. This key is labeled RETURN on the Teletype keyboard. The user must depress the Carriage Return key after each command to inform QED the command is terminated. The computer then upspaces the paper automatically. The Ⓕ notation represents the Line Feed key.
4. Throughout this manual, QED commands have been written in full. However, QED will also accept an abbreviated form of the commands. The user need actually type only the first letter rather than the whole command.

## OPERATING PROCEDURES

The standard procedure for gaining access to an SDS 940 time-sharing computer center from a remote Teletype terminal is described in the SDS 940 Terminal User's Guide, Publication No. 90 11 18. This publication also includes information concerning the 940 Executive System and the calling of the various subsystems available to the terminal user. The following paragraphs summarize the standard procedures as they apply to QED users.

### LOG IN

To gain access to the computer, the following operating sequence is performed:

1. If the FD-HD (Full Duplex-Half Duplex) switch is present, turn the switch to FD. This is a toggle switch with two locking positions. When the Teletype is not connected to the computer (sometimes called the Local Mode), this switch must be in the HD position. When the Teletype is connected to the computer, this switch must be in the FD position.
2. Press the ORIG (originate) key, which is located at the lower right corner of the console, to obtain a dial tone before dialing the computer center.

3. Dial the computer center number. When the computer accepts your call, the ringing will change to a high-pitched tone. Then, a request for the user to log in will appear on the Teletype.

PLEASE LOG IN:

4. The user must then type his account number, password, name and project code (if applicable) in the following format:

PLEASE LOG IN: number password;name;proj. code  $\text{\textcircled{RET}}$

Only persons who know the account number, password, and name may log in under that particular combination. The following examples all illustrate acceptable practice:

PLEASE LOG IN: A7PASS;1234;QED  $\text{\textcircled{RET}}$

PLEASE LOG IN: C8WORD;0001;MANUAL  $\text{\textcircled{RET}}$

PLEASE LOG IN: F5PW;PSEUDO;  $\text{\textcircled{RET}}$

The optional 1-12 character project code is provided for installations that have several programmers using the same account number. The project code is not checked for validity.

If the user does not correctly type his account number, password, and name within a minute and a half, a message is transmitted instructing him to call the computer center for assistance. The computer will then disconnect the user, and the dial and log-in procedure will have to be repeated.

5. If the account number, password (nonprinting), and name are accepted by the computer, it will print

READY, the date, and the time on one line and a dash at the beginning of the following line:

READY date,time

The dash indicates that the 940 Executive is ready to accept a command.<sup>f</sup>

In response to the dash, the user types

QED  $\text{\textcircled{RET}}$

When ready to accept commands, QED responds with an asterisk.

### ESCAPE

The ESCAPE  $\text{\textcircled{ESC}}$  key<sup>tt</sup> may be used at almost any time. It causes the subsystem to abort the current operation and ask for a new command. Striking the  $\text{\textcircled{RET}}$  key before terminating a command with  $\text{\textcircled{ESC}}$  aborts the command.

### EXIT AND CONTINUE

-CONTINUE  $\text{\textcircled{RET}}$

Depressing the  $\text{\textcircled{ESC}}$  key several times in succession returns control to the Executive, which responds with a dash (-). If the user wants to return to QED without losing his program, and if he has not subsequently called another subsystem (e.g., BASIC, TAP, CAL), he may type CONTINUE. The computer will type QED and return to it without any initialization. Meanwhile, nothing in core is destroyed.

### LOG-OUT

When the user wishes to be disconnected from the computer, he depresses  $\text{\textcircled{ESC}}$  several times in succession to return to the Executive and then types:

-LOGOUT  $\text{\textcircled{RET}}$

or

-EXIT  $\text{\textcircled{RET}}$

The computer will respond by printing the amount of computer time and connect (line) time charged to the user's account since the previous log-in procedure was completed.

<sup>f</sup>In some 940 time-sharing systems the commercial "at" sign, @, is used to indicate that the 940 Executive is ready to accept a command.

<sup>tt</sup>In some 940 time-sharing system configurations the RUBOUT or ALT MODE key is used instead of the ESCAPE key. Where ESC appears in this manual, RUBOUT or ALT MODE may be substituted.

## 2. ENTERING TEXT INTO QED

In order to edit text using the 940's editor, the text or program must first be entered into the QED subsystem. The methods available for this are

1. Preparing paper tape off-line and then reading it from the Teletype.
2. Reading in from a previously prepared file.
3. Creating and entering new text on-line from the Teletype.

### READING A PAPER TAPE

To prepare paper tape off-line, the user must

1. Set the Teletype in the local mode.
2. Set the half duplex mode.
3. Turn the paper tape punch on.

If a logical line comprises several physical lines of typing, all lines except the last must end with a line feed followed by a carriage return. The last physical line must end with a carriage return followed by a line feed. The resulting paper tape may then be read into QED by using the command

```
*READ TELETYPE (RET)
```

QED waits for the user to place the paper tape in the reader and press the reader switch on. When the entire tape has been read, the user depresses the CTRL key and D key simultaneously. (Called Control D, written as D<sup>c</sup>.) This terminates the READ command. The user may have punched D<sup>c</sup> onto the paper tape. If so, he need not type it manually. If QED contained text before the READ command, new text will follow the old text.

### READING A FILE

Similarly, text may be entered into QED from a file previously prepared by using either the 940 Executive or one of the subsystems. To read a file the user types

```
*READ /FILENAME/ (RET)
```

This command will bring in all of the text included in the specified file. Again, if QED contained text before the READ command was issued, the new text will follow the old text.

### CREATING NEW TEXT

It is often convenient to enter new text directly from the Teletype into QED. To do this the user types

```
*APPEND (RET)
```

After receiving the command, QED waits for the new text. The user may type any number of lines, terminating each line with a carriage return (RET). The computer automatically supplies the line feed (LF). When all the desired text has been typed, the user must then type D<sup>c</sup> (Control D) to terminate the APPEND command. If any text was present in QED before using the APPEND command, the new text is appended to it.

The user may see all of the text in QED by typing /. This will cause the computer to print all of the text entered into QED so far, as illustrated below. (The following example is continued and expanded throughout most of this manual. The user may find it helpful to be on-line and working the example(s) as he reads the manual. Note again that underlined copy is that which is generated by the computer.)

```
*APPEND (RET)
TO CREATE NEW TEXT, USE APPEND. Dc
*/
TO CREATE NEW TEXT, USE APPEND.
*APPEND (RET)
NEW TEXT WILL FOLLOW ALL OLD TEXT. Dc
*/
TO CREATE NEW TEXT, USE APPEND.
NEW TEXT WILL FOLLOW ALL OLD TEXT.
*APPEND (RET)
USE / TO SEE CONTENTS
OF MAIN TEXT BUFFER. Dc
*/
TO CREATE NEW TEXT, USE APPEND.
NEW TEXT WILL FOLLOW ALL OLD TEXT.
USE / TO SEE CONTENTS
OF MAIN TEXT BUFFER.
```

### TEXT ADDRESSING

The place where QED keeps all the entered text is called the main text buffer. The text is stored in lines, where a line is simply all the characters between two carriage returns. Most QED commands work with one line or a range of lines at a time. The user has several ways of referencing a line: by position, by current line, by last line, by labels, and by text. These are discussed individually below.

#### ADDRESSING BY POSITION

A line may be referenced by its physical position in the main text buffer. For example, the tenth line is line 10, the first line is line 1, and so on.

```
*1/
TO CREATE NEW TEXT, USE APPEND.
*3/
USE / TO SEE CONTENTS
*3,4/
USE / TO SEE CONTENTS
OF MAIN TEXT BUFFER.
```

Note that two addresses separated by a comma denote a range of lines.



## ADDRESSING BY DOT

A second way of referring to a line is by using the dot (typed as a period on the Teletype). The dot is the address of the current line, i. e., the line accessed or referenced most recently. Each time a line is accessed, that line then becomes the current line and is addressable by dot.

```
*3/  
USE / TO SEE CONTENTS  
*./  
USE / TO SEE CONTENTS  
*3,4/  
USE / TO SEE CONTENTS  
OF MAIN TEXT BUFFER.  
*./  
OF MAIN TEXT BUFFER.
```

Note that when a range is referenced before using the dot, the last line of the range becomes the current line.

## ADDRESSING BY DOLLAR SIGN

The dollar sign refers to the last line in the main text buffer. It is automatically updated each time additional text is appended to the main text buffer.

```
*$/  
OF MAIN TEXT BUFFER.  
*APPEND (*)  
ADDRESS LINES BY POSITION, ., $, LABEL OR TEXT. Dc  
*$/  
ADDRESS LINES BY POSITION, ., $, LABEL OR TEXT.  
*/  
TO CREATE NEW TEXT, USE APPEND.  
NEW TEXT WILL FOLLOW ALL OLD TEXT.  
USE / TO SEE CONTENTS  
OF MAIN TEXT BUFFER.  
ADDRESS LINES BY POSITION, ., $, LABEL OR TEXT.
```

## ADDRESSING BY LABELS

When the number of lines in the main text buffer makes counting difficult, the user can initiate a search for the line containing a label by putting the label name in colons, e. g., :label name:. A label is any series of characters beginning in column one on the Teletype with the requirement that they be followed by a character which is not a letter or digit. If the user desires to use more than one word or series of characters, all spaces must be included between the colon or the address is invalid. The computer's search for a label always begins with the first line after the current line, continues to the end of the text, then starts again at the beginning of the text and continues down through the current line (or to the starting point). If there is more than one occurrence of a label, the first line found containing the label will be printed.

```
*:USE:/  
USE / TO SEE CONTENTS  
*:USE:,:OF:/  
USE / TO SEE CONTENTS  
OF MAIN TEXT BUFFER.  
*:OF:,$/  
OF MAIN TEXT BUFFER.  
ADDRESS LINES BY POSITION, ., $, LABEL OR TEXT.
```

## ADDRESSING BY ARBITRARY TEXT

If no unique label exists on a particular line, the user can initiate a search for the line containing specific text by putting the text in brackets, e. g., [text]. The specified text may occur anywhere in the line. (Note that [ is obtained by typing Shift K and ] is obtained by typing Shift M.) As with labels, the search begins with the first line after the current line.

```
*[CREATE]/  
TO CREATE NEW TEXT, USE APPEND.  
*[SEE] , [LINES] /  
USE / TO SEE CONTENTS  
OF MAIN TEXT BUFFER.  
ADDRESS LINES BY POSITION, ., $, LABEL OR TEXT.
```

All searches can be started at any line, rather than at the line after the current one, by putting the starting line immediately before the search construct.

```
*:OF:[.] /  
OF MAIN TEXT BUFFER.
```

## MISCELLANEOUS ADDRESS NOTATIONS

QED also allows the user to add to and subtract from a line address. The following examples are based on the contents of the main text buffer developed thus far.

```
*$-2/  
USE / TO SEE CONTENTS  
*:TO:+1/  
NEW TEXT WILL FOLLOW ALL OLD TEXT.  
*[CREATE] , [BUFFER] -2/  
TO CREATE NEW TEXT, USE APPEND.  
NEW TEXT WILL FOLLOW ALL OLD TEXT.
```

By depressing just a Line Feed in response to the QED asterisk, the user commands the computer to print the line after the current line. In this case, the Line Feed is simply an abbreviation of .+1/. For example:

```
*.+1/  
USE / TO SEE CONTENTS  
*(†)  
OF MAIN TEXT BUFFER.
```

Similarly, the upward arrow (†) causes the computer to print the line before the current line, i. e., it is the equivalent of the .-1/ command.

To obtain a position address for a line, the user may type an equals sign following any legal address:

```
*:TO:=1  
*[MAIN]=4  
*$=5
```

It is often convenient to use \$= to find out how many lines have been entered into QED.

Similarly, a user can obtain the label of a line by typing a left-pointing arrow following any legal address which references the line. This is convenient if he is working with a large block of text and wants to know, for example, which line is line 150.

```
*[MAIN] ← :OF:
*2 ← :NEW:
```

## CORRECTING TYPING ERRORS

If a typing error occurs while entering text, there are several methods available for correcting the errors immediately.

**A<sup>C</sup>** The A<sup>C</sup> (Control A) prints an upward arrow (↑) (not to be confused with Shift N which is used to print the previous line) to give the user an indication that he has used the A<sup>C</sup>. Each time the A<sup>C</sup> is typed, the last character on the line is erased (internally). Repeated use of the A<sup>C</sup> deletes several characters from the line.

```
*APPEND (REF)
CONTRLACOL A DELETES A CHARTACACTER. DC      As typed
CONTRL↑OL A DELETES A CHART↑ACTER.           As printed
*$/
CONTROL A DELETES A CHARACTER.
```

**W<sup>C</sup>** The W<sup>C</sup> (Control W) prints a backward slash (\) on the Teletype to inform the user that he has used a W<sup>C</sup>. This editing character deletes the word which was typed immediately preceding the W<sup>C</sup> and is useful when several characters in a row have been incorrectly typed. The first blank encountered preceding a nonblank character terminates the deletion; that blank is not deleted. The A<sup>C</sup> used repeatedly would accomplish the same function as the W<sup>C</sup>.

```
*APPEND (REF)
CONTROL W DLTESWCDELETES A WORD. DC          As typed
CONTROL W DLTES\DELETES A WORD.              As printed
*$/
CONTROL W DELETES A WORD.
```

**Q<sup>C</sup>** The Q<sup>C</sup> (Control Q) prints a left-pointing arrow (←) to provide a record that the Q<sup>C</sup> was used. Immediately after the arrow is printed, the carriage returns, the paper feeds one line and the print head is positioned to start a new line.

The logical line on which the Q<sup>C</sup> was given is erased. The user is in the same position as having just given the carriage return at the end of the previous logical line.

Repeated use of the Q<sup>C</sup> will delete several lines in the same fashion that repeated use of the A<sup>C</sup> will delete several characters.

```
*APPEND (REF)
CONTROL Q DELETESQC                               As typed
CONTROL Q DELETES ←                               As printed
TO DELETE THE CURRENT LINE TYPE CONTROL Q. DC
*$/
TO DELETE THE CURRENT LINE TYPE CONTROL Q.
```

## TABS

QED provides tabulating capabilities for indenting, typing figures in columns, etc. Four tabs are automatically set at positions 8, 16, 32, and 40 on the Teletype. They may be used at any time by depressing Control I (I<sup>C</sup>).

If the user wishes to change any or all of the tabs, he types TABS. QED will now accept up to six new tab positions. For example:

```
*TABS (REF)
7 (REF)
*
```

This sets a tab at position 7 and removes the other three tabs. To change all the tabs, the user might type

```
*TABS (REF)
7,15,30,45 (REF)
*
```

## MAIN TEXT BUFFER DIAGNOSTICS

There are four QED diagnostics that can occur when using the main text buffer. These are listed below and an explanation of the causes and the recommended solution are also given.

### NEARLY FULL

The APPEND command actually reads the text into string buffer 0 (see Chapter 4). When the D<sup>C</sup> is given, the APPEND is terminated and the text is transferred to the main text buffer. By reading into string buffer 0, the APPEND command can allow editing during text input.

When approximately 1400 characters have been typed into QED with one APPEND command, string buffer 0 will be nearly full. The diagnostic NEARLY FULL then is typed out. The correct action is to type a D<sup>C</sup> to end the APPEND command. Another APPEND can then be given.

It is good practice to write the contents of the main text buffer out onto a disc file after ending the APPEND command. If the contents of the main text buffer are subsequently destroyed, all that is required is to read the disc file back into QED.

### NO ROOM

The memory capacity of the QED main text buffer has been exceeded. When the user gets this diagnostic, the text must be broken into two or more segments and handled separately.

### FATAL ERROR

A QED failure has occurred. The cause could be an unusual combination of events that QED could not anticipate.

The user should try to save any main text buffer material he may have, then log out, log back in, and try again. If the failure is repeated, the computer center should be called to obtain technical assistance.

## EDIT TERMINATED

The diagnostic NEARLY FULL is typed out as a warning when string buffer 0 is nearly full. The user should terminate the

APPEND with a D<sup>C</sup> at that point. If the APPEND is not terminated by the user, QED will terminate the APPEND, type out the diagnostic EDIT TERMINATED, and return control to the user with an asterisk (\*). Another APPEND can then be given.

## 3. TEXT MODIFICATION

After text has been entered into the main text buffer of QED the user often realizes he needs to make several insertions, deletions, or replacements to complete his work. He can accomplish this simply and quickly by using the INSERT, DELETE, CHANGE, EDIT, and MODIFY commands.

**a<sub>1</sub>INSERT** QED allows the user to insert one or more lines of text into the main text buffer. The user finds the line that the new text will precede and types any legal address of the line (a<sub>1</sub>) and INSERT. The computer will then wait for the text that is to be inserted. A D<sup>C</sup> terminates the inserted text.

```
*1INSERT (REF)
QED IS A POWERFUL EDITOR. DC
*/
QED IS A POWERFUL EDITOR.
TO CREATE NEW TEXT, USE APPEND.
NEW TEXT WILL FOLLOW ALL OLD TEXT.
USE / TO SEE CONTENTS
OF MAIN TEXT BUFFER.
ADDRESS LINES BY POSITION, ., $, LABEL OR TEXT.
CONTROL A DELETES A CHARACTER.
CONTROL W DELETES A WORD.
TO DELETE THE CURRENT LINE TYPE CONTROL Q.
```

Note again that the print command (/) without line addresses prints the entire main text buffer. The following example is a further illustration of the INSERT command.

```
*:USE:INSERT (REF)
TO INSERT NEW TEXT, USE INSERT. DC
*/
TO INSERT NEW TEXT, USE INSERT.
```

Note that after using the INSERT command, the current line becomes the line that was inserted.

```
*1,:USE:/
QED IS A POWERFUL EDITOR.
TO CREATE NEW TEXT, USE APPEND.
NEW TEXT WILL FOLLOW ALL OLD TEXT.
TO INSERT NEW TEXT, USE INSERT.
USE / TO SEE CONTENTS
```

**a<sub>1</sub>,a<sub>2</sub>DELETE** To delete one or more lines, the user types a line address, a<sub>1</sub> (or two line addresses separated by a comma), and the word DELETE. After using this command, the current line becomes the line preceding the deleted text. If the last line of the main text buffer is deleted, the \$ is automatically

reassigned. The following examples illustrate these features of the DELETE command.

```
*7,$DELETE (REF)
*.=6
*/
QED IS A POWERFUL EDITOR.
TO CREATE NEW TEXT, USE APPEND.
NEW TEXT WILL FOLLOW ALL OLD TEXT.
TO INSERT NEW TEXT, USE INSERT.
USE / TO SEE CONTENTS
OF MAIN TEXT BUFFER.
*$=6
*$DELETE (REF)
*$=5
```

**a<sub>1</sub>,a<sub>2</sub>CHANGE** The user may replace one or more lines of text with new text. Several lines may replace one line, or one line may replace several. New text is always terminated by a Control D.

```
*1CHANGE (REF)
QED IS THE EDITOR OF THE SDS 940.
TO REPLACE A LINE, USE CHANGE. DC
*/
QED IS THE EDITOR OF THE SDS 940.
TO REPLACE A LINE, USE CHANGE.
TO CREATE NEW TEXT, USE APPEND.
NEW TEXT WILL FOLLOW ALL OLD TEXT.
TO INSERT NEW TEXT, USE INSERT.
USE / TO SEE CONTENTS
*2,$CHANGE (REF)
THE USER CAN INSERT, DELETE AND CHANGE LINES. DC
*/
QED IS THE EDITOR OF THE SDS 940.
THE USER CAN INSERT, DELETE AND CHANGE LINES.
```

Although lines may be completely retyped to correct errors by using the CHANGE, DELETE, or INSERT commands, the EDIT and MODIFY commands give the user a faster and easier method for editing lines.

**a<sub>1</sub>,a<sub>2</sub>EDIT** This command is especially useful when the user is working with a large amount of text. Once a user is in the edit mode, several control characters (discussed below) may be used to edit the lines (i. e., to delete, insert, or change characters in it) without retyping that part of the line which is correct. The most frequently used form of the EDIT command is a<sub>1</sub>EDIT wherein the computer prints out line a<sub>1</sub> and waits for it to be edited. The edited copy then replaces the original copy in the main text buffer.

The general form of the EDIT command,  $a_1, a_2$ EDIT, permits repeated single line edits. The first line addressed ( $a_1$ ) is typed out, and when the edit of that line is terminated, the next line is typed out for editing. When the edit of the last line addressed ( $a_2$ ) is terminated, the new lines replace the original lines in the main text buffer.

**$a_1, a_2$ MODIFY** The MODIFY command works exactly like the EDIT command, but does not print out the line or range of lines to be edited. This command can be very useful to experienced users who do not need to see the line printed out for every edit.

## CONTROL CHARACTERS

In this discussion of the control characters available for editing or modifying lines, "original line" means the line to be edited. "New line" means the line as it undergoes editing or modifying. When the editing is complete, the new line replaces the original line. For the purposes of this discussion, we shall use the EDIT command only. In the first set of examples, assume the contents of the main text buffer are

```
*/
THE SDS 940 OFFERS MANY LANGUAGES.
THE TEST EDITOR OF THE 940 IS QED.
```

**$C^c$**  The Control C copies one character from the original line to the new line. If used several times in succession, the Control C copies several characters from the original line to the new line.

**$D^c$**  Control D copies the rest of the original line to the new line and ends the edit mode. The remainder of the original line is typed as it is being copied to the new line.

```
*2EDIT (RET)
THE TEST EDITOR OF THE 940 IS QED.
C^cC^cC^cC^cC^cXD^c
THE TEST EDITOR OF THE 940 IS QED.
```

Typed by user

**$Z^cx$**  Control Z causes QED to copy from the original line into the new line all characters through the character x. If the character x does not appear in the original line, QED rings the bell and takes no further action.

```
*2EDIT (RET)
THE TEST EDITOR OF THE 940 IS QED.
Z^cEZ^cEXD^c
THE TEST EDITOR OF THE 940 IS QED.
```

Typed by user

**$O^cx$**  Control O causes QED to copy from the original line into the new line all characters up to but not including the character x.

```
*2EDIT (RET)
THE TEST EDITOR OF THE 940 IS QED.
O^cSXD^c
THE TEST EDITOR OF THE 940 IS QED.
```

Typed by user

**$S^c$**  Control S is the same as  $C^c$  except that a character is deleted (i. e., skipped) rather than copied from the original line, for each  $S^c$  used. A % is printed for each character skipped.

**$F^c$**  Control F is the same as  $D^c$  except that the remainder of the old line is not typed as it is being copied to the new line.

```
*1EDIT (RET)
THE SDS 940 OFFERS MANY LANGUAGES.
O^cS S^cS^cS^cS^cF^c
THE %%%%
*1/
THE 940 OFFERS MANY LANGUAGES.
```

Typed by user

**$X^cx$**  Control X is the same as  $Z^c$  except characters are deleted from the original line up through the character x and % signs are printed for each character deleted.

```
*1EDIT (RET)
THE SDS 940 OFFERS MANY LANGUAGES.
O^cS X^c F^c
THE %%%%
*1/
THE 940 OFFERS MANY LANGUAGES.
```

Typed by user

**$P^cx$**  Control P is the same as  $O^c$  except that characters are deleted from the original line up to but not including the character x, and % signs are printed for each character deleted.

```
*1EDIT (RET)
THE SDS 940 OFFERS MANY LANGUAGES.
O^cSP^c9F^c
THE %%%%
*1/
THE 940 OFFERS MANY LANGUAGES.
```

Typed by user

**$K^c$**  The Control K is a "delete and document" command. The first Control K given prints a quote mark (") on the Teletype to inform the user that the delete and document mode has been entered. The second Control K ends the mode and prints another quote mark. Between the two Control K's, the user can type anything. The most common use is to delete characters from the original line and print them on the Teletype so the user can see what has been deleted. Used in the APPEND mode,  $K^c$  is a means by which the user can write comments on the Teletype paper without affecting the main text buffer.

```
*APPEND (RET)
CONTROL K PRINTS THAT WHICH IS BEING DELETED. D^c
*$EDIT (RET)
CONTROL K PRINTS THAT WHICH IS BEING DELETED.
Z^cKK^cZ^cGK^cD^c
CONTROL K" PRINTS THAT WHICH IS BEING" DELETED.
*$/
CONTROL K DELETED.
```

Typed by user

**$E^c$**  In many cases, it is necessary to insert characters into a line. QED provides a control character,  $E^c$ , to do

this without retyping the line. Control E is used to insert new text anywhere in the original line. At the first use of E<sup>C</sup>, a < sign is printed, at which point the user types in his insertion. He then types a second E<sup>C</sup>, which prints a > sign and informs the computer that he has completed inserting text. Any characters typed between the E<sup>C</sup> control characters are inserted in the new line without replacing any from the original line. To illustrate the operation of Control E, we can insert "SDS" into line 2.

```
*2EDIT (REF)
THE TEXT EDITOR OF THE 940 IS QED.
0C9ECSDS ECDC
THE TEXT EDITOR OF THE < SDS >940 IS QED.
*2/
THE TEXT EDITOR OF THE SDS 940 IS QED.
```

Typed by user

**H<sup>C</sup>** The Control H copies the remainder of the original line to the new line and types it on the Teletype. It does not end the edit. QED is positioned at the end of the line and editing may continue.

```
*1EDIT (REF)
THE 940 OFFERS MANY LANGUAGES.
HC QED IS ONE.
THE 940 OFFERS MANY LANGUAGES. QED IS ONE.
```

Typed by user

**U<sup>C</sup>** The Control U copies characters from the original line to the new line up to, but not including, the next tab stop. The tab stops are initialized by QED at print positions 8, 16, 32, and 40. They can be changed by use of the TABS command as discussed in Chapter 2.

```
*APPEND (REF)
1234567890123456789012345678901234567890DC
*$EDIT (REF)
1234567890123456789012345678901234567890
UCABCDEUCFGHIJUCLMNOPUCQ
1234567ABCDE345FGHIJ12345678901LMNOP7890
*$/
1234567ABCDE345FGHIJ12345678901LMNOP7890
```

Typed by user

**N<sup>C</sup>** Control N prints ↑ and "backs up" one character position on both the new and original lines. Successive use of the control N backs up a corresponding number of characters.

```
*APPEND (REF)
CONTROL N AND CONTROL A ECHO A ↑ DC
*$EDIT (REF)
CONTROL N AND CONTROL A ECHO A ↑
ZCDNCNCNC,XCDDC
CONTROL N AND ↑↑↑, %% CONTROL A ECHO A ↑
*$/
CONTROL N, CONTROL A ECHO A ↑
```

Typed by user

**R<sup>C</sup>** The R<sup>C</sup> gives a carriage return and line feed. Then it retypes the line up to the point where it was given. It types the line with all editing corrections made so that the user can see exactly what he has in the line at that point. It is useful when a lot of editing has been done on a line and the user is not sure what is in the line or what the format of the line is.

```
*APPEND (REF)
THIS LINE WILL BE EDITED. DC
*$EDIT (REF)
THIS LINE WILL BE EDITED.
OCWECOF TEXT ECRC
THIS LINE <OF TEXT >
THIS LINE OF TEXT
```

Typed by user

**T<sup>C</sup>** Control T is used to retype the remainder of the original line that is yet to be edited and the new line as far as it has been created. Control T aligns the original and new lines properly.

```
*APPEND (REF)
CONTROL T,LIKE CONTROL R,ALIGNS...DC
*$EDIT (REF)
CONTROL T,LIKE CONTROL R,ALIGNS...
ZC,ECUNECTC
CONTROL T,<UN>
LIKE CONTROL R,ALIGNS...
CONTROL T,UN
```

**Y<sup>C</sup>** The control Y copies the remainder of the original line to the new line without typing it on the Teletype (like control F), and positions QED at the beginning of the line. It then allows the edit to continue with the new line being used as the original line. This is useful when the user has performed some editing on a line and it is then realized that some additional editing is needed near the beginning of the line. A repetition of the command EDIT or MODIFY will be unnecessary.

```
*APPEND (REF)
CONTROL Y COPIES THE ORIGINAL LINE. DC
*$EDIT (REF)
CONTROL Y COPIES THE ORIGINAL LINE.
ZCHCCEC REST OF ECZCLYC
CONTROL Y COPIES THE < REST OF > ORIGINAL LINE.
ZCFEC THEECDC
CONTROL Y COPIES THE REST OF < THE > ORIGINAL.
*$/
CONTROL Y COPIES THE REST OF THE ORIGINAL LINE.
```

Typed by user

Typed by user

**M<sup>C</sup>** Control M is synonymous with the Carriage Return key. It ends the line and edit mode.

**J<sup>C</sup>** Control J is synonymous with the Line Feed key. It continues the logical line on the next printing line.

## 4. STRING BUFFERS

In addition to the main text buffer, QED has ten string buffers numbered from 0 to 9. These buffers can hold lines of text just as the main text buffer does. They are very similar to the main text buffer except that special commands are available to use the string buffers in conjunction with the main text buffer. String buffers are very useful for moving large pieces of data from one location in the main text buffer to another. In addition, the contents of a string buffer may be inserted or appended to the main text buffer. The string buffers are particularly helpful when a particular line or command is used frequently throughout a program.

**bLOAD** To enter text directly into a string buffer, the user types the number of the buffer (b), LOAD, and a Carriage Return. The computer will issue a line feed and wait for the text. The text is terminated with a Control D (D<sup>c</sup>).

```
*1LOAD (RET)
TO LOAD A BUFFER USE LOAD. Dc
```

A buffer may contain many lines of text. However, each LOAD command clears the buffer before loading.

**a<sub>1</sub>,a<sub>2</sub>;bLOAD** To load a string buffer with a line of text from the main text buffer, the user types the line address, a semicolon (;), buffer number, LOAD, and a Carriage Return. An example of this use of the LOAD command is given below. In the first part of the example, we look at the contents of the main text buffer.

```
*1,3/
QED IS A POWERFUL EDITOR.
TO CREATE NEW TEXT, USE APPEND.
NEW TEXT WILL FOLLOW ALL OLD TEXT.
*1;5LOAD (RET)
```

Upon receiving the Carriage Return, QED loads buffer 5 with the contents of line 1.

A range of lines may be specified by typing two line addresses separated by a comma. For example:

```
*2,3;8LOAD (RET)
```

The entire contents of the main text buffer can be loaded into a string buffer by omitting the line address before the semicolon. For example:

```
*;9LOAD (RET)
```

**bBUFFER** To print the contents of a string buffer, the user types the buffer number, BUFFER, and a Carriage Return. For example:

```
*5BUFFER (RET)
"QED IS A POWERFUL EDITOR.
"
```

Note that the contents of the string buffers are always printed in quotes.

```
*8BUFFER (RET)
"TO CREATE NEW TEXT, USE APPEND.
NEW TEXT WILL FOLLOW ALL OLD TEXT.
"
*9BUFFER (RET)
"QED IS A POWERFUL EDITOR.
TO CREATE NEW TEXT, USE APPEND.
NEW TEXT WILL FOLLOW ALL OLD TEXT.
"
```

**bKILL** The contents of a buffer may be erased by typing the buffer number, KILL, and a Carriage Return. For example:

```
*5KILL (RET)
*8KILL (RET)
*9KILL (RET)
```

**B<sup>c</sup>** The contents of a string buffer may be inserted or appended to text in the main text buffer by calling for it with B<sup>c</sup>. The B<sup>c</sup> prints a # to give the user an indication that B<sup>c</sup> was typed. For example:

```
*1LOAD (RET)
THERE ARE TEN STRING BUFFERS. Dc
*APPEND (RET)
Bc1 Dc
#1
*/
QED IS A POWERFUL EDITOR.
TO CREATE NEW TEXT, USE APPEND.
NEW TEXT WILL FOLLOW ALL OLD TEXT.
THERE ARE TEN STRING BUFFERS.
```

Typed by user

Similarly, text is inserted by issuing the INSERT command.

```
*2LOAD (RET)
THERE IS ONE MAIN TEXT BUFFER. Dc
*2INSERT (RET)
Bc2Dc
#2
*1,3/
QED IS A POWERFUL EDITOR.
THERE IS ONE MAIN TEXT BUFFER.
TO CREATE NEW TEXT, USE APPEND.
```

Typed by user

The B<sup>c</sup> command can also be used with CHANGE.

String buffers may contain commands, i. e., the user may want to load a string buffer with a command or sequence of commands that is used often in his program or text. Then, instead of typing the command over and over, he need only call upon the string buffer. The B<sup>c</sup> command is used to

cause the command in the buffer to execute, as shown in the following example:

```
*1LOAD (RET)
2/ DC
*BC1                               Typed by user
*#1
THERE IS ONE MAIN TEXT BUFFER.
*2LOAD (RET)
1, $D DC
*BC2 (RET) Typed by user
*#2
*/
*
-                               Main text buffer is now empty.
```

**a<sub>1</sub>, a<sub>2</sub>; bGET** The GET command is used to move blocks of text from one position to another. This command is equivalent to a<sub>1</sub>, a<sub>2</sub>; bLOAD together with a<sub>1</sub>, a<sub>2</sub>DELETE. The general form of the command deals with several lines in the main text buffer where a<sub>1</sub> is the starting line address, a<sub>2</sub> is the ending line address, and b is the number of the string buffer into which the text is loaded. The a<sub>1</sub>; bGET form of the command deals with one line (a<sub>1</sub>) in the main text buffer and ; bGET deals with the whole main text buffer. The text in the specified range is loaded into the string buffer and deleted from the main text buffer. The user may then move the text from the string buffer into any other position by using such previously discussed commands as INSERT and CHANGE.

```
*APPEND (RET)
LINE1
LINE2
LINE3
LINE4
LINE5 DC
*2, 4; 2GET (RET)
*1INSERT (RET)
BC2 DC                               Typed by user

#2
*/
LINE2
LINE3
LINE4
LINE1
LINE5
```

**SUBSTITUTE** The SUBSTITUTE command is used to replace a specific string of characters in a designated range with a new string of characters. This is particularly useful when the user has consistently misspelled a word, used an illegal variable name in his program or, in general, has repeatedly made an error that he wishes to correct. The format of the command is:

```
*a1, a2SUBSTITUTE
" TEXT TO BE INSERTED DC "FOR" TEXT TO BE REPLACED DC "
WAIT? NO } alternatives
      YES }
      ONCE }
Number of substitutions
```

The general form of the command causes the substitution to be made in the range of lines addressed by a<sub>1</sub> through a<sub>2</sub>. The form a<sub>1</sub>SUBSTITUTE causes substitution only into line a<sub>1</sub> and the form SUBSTITUTE causes substitution throughout the whole main text buffer.

If the user types N (QED types O) to the question WAIT?, QED substitutes the 'TEXT TO BE INSERTED' for every occurrence of 'TEXT TO BE REPLACED' in the designated range. If the user types Y (QED types ES) to the question WAIT?, QED types out each line that contains the text to be replaced. At this point, the user types S if the substitution is desired or any other character (excluding S) if he does not want the substitution made at that point. In either case, QED continues to search through the remainder of the designated range of lines.

If the user types O (QED types NCE) to the question WAIT?, QED does the substitution once in the first line within the designated range that contains the text to be replaced.

String buffers 0 and 1 are destroyed by the SUBSTITUTE command, since the 'TEXT TO BE INSERTED' and 'TEXT TO BE REPLACED' are actually loaded into string buffers 1 and 0, respectively, by QED. QED then references the two string buffers while performing the substitutions. QED prints the number of substitutions made before returning control to the user.

```
*APPEND (RET)
THIS IS LINE1
THIS IS LINE2
LINE3 IS THE END DC
*1, 2SUBSTITUTE (RET)
MAY BEDC IS DC                               Typed by user
" MAY BE" FOR " IS"
WAIT? NO
2
*/
THIS MAY BE LINE1
THIS MAY BE LINE2
LINE3 IS THE END
*2SUBSTITUTE (RET)
(THIS WAS AN 'E') DC E DC                               Typed by user
"(THIS WAS AN 'E')" FOR "E"
WAIT? YES
THIS MAY BE LINE2
S                               Occurrence is E in 'BE'
User typed 'S' for substitute
THIS MAY B(THIS WAS AN 'E') LINE2           E in 'LINE2'
(R)                                           User typed (RET) - no substitute
```

```
1
*/
THIS MAY BE LINE1
THIS MAY B(THIS WAS AN 'E') LINE2
LINE3 IS THE END
*SUBSTITUTE (RET)
LINKDC LINEDC                               Typed by user
"LINK" FOR "LINE"
WAIT? ONCE
1
*/
THIS MAY BE LINK1
THIS MAY B(THIS WAS AN 'E') LINE2
LINE3 IS THE END
```

**V<sup>C</sup>** To include a control character in the text in a string buffer, V<sup>C</sup> must be used. V<sup>C</sup> instructs QED to accept the next character (which will be a control character) as a literal character rather than performing the action that the control character would normally cause. The control character typed after the V<sup>C</sup> will be printed with an ampersand (&) before it. For example, to load a string buffer with D<sup>C</sup> the user types

```
*1LOAD (REF)
VCDC DC          Typed by user
&D
```

Note that the second D<sup>C</sup> is used to terminate the LOAD command.

A string buffer may contain a B<sup>C</sup>. This will cause a buffer to "call" upon another buffer.

```
*4LOAD (REF)
VCBC5 DC          Typed by user
&B5
```

Then, if B<sup>C</sup>4 were typed, buffer 4 would activate the command in buffer 5. Consider the following, more complex example.

Let us say that text was prepared off-line by placing the Teletype in the local mode and punching paper tape. This tape is then read into the QED main text buffer and corrections to the text are made on-line. Because the READ command does not recognize the editing control characters A<sup>C</sup>, W<sup>C</sup>, and Q<sup>C</sup>, they cannot be included on the paper tape or in a disc file. Therefore, we will have to provide for editing indirectly.

Let us set the standard that whenever a character is erroneously typed on the Teletype tape it will be followed by a % sign to indicate at a later time that it should be deleted. (The % may be used iteratively.) Text prepared in this manner might appear as follows as it comes into the main text buffer

```
THIS ISS% LNE%INE ONE
```

Now, to automatically delete both the % and the preceding character from the text, we will use string buffer number 3. We do this by loading buffer 3 with the commands to do the following.

1. Search for a line with a %, [%]
2. Go to the edit mode via MODIFY
3. Duplicate the line out to the %, Z<sup>C</sup>%

4. Delete the % and previous character, A<sup>C</sup>A<sup>C</sup>
5. Duplicate the rest of the line, F<sup>C</sup>
6. Call on buffer 3 again (to search for the next %), B<sup>C</sup>3
7. Terminate the buffer string with D<sup>C</sup>

This is achieved by the command

```
*3LOAD (REF)
[%]M (REF)
VCZC%VCACVCACVCFCVCBC3DC          Typed by user
&Z%&A&A&F&B3
```

Note that the V<sup>C</sup> is used so that the control commands will be placed in the string buffer and will not be executed immediately.

Having loaded the string buffer with these directions, we can respond to the QED asterisk with a B<sup>C</sup>3 to call the string buffer. This causes the commands in the buffer to be executed. The text in the main text buffer would then appear as follows.

```
THIS IS LINE ONE
```

**L<sup>C</sup>** Control L is used to load part of a line into string buffer 1 at the same time it is being input to the main text buffer. Using this facility while entering the text, the user can type L<sup>C</sup> (prints [ ) at the beginning of a phrase which will be used frequently and another L<sup>C</sup> (prints ] ) at the end of the phrase. Subsequently, instead of typing the phrase, the user need only type B<sup>C</sup>1 to insert the phrase into the text.

```
*APPEND (REF)
THIS STATEMENT IS LC FREQUENTLY USED LC DC
*$EDIT (REF)          Typed by user
THIS STATEMENT IS FREQUENTLY USED
ZCSECBC1ECDC          Typed by user
THIS <# 1 > STATEMENT IS FREQUENTLY USED
*$/
THIS FREQUENTLY USED STATEMENT IS FREQUENTLY USED
```

## STRING BUFFER DIAGNOSTICS

### FULL

The capacity of each string buffer is approximately 1400 bytes. The FULL diagnostic is typed out only when the string buffer capacity has been reached. If this diagnostic is encountered, the text can be safely kept in the string buffer by immediately typing a D<sup>C</sup>. The remainder of the text should be put into another of the available string buffers.



## 5. TEXT OUTPUT

QED is used most often to prepare and edit programs for one of the SDS 940 subsystems. Once a program is edited it must be transferred from the QED main text buffer to the appropriate subsystem for execution. This is done by first outputting the contents of the main text buffer to a disc file or to paper tape. The user then enters the subsystem via the 940 Executive and loads the program from the disc file or paper tape. The program may also be printed on the Teletype, but this method does not permit the user to load the program into the subsystem.

It should be noted that the user loses his edited copy if he fails to write it on a file or paper tape before moving on to another subsystem or logging out.

### OUTPUT TO A FILE

The following command outputs text from the main buffer to a user's disc file.

```
*a1,a2WRITE/FILENAME/␣
```

where

a<sub>1</sub> and a<sub>2</sub> are the beginning and ending addresses of the lines to be output, and FILENAME designates either an existing file or a new one. If address a<sub>2</sub> is missing, only line a<sub>1</sub> is output. If both addresses are missing, the contents of the entire main buffer are output.

To make sure that the user does not inadvertently erase a file, QED prints out either NEW FILE or OLD FILE on the next line.

```
*WRITE /FILENAME/␣  
NEW FILE␣
```

OLD FILE reminds the user that the file named in the command has been previously used and contains text which may need to be saved. NEW FILE tells the user that he is creating a new file under the name given. In either case, the user confirms this by striking a␣. If he decides that he does not want to use that particular file, he can abort the command by striking the␣ key once. He can then repeat the command using the proper file name.

The WRITE command does not erase any of the material written from the main text buffer. If the user wants to have the main text buffer emptied, he must use 1,\$DELETE as discussed in Chapter 3.

After the file has been written, QED informs the user how many words of text were written. The number of words times three gives the number of characters that were written. QED suppresses blanks if there are two or more in a row. Thus, the number of characters read into QED (from the Teletype, paper tape, or a file not written by QED) and the number of characters written by QED will seldom be the same.

The 940 Executive and other subsystems can read the files QED creates and likewise QED can read the symbolic files created by the other subsystems.

The statement of the number of words written by QED tells just the amount of text which was output and will always be less than the number given by the Executive command "FILES:" (see Terminal Users' Guide), since the FILES: command includes the index block (256 words) in addition to the text and any unused space in its data record (up to 255 words).

```
*APPEND␣  
THIS LINE WILL BE WRITTEN ONTO FILE /QED/. DC  
*WRITE /QED/␣  
NEW FILE␣  
15 WORDS.  
*␣  
*␣  
-FILES:␣
```

```
23,512 /QED/     Size of file /QED/ is 512 words
```

### OUTPUT TO TELETYPE

The slash (/) command for Teletype output was discussed earlier, to facilitate an orderly development in the manual. This command is discussed again here with the other commands available to the user for Teletype output.

```
*a1,a2/     This command prints all lines in the range a1  
through a2. If a2 is missing (i. e., a1/), only line a1 is  
printed. If a1 and a2 are missing (i. e., /), the statement  
is equivalent to 1,$/ whereby the complete main text buffer  
is printed.
```

The slash command differs from most commands in that no Carriage Return is required. Immediately after the slash is typed, QED starts printing at the beginning of the next line.

```
*a1,a2PRINT␣     The line addressing works the same  
with the PRINT as with the slash.
```

The PRINT command is used when neatly formatted output is desired. After the Carriage Return is given, QED asks the question DOUBLE?, to which the user responds with N (QED types O) or Y (QED types ES), signifying whether the printing is to be single spaced (NO) or double spaced (YES). The typing is so formatted that the paper can be torn out of the Teletype, punched with three holes, and put into a notebook.

```
*␣
```

```
* ↑     These commands print one logical line and, like  
the slash command, do not require a Carriage Return to com-  
mence. When the user types the Line Feed, QED types the  
line addressed by .+1 on the next line. If the user types ↑,  
the line .-1 is typed.
```

## OUTPUT TO PAPER TAPE

To punch a paper tape on the Teletype for rereading into an SDS 940 system at a later time, the user should use the slash command. Before typing the slash, the user must turn the Teletype paper tape punch on.

The resultant paper tape will have a / on the first line and an \* on the last line. After reading the paper tape into the main text buffer of QED with the READ TELETYPE command, the user can type

```
*1DELETE (RT)
```

```
*$DELETE (RT)
```

to delete the / and \* from the first and last line, respectively.

## APPENDIX. QED SUMMARY

### Line Addressing Methods

1. By position (line number).
2. By . (symbol for current line).
3. By \$ (symbol for last line).
4. By :label: (where label is the first word or string of characters in a line).
5. By [text] (where text is a unique word or string of characters in a line).
6. By adding or subtracting lines from any of the above addresses, e. g., [text] +2.

### COMMANDS (may be abbreviated to first letter)

TABS <sup>(RET)</sup>	Allows the user to set up to six new tab stops.
READ/filename/ <sup>(RET)</sup>	Reads text from the specified file and appends it to the contents of the main text buffer.
a <sub>1</sub> APPEND <sup>(RET)</sup>	Appends, after line a <sub>1</sub> , the text typed in after the Carriage Return. The new text entered must be terminated with a D <sup>C</sup> .
a <sub>1</sub> INSERT <sup>(RET)</sup>	Inserts, before line a <sub>1</sub> , text typed in after the Carriage Return. The new text must be terminated with a D <sup>C</sup> .
a <sub>1</sub> , a <sub>2</sub> DELETE <sup>(RET)</sup>	Deletes lines a <sub>1</sub> through a <sub>2</sub> .
a <sub>1</sub> , a <sub>2</sub> CHANGE <sup>(RET)</sup>	Replaces lines a <sub>1</sub> through a <sub>2</sub> with the text typed in after the Carriage Return. The new text must be terminated with a D <sup>C</sup> .
a <sub>1</sub> , a <sub>2</sub> EDIT <sup>(RET)</sup>	Prints line a <sub>1</sub> and allows it to be edited. Line a <sub>1</sub> + 1 is then printed and edited. This continues until line a <sub>2</sub> has been printed and edited.
a <sub>1</sub> , a <sub>2</sub> MODIFY <sup>(RET)</sup>	Does not print lines a <sub>1</sub> through a <sub>2</sub> , but waits for each to be reconstructed via editing.
a <sub>1</sub> , a <sub>2</sub> ;bLOAD <sup>(RET)</sup>	Loads buffer b with lines a <sub>1</sub> through a <sub>2</sub> of the main text buffer.
bLOAD <sup>(RET)</sup>	Loads buffer b with the text that follows the Carriage Return. The text entered must be terminated with a D <sup>C</sup> .
bBUFFER <sup>(RET)</sup>	Prints contents of buffer b.
bKILL <sup>(RET)</sup>	Erases contents of buffer b.
a <sub>1</sub> , a <sub>2</sub> ;bGET <sup>(RET)</sup>	Loads buffer b with lines a <sub>1</sub> through a <sub>2</sub> of the main text buffer and deletes those lines from the main text buffer.
a <sub>1</sub> , a <sub>2</sub> WRITE/filename/ <sup>(RET)</sup>	Writes lines a <sub>1</sub> through a <sub>2</sub> into the specified file.
a <sub>1</sub> , a <sub>2</sub> PRINT <sup>(RET)</sup>	Prints lines a <sub>1</sub> through a <sub>2</sub> of the main text buffer in a neat format.
a <sub>1</sub> , a <sub>2</sub> / (Line Feed)	Prints lines a <sub>1</sub> through a <sub>2</sub> of the main text buffer. Prints the next line.
↑	Prints the preceding line.
a <sub>1</sub> =	Prints the position address of line a <sub>1</sub> .
a <sub>1</sub> ←	Prints the label of line a <sub>1</sub> .
a <sub>1</sub> , a <sub>2</sub> SUBSTITUTE <sup>(RET)</sup>	Replaces a specific string of characters in line a <sub>1</sub> through a <sub>2</sub> of the main text buffer with a new string of characters.

### EDITING CONTROL CHARACTERS<sup>†</sup> (used at any time other than command mode)

A <sup>C</sup> (Prints ↑)	Deletes preceding character.
Q <sup>C</sup> (Prints ↔)	Deletes current line.
W <sup>C</sup> (Prints \)	Deletes preceding word.
I <sup>C</sup>	Tabs to next tab stop. In EDIT and MODIFY modes, all characters up to the next tab stop will be replaced with blanks.
M <sup>C</sup>	Synonymous with the Carriage Return.
J <sup>C</sup>	Synonymous with the Line Feed.

### EDITING CONTROL CHARACTERS (used in EDIT and MODIFY modes)

C <sup>C</sup>	Copies the next character.
D <sup>C</sup>	Finishes a line edit by copying and printing remainder of line.
Z <sup>C</sup> x	Copies all characters through x.
O <sup>C</sup> x	Copies all characters to but not including x.
S <sup>C</sup> (Prints %)	Deletes the next character.
F <sup>C</sup>	Finishes a line edit by copying but not printing remainder of line.
X <sup>C</sup> x (Prints %'s)	Deletes all characters through x.
P <sup>C</sup> x (Prints %'s)	Deletes all characters to but not including x.
K <sup>C</sup> (Prints ")	Facilitates printing of characters that are being deleted.
E <sup>C</sup> (Prints <)	Allows characters to be inserted in a line. Another E <sup>C</sup> after insertion prints >.
H <sup>C</sup>	Copies a line up to but not including the carriage return.
U <sup>C</sup>	Copies and prints all characters up to the next tab stop.
N <sup>C</sup> (Prints ↑)	Backs up one character on new and original line.
R <sup>C</sup>	Retype line being edited up to the point where R <sup>C</sup> was typed.
T <sup>C</sup>	Retype remainder of original lines from point where T <sup>C</sup> was typed on one line and line being edited on the next line. Two lines will be aligned properly.
Y <sup>C</sup>	Copies the remainder of the original line to the new line without printing it and positions QED at the beginning of the next line ready to accept editing control characters.

### EDITING CONTROL CHARACTERS (used with string buffers)

B <sup>C</sup> (Prints #)	Indicates that the next character designates the number of the buffer whose contents should be entered into the text.
L <sup>C</sup> (Prints [)	Used to load part of a line into string buffer 1 at the same time it is being input into the main text buffer. The second L <sup>C</sup> prints ].
V <sup>C</sup> (following control character will be preceded by &)	Indicates that the control character which follows is to be taken literally.

---

<sup>†</sup>Control Characters are obtained by simultaneously depressing the control key (CTRL) and the alphabetic character.



701 South Aviation Blvd./El Segundo, California 90245

**EXECUTIVE OFFICES  
AND OPERATING DIVISIONS**  
701 South Aviation Blvd.  
El Segundo, Calif. 90245  
(213) 679-4511

**EASTERN SYSTEMS DEPT.**  
12150 Parklawn Drive  
Rockville, Maryland 20852  
(301) 933-5900

**PRINTED CIRCUITS DEPT.**  
600 East Bonita Avenue  
Pomona, Calif. 91767  
(714) 628-7371

**TECHNICAL TRAINING**  
5250 West Century Blvd.  
Los Angeles, California 90045  
(213) 679-4511

**SALES OFFICES**

**Eastern**

Maryland Engineering Center  
12150 Parklawn Drive  
Rockville, Maryland 20852  
(301) 933-5900

69 Hickory Drive  
Waltham, Mass. 02154  
(617) 899-4700

Brearley Office Building  
190 Moore Street  
Hackensack, New Jersey 07601  
(201) 489-0100

1301 Avenue of the Americas  
New York City, N.Y. 10019  
(212) 765-1230

673 Panorama Trail West  
Rochester, New York 14625  
(716) 586-1500

P.O. Box 168  
1260 Virginia Drive  
Fort Washington Industrial Park  
Fort Washington, Pa. 19034  
(215) 643-2130

**Southern**

Suite 620  
State National Bank Bldg.  
200 W. Court Square  
Huntsville, Alabama 35801  
(205) 539-5131

Orlando Executive Center  
1080 Woodcock Road  
Orlando, Florida 32803  
(305) 841-6371

2964 Peachtree Road, N.W.  
Suite 350  
Atlanta, Georgia 30305  
(404) 261-5323

Suite 311-B  
First National Bank Office Bldg.  
7809 Airline Highway  
Metairie, Louisiana 70003  
(504) 721-9172

8383 Stemmons Freeway  
Suite 233  
Dallas, Texas 75247  
(214) 637-4340

3411 Richmond Avenue  
Suite 202  
Houston, Texas 77027  
(713) 621-0220

**Midwest**

2720 Des Plaines Avenue  
Des Plaines, Illinois 60018  
(312) 824-8147

17500 W. Eight Mile Road  
Southfield, Michigan 48076  
(313) 353-7360

4367 Woodson Road  
St. Louis, Missouri 63134  
(314) 423-6200

Seven Parkway Center  
Suite 238  
Pittsburgh, Pa. 15220  
(412) 921-3640

**Western**

1360 So. Anaheim Blvd.  
Anaheim, Calif. 92805  
(213) 865-5293

5250 West Century Blvd.  
Los Angeles, California 90045  
(213) 679-4511

505 W. Olive Avenue  
Suite 300  
Sunnyvale, Calif. 94086  
(408) 736-9193

World Savings Bldg.  
Suite 401  
1111 So. Colorado Blvd.  
Denver, Colo. 80222  
(303) 756-3683

Fountain Professional Bldg.  
9004 Menaul Blvd., N.E.  
Albuquerque, N.M. 87112  
(505) 298-7683

Dravo Bldg., Suite 501  
225 108th Street, N.E.  
Bellevue, Wash. 98004  
(206) 434-3991

**Canada**

864 Lady Ellen Place  
Ottawa 3, Ontario  
(613) 722-8387

**England**

London Branch  
I.L.I. House  
Olympic Way  
Wembley Park  
Middlesex

**INTERNATIONAL  
REPRESENTATIVES**

**France**

Compagnie Internationale  
pour l'Informatique, C.I.I.

**EXECUTIVE AND  
SALES OFFICES**  
66-68 Route de Versailles  
78 - Louveciennes  
951 86 00 (Paris area)

**MANUFACTURING  
AND ENGINEERING**  
Rue Jean Jaures  
Les Clayes-sous-Bois  
Seine et Oise  
950 94 00 (Paris area)

**Israel**

Elbit Computers Ltd.  
Subsidiary of Elron  
Electronic Industries Ltd.  
88 Hagiborim Street  
Haifa  
6 4613

**Japan**

F. Kanematsu & Co. Inc.  
Central P.O. Box 141  
New Kaijo Building  
Marunouchi, Chiyoda-Ku  
Tokyo