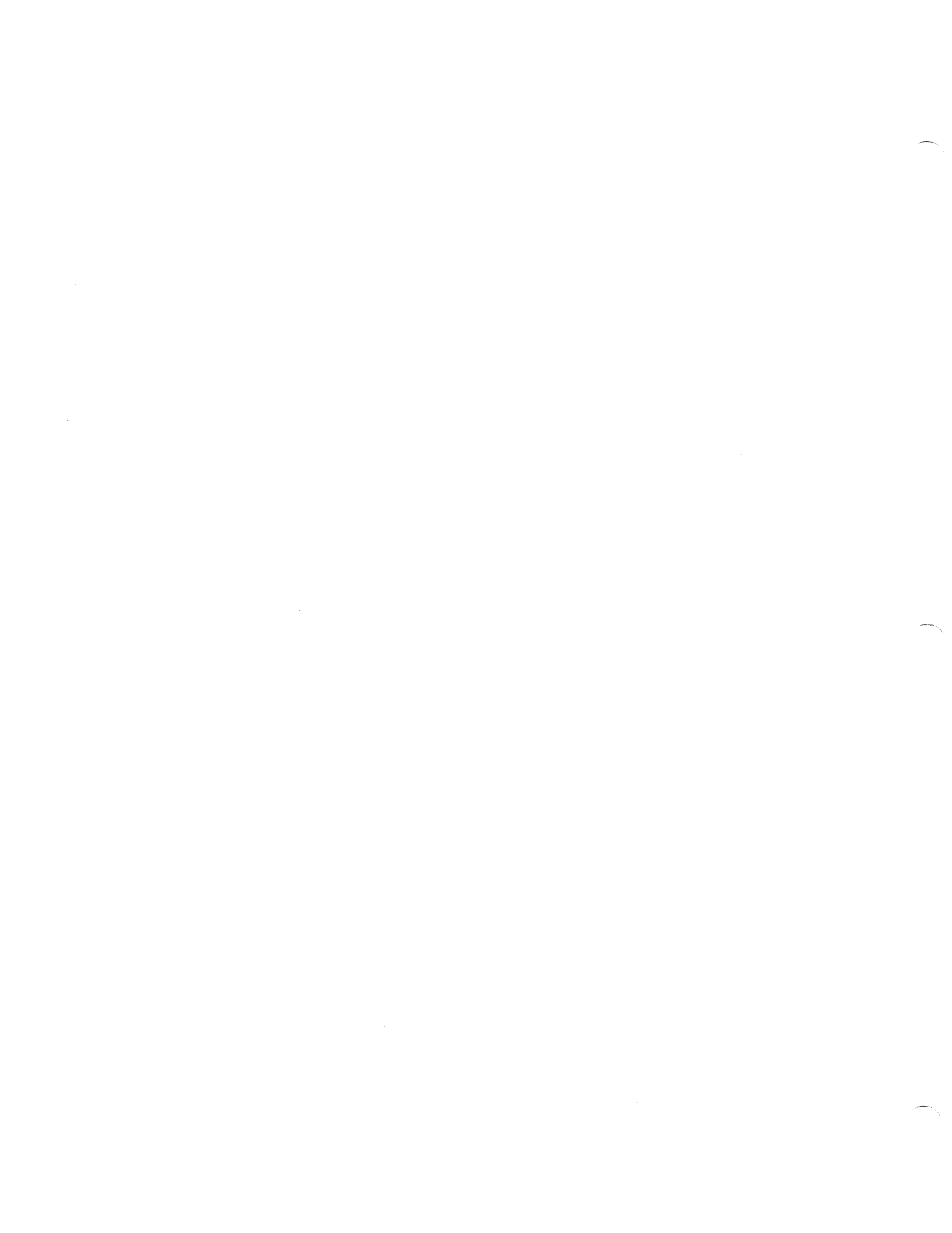


SED Text Editor  
User's Manual  
(AOS and AOS/VS)



**SED  
Text Editor  
User's Manual  
(AOS and AOS/VS)**

093-000249-02

*For the latest enhancements, cautions, documentation changes, and other information on this product, please see the Release Notice (085-series) supplied with the software.*

Ordering No. 093-000249  
©Data General Corporation, 1980, 1983, 1984  
All Rights Reserved  
Printed in the United States of America  
Revision 02, March 1984  
Licensed Material - Property of Data General Corporation

## NOTICE

DATA GENERAL CORPORATION (DGC) HAS PREPARED THIS DOCUMENT FOR USE BY DGC PERSONNEL, LICENSEES, AND CUSTOMERS. THE INFORMATION CONTAINED HEREIN IS THE PROPERTY OF DGC; AND THE CONTENTS OF THIS MANUAL SHALL NOT BE REPRODUCED IN WHOLE OR IN PART NOR USED OTHER THAN AS ALLOWED IN THE DGC LICENSE AGREEMENT.

DGC reserves the right to make changes in specifications and other information contained in this document without prior notice, and the reader should in all cases consult DGC to determine whether any such changes have been made.

THE TERMS AND CONDITIONS GOVERNING THE SALE OF DGC HARDWARE PRODUCTS AND THE LICENSING OF DGC SOFTWARE CONSIST SOLELY OF THOSE SET FORTH IN THE WRITTEN CONTRACTS BETWEEN DGC AND ITS CUSTOMERS. NO REPRESENTATION OR OTHER AFFIRMATION OF FACT CONTAINED IN THIS DOCUMENT INCLUDING BUT NOT LIMITED TO STATEMENTS REGARDING CAPACITY, RESPONSE-TIME PERFORMANCE, SUITABILITY FOR USE OR PERFORMANCE OF PRODUCTS DESCRIBED HEREIN SHALL BE DEEMED TO BE A WARRANTY BY DGC FOR ANY PURPOSE, OR GIVE RISE TO ANY LIABILITY OF DGC WHATSOEVER.

This software is made available solely pursuant to the terms of a DGC license agreement which governs its use.

**CEO, DASHER, DATAPREP, ECLIPSE, ENTERPRISE, INFOS, microNOVA, NOVA, PROXI, SUPERNOVA, PRESENT, ECLIPSE MV/4000, ECLIPSE MV/6000, ECLIPSE MV/8000, TRENDVIEW, SWAT, GENAP, and MANAP** are U.S. registered trademarks of Data General Corporation, and **AZ-TEXT, DG/L, DG/GATE, DG/XAP, ECLIPSE MV/10000, GW/4000, GDC/1000, REV-UP, XODIAC, DEFINE, SLATE, microECLIPSE, DESKTOP GENERATION, BusiPEN, BusiGEN and BusiTEXT** are U.S. trademarks of Data General Corporation.

SED  
Text Editor  
User's Manual  
(AOS and AOS/VS)  
093-000249

Revision History:

Original Release - November 1980

First Release - August 1983

Second Revision - March 1984

Effective with:

AOS Rev. 4.10

AOS/VS Rev. 1.60

### CONTENT UNCHANGED

The content in this revision is unchanged from 093-000249-00. This revision changes only printing and binding details.

# Preface

This manual shows you how to create and modify text files using the SED Text Editor. We assume that you are familiar with AOS and AOS/VS file structure and have some experience using the Command Line Interpreter (CLI).

In addition to the manual, you'll need the SED Text Editor templates. To order additional templates, use the following numbers:

DASHER® D200, D400, D410, D450, D460, G300 template (093-000256-02)  
DASHER® D1, D2 template (093-000248-02)

## Some Helpful Manuals

If you are not familiar with text editing and have little or no experience working at a console, you will find it helpful to read Chapters 1 through 3 of *Learning to Use Your AOS/VS System* (069-000031) or Chapters 1 through 3 of *Learning to Use Your Advanced Operating System (AOS)* (069-000018). If you are not familiar with the CLI, read the *Command Line Interpreter (CLI) User's Manual (AOS and AOS/VS)* (093-000122).

## How to Use This Manual

If you are new to the world of text editing, we recommend that you read each chapter in sequence and work through the editing sessions at your terminal. This will give you a good basis for creating and editing your own text files. You can then refer to Chapter 4 during future editing sessions for a summary of each SED command.

If you are a seasoned text editor user or you have had experience using SED, you may want to skip to Chapter 4, which contains a summary of all of SED's commands and arguments.

## The Organization of This Manual

Chapter 1 introduces you to the terms we use throughout this manual. It describes SED screen displays, the terminal keyboard, and how to use the function keys. It also tells you how to correct mistakes on the command line and in your text.

Chapter 2 shows you how to execute SED and to perform elementary editing tasks with SED's most frequently used commands.

Chapter 3 describes how to perform more complicated editing tasks and how to change the way SED displays your file on the screen.

Chapter 4 is the command reference chapter. It summarizes SED commands by category in a table, then lists each command alphabetically.

Appendix A describes differences in the DASHER® display terminals.

Appendix B lists SED error messages alphabetically.

Appendix C contains the ASCII character set.

## Reader, Please Note:

We use these conventions for command formats in this manual:

COMMAND required [optional] ...

Where	Means
COMMAND	You must enter the command (or its accepted abbreviation) as shown.
required	You must enter some argument (such as a filename). Sometimes, we use:

{ required<sub>1</sub> }  
{ required<sub>2</sub> }

which means you must enter *one* of the arguments. Don't enter the braces; they only set off the choice.

*[optional]* You have the option of entering this argument. Don't enter the brackets; they only set off what's optional.

... You may repeat the preceding entry or entries. The explanation will tell you exactly what you may repeat.

Additionally, we use certain symbols in special ways:

### **Symbol Means**

- }) Press the NEW LINE or carriage return (CR) key on your terminal's keyboard.
- Be sure to put a space here. (We use this only when we must; normally, you can see where to put spaces.)

All numbers are decimal unless we indicate otherwise; e.g., 35g.

Finally, in examples we use

**THIS TYPEFACE TO SHOW YOUR ENTRY;**  
*THIS TYPEFACE FOR SYSTEM QUERIES AND RESPONSES.*

) is the CLI prompt.

## **Contacting Data General**

- If you have comments on this manual, please use the prepaid Remarks Form that appears after the Index. We want to know what you like and dislike about this manual.
- If you need additional manuals, please use the enclosed TIPS order form (USA only) or contact your Data General sales representative.
- If you experience software problems, please notify Data General Systems Engineering.

End of Preface

# Contents

## Chapter 1 - Introduction

What Is SED? .....	1-1
What Does SED Require? .....	1-1
How Do I Work with SED? .....	1-1
SED Commands and Modes .....	1-1
Units of Text .....	1-1
Character .....	1-1
Character Position .....	1-1
Line .....	1-1
Current Line .....	1-2
Screen .....	1-2
Page .....	1-2
File .....	1-2
The DASHER® Terminals .....	1-2
The Main Keypad .....	1-2
The Cursor Control Keypad .....	1-4
The Numeric Keypad .....	1-4
Function Keys .....	1-4
The SED Template .....	1-4
Using the Function Keys .....	1-4
SED Screen Displays .....	1-5
Text Display .....	1-6
File Status Display .....	1-6
Function Key Display .....	1-6
CLI Display .....	1-6
HELP Display .....	1-7
SPELL Display .....	1-7
Refreshing the Screen .....	1-7
Arguments to Commands .....	1-8
Address .....	1-8
Range .....	1-9
Source .....	1-9
Destination .....	1-9
Searchstring .....	1-9
Making Mistakes .....	1-10
Typing Mistakes .....	1-10
Command Line Mistakes .....	1-10
Function Key Mistakes .....	1-10
Summary .....	1-10

## Chapter 2 - Elementary Editing With SED

Using the Examples .....	2-1
File Naming Conventions .....	2-1
Executing SED .....	2-1
Entering SED Commands .....	2-2
Case .....	2-2
Multiple Command Lines .....	2-2
Abbreviations .....	2-2
Adding to a New or Existing File (APPEND) .....	2-2
Adding Text From Your Terminal Keyboard .....	2-2
Adding Text From Another File .....	2-3
Adding a Range of Lines .....	2-3
Displaying a Range of Text (LIST) .....	2-3
Displaying Lines in the Current Page .....	2-3
Displaying Another Page .....	2-3
Displaying Lines Around the Current Line (VIEW) .....	2-4
Getting Help (HELP) .....	2-4
Changing the Current Line (The Screen Position Function Keys and POSITION) .....	2-4
The Screen Position Function Keys .....	2-4
POSITION .....	2-5
Control Characters .....	2-6
CTRL-F .....	2-6
CTRL-B .....	2-6
CTRL-A .....	2-6
CTRL-E .....	2-6
CTRL-U .....	2-6
CTRL-S .....	2-6
CTRL-Q .....	2-6
CTRL-C CTRL-A .....	2-6
Control Characters or Cursor Control Keys .....	2-6
CTRL-J or NEW LINE .....	2-7
CTRL-M or CR .....	2-7
CTRL-H or HOME .....	2-7
CTRL-X or → .....	2-7
CTRL-Y or ← .....	2-7
CTRL-I or TAB .....	2-7
Editing Control Characters .....	2-7
Changing Text (MODIFY) .....	2-7
Modifying the Current Line .....	2-7
Modifying a Range of Text .....	2-8
Inserting Lines of Text (INSERT) .....	2-8
Inserting Text from Your Terminal Keyboard .....	2-8
Inserting Text from Another File .....	2-9
Inserting a Range of Lines .....	2-9
Moving Text (MOVE) .....	2-9
Moving Text Within the Current Page .....	2-10
Moving Text Onto Another File .....	2-10
Moving Text Onto Another Page .....	2-10
Duplicating Text (DUPLICATE) .....	2-10
Duplicating Text Within the Current Page .....	2-10
Duplicating Text Onto Another File .....	2-11
Duplicating Text Onto Another Page .....	2-11
Updating Your File (SAVE) .....	2-11



Deleting and Restoring Text (DELETE and UNDO) .....	2-11
DELETE .....	2-11
Deleting the Current Line .....	2-12
Deleting a Range of Lines .....	2-12
UNDO .....	2-12
Locating Text (FIND and BACKFIND) .....	2-12
FIND .....	2-12
Locating Text Within the Current Page .....	2-13
Locating Text Across Page Boundaries .....	2-13
Locating Subsequent Occurrences of Text .....	2-13
BACKFIND .....	2-13
Substituting Text (SUBSTITUTE) .....	2-14
Substituting Text Within the Current Page .....	2-14
Substituting Text Across Page Boundaries .....	2-14
Replacing Lines of Text (REPLACE) .....	2-14
Terminating This Editing Session (BYE) .....	2-15
Saving a Backup Copy .....	2-15
Continuing with SED .....	2-16
Removing PERMANENCE .....	2-16
Congratulations! .....	2-17

### Chapter 3 - Advanced Editing with SED

SED Files .....	3-1
The Edit File .....	3-1
The .SV File .....	3-1
The .BU File .....	3-1
The .ED File .....	3-1
The .SC File .....	3-1
The .S2 File .....	3-1
Executing SED with Command Switches .....	3-2
/ED=directory .....	3-2
/NO_ED .....	3-2
/PROFILE=pathname .....	3-2
/WORK=directory .....	3-2
/NO_SCREEN .....	3-3
/NO_FORM_FEEDS .....	3-3
Editing an Existing File .....	3-3
Display Settings (DISPLAY, SET, and CLEAR) .....	3-4
DISPLAY .....	3-4
Changing Display Settings .....	3-4
SET and CLEAR .....	3-5
VIEW_RANGE .....	3-5
DISPLAY_MODE .....	3-5
TYPER_MODE .....	3-7
BLANK_MODE .....	3-8
PRELOAD_OLD_LINE .....	3-8
SKIP_WHITE_SPACE .....	3-8
UPPERCASE_SHIFT_CHARACTER x .....	3-8
ESCAPE_CHARACTER x .....	3-9
NEW_LINE_IS_VIEW .....	3-9
ESCAPE_IS_VIEW .....	3-9
LINE_NUMBER_DISPLAY .....	3-9

Defining and Clearing Function Keys (SET and CLEAR) .....	3-9
SET .....	3-10
CLEAR .....	3-11
Creating and Removing a Page Break (SPLIT and JOIN) .....	3-12
SPLIT .....	3-12
Creating a Page Containing Text .....	3-12
Displaying Page Information .....	3-12
Creating a Blank Page .....	3-12
JOIN .....	3-13
Cutting a Line and Merging Lines (CUT and PASTE) .....	3-14
CUT .....	3-14
PASTE .....	3-14
Executing CLI Commands (DO and CLI) .....	3-15
DO .....	3-15
DO FILESTATUS .....	3-15
DO TYPE .....	3-16
DO XEQ SED .....	3-16
CLI .....	3-16
Changing Your Working Directory (DIRECTORY) .....	3-17
Executing a File (EXECUTE) .....	3-17
Printing Text on @LIST (PRINT) .....	3-18
Setting the Listfile .....	3-18
Printing a Range of Lines .....	3-18
Printing a Single Page .....	3-18
Getting Spelling Assistance (SPELL) .....	3-19
Adding to SED's Dictionary .....	3-19
Terminating This Editing Session (BYE and ABANDON) .....	3-19
BYE .....	3-19
ABANDON .....	3-19
Continuing with SED .....	3-20
Once Again, Congratulations! .....	3-20

## Chapter 4 - SED Commands and Arguments

EXECUTE SED .....	4-7
ABANDON .....	4-8
address .....	4-8
APPEND .....	4-9
BACKFIND .....	4-10
BYE .....	4-10
CLEAR .....	4-11
CLI .....	4-12
CUT .....	4-12
DELETE .....	4-13
destination .....	4-13
DIRECTORY .....	4-14
DISPLAY .....	4-14
DISPLAY FUNCTION .....	4-15
DO .....	4-15
DUPLICATE .....	4-16
EXECUTE .....	4-16
FIND .....	4-17
HELP .....	4-17
INSERT .....	4-18
JOIN .....	4-19
LIST .....	4-19
MODIFY .....	4-20
MOVE .....	4-21
PASTE .....	4-21
POSITION .....	4-22
PRINT .....	4-23
range .....	4-23
REPLACE .....	4-24
SAVE .....	4-25
searchstring .....	4-25
SET .....	4-26
source .....	4-27
SPELL .....	4-27
SPLIT .....	4-28
SUBSTITUTE .....	4-28
UNDO .....	4-29
VIEW .....	4-29

## Appendix A - Terminal Differences

The DASHER D1 Terminal .....	A-1
The DASHER D400 and D450 Terminals .....	A-1
The DASHER G300 Terminal .....	A-2

## Appendix B - SED Messages

## Appendix C - ASCII Chart

# Tables

## Table

2-1	Summary of Screen Position Function Keys .....	2-5
3-1	Summary of SED Display Settings .....	3-6
3-2	DISPLAY_MODE Options .....	3-7
4-1	SED Command Summary .....	4-2
A-1	Local Function Keys on the D400 and D450 Keyboards .....	A-2

# Illustrations

## Figure

1-1	The DASHER® D2 and D200 Display Terminal Keyboards .....	1-3
1-2	Function Keys and the SED Templates .....	1-5
1-3	HELP Display of SED Commands and Keywords .....	1-8
3-1	Function Keys and the SED Templates .....	3-10
A-1	DASHER D400/D450 Terminal Keyboard and Template .....	A-1
A-2	DASHER G300 Display Terminal Keyboard and Template .....	A-2
C-1	ASCII Character Set .....	C-1

# Chapter 1

## Introduction

This chapter presents some information you need before beginning your first editing session with SED. Many of the terms defined in this chapter may be new to you. We suggest that you read it through for general content and leave the details for later. You can refer back to specific terms as you meet them during the editing sessions in Chapters 2 and 3.

### What Is SED?

Data General's SED (Screen EDitor) is a screen-oriented text editor that allows you to modify existing text files or to create new ones. The text may be as varied as reports, computer programs, business correspondence, and novels. With SED commands you can

- display existing text
- add new text
- change existing text
- delete old text
- move pieces of text
- copy pieces of text

SED allows you to choose among several ways of displaying your text. You can define function keys to execute commands that meet your editing needs. You can add text from other files to the file you are currently editing and copy text in your current file onto other files.

### What Does SED Require?

You need some understanding of AOS/VS or AOS file structure, because you need to tell SED where to find or create your text files. You also need some experience using the CLI in order to perform tasks such as renaming files and moving files to different directories. If you have not done so already, read Chapters 1 through 3 of *Learning to Use Your AOS/VS System* or Chapters 1 through 3 of *Learning to Use Your Advanced Operating System (AOS)*.

### How Do I Work with SED?

You log on to an AOS/VS or AOS system at a DASHER® display terminal. To enter SED, you give the CLI command XEQ SED. To create and modify a text file,

you use SED commands. When you have finished editing your file, you type BYE and press the NEW LINE key, which places you back in the CLI. It's as easy as that!

### SED Commands and Modes

Commands are keywords that tell SED what you want it to do. When you first enter SED, you are in *command mode*. In this mode you can give any SED command. The APPEND, MODIFY, INSERT, and REPLACE commands place you in *keyboard input mode*, where you can work directly on text. You can give many commands in keyboard input mode by using the function keys. During a SED session, you can go back and forth between command mode and keyboard input mode as often as you wish.

### Units of Text

In this manual we use the following terms to refer to different units of text.

#### Character

Characters include letters of the alphabet, the digits 0 through 9, punctuation marks, spaces, and the following special characters:

@ # \$ % ^ & \* \_ = + ' ~ [ ] \ | { } < > /

#### Character Position

Each character, except a tab, occupies a character position on your screen. A tab occupies 8 character positions.

#### Line

A line is a series of characters ending with NEW LINE. SED allows 255 characters per line. If you exceed this number, SED displays an error message.

You have the option of displaying your text with or without *line numbers*. Line numbers serve only as reference points for you; they are not actually part of the file — you cannot edit them. As you modify text in a file, SED automatically updates the line numbers to reflect any deletions or additions.

## Current Line

In command mode, the current line is the line of text that is bright. In keyboard input mode, the current line is bright and also contains the cursor. Many SED commands act on the current line and/or establish a new current line when you execute them. The SET DISPLAY\_MODE command, explained in Chapter 3, allows you to change the way SED displays the current line.

## Screen

The terminal screen holds 72 characters across if you're displaying line numbers, or 80 characters across if you're not displaying line numbers. SED displays up to 21 lines of text on your terminal screen. You can change the number of lines displayed around the current line with the SET VIEW\_RANGE command.

## Page

A page is a series of lines. The maximum number of lines on a page is 1023, and the maximum number of pages in any one file is 255. You can change page lengths by splitting and combining pages with the SPLIT and JOIN commands.

## File

A SED file contains the characters, lines, and pages of text you enter. We call this file your *edit file*. You can tell SED to create a copy of your edit file in case your system fails, and to save your original edit file as a backup. We explain these files in the sections on the SAVE and BYE commands in Chapter 2. SED automatically creates an auxiliary file and two temporary files which you will learn about in Chapter 3.

## The DASHER Terminals

The computer terminal has two parts: the screen and the keyboard. The terminal screen displays messages, the command line, and shows you what you're typing. You type text and commands on the keyboard.

Data General has several different terminals: the DASH-ER D1, D2, D200, G300, D400, and D450 display terminals. Although for the most part it does not matter which terminal you are using, we assume you are using a DASHER D2 or D200 terminal. If you are not, see Appendix A for any differences.

Figure 1-1 shows the DASH-ER D2 and D200 keyboards. Each keyboard has

- a typewriter-style main keypad
- a cursor control keypad
- a numeric keypad
- a top row of function keys.

## The Main Keypad

The main keypads of the D2 and D200 keyboards contain the standard alphabetic and numeric keys, punctuation marks, and special characters. In addition, there are several keys that perform special functions.

The NEW LINE key serves two functions. When you are in command mode, NEW LINE tells SED to execute the command you have typed on the command line. When you are in keyboard input mode, NEW LINE enters a line of text into a temporary version of your edit file and moves the cursor to the beginning of the next line.

The CR (Carriage Return) key performs the same functions as the NEW LINE key, *except* CR erases all characters from the cursor to the end of the line before it moves the cursor to the beginning of the next line.

The ESC (ESCAPE) key, marked BREAK ESC on the D200 terminal, terminates keyboard input mode, and returns you to command mode. After adding or changing text with the APPEND, INSERT, MODIFY, or REPLACE command, press the ESC key. If you press ESC after editing characters in a line, SED erases the changes you made. If you press NEW LINE and then ESC, SED stores the line with the changes you made. When adding new lines of text, press NEW LINE at the end of each line that you type, including the last line. If you press ESC after typing your last line of text, SED erases the last line. If you press NEW LINE and then ESC, SED stores your last line just as you typed it. Therefore, if you decide that you do not want to keep the changes you made in a line or the last line of text you added, simply press ESC.

The DEL (DELEte) key, on the right side of the main keypad, erases the character to the left of the cursor and closes up the line. Because SED treats a tab as a single character, pressing the DEL key erases a tab stop to the left of the cursor.

The ALPHA LOCK key is a toggle key that allows you to enter alphabetic characters in uppercase or lowercase. If the characters you are typing are all lowercase, pressing the ALPHA LOCK key causes all subsequent characters to be uppercase. If the characters you are typing are all uppercase, pressing the ALPHA LOCK key causes all subsequent characters to be lowercase. The ALPHA LOCK key does not affect the numeric or punctuation keys.

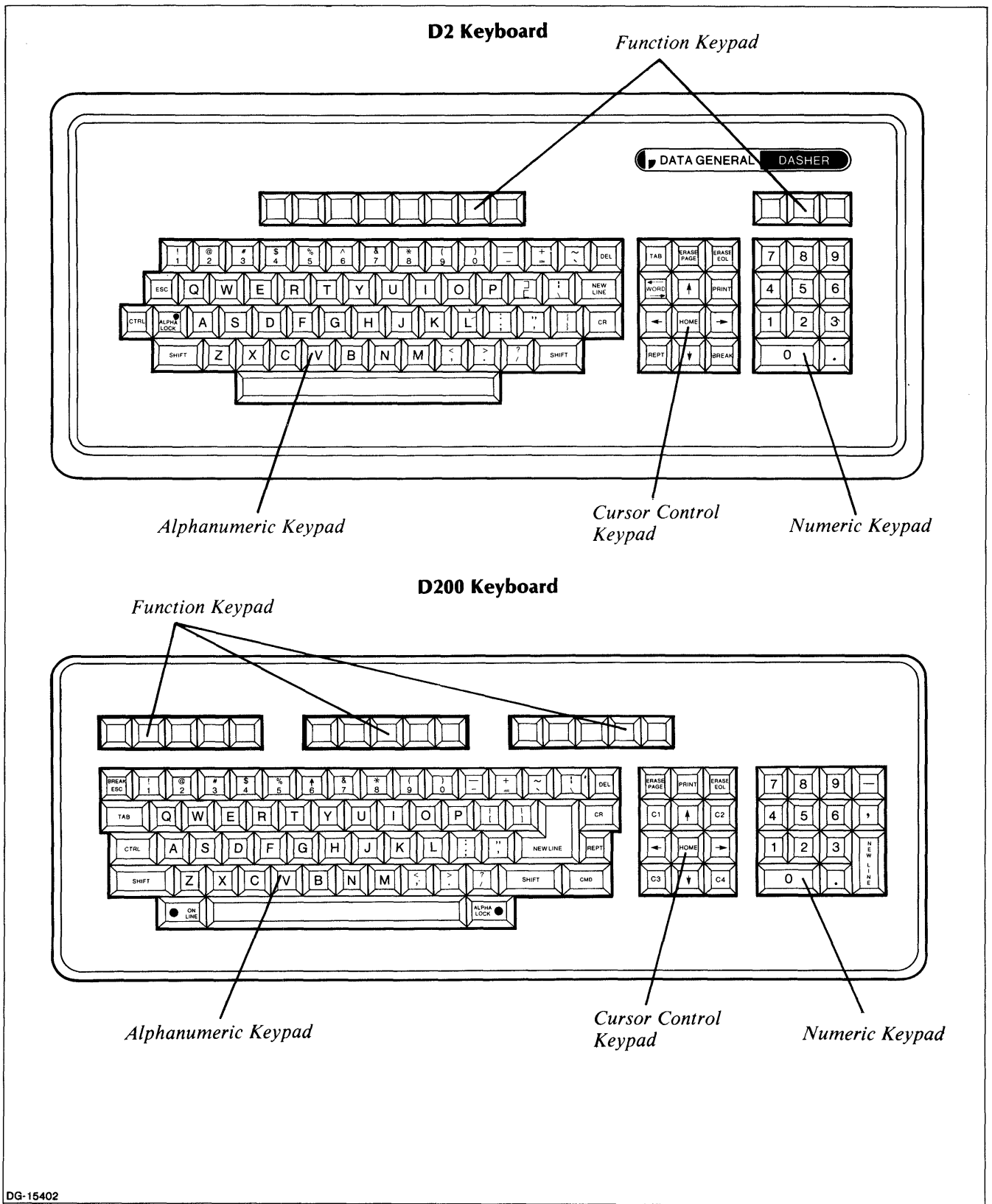


Figure -1. The DASHER D2 and D200 Display Terminal Keyboards

The TAB key moves the cursor to the next tab stop and inserts a tab character in the line. Tab stops are in columns 9, 17, 25, 33, 41, 49, 57, 65, and 73. If there are text characters between the cursor and the next tab stop, SED moves those characters to the right in addition to inserting the tab character and moving the cursor. (On the DASHER D2 terminal, the TAB key is on the cursor control keypad.)

The CTRL (ConTRoL) key is like a second SHIFT key; it allows a key to have a third value or function. To enter a control character, press and hold the CTRL key and press another key. Chapter 2 contains a description of each control character.

## The Cursor Control Keypad

The cursor control keypad, located to the right of the main keypad, contains keys that move the cursor within your text.

The ERASE EOL (ERASE End Of Line) key erases characters, beginning with the current cursor position, to the end of the line.

The uparrow (↑) moves the cursor up one line while remaining in the same column.

The downarrow (↓) moves the cursor down one line while remaining in the same column.

The leftarrow (←) moves the cursor to the left one character position.

The rightrightarrow (→) moves the cursor to the right one character position.

The HOME key moves the cursor to the first column on the line.

The REPT (REPeaT) key, used in combination with another key, inputs the other character as long as you hold down the two keys. On the D200 keyboard, the REPT key is on the right side of the main keypad.

## The Numeric Keypad

The numeric keypad is located to the right of the cursor control keypad. You have the option of entering digits with these keys or with the numeric keys on the main keypad. If you're entering just numbers, you may find it easier to use the numeric keypad.

## Function Keys

Across the top of your DASHER D2 or D200 keyboard is a row of function keys. Each function key can store one or more SED commands. Using the function keys can save you time and make your editing tasks easier in at least two ways:

- They allow you to enter frequently used SED commands with a minimum of keystrokes. A function key will act just as if you had typed its stored command on the command line and pressed NEW LINE.
- You can use the function keys in keyboard input mode as well as in command mode. This means that when you are in keyboard input mode, you can enter one SED command after another without returning to command mode each time. You must return to command mode (by pressing the ESC key) after using the INSERT and REPLACE commands and when you want to enter a command with a special argument. We explain these exceptions in Chapters 2 and 3.

Several of the function keys, called screen position keys, allow you to change the current line. You will learn more about the screen position keys in Chapter 2. Other function keys contain stored definitions of the most frequently used SED commands. You will use these keys during the editing sessions in Chapters 2 and 3.

## The SED Template

To help you distinguish the function key commands, a cutout card called a template fits over the function keys. (If you do not have the proper templates, see the Preface for ordering information.) Figure 1-2 shows the templates placed over the DASHER D2 and D200 function keys.

As the template illustrates, SED predefines many of the function keys for you (the labeled boxes). You can define the remaining function keys (the blank boxes) and change the definitions of all of the predefined keys with the SET and CLEAR commands which we explain in Chapter 3. The following seven keys are exceptions to the rules: UP ONE SCREEN, DOWN ONE SCREEN, GO TO PREVIOUS POSITION, TOP OF SCREEN, BOTTOM OF SCREEN, MIDDLE OF SCREEN, and MIDDLE LINE ON PAGE. You cannot clear them and their definitions do not appear in the function key display.

## Using the Function Keys

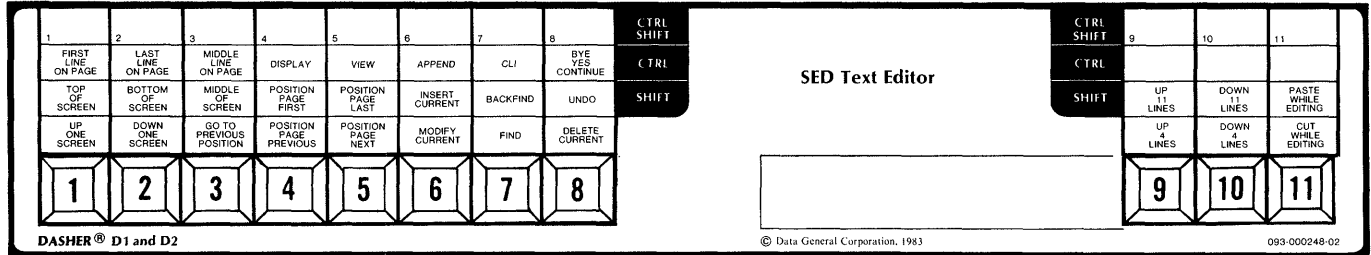
Referring to your template, you see that it is divided into four rows of boxes. Boxes marked CTRL SHIFT, CTRL, and SHIFT label the top three rows. The bottom row is unlabeled.

To enter a command in the bottom row, simply press the function key below it. For example, if you want to enter the UP ONE SCREEN command, press the first function key on the left.

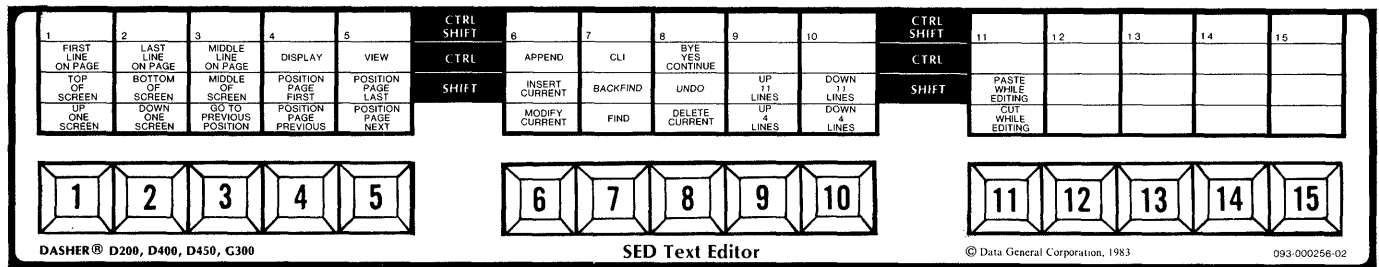
To enter a command in the next row up (the SHIFT row), press and hold one of the SHIFT keys and press the function key below the box. For example, if you want to enter the TOP OF SCREEN command, press and hold a SHIFT key and press the first function key on the left.



## D2 Terminal



## D200 Terminal



SD-02466

Figure 1-2. Function Keys and the SED Templates

To enter a command in the CTRL row, press and hold the CTRL key and press the function key below the box. For example, if you want to enter the FIRST LINE ON PAGE command, press and hold the CTRL key and press the first function key on the left.

To enter a command in the top row, press and hold both the CTRL and SHIFT keys and press the function key below the box. For example, let's assume you have defined the first function key on the CTRL SHIFT row to execute the HELP command. When you want to execute the HELP command, press and hold the CTRL and SHIFT keys at the same time and press the first function key on the left.

## SED Screen Displays

After you enter SED from the CLI, SED displays its *command line* at the top of your screen. Here is an example of a SED command line:

```
view
*_
_____
page 1 line 1
```

The command line remains at the top of your screen until you end the editing session.

A *cursor* indicates your position on the screen. The cursor is a white rectangle on the DASHER D200, D400, D450, and G300 terminals and a blinking underscore on the DASHER D1 and D2 terminals. Typing a character puts that character at the cursor location and moves the cursor one position to the right. In this manual, we use an underscore ( `_` ) to represent the cursor.

At the beginning of every editing session and while you're in command mode, the cursor is on the command line to the right of SED's prompt ( `*` ). In keyboard input mode, the cursor marks your current position on a line of text. Above its prompt, SED displays the previous command.

SED provides six types of screen displays that appear below the command line. A sample screen accompanies the explanation of each type of screen display below.

## Text Display

SED displays up to 21 lines of text at a time. When you create a new file, SED displays only the command line until you add text from the keyboard or from another file. When you open an existing file for editing, SED automatically displays a screen of text. You will learn several ways of changing text display in the editing sessions in Chapters 2 and 3.

*position -1*

\*

*page 4 line 21*

---

18 *This is the way lines of text*  
19 *appear on your screen. The line*  
20 *numbers on the left serve as*  
21 *reference points for you. The*  
22 *current line is line 21, where*  
23 *the cursor is positioned.*

## File Status Display

The file status display contains information about your current file, including:

- the complete pathname of the current file
- the current page number
- the current line number
- the last page number in the current file
- the last line number on the current page
- each display setting and its current state
- a statement when SED is displaying line numbers

You list this file status information with the DISPLAY command.

*display*

\*\_

*page 4 line 22*

---

*Edit file name - :UDD:USER:FILE*

*Current page: 4 Last page: 7*  
*Current line: 22 Last line: 23*  
*View range: 10*  
*Display mode: 0*  
*Typewriter mode: ON*  
*Blank mode: OFF*  
*Preload old line: ON*  
*Skip white space: ON*  
*Upper mode: OFF*  
*Escape mode: OFF*  
*New\_line is view: ON*  
*Escape is view: ON*

*Line numbers being displayed*

## Function Key Display

The function key display lists the current command definition of most of the function keys at the top of the DASHER D2 and D200 keyboards. SED predefines most of these keys for you. You may change the definitions of the predefined keys if you wish, and you may define the remaining function keys to suit your own editing needs. We show you how to do this in Chapter 3. You list the function key display with the DISPLAY FUNCTION command.

*display function*

\*\_

*page 4 line 22*

---

*Function keys currently defined*

4 *"position page previous"*  
5 *"position page next"*  
6 *"modify current"*  
7 *"find"*  
8 *"delete current"*  
9 *"position -4"*  
10 *"position +4"*  
11 *"cut\_line current at cursor"*  
Shift 4 *"position page first"*  
Shift 5 *"position page last"*  
Shift 6 *"insert current"*  
Shift 7 *"backfind"*  
Shift 8 *"undo"*  
Shift 9 *"position -11"*  
Shift 10 *"position +11"*  
Shift 11 *"paste current next at cursor"*  
Ctrl 1 *"position 1"*  
Ctrl 2 *"position last"*  
Ctrl 4 *"display"*  
Ctrl 5 *"view:"*  
Ctrl 6 *"append"*  
Ctrl 7 *"cli"*  
Ctrl 8 *"bye yes continue"*

Because the function key display is longer than 21 lines, the display scrolls on your screen. You can use CTRL-S to freeze the display and CTRL-Q to continue it. See Chapter 2 for information on control characters.

## CLI Display

SED's CLI and DO commands allow you to execute CLI commands during a SED session. Both of these commands create a son process and allow you to display CLI information. To use CLI and DO, your user profile must allow you to create at least two son processes. Your system manager establishes your user profile.

The CLI and DO commands differ in their purpose and in the way you execute them. You will learn about these differences in Chapter 3.

This is a sample screen of SED's CLI command:

```
cli
*_
```

---

*page 3 line 49*

```
AOS/VS CLI REV 01.50 10-JUN-82 16-45-57
Son of SED) FILESTATUS )
DIRECTORY :UDD:MYDIR
PROJECT  FORMS  PLANS2
PLANS1   SAMPLE FORMS.ED
FORMS.SC FORMS.S2
Son of SED) QPRINT FORMS )
QUEUED, SEQ= 1576, QPRI= 127
Son of SED) BYE )
```

```
AOS/VS CLI TERMINATING 10-JUN-82 16-46-46
```

This is a sample screen of SED's DO command:

```
DO FILESTATUS
*_
```

---

*page 1 line 105*

```
DIRECTORY :UDD:MYDIR
```

```
ALPHA      GAMMA      EXAMPLE3.ED
BETA       EXAMPLE3    EXAMPLE.SC
REPORT     EXAMPLE.S27
```

## HELP Display

SED provides a HELP command that displays information about all the SED commands and vocabulary. You can display a list of all commands and keywords by category, or display information on a particular command or keyword. You will learn more about the HELP command in Chapter 2.

Figure 1-3 shows the list of SED commands and keywords by category.

## SPELL Display

With SED's SPELL command you can ask for spelling assistance during an editing session. SED lists words that begin with a string of letters that you specify. Chapter 3 contains the details of the SPELL command.

## SPELL CERT

\*\_ *page 3 line 403*

---

```
certain
certainly
certainties
certainty
certes
certifiable
certifiably
certificate
certificates
certification
certificatory
certified
certifier
certifies
certify
certifying
certiorari
certitude
certitudes
```

## Refreshing the Screen

The following commands change the display on your terminal screen: DISPLAY, DISPLAY FUNCTION, CLI, DO, HELP, SPELL, LIST, and SUBSTITUTE. When you have finished reading the information on your screen, you will want to refresh the screen display. You can either redisplay a range of lines before and after the current line or execute another SED command.

To redisplay a range of lines around the current line, press one of the following keys:

- the NEW LINE key
- the ESC key
- the ERASE PAGE key
- the VIEW function key

If the display settings NEW\_LINE\_IS\_VIEW and ESCAPE\_IS\_VIEW are OFF, the NEW LINE key and the ESC key do not refresh the screen. See Chapter 3 for more information on these display settings.

You can also refresh the screen by executing another command. SED implements the command and changes the screen display to reflect the command you executed.

```

HELP
*
-----
page 2 line 377
-----
***** C O M M A N D S *****

ESCAPES  ADD TEXT  CHANGE TEXT  DELETE TEXT  LISTINGS  POSITIONING
-----  -----  -----  -----  -----  -----
EXECUTE  APPEND    MODIFY      DELETE      LIST      POSITION
HELP     INSERT    REPLACE     MOVE        VIEW      FIND
SAVE     DUPLICATE  SUBSTITUTE  JOIN        PRINT     BACKFIND
          UNDO      CUT
          PASTE

EXITING  MISC      HELP WORDS
-----  -----  -----
ABANDON  CLEAR     CURSOR_CONTROL  ADDRESS
BYE      DIRECTORY RANGE          SOURCE
CLI      DISPLAY  SEARCH_STRING  DESTINATION
DO       SET       KEYS           SYNTAX
          SPELL     SWITCHES

Type HELP followed by a command or keyword you want to know about.

```

Figure 1-3. HELP Display of SED Commands and Keywords

## Arguments to Commands

You can customize certain SED commands by using arguments. While commands tell SED what action to take, arguments tell SED what to act upon. Arguments can be page numbers, line numbers, keywords, searchstrings, or pathnames. Some SED commands require arguments, others have optional arguments, and still others accept no arguments at all. You type arguments on the command line following the command; you must separate each argument from its preceding command (or argument) with a space.

We use the following general terms to describe the different types of arguments: address, range, source, destination, and searchstring. You will see these words used in the command formats throughout this manual. In each of the examples in the following sections, the first word is a SED command and the remainder of the line is the argument.

### Address

**address** specifies a particular line or page. It consists of an optional word, **PAGE**, followed by a modifier. If you omit the word **PAGE**, SED searches for the line within the current page.

The modifier can be

- a number without a plus or minus sign, which means the actual page or line number.
- a number preceded by a plus or minus sign, which means the specified number of pages or lines after or before the current page or line.
- one of the following keywords: **CURRENT**, **PREVIOUS**, **NEXT**, or **LAST**.

If you omit the modifier, SED assumes the current line.

Examples:

**POSITION LINE 47**

or

**POSITION 47**

Set the current line to line 47 of the current page.

**POSITION PAGE 3**

Set the current line to line 1 of page 3.

**POSITION -3**

Set the current line to three lines previous.

**POSITION +5**

Set the current line to 5 lines ahead.

**POSITION PREVIOUS**

Set the current line to the previous line.

**POSITION PAGE LAST**

Set the current line to line 1 of the last page.

You cannot specify both a page and a line argument at the same time. If you do, SED displays the following error message:

*Invalid extra arguments in command; correct the command:*

## Range

**range** specifies a contiguous group of lines or pages. You can enter a range in numbers or with the keywords ALL, REMAINING, TO, or FOR. The range can be any one of the following formats:

ALL [PAGES]  
REMAINING [PAGES]  
address  
address [TO] address  
address FOR number

Examples:

LIST ALL  
Display all lines of the current page.

LIST ALL PAGES  
Display the entire contents of the file.

FIND "debit" REMAINING PAGES  
Search for the first occurrence of the word debit from the current to the last page in the file.

INSERT 35  
Insert any number of lines typed on the terminal keyboard before line 35.

LIST 47 TO 60  
or  
LIST 47 60  
List lines 47 through 60 of the current page.

DELETE CURRENT TO LAST  
or  
DELETE CURRENT LAST  
Delete lines ranging from the current line to the last line of the current page.

REPLACE 6 FOR 3  
Delete lines 6, 7, and 8, and replace them with any number of lines typed on the terminal keyboard.

## Source

**source** specifies a particular place from which to get text. If you specify a pathname, it must already exist, and you must precede it with FROM. If you omit a pathname, SED uses your keyboard as a default source for text.

Examples:

APPEND  
Append text typed on the terminal keyboard to the end of the current page.

## APPEND FROM OTHERFILE

Append the entire contents of file OTHERFILE to the end of the current page.

## INSERT FROM TABLE 1 23

Insert lines 1 through 23 of the file TABLE before the current line.

## Destination

**destination** specifies a new location for a block of text. It has two formats:

range ONTO pathname  
range { BEFORE } address  
          { AFTER }

The first format places a block of text onto the file specified by pathname. If the file exists, SED appends the text to the contents; if the file doesn't exist, SED creates it and appends the text.

The second format places a block of text before or after an address in the current page.

Examples:

DUPLICATE 11 23 ONTO TEMP  
Copy lines 11 through 23 onto file TEMP. Create TEMP first if it doesn't exist.

MOVE 36 BEFORE 104  
Move line 36 before line 104 in the current page.

## Searchstring

**searchstring** specifies a particular word or phrase in the text. You must enclose the searchstring in quotation marks or apostrophes if it contains spaces or special characters other than a dollar sign or underscore, or if you want a case-sensitive search. Unless you specify a range, SED searches for the searchstring from the current line to the end of the current page.

Examples:

FIND quantity  
Search for the first occurrence of the word "quantity" from the current line to the last line in the current page. The search is not case-sensitive, because quantity is not enclosed in quotation marks. Therefore, SED locates the first occurrence of the word whether it is quantity, QUANTITY, Quantity, or even quaNTity.

FIND "and in summary"  
Search for the first occurrence of the phrase and in "summary" from the current line to the last line in the current page.

You can enclose a searchstring either in quotation marks (" ") or apostrophes ( ' '). Use apostrophes if the

searchstring contains quotation marks, and use quotation marks if the searchstring contains apostrophes.

## Making Mistakes

When first working with SED, you will make mistakes. Don't worry about them. It's better to make mistakes during the sample sessions and learn what to expect than it is to make them later during serious editing sessions.

## Typing Mistakes

Everyone makes typing mistakes, and SED provides several ways of correcting them. You can position the cursor and type over the the incorrect characters. You can erase the preceding character and move the cursor to the left one character position by pressing the DEL key. You can erase the character to the right of the cursor by pressing the space bar. You can erase a sequence of characters by pressing the DEL and REPT keys or the space bar and the REPT key at the same time.

## Command Line Mistakes

If you make a mistake entering a command, SED displays an error message above the command line. It also redisplay the command, positioning the cursor at the first character in the command. For example, if you want to clear your text display of line numbers, you might type:

```
*CL LINE )
```

*Command not unique; correct the command:*

```
*CL LINE
```

SED displays the error message because CL is an invalid abbreviation for CLEAR. You may choose to cancel the command or modify it. To erase the entire command, press the ERASE EOL key. To correct the command, use the methods described in the previous section on typing mistakes.

Appendix B lists SED error messages alphabetically, and provides information on the cause of each message.

## Function Key Mistakes

If SED displays an error message when you press a function key, it also displays the function key command on the command line. For example, if your current line is line 1 and you press the UP 4 LINES function key, SED displays:

*Line doesn't exist on this page, correct the command:*

```
*position -4
```

To correct this mistake, you can either edit the command or erase it by pressing the ERASE EOL key. If you erase the command, you can then press another function key (POSITION PAGE PREVIOUS, perhaps) or type a command and press the NEW LINE key.

NOTE: If there is a command on the command line, you must erase the command before pressing a function key.

## Summary

In this chapter, we introduced you to SED. We defined terms you need to know and introduced you to SED modes, commands, and arguments. We also showed you what the keyboards and the different types of screen displays look like.

In the next chapter, we tell you how to execute SED, and how to move around in your file and perform simple editing tasks.

End of Chapter

# Chapter 2

## Elementary Editing With SED

This chapter contains explanations of the most frequently used SED commands, the screen position keys, and control characters. With the explanation of each SED command in this chapter, you will find an example to try at your terminal. The editing steps and commands are in sequence, as they might occur during a typical session.

### Using the Examples

We show the commands you type and the keys you press, like this:

```
DUPLICATE 1 3 ONTO NEWFILE ↵
```

Press the MODIFY CURRENT function key.

The curved arrow (↵) at the end of a line means press the NEW LINE key. A box around letters or a symbol means press that key. For example:

ESC means press the ESC key.

The information that SED displays looks like this:

```
File NEWFILE created
```

Each section that explains a SED command has a general description of the command, followed by the complete format of that command. After an explanation of the command format, there is an example or two for you to try at your terminal. The remainder of each section contains explanations and examples of other command features that you may want to use in future editing sessions.

### File Naming Conventions

Each time you execute SED, you must give the name of a file to edit. When you are naming a SED file (or renaming a file with the CLI RENAME command), there are certain CLI naming conventions you must follow

- The name can contain from 1 through 31 characters.
- The characters can be:  
Letters of the alphabet, A to Z, both upper- and

lowercase

Digits, 0 to 9

The following special characters: ? . \$ \_

### Executing SED

The first step in editing with SED is to execute SED from the CLI. Next to the CLI prompt type

```
) XEQ SED ↵
```

SED asks for a filename to edit. Type the name of the file you want to create or edit. In this sample session, let's use the filename SAMPLE. (If you already have a file named SAMPLE, type a different filename.)

```
Name of file to edit: SAMPLE ↵
```

OR you can type the filename on the same line with the XEQ SED command:

```
) XEQ SED SAMPLE ↵
```

SED then displays its revision number and the filename of your input file. It will look similar to this:

```
SED Rev 4.10; Input file - SAMPLE
```

Because SAMPLE is the name of a new file, SED asks if you want to create the file. Whenever SED asks a question, type Y↵ for YES, N↵ for NO or simply ↵ for NO. In this case, type Y↵ because you want SED to create SAMPLE.

```
Do you want SAMPLE to be created? Y ↵
```

When SED is ready to begin the editing session, it displays its prompt, an asterisk (\*), on the command line at the top of your screen.

```
view
```

```
*_
```

```
page 1 line 1
```

The cursor is in the character position to the right of the SED prompt. At the other end of the line, SED displays the current page number and the current line number. SED displays the previous command above the prompt. When you begin a SED session, the previous command is always VIEW.

You are now in command mode. Whenever the cursor is on the command line, you are in command mode. In this

mode, you can enter any SED command, either by pressing a function key or by typing the command on the command line.

## Entering SED Commands

To enter many SED commands, you can either press a function key or type the command on the command line and press the NEW LINE key. After you type a command on the command line, you must press the NEW LINE key to execute the command. We show the NEW LINE symbol (␣) in the examples, but we do not continue telling you to press NEW LINE. In the examples in this manual, we use the function keys whenever possible because they make editing faster and easier.

Some commands, such as MOVE and DUPLICATE, require arguments, while other commands have optional arguments, and still others accept no argument at all. When you want to execute a command with an argument, you must type the command and the argument on the command line and press NEW LINE. If you want to enter the same command and argument several times during a particular session, you can define a function key to execute that command and argument. Chapter 3 contains information on defining function keys.

### Case

SED accepts commands and argument keywords typed in either upper- or lowercase. For example, SED accepts MOVE, Move, and move. For clarity, we show all commands and keywords in uppercase. SED stores your *text* the way you enter it.

### Multiple Command Lines

You can type one or more commands on the command line before pressing NEW LINE. If you type more than one command, separate the commands with semicolons. For example:

```
*FIND "COMPUTE SUB-TOTAL"; MODIFY ␣
```

### Abbreviations

You can abbreviate most SED commands and all argument keywords. SED accepts the fewest characters that uniquely identify the command or argument keyword. For example, J is a valid abbreviation for the JOIN command since it is the only command that begins with J. CL is not a valid abbreviation because both the CLEAR and CLI commands begin with CL. For clarity, we do not abbreviate commands and keywords in this manual.

The best way to learn about entering SED commands is to try it. We will begin by adding text to the file SAMPLE.

## Adding to a New or Existing File (APPEND)

The APPEND command allows you to enter text into a new file or to add text to the end of the current page in an existing file. You can either type the text in at your terminal keyboard, or tell SED to append a copy of a complete file or a range of lines from a single-page file.

```
APPEND [FROM pathname] [range]
```

Typing APPEND without an argument, or pressing the APPEND function key allows you to add text from your terminal keyboard. Typing APPEND with the FROM pathname argument or FROM pathname and a range instructs SED to add text from another file.

### Adding Text From Your Terminal Keyboard

To enter text from your terminal keyboard, press the APPEND function key.

SED displays a line number (1) at the left margin of the line below the command line. The cursor moves to the first character position on line number 1.

You are now in keyboard input mode. Whenever the cursor is on a line below the command line, you are in keyboard input mode. In this mode, you can type lines of text and modify them.

At the end of each line of text, press the NEW LINE key. This instructs SED to store the line exactly as you typed it.

Type the following lines exactly as they are. We will fix the mistakes later, using other SED commands and control characters.

```
Heer is the first line of text. ␣  
This is thr second line. ␣  
This lne has three mistake ␣  
Here is the sixth line. ␣  
We will move this fifth line. ␣
```

To return to command mode, press the ESC key. Remember to press NEW LINE before you press ESC, or you will lose your last line of text.



After you press ESC to return to command mode, SED displays APPEND (the previous command) above the prompt. The cursor moves to the command line next to the prompt. The current line is the line that is bright (line 5). Your screen display now looks like this:



append

\*\_

page 1 line 5

```
1 Heer is the first line of text. }
2 This is thr second line. }
3 This lne has three mistake }
4 Here is the sixth line. }
5 We will move this fifth line. }
```

You have just appended five lines of text to the temporary version of SAMPLE and returned to command mode. You do not always have to return to command mode after appending text. We do it here because you must type the next command we're going to use (LIST) on the command line.

During future editing sessions, you will want to add text to the end of the current page. (The current page is the page that contains the current line. If you have not used the SPLIT command to create a page break, your file has only one page.) No matter where the cursor and current line are, you press the APPEND function key to add text to the end of the page. The cursor then moves to the first character position on the line following the last line of text. If you are displaying line numbers, SED displays the next sequential number at the left margin.

### Adding Text From Another File

When you want to add the contents of another file to the end of the page you are currently editing, type APPEND FROM and a pathname argument on the command line. For example:

```
*APPEND FROM OTHERFILE }
```

SED finds the source file specified by pathname, in this case OTHERFILE, and appends its contents to the end of the current page. This command does not change the contents of the named source file. SED sets the current line to the last line of text appended to the current page.

### Adding a Range of Lines

You can append part of a single-page file by typing APPEND with a pathname argument and a range of lines. The range is the first and last line numbers of the text you want to add. For example:

```
*APPEND FROM SOURCE 23 45 }
```

This command adds lines 23 through 45 of a file named SOURCE to the end of the page you are currently editing.

Remember that SED has a maximum capacity of 1023 lines per page. If the text you append from another file causes the current page to exceed this maximum, SED creates a page break in your file. SED appends the text to the end of the current page until it reaches 1023 lines. It then creates a page break and appends the remaining text to the new page. SED renumbers the pages in your file to reflect the new page it created.

## Displaying a Range of Text (LIST)

The LIST command displays a range of text on your terminal screen without changing the current line.

```
LIST [range]
```

range can be

- line numbers within the current page
- a single page argument
- the keywords CURRENT, PREVIOUS, NEXT, LAST, ALL, REMAINING, FOR

If you do not specify a range, SED displays the current line. If the range is greater than the maximum that your display allows, the text will scroll on your screen. You can freeze the display by pressing CTRL-S. When you have finished reading the screen, and you want to continue the display, press CTRL-Q. See the section on Control Characters in this chapter for more information on CTRL-S and CTRL-Q.

### Displaying Lines in the Current Page

To display lines 1 through 3 of the text you just appended, type

```
*LIST 1 3 }
```

```
LIST 1 3
```

\*\_

page 1 line 5

```
1 Heer is the first line of text.
2 This is thr second line.
3 This lne has three mistake
```

To display all of the text on the current page, type

```
*LIST ALL }
```

```
LIST ALL
```

\*\_

page 1 line 5

```
1 Heer is the first line of text.
2 This is thr second line.
3 This lne has three mistake
4 Here is the sixth line.
5 We will move this fifth line.
```

### Displaying Another Page

When you have a file that contains more than one page, the range can be a different page. For example:

```
*LIST PAGE 7 }
```

1 This is line 1 of page 7.

.

.

.

18 This is line 18 of page 7.

NOTE: The LIST command *does not* change the current line. To redisplay a range of lines around the current line, press one of the following keys: NEW LINE, ESC, ERASE PAGE, or the VIEW function key. Refer to Chapter 1 for more information on refreshing the screen. We explain the VIEW command in the next section.

## Displaying Lines Around the Current Line (VIEW)

The VIEW command allows you to display a range of lines before and after the current line. You can press the VIEW function key or type the VIEW command on the command line.

### VIEW

When you execute the VIEW command, SED displays the maximum of 21 lines of text on your screen, 10 lines before and 10 lines after the current line. If there aren't a total of 10 lines on either side of the current line in your file, SED displays as many lines as there are.

The following commands clear your screen of text to display other information or text: DISPLAY, DISPLAY FUNCTION, CLI, DO, HELP, LIST, and SUBSTITUTE. When you have finished reading the information these commands display and you want to continue editing, you can refresh the screen with the VIEW command.

As you learned in Chapter 1, you can also use the NEW LINE key and the ESC key to refresh the screen when the cursor is on an empty command line. But if you have cleared the NEW\_LINE\_IS\_VIEW and the ESCAPE\_IS\_VIEW display settings, NEW LINE and ESC will not refresh your screen. (See Chapter 3 for more information on these display settings.) In this case, the only way to display a range of lines around the current line is with the VIEW function key or command.

You can reduce the number of lines SED displays before and after the current line with the SET VIEW\_RANGE command. (Chapter 3 contains information on the SET VIEW\_RANGE command.) If you do so, the VIEW command displays the reduced number of lines on either side of the current line. Viewing less than the maximum number of lines saves time, especially on a slow system or a low baud rate terminal.

## Getting Help (HELP)

SED has a HELP command to give you information about all SED commands, arguments, and other subjects such as function keys. If you are in doubt about one of these, you can ask SED to display an explanation of the topic on your screen.

```
HELP [command]
      [keyword]
```

To display a list of all SED commands and keywords by category, type HELP without an argument:

```
HELP )
```

To display information about a specific command or keyword, type HELP and the command or keyword. For example:

```
HELP SYNTAX )
```

SED displays a screen of information about abbreviations, SED modes, and multiple command lines.

To refresh the screen press NEW LINE, ESC, or the VIEW function key.

Some of the HELP information is longer than your screen display allows, and the HELP text will scroll on your screen. As with the LIST command, you can freeze the display by pressing CTRL-S. To continue the display, press CTRL-Q.

## Changing the Current Line (The Screen Position Function Keys and POSITION)

### The Screen Position Function Keys

Several of the predefined function keys, called screen position function keys, allow you to change the current line quickly and easily. You can use the screen position function keys in both command mode and keyboard input mode. Table 2-1 lists each screen position key and the action it performs.

Let's change your current line to line 1. Press the UP 4 LINES function key.

```
position -4
```

\*\_

page 1 line 1

- 1 Heer is the first line of text.
- 2 This is thr second line.
- 3 This lne has three mistake
- 4 Here is the sixth line.
- 5 We will move this fifth line.

**Table 2-1. Summary of Screen Position Function Keys**

Screen Position Key	Action
GO TO PREVIOUS POSITION	Sets the current line to the previous current line. This key stores the previous 15 line positions.
UP 4 LINES UP 11 LINES DOWN 4 LINES DOWN 11 LINES	Sets the current line to the fourth line back. Sets the current line to the eleventh line back. Sets the current line to the fourth line ahead. Sets the current line to the eleventh line ahead.
TOP OF SCREEN MIDDLE OF SCREEN BOTTOM OF SCREEN UP ONE SCREEN DOWN ONE SCREEN	Sets the current line to the first line on the screen. Sets the current line to the middle line on the screen. Sets the current line to the last line on the screen. Displays the previous screen of text and sets the current line to the middle line on the screen. Displays the next screen of text and sets the current line to the middle line on the screen.
FIRST LINE ON PAGE MIDDLE LINE ON PAGE LAST LINE ON PAGE	Sets the current line to the first line on the page. Sets the current line to the middle line on the page. Sets the current line to the last line on the page.
POSITION PAGE PREVIOUS POSITION PAGE FIRST POSITION PAGE NEXT POSITION PAGE LAST	Sets the current line to the first line on the previous page. Sets the current line to the first line on the first page. Sets the current line to the first line on the next page. Sets the current line to the first line on the last page.

In addition to the screen position keys, you can change the current line in both modes with the uparrow and downarrow keys on the cursor control keypad. The NEW LINE key changes the current line in keyboard input mode only.

## POSITION

You can also change the current line by typing the POSITION command and an address on the command line.

### POSITION address

address can be

- a line number within the current page
- a single page argument
- the keywords PREVIOUS, NEXT, LAST

If you specify a page address, SED sets the current line to the first line on that page.

Examples:

**\*POSITION 182 )**

SED sets the current line to line 182 in the current page.

**\*POSITION PAGE 5 )**

SED sets the current line to line 1 in page 5.

**\*POSITION LAST )**

SED sets the current line to the last line in the current page.

## Control Characters

Many of the control characters help you modify the current line and the command line; you can use control characters in command mode and in keyboard input mode. They allow you to move the cursor back and forth on a line, insert characters, and erase characters. To enter a control character, you press and hold the CTRL key and then press another key. In this manual, we show control characters as CTRL-character (e.g., CTRL-F).

### CTRL-F

CTRL-F moves the cursor forward to the first character of the next word. To find the beginning of the next word, the cursor moves right to the next nonblank character after any number of tabs and/or spaces.

### CTRL-B

CTRL-B moves the cursor backward to the last character of the previous word. To find the end of the previous word, the cursor moves left to the next nonblank character after any number of tabs and/or spaces.

### CTRL-A

In command mode, CTRL-A moves the cursor to the character position following the last character you have typed on the command line. If the command line is empty, CTRL-A redisplay the previous command.

In keyboard input mode, CTRL-A can perform one of two actions, depending on the command you used to enter keyboard input mode. If you entered keyboard input mode with the MODIFY command, CTRL-A moves the cursor to the character position following the last character on the line. If you entered keyboard input mode with the APPEND, INSERT, or REPLACE command, SED saves a copy of the previous current line. If the current line is blank, CTRL-A copies the entire previous current line onto the current line. If the current line contains text

but is shorter than the previous current line, CTRL-A appends what will fit of the end of the previous line to the current line. The cursor moves to the character position following the copied text. For example, let's say you enter keyboard input mode by pressing the APPEND function key, and then you type

```
1 This report contains )
```

```
2 _
```

Then, when you press CTRL-A, SED copies the previous line onto the current line. Your text display looks like this:

```
1 This report contains
```

```
2 This report contains_
```

NOTE: Changing the default setting of the PRELOAD\_OLD\_LINE display setting alters the performance of CTRL-A in keyboard input mode. See Chapter 3 for more information on this display setting.

### CTRL-E

CTRL-E allows you to open a space before the cursor where you can insert one or more characters. Pressing CTRL-E again closes up the line.

### CTRL-U

CTRL-U deletes all characters on the line.

### CTRL-S

CTRL-S freezes the display of information on the screen. This is useful when you want to read information that is scrolling on your screen.

### CTRL-Q

CTRL-Q cancels CTRL-S and continues displaying information on the screen.

### CTRL-C CTRL-A

CTRL-C CTRL-A interrupts the command that SED is currently executing and displays *\*INTERRUPTED\** on your terminal screen.

## Control Characters or Cursor Control Keys

You have the option of using the following control characters or the keys on the main keypad and the cursor control keypad that perform the same functions.

## CTRL-J or NEW LINE

CTRL-J performs the same function as the NEW LINE key. In command mode, it tells SED to execute the command you have typed on the command line. In keyboard input mode, CTRL-J enters a line of text into a temporary version of your file and moves the cursor to the beginning of the next line.

## CTRL-M or CR

CTRL-M performs the same function as the CR key. It erases all characters from the cursor to the end of the line and moves the cursor to the leftmost position of the next line.

## CTRL-H or HOME

CTRL-H performs the same function as the HOME key. It moves the cursor to the first character position on the line.

## CTRL-X or →

CTRL-X performs the same function as the rightrightarrow key. It moves the cursor to the right one character.

## CTRL-Y or ←

CTRL-Y performs the same function as the leftarrow key. It moves the cursor to the left one character.

## CTRL-I or TAB

CTRL-I performs the same function as the TAB key. It moves the cursor to the next tab stop and inserts a tab character in the line. Tab stops are in columns 9, 17, 25, 33, 41, 49, 57, 65, and 73. If there are text characters between the cursor and the next tab stop, SED moves those characters to the right in addition to inserting the tab character and moving the cursor.

## Editing Control Characters

If you type a control character that appears in your text, you can edit it as you would any other character. For example, if you press CTRL-R, it appears as ^R in your text. To erase it, you can use the DEL key if the cursor is to the right of it, or type over it or press the space bar if the cursor is to the left of it.

## Changing Text (MODIFY)

The MODIFY command allows you to change lines of text within the current page. The MODIFY command places you in keyboard input mode and positions the cursor at the first character on the current line or on the first line you specify by *range*. To move back and forth on the line, use control characters and the right- and leftarrow keys.

MODIFY [*range*]

After you modify each line, press the NEW LINE key, the up- or downarrow keys, or any of the screen position keys to enter the changes. SED does not store the changes in a line until you leave the line.

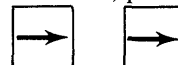
To return to command mode, press the ESC key. When you press NEW LINE after modifying the last line in the current page, SED automatically returns you to command mode. If you press ESC before leaving a line on which you have made changes, SED stores the line with no changes.

## Modifying the Current Line

When you press the MODIFY CURRENT function key or type the MODIFY command without an argument, SED moves the cursor to the beginning of the current line.

To move the cursor to the beginning of the current line (line 1) in your text, press the MODIFY CURRENT function key.


You can correct some errors by simply typing over the incorrect characters. To position the cursor at the second *e* in *Heer*, press



Type re )

The current line is now line 2. (When you enter keyboard input mode with the MODIFY command, SED does not update the page and line numbers displayed on the command line until you press a function key or the ESC key.)

In many cases, you will find more than one way to position the cursor where you want it on the current line. For example, to position the cursor at the *r* in *thr*, you can press


CTRL-F CTRL-F  

OR you can press

CTRL-F CTRL-F CTRL-F CTRL-B

Now that you have the cursor positioned, type `e` ↵

The current line is now line 3, which contains three mistakes. To position the cursor at the *n* in *line* and open a space for insertion, press

CTRL-F  CTRL-E

To insert the missing letter, type `i`

To close up the line, press CTRL-E again.

The other two mistakes in line 3 are at the end of the line. To position the cursor there, press CTRL-A.

Type `s`. ↵

Your screen display now looks like this:

*position -4*

\*

*page 1 line 1*

---

```
1 Here is the first line of text.
2 This is the second line.
3 This line has three mistakes.
4 Here is the sixth line.
5 We will move this fifth line.
```

The current line is line 4 which, as you've probably noticed, says it's the sixth line. We will fix this error by inserting a line with the INSERT command and by moving line 5 with the MOVE command.

## Modifying a Range of Text

To modify a range of text, type the MODIFY command and a range argument on the command line and press NEW LINE. The range can be:

- line numbers within the current page
- a single page argument
- the keywords CURRENT, PREVIOUS, NEXT, LAST, ALL, REMAINING, FOR

SED positions the cursor at the first character on the first line of the range. You can then make changes to each line in the range, pressing NEW LINE to enter the changes. When you press NEW LINE after changing the last line in the range, SED automatically returns you to command mode. If you press the uparrow key to position to the line before the first line in the range, SED automatically returns you to command mode.

Examples:

\*MODIFY 5 12 ↵

SED moves the cursor to the beginning of line 5. You can modify each line through line 12. Press the NEW LINE key to enter the changes on each line. When you press NEW LINE after modifying line 12, SED returns you to command mode.

\*MODIFY PAGE 3 ↵

If the range you specify is a page, SED positions the cursor at the first character on the first line of the page. You can modify each line in the page, if you wish. When you press NEW LINE after modifying the last line in the page, SED returns you to command mode.

\*MODIFY REMAINING ↵

SED moves the cursor to the beginning of the line following the current line. When you press NEW LINE after modifying the last line in the page, SED returns you to command mode.

## Inserting Lines of Text (INSERT)

The INSERT command allows you to insert lines of text *before* a specified line in the current page. You can either type the text in at your terminal keyboard or tell SED to insert a copy of a complete file or part of a single-page file.

INSERT [*address*] [*FROM pathname*] [*range*]

*address* can be:

- a line number within the current page
- a single page argument
- the keywords PREVIOUS, NEXT, LAST

If you specify a page address, SED inserts the text before the first line on the page. When you press the INSERT CURRENT function key or type the INSERT command without an address, SED inserts the text before the current line.

You tell SED you've finished inserting lines of text from your terminal keyboard by pressing the ESC key. At the end of any insertion, SED automatically renumbers the lines and sets the current line to the line following the inserted text.

## Inserting Text from Your Terminal Keyboard

To insert the missing fourth line of text, press the INSERT CURRENT function key.

SED moves the current line and all following lines down on your screen. This opens a *window* in which you can type the missing line of text. SED replaces the line

numbers below the window with asterisks and displays the next sequential line number at the top of the window. (If you are not displaying line numbers, SED does not display the asterisks or the next line number.) The cursor moves to the first character position on the line, where you can begin typing the text you want to insert.

Your screen display now looks like this:

```

modify current
*
_____ page 1 line 4
1 Here is the first line of text.
2 This is the second line.
3 This line has three mistakes.
4 —
* Here is the sixth line.
* We will move this fifth line.

```

Type Now you have a fourth line. )

To close up the insert window and return to command mode, press the ESC key. Remember to press NEW LINE before you press ESC or you will lose the line you just inserted.



You have just inserted a line of text and returned to command mode. Your screen display now looks like this:

```

insert current
*—
_____ page 1 line 5
1 Here is the first line of text.
2 This is the second line.
3 This line has three mistakes.
4 Now you have a fourth line.
5 Here is the sixth line.
6 We will move this fifth line.

```

## Inserting Text from Another File

During future editing sessions, you may want to insert a copy of another file. To do this you must type the INSERT command on the command line, using the following format:

```
INSERT [address] FROM pathname
```

Examples:

```
*INSERT 16 FROM SOURCEFILE )
```

This command instructs SED to insert the entire contents of SOURCEFILE before line 16 on the current page. The INSERT command does not change the contents of the source file.

```
*INSERT PAGE 3 FROM SOURCEFILE )
```

This command inserts the entire contents of SOURCEFILE before the first line on page 3.

## Inserting a Range of Lines

You can insert a range of lines from a one-page file. To do so, type the INSERT command on the command line using the following format:

```
INSERT [address] FROM pathname range
```

The range is the first and last line numbers of the text you want to insert. For example:

```
*INSERT 21 FROM SOURCEFILE 6 12 )
```

This command instructs SED to insert lines 6 through 12 of SOURCEFILE before line 21 on the current page.

Since SED has a maximum capacity of 1023 lines per page, SED creates a page break in your file if the text you inserted would cause the current page to exceed 1023 lines. SED creates the page break before the inserted text and renumbers the pages in your file.

## Moving Text (MOVE)

The MOVE command allows you to move lines of text from one location to another. SED deletes the lines you specify from their original location and inserts them at the destination. The destination can be within the current page or another file. Because the MOVE command requires an argument, you must type it on the command line.

```
MOVE [range] [BEFORE address
AFTER address
ONTO pathname]
```

range can be:

- line numbers within the current page
- a single-page argument (only when you are moving text onto another file)
- the keywords CURRENT, PREVIOUS, NEXT, LAST, ALL, REMAINING, FOR

If you omit the range, SED moves the current line.

When the move is complete, SED automatically renumbers the lines and sets the current line to the line following the moved text. The cursor is at the command line.

## Moving Text Within the Current Page

To move lines within the current page, type the **MOVE** command with a **BEFORE** or **AFTER** argument, using the following format:

```
MOVE range [BEFORE | AFTER] address
```

SED then copies the range of text and places it before or after the line specified by **address**. SED then deletes the range of text from its original location. The **address** can be

- a line number within the current page
- the keywords **CURRENT**, **PREVIOUS**, **NEXT**, **LAST**

To move line 6 before line 5 of your text, type

```
*MOVE 6 BEFORE 5 ↓
```

```
MOVE 6 BEFORE 5
```

```
*_ page 1 line 6
```

```
1 Here is the first line of text.
2 This is the second line.
3 This line has three mistakes.
4 Now you have a fourth line.
5 We will move this fifth line.
6 Here is the sixth line.
```

## Moving Text Onto Another File

When you want to move lines of text onto the end of another file, use this format:

```
MOVE range ONTO pathname
```

SED copies the range of text and places it at the end of the last page in the file specified by **pathname**. SED then deletes the range of text from its original location. You cannot specify an address within **pathname**.

If the file specified by **pathname** does not already exist, SED creates it in your working directory and then moves the range of text onto it. SED displays a message to let you know that it has created the file. For example:

```
*MOVE 1 32 ONTO EXAMPLE.2 ↓
```

```
File EXAMPLE.2 created
```

```
*
```

SED first creates the file **EXAMPLE.2**, then copies lines 1 through 32 onto it, and finally deletes lines 1 through 32 from their original location.

## Moving Text Onto Another Page

You cannot move text to another page in the current file directly. To do so indirectly:

- Move the range of text onto a new file.
- Position to the page where you want to move the text.
- Insert the new file where you want it on that page.

For example, let's assume you want to move lines 17 through 45 on page 2 before line 6 on page 4. Your current page is page 2. First, you would type

```
*MOVE 17 45 ONTO EXAMPLE.3 ↓
```

```
File EXAMPLE.3 created
```

```
*
```

Then position to page 4 by pressing the **POSITION PAGE NEXT** function key twice or by typing

```
*POSITION PAGE 4 ↓
```

Finally, insert the text from **EXAMPLE.3** by typing

```
*INSERT 6 FROM EXAMPLE.3 ↓
```

Later, you can delete **EXAMPLE.3** with the CLI **DELETE** command.

## Duplicating Text (DUPLICATE)

The **DUPLICATE** command allows you to copy lines of text from one location to another. Unlike the **MOVE** command, the **DUPLICATE** command *does not* delete the lines you specify from their original location. The destination can be within the current page or the end of another file. Because the **DUPLICATE** command requires an argument, you must type it on the command line.

```
DUPLICATE [range] [BEFORE address | AFTER address | ONTO pathname]
```

*range* can be

- line numbers within the current page
- a single page argument (only when you are moving text onto another file)
- the keywords **CURRENT**, **PREVIOUS**, **NEXT**, **LAST**, **ALL**, **REMAINING**, **FOR**

If you omit the range, SED duplicates the current line.

At the end of any duplication, SED automatically updates the line numbers and sets the current line to the last line in the range of text to be duplicated. The cursor is at the command line.

## Duplicating Text Within the Current Page

To duplicate lines within the current page, type the **DUPLICATE** command using the following format:

```
DUPLICATE range [BEFORE | AFTER] address
```



SED copies the range of text and places it before or after the line specified by address. address can be

- a line number within the current page
- the keywords CURRENT, PREVIOUS, NEXT, LAST

To duplicate lines 2 and 3 of your text and place them after the last line, type

```
*DUPLICATE 2 3 AFTER LAST )
```

```
DUPLICATE 2 3 AFTER LAST
```

```
*_ page 1 line 3
```

- 1 Here is the first line of text.
- 2 This is the second line.
- 3 This line has three mistakes.
- 4 Now you have a fourth line.
- 5 We will move this fifth line.
- 6 Here is the sixth line.
- 7 This is the second line.
- 8 This line has three mistakes.

Your current line is now line 3.

## Duplicating Text Onto Another File

If you want to duplicate lines of text onto *another* file, type the DUPLICATE command using the following format:

```
DUPLICATE range ONTO pathname
```

SED copies the range of text to the end of the file specified by pathname. You cannot specify an address within pathname.

If the file does not exist, SED creates it in your working directory and then adds the duplicate lines of text. SED displays a message to let you know that it has created the file named in pathname. For example:

```
*DUPLICATE 10 17 ONTO OTHER )  
File OTHER created  
*
```

## Duplicating Text Onto Another Page

You cannot duplicate text to another page in the same file directly. To do so indirectly:

- Duplicate the range of text onto a new file.
- Position to the page where you want the copied text.
- Insert the new file where you want it on that page.

For example, assume you want an elaborate table to appear several times in a file. You create the table once within the current file or make it a separate file. Then duplicate the table onto a new file:

```
*DUPLICATE 46 85 ONTO A1FILE )  
FILE A1FILE CREATED
```

Then, whenever you want the table in your text, position to the appropriate page and line and insert the file:

```
*INSERT FROM A1FILE )
```

## Updating Your File (SAVE)

If your system fails, you may lose the changes you have made during the current editing session. To prevent this you should update your file from time to time. You can update a file by terminating the editing session with the BYE command and then re-executing SED, but this takes time. It is easier and faster to use the SAVE command.

The SAVE command creates a copy of your file, including all changes you made in the current editing session, and gives the file the name of the edit file with the .SV extension. When you type the SAVE command during an editing session, SED overwrites the previous .SV file (if one exists), updating it.

To enter the SAVE command, you can type it on the command line and press NEW LINE:

```
*SAVE )
```

or press a function key that you have defined to execute the SAVE command. (You will learn how to define function keys in Chapter 3.)

Then, if your system fails, you have the .SV copy of your file. You can edit the .SV file, but you might find it easier to rename the .SV file with the CLIRENAME command.

## Deleting and Restoring Text (DELETE and UNDO)

### DELETE

The DELETE command allows you to remove lines of text from a file.

```
DELETE [range]
```

When SED has completed the deletion, it automatically renumbers the lines and sets the current line to the line following the deleted text. If the last line deleted is the last line in the page, SED sets the current line to the line immediately before the range of deleted text. If you delete an entire page, SED renumbers the pages and sets the current line to the last line in the page before the one deleted.

## Deleting the Current Line

To delete the current line, you can press the DELETE CURRENT function key or type the DELETE command on the command line without a range.

Let's delete line 7 of your text. To make line 7 the current line, press the DOWN 4 LINES function key. To delete line 7, press the DELETE CURRENT function key.

*delete current*

\*\_ page 1 line 7

---

```
1 Here is the first line of text.
2 This is the second line.
3 This line has three mistakes.
4 Now you have a fourth line.
5 We will move this fifth line.
6 Here is the sixth line.
7 This line has three mistakes.
```

Your current line is now line 7.

## Deleting a Range of Lines

To delete a range of text, type the DELETE command with a range argument.

*range* can be

- line numbers within the current page
- a single page argument
- the keywords PREVIOUS, NEXT, LAST, ALL, REMAINING, FOR

Examples:

\*DELETE 112 120 }

SED deletes line 112 through line 120 in the current page and sets the current line to the line immediately following the last line deleted.

\*DELETE PAGE 7 }

SED deletes all of page 7, renumbers the pages, and sets the current line to the last line in page 6.

\*DELETE REMAINING }

SED deletes the remaining lines in the current page, beginning with the line following the current line to the last line in the page. The current line does not change.

## UNDO

The UNDO command allows you to restore the last range of text you deleted with the DELETE command. UNDO restores only the most recent deletion during the current editing session. Therefore, entering several UNDO commands in a row will not restore text deleted earlier.

**UNDO**

The UNDO command never takes an argument.

When SED has completed the command, it automatically renumbers the lines and sets the current line to the first line restored. If the restored text is an entire page or includes the last line in the page, SED does not change the current line. When restoring an entire page, SED inserts the text at the beginning of the next page. To restore the page break, use the SPLIT command, explained in Chapter 3.

To restore the line of text you just deleted, press the UNDO function key.

*undo*

\*\_ page 1 line 7

---

```
1 Here is the first line of text.
2 This is the second line.
3 This line has three mistakes.
4 Now you have a fourth line.
5 We will move this fifth line
6 Here is the sixth line.
7 This is the second line.
8 This line has three mistakes.
```

Your current line is now line 7.

## Locating Text (FIND and BACKFIND)

### FIND

The FIND command allows you to find a word or phrase and set the current line to the line in which it is located.

FIND "searchstring" [*range*]

*searchstring* is the word or phrase you wish to locate. You must enclose the searchstring in quotation marks or apostrophes if it contains blank spaces, symbols other than a dollar sign or underscore, or if you want a case-sensitive search. Use apostrophes if the searchstring contains quotation marks, and vice versa.

*range* can be

- line numbers within the current page
- a single page argument
- a series of pages
- the keywords CURRENT, PREVIOUS, NEXT, LAST, ALL, ALL PAGES, REMAINING, REMAINING PAGES, FOR

The FIND command instructs SED to search the specified range for the first occurrence of *searchstring*. If you omit the range, SED searches from the line following the current line to the end of the page.

The cursor remains at the command line. If you enter the MODIFY command immediately after entering the

FIND command, SED positions the cursor at the first character in the searchstring.

If SED cannot find the searchstring within the range you specify, it displays the following error message:

*String not found, correct the command:*

### Locating Text Within the Current Page

To locate a word or phrase after the current line in the page, type the FIND command and the searchstring without a range argument. Let's find the first occurrence of *Here is* in your text. Type

```
*FIND 'Here is' )
```

*String not found, correct the command:*

```
*FIND 'Here is'
```

-

SED displays the error message, because *Here is* does not occur after the current line. Let's modify the command to instruct SED to search the entire page. Move the cursor to the end of the command by pressing CTRL-A.

Press the space bar once and type

```
ALL )
```

The argument keyword ALL instructed SED to search for the first occurrence of *Here is* from the first through the last lines in the current page. SED sets the current line to line 1, which contains the first occurrence of *Here is* in your text.

### Locating Text Across Page Boundaries

To locate the first occurrence of a word or phrase on another page in your file, type the FIND command with a single page argument, a series of pages, ALL PAGES, or REMAINING PAGES.

Examples:

```
*FIND 'Comments:' PAGE 6 )
```

SED searches for the first occurrence of *Comments:* from the first through the last line on page 6.

```
*FIND 'Comments:' PAGE 3 7 )
```

SED searches for the first occurrence of *Comments:* from the first line on page 3 through the last line on page 7.

```
*FIND 'Comments:' ALL PAGES )
```

SED searches for the first occurrence of *Comments:* from the first through the last pages.

```
*FIND 'Comments:' REMAINING PAGES )
```

SED searches for the first occurrence of *Comments:* from the line following the current line through the last line on the last page.

### Locating Subsequent Occurrences of Text

When you type the FIND command, SED locates only the first occurrence of *searchstring* within the range you specify. You may want to locate other occurrences of *searchstring* that come after the first.

**Within the Current Page** – The FIND function key stores the last searchstring that you typed on the command line with the FIND command. You can press the FIND function key to locate subsequent occurrences of the stored searchstring within the current page.

To find the next occurrence of *Here is* in your text press the FIND function key. SED sets the current line to line 6.

If you enter keyboard input mode after typing the FIND command and then press the FIND function key, SED positions the cursor at the first character in the next occurrence of the searchstring.

**Across Page Boundaries** – The FIND command with the ALL PAGES argument locates only the first occurrence of the searchstring in the file. To locate the next occurrence on another page in the file, you must reenter the FIND command with the REMAINING PAGES argument, a single page argument, or a range of pages.

### BACKFIND

The BACKFIND command allows you to find a specified word or phrase in the current page, beginning at the line before the current line and working *backwards*. SED sets the current line to the line containing the next previous occurrence of the word or phrase.

BACKFIND "searchstring"

As with the FIND command, you must enclose the searchstring in quotation marks or apostrophes if it contains blank spaces, symbols other than a dollar sign or underscore, or if you want a case-sensitive search.

The BACKFIND command does not take a range argument. The cursor remains at the command line. If SED cannot find the searchstring, it displays the following error message:

*String not found, correct the command:*

To find the next previous occurrence of *this* in your text, type

```
*BACKFIND this )
```

SED sets the current line to line 5.

The BACKFIND function key stores the last searchstring you typed on the command line with either the

BACKFIND or the FIND command. To locate the next previous occurrence of the stored searchstring, you press the BACKFIND function key.

To find the next previous occurrence of *this*, press the BACKFIND function key. SED sets the current line to line 3.

## Substituting Text (SUBSTITUTE)

The SUBSTITUTE command allows you to substitute a word or phrase for another word or phrase within a specified range.

SUBSTITUTE "searchstring 1" [FOR] "searchstring 2"  
[range]

SED searches for each occurrence of searchstring 2 within the range you specify and replaces it with searchstring 1.

You must enclose the searchstrings in quotation marks or apostrophes if they contain blank spaces or symbols other than a dollar sign or underscore. If you do not enclose the first searchstring, SED translates it to uppercase. If you do not enclose the second searchstring, the substitution is not case sensitive. That is, the first searchstring replaces the second searchstring regardless of the case of the letters in the second searchstring. For example, if you type

SUBSTITUTE "that" which

SED substitutes *that* for which, WHICH, Which, and even whIch.

range can be:

- line numbers within the current page
- a single page argument
- a series of pages
- the keywords CURRENT, PREVIOUS, NEXT, LAST, ALL, ALL PAGES, REMAINING, REMAINING PAGES, FOR

If you omit the range, SED makes the substitution from the current line to the end of the current page.

SED displays each line in which it made a substitution and sets the current line to the last line in which it made a substitution. To redisplay a range of lines around the current line or change the current line, use one of the methods for refreshing the screen explained in Chapter 1.

### Substituting Text Within the Current Page

To substitute the word *had* for *has* in your text, type

```
*SUBSTITUTE 'had' 'has' ALL )
```

```
SUBSTITUTE 'had' 'has' ALL
```

```
*_
```

page 1 line 8

```
3 This line had three mistakes.
```

```
8 This line had three mistakes.
```

SED makes the substitution in lines 3 and 8 and sets the current line to line 8. To redisplay all of your text, press NEW LINE, ESC, ERASE PAGE, or the VIEW function key.

### Substituting Text Across Page Boundaries

To substitute a word or phrase on other pages in your file, type the SUBSTITUTE command with a single page argument, a series of pages, ALL PAGES, or REMAINING PAGES.

Examples:

```
*SUBSTITUTE 'occurrence' 'occurrence' PAGE 3 )  
SED replaces occurrence with the correct spelling of occurrence each time it appears in page 3.
```

```
*SUBSTITUTE 'occurrence' 'occurrence' PAGE 4 7 )  
SED makes the substitution each time it finds occurrence from the beginning of page 4 to the end of page 7.
```

```
*SUBSTITUTE 'occurrence' 'occurrence' ALL PAGES )  
SED replaces each instance of occurrence with occurrence from the first to the last page in your file.
```

```
*SUBSTITUTE 'occurrence' 'occurrence' REMAINING PAGES )  
SED substitutes occurrence for occurrence from the line following the current line to the end of the file.
```

## Replacing Lines of Text (REPLACE)

The REPLACE command allows you to delete a range of text and replace it with text you type at your terminal keyboard.

```
REPLACE [range]
```

range can be

- line numbers within the current page
- a single page argument
- the keywords PREVIOUS, CURRENT, NEXT, LAST, ALL, REMAINING, FOR

If you omit the range, SED replaces the current line.

The range is the lines you want to delete. You can replace the deleted lines with any number of lines typed at your terminal keyboard as long as you do not exceed 1023 lines in the current page. If this occurs, SED displays the following error message:

Page exceeded 1023 line limit, line not accepted on this page, use 'SPLIT'

Use the SPLIT command, described in Chapter 3, to create a new page.

To terminate the REPLACE command and return to command mode, press the ESC key. Remember to press NEW LINE before you press ESC, or you will lose the last replacement line you typed. When you terminate the REPLACE command, SED automatically renumbers the lines and sets the current line to the line following the last line of replacement text.

To replace lines 5 and 6 of your text, type

```
*REPLACE 5 6 )
```

view

```
*REPLACE 5 6 page 1 line 8
```

---

```
1 Here is the first line of text.
2 This is the second line.
3 This line had three mistakes.
4 Now you have a fourth line.
5 —
* This is the second line.
* This line had three mistakes.
```

SED moves lines 7 and 8 down on your screen and replaces their line numbers with asterisks. (If you are not displaying line numbers, SED does not display the asterisks, either.) SED then deletes lines 5 and 6 and moves the cursor to the first character position on line 5. Now, type

This is replacement text. )

view

```
*REPLACE 5 6 page 1 line 8
```

---

```
1 Here is the first line of text.
2 This is the second line.
3 This line had three mistakes.
4 Now you have a fourth line.
5 This is replacement text.
6 —
* This is the second line.
* This line had three mistakes.
```

When you press NEW LINE at the end of line 5, SED moves the cursor to the first character position on line 6. Now type

This will be line 1 of page 2. )



```
REPLACE 5 6
```

```
*—
```

*page 1 line 7*

---

```
1 Here is the first line of text.
2 This is the second line.
3 This line had three mistakes.
4 Now you have a fourth line.
5 This is replacement text.
6 This will be line 1 of page 2.
7 This is the second line.
8 This line had three mistakes.
```

You have just replaced two lines of text and returned to command mode.

## Terminating This Editing Session (BYE)

The BYE command terminates the editing session and returns you to the CLI. SED updates your file by incorporating all the changes you made during this session.

To terminate your current editing session, type

```
*BYE )
```

If you are closing a new file, SED displays this message:

```
Output file - :UDD:USERNAME:SAMPLE
)
```

If you did not change the file during the session, SED displays no message:

```
*BYE )
```

```
)
```

## Saving a Backup Copy

If the file existed before this editing session, and you changed it during this session, SED asks if you want to save the original as a backup. You may want to keep a backup copy of your file if you're not sure you like the changes you made, or if you want two versions of the file.

```
*BYE )
```

*Do you want to save the original file as a backup file?*

If you answer N) or just ), SED displays the output file message and returns you to the CLI. If you answer Y), SED creates a backup file named filename.BU, deleting any existing version of filename.BU first. The .BU file contains a copy of filename as it existed before the current editing session. SED always places the backup file in the original directory. For example:

```
) XEQ SED EXAMPLE )
* (You edit EXAMPLE.)
*BYE )
Do you want to save the original file as a backup
file? Y )
Output file - :UDD:JOHN:EXAMPLE
Backup file - :UDD:JOHN:EXAMPLE.BU
)
```

The next time you edit EXAMPLE, and ask for a backup file, SED deletes the old backup file.

## Continuing with SED

If you want to finish editing one file and start editing another one, you can save some time by using the argument CONTINUE with the BYE command. If you want SED to create a backup file and then to continue editing, press the BYE YES CONTINUE function key. Or you can type the command on the command line. Type BYE, either YES or NO, and CONTINUE all on the same line and press NEW LINE. After SED closes the first file, it asks you for the name of the next file you want to edit. For example:

```
*BYE YES CONTINUE )
Output file - :UDD:JOHN:EXAMPLE
Backup file - :UDD:JOHN:EXAMPLE
Name of file to edit: —
```

You type the name of the next file you want to edit and press NEW LINE. Using the argument CONTINUE saves time, because the computer does not have to recreate the SED process.

## Removing PERMANENCE

If you used the CLI PERMANENCE command to protect an existing edit file or backup file from accidental deletion, SED cannot replace the original edit file or backup file with their updated versions. Therefore, SED displays an error message when you try to terminate the session with the BYE command.

You can correct this error by using the argument PERM\_OFF with the BYE command. For example, let's say that you set PERMANENCE ON for the files INTRO and INTRO.BU

```
) PERMANENCE INTRO ON )
) PERMANENCE INTRO.BU ON )
```

Then you edit the file INTRO

```
) XEQ SED INTRO )
* (You make changes to the file.)
```

When you have finished editing INTRO and you type

```
*BYE
Do you want to save the original file as a backup file?
Y )
```

OR

```
*BYE Y )
```

SED displays the following error message and redisplay the command:

*Backup file has permanence set, correct the command:*

If you type BYE) and answer N) to SED's question, or you type BYE N), SED displays this message:

*Edit file has permanence set, correct the command:*

To correct any of these BYE commands, type PERM\_OFF on the same line with the BYE command. SED then permits you to terminate the session and updates the appropriate files. For example:

```
*BYE Y PERM_OFF )
Previous backup file deleted.
Output file - :UDD:USERNAME:INTRO
Backup file - :UDD:USERNAME:INTRO.BU
)
```

If you made no changes to a file that has permanence set, SED allows you to terminate the session by typing BYE alone.

If you want to turn permanence ON again for these files, you must retype the CLI PERMANENCE command.

If you want to close one file and start editing another, you can type the argument CONTINUE on the same line with BYE, Y or N, and PERM\_OFF. SED closes the current file (with or without creating or updating the backup file) and asks for the name of the next file to edit.

SED accepts the arguments to the BYE command in any order. For example, all of the following commands are valid:

```
*BYE Y PERM_OFF CONTINUE )
*BYE N CONTINUE PERM_OFF )
*BYE PERM_OFF CONTINUE Y )
*BYE CONTINUE N PERM_OFF )
```

## **Congratulations!**

You have just completed your first editing session with SED. During this session, you learned to create a text file with SED, to add text to it, and to make many kinds of changes. You've now had experience with SED's most frequently used commands.

In the next chapter, you will learn about SED's advanced commands and features. You will reedit the file SAMPLE during a second session with SED.

End of Chapter





# Chapter 3

## Advanced Editing with SED

This chapter contains the advanced commands and features of SED. It explains switches you can use when executing SED, display settings, defining function keys, and SED's advanced editing commands. You will have a second SED session with the file SAMPLE in which you can try out many of the commands covered in this chapter.

### SED Files

In Chapters 1 and 2 you learned about three files SED creates: the edit file, the .SV file, and the .BU file. We'll define those again here, plus introduce three other files that SED uses: an auxiliary file (.ED) and two temporary files (.SC and .S2).

#### The Edit File

The *edit file* contains the text you enter. It has the filename you type when you execute SED from the CLI. During the editing session, SED stores all of your additions and modifications in the temporary .SC file. When you end the editing session with the BYE command, SED combines the changes in the .SC file and the original edit file (as it existed before the current editing session) into another temporary file (the .S2 file). If you ask for a backup file, SED renames the original edit file, giving it the name of the original edit file with the .BU extension. If you instruct SED not to save a backup file, SED deletes the original edit file. SED then renames the .S2 file, giving it the name of the original edit file, and deletes the temporary .SC and .S2 files.

#### The .SV File

The .SV file contains a copy of your edit file, including all of the changes you made up to the time you entered the SAVE command. The .SV file has the name of your edit file and the .SV extension. Whenever you execute the SAVE command during a session, SED overwrites the .SV file with the changes you made since executing the previous SAVE command. If your system fails, you can edit the .SV file, but you may prefer to first rename the .SV file (with the CLI RENAME command) and then edit it.

#### The .BU File

The .BU file contains a copy of your edit file as it existed before you began the current editing session. SED created the .BU file when you terminated the previous editing session by typing BYE Y) or by typing BYE) and answering Y) when SED asked if you wanted a backup file. The .BU file has the name of your edit file with the .BU extension. You may choose to create a .BU file if you want two versions of the file or if you are not sure you want to keep the changes you made.

#### The .ED File

For every edit file that SED creates, it also creates an auxiliary or .ED file that stores display settings. Display settings, which we explain later in this chapter, affect the way in which SED displays your text during an editing session. They do such things as display line numbers and display the current line bright and the rest dim.

When you terminate an editing session with the BYE command, SED updates the .ED file to include any changes you made to the display settings during the session. At the same time, SED stores the current page and line positions in the .ED file. The next time you edit the file, SED uses the updated information in the .ED file to display your text. The .ED file has the name of the edit file with the .ED extension.

#### The .SC File

The .SC file is a temporary file that stores the changes you make *while* you are editing. SED creates the .SC file when you execute SED and gives it the name of the edit file with the .SC extension. Each time you press the NEW LINE key after adding or changing a line of text, SED stores that line in the .SC file. When you terminate the session with the BYE command, SED combines the changes in the .SC file with the original edit file to make the .S2 file. SED then deletes the .SC file.

#### The .S2 File

The .S2 file is a temporary file that contains the updated version of your edit file after you type the BYE command. SED creates the .S2 file when you execute SED and gives

it the name of the edit file with the .S2 extension. When you type the BYE command, SED combines the changes in the .SC file with the edit file and stores them in the .S2 file. SED renames the .S2 file, giving it the filename you typed when you executed SED. SED then deletes the .S2 file.

## Executing SED with Command Switches

In Chapter 2 you learned how to execute SED. You may choose to execute SED as described there. But you can keep neater directories, change the way SED displays your files, and improve system response time by taking advantage of one or more of SED's command switches.

A command switch modifies *how* you execute SED. SED provides six command switches that you can add to the XEQ SED command.

```
XEQ SED [ /ED=directory
          /NO_ED
          /PROFILE=pathname pathname
          /WORK=directory
          /NO_SCREEN
          /NO_FORM_FEEDS ]
```

### /ED=directory

As you learned earlier in this chapter, for every file SED creates, it also creates a .ED file that stores display settings. The /ED=directory switch allows you to choose the directory that will store your .ED files. Some users feel that a proliferation of .ED files clutters their directories. Using the /ED=directory switch allows you to keep neater directories and to save space in your working directory.

```
) XEQ SED/ED=:UDD:JOHN:NEAT EXAMPLE )
```

This command executes SED on the file EXAMPLE and instructs SED to store the file EXAMPLE.ED in the directory :UDD:JOHN:NEAT.

### /NO\_ED

You may choose not to create a .ED file at all by specifying the /NO\_ED switch when you execute SED. If you do this, SED will still work, but it will take a bit longer to begin an editing session. Using the /NO\_ED switch saves disk space since your system does not have to store a .ED file. To instruct SED not to create a .ED file, type

```
) XEQ SED/NO_ED EXAMPLE )
```

This command executes SED from the CLI and instructs SED not to create a .ED file for the file EXAMPLE.

### /PROFILE=pathname

The /PROFILE=pathname switch allows you to execute a file containing SED commands when you begin an editing session. This is a fast and easy way to execute SED commands that change the display settings and define function keys. (We explain these commands in detail later in this chapter.) You create the file ahead of time and specify its pathname in the /PROFILE=pathname switch. For example, you could create such a file and call it FOO:

```
) XEQ SED FOO )
Do you want file FOO to be created? Y )
```

When SED displays its command line, you press the APPEND function key and then type

```
CLEAR LINE_NUMBER_DISPLAY; )
SET DISPLAY_MODE 2; )
SET FUNCTION CTRL 9 "SAVE" )
```



```
*BYE )
)_
```

FOO contains commands that instruct SED to display your text without line numbers, to display all lines bright but place a dimmed asterisk before the current line, and to define the ninth function key on the CTRL row to execute the SAVE command.

If you want to execute the commands in FOO when you begin an editing session on the file FORMS, for example, you type

```
) XEQ SED/PROFILE=FOO FORMS )
```

When SED displays its command line, it flashes the commands contained in FOO on the command line.

When creating a file of SED commands, it is good practice to type out all of the commands and keywords, using no abbreviations. This will save you time and inconvenience if future revisions of SED have new commands and argument keywords that make the current abbreviations invalid; you will not have to edit the commands in the files you have already created.

### /WORK=directory

The /WORK=directory switch stores the temporary .SC file SED creates in a directory you name. If your system has a fixed-head disk, your system manager can edit the SED.CLI macro to make the directory a file on the fixed-head disk. This improves system response somewhat

because it allows SED faster access to the .SC file. Using the /WORK=directory switch also saves space in your working directory.

```
) XEQ SED /WORK=:UDD:WOOD:TEMPS EXAMPLE )
```

This command executes SED on the file EXAMPLE and instructs SED to store the file EXAMPLE.SC in the directory :UDD:DON:TEMPS.

## /NO\_SCREEN

When you specify the /NO\_SCREEN switch, SED will not automatically update the screen. This is useful for both low baud rate terminals and for hard copy terminals. If you use the /NO\_SCREEN switch, SED displays only its asterisk prompt at the beginning of the session.

You must tell SED to display lines of text you want to see, using the VIEW or LIST command. With the VIEW command, SED displays two lines before and two lines after the current line. (You can change the number of lines SED displays with the SET VIEW\_RANGE command, explained later in this chapter.) With the LIST command, you can instruct SED to display a range of lines without changing the current line.

When you enter keyboard input mode with the APPEND, INSERT, or REPLACE command, SED displays a line number (if you're displaying line numbers) and the cursor. When you enter keyboard input mode with the MODIFY command, SED displays the line number of the current line or the line you specified and the text on that line.

When you change the current line in command mode, SED changes the current line but does not display it. When you change the current line in keyboard input mode, SED displays the new current line.

```
) XEQ SED /NO_SCREEN EXAMPLE )
```

This command executes SED on the file EXAMPLE and instructs SED not to automatically update the screen.

## /NO\_FORM\_FEEDS

The /NO\_FORM\_FEEDS switch instructs SED to remove all page breaks (form feeds) from your file when you terminate the editing session. After editing the file, type BYE to end the session. SED removes all page breaks you created with the SPLIT command, whether you created them during this session or during a previous session.

```
) XEQ SED /NO_FORM_FEEDS EXAMPLE )
```

This command executes SED on the file EXAMPLE and instructs SED to remove all page breaks from EXAMPLE when you end the editing session.

## Editing an Existing File

In this chapter, you will edit an existing file, SAMPLE, which you created during your first SED session in Chapter 1. Let's purposely misspell SAMPLE and see what happens. Type

```
) XEQ SED ZAMPLE )
```

*Do you want ZAMPLE to be created?*

When you misspell the name of an existing file, SED asks if you want to create the file you named. Because SED cannot find the file, it assumes you want to create it. Since you do not want to create ZAMPLE, type

```
N ) or just )
```

SED then asks if you want to start over again. Since you do, type Y) and when SED asks for the name of the file to edit, type SAMPLE).

*Start over? Y )*

*Name of file to edit: SAMPLE )*

If you are searching for an existing file and you did not misspell its filename, SED's question about creating the file means the file isn't in your working directory or you typed the wrong pathname. In this case, type N). When SED asks if you want to start over, type N). SED returns you to the CLI where you can try another directory.

When you give SED the name of an existing file that it can locate, SED displays its revision number and the complete pathname of your input file:

```
SED Rev 4.10; Input file - :UDD:USERNAME:SAMPLE
```

When SED is ready to begin the session, it displays the command line and a screen of text.

```
view
```

```
*_
```

```
page 1 line 7
```

- 1 *Here is the first line of text.*
- 2 *This is the second line.*
- 3 *This line had three mistakes.*
- 4 *Now you have a fourth line.*
- 5 *This is replacement text.*
- 6 *This will be line 1 of page 2.*
- 7 *This is the second line.*
- 8 *This line had three mistakes.*

You are now ready to reedit the file SAMPLE.

## Display Settings (DISPLAY, SET, and CLEAR)

Display settings determine the way SED displays your text on the screen and the functionality of certain SED features and keys. Display settings apply only when you are using SED; they do not affect the contents of your edit file.

In each .ED file, SED stores 11 display settings for the corresponding edit file. They are:

VIEW\_RANGE  
DISPLAY\_MODE  
TYPER\_MODE  
BLANK\_MODE  
PRELOAD\_OLD\_LINE  
SKIP\_WHITE\_SPACE  
UPPERCASE\_SHIFT\_CHARACTER  
ESCAPE\_CHARACTER  
NEW\_LINE\_IS\_VIEW  
ESCAPE\_IS\_VIEW  
LINE\_NUMBER\_DISPLAY

### DISPLAY

The DISPLAY command allows you to list the current display settings and other file status information on your terminal screen. You can either press the DISPLAY function key or type the DISPLAY command on the command line. The file status display includes the following:

- the complete pathname of the file
- the current page number
- the current line number
- the last page number in the file
- the last line number on the current page
- each display setting and its current state
- a statement when SED is displaying line numbers

To display information about your file, press the DISPLAY function key.

display

\*—

page 1 line 7

Edit file name - :UDD:USERNAME:SAMPLE

Current page: 1 Last page: 1  
Current line: 7 Last line: 8

View range: 10  
Display mode: 0

Typewriter mode: ON  
Blank mode: OFF

Preload old line: ON  
Skip white space: ON

Upper mode: OFF  
Escape mode: OFF

New line is view: ON  
Escape is view: ON

Line numbers being displayed

NOTE: The line marked Escape mode displays information about the ESCAPE\_CHARACTER display setting, and the line marked Upper mode displays information about the UPPERCASE\_SHIFT\_CHARACTER display setting. If you're displaying your text without line numbers, SED deletes the last line of the file status display.

You may find it helpful to keep the file status display on your screen while you read the remaining sections on display settings. When you want to refresh your screen, press NEW LINE, ESC, ERASE PAGE, or the VIEW function key.

### Changing Display Settings

SED uses default (standard) display settings unless you choose to change them. To change the display settings, you always use the SET and CLEAR commands. However, SED provides three ways to execute SET and CLEAR commands:

- During an editing session, you can change display settings one at a time using the SET and CLEAR commands, which we explain in the next section.

- You can create a separate file containing SET and CLEAR commands that change display settings and specify the file's pathname in the /PROFILE=pathname switch when you execute SED from the CLI. This allows you to change display settings at the beginning of an editing session. It's a convenient way to set the display settings the way you want them for a new file. For more information on the /PROFILE=pathname switch, see the section on Command Switches in this chapter.
- During an editing session, you can change several display settings all at once with SED's EXECUTE command. As with the /PROFILE=pathname switch, you create a separate file containing SET and CLEAR commands. But in this case, you specify the file's pathname as an argument to SED's EXECUTE command during the editing session. We explain the details of SED's EXECUTE command later in this chapter.

Whenever you change display settings, SED updates the .ED file to include the changes. The next time you edit the file, SED uses the updated .ED file to determine how to display your text.

Table 3-1 lists the default settings and summarizes the functions of the display settings.

## SET and CLEAR

The SET and CLEAR commands allow you to change the display settings. You can execute the SET and CLEAR commands one at a time or execute a file containing several SET and CLEAR commands. Except for VIEW\_RANGE and DISPLAY\_MODE, you change a display setting from OFF to ON using the SET command and from ON to OFF using the CLEAR command. To change the VIEW\_RANGE and DISPLAY\_MODE settings, you always use the SET command. You can abbreviate each display setting to the fewest number of characters that uniquely identifies it.

When using the SET command, you change some display settings by typing SET and the name of the display setting. You change others by typing SET, the name of the display setting, and a number or a character.

SET	}	VIEW_RANGE n
		DISPLAY_MODE n
		TYPYR_MODE
		BLANK_MODE
		PRELOAD_OLD_LINE
		SKIP_WHITE_SPACE
		UPPERCASE_SHIFT_CHARACTER x
		ESCAPE_CHARACTER x
		NEW_LINE_IS_VIEW
		ESCAPE_IS_VIEW
LINE_NUMBER_DISPLAY		

When using the CLEAR command to change display settings, you type CLEAR and the name of the display setting.

CLEAR	}	TYPYR_MODE
		BLANK_MODE
		PRELOAD_OLD_LINE
		SKIP_WHITE_SPACE
		UPPERCASE_SHIFT_CHARACTER
		ESCAPE_CHARACTER
		NEW_LINE_IS_VIEW
		ESCAPE_IS_VIEW
LINE_NUMBER_DISPLAY		

The SET and CLEAR commands do not change the current line position. The cursor remains at the command line.

We give examples of the SET and CLEAR commands with the explanation of each display setting that follows.

## VIEW\_RANGE

The VIEW\_RANGE display setting determines the number of lines that SED displays before and after the current line. The number of lines can be from 1 to 10. The default setting is 10, which means that SED displays 21 lines of text on your screen, 10 before and 10 after the current line. Reducing the VIEW\_RANGE setting is helpful if you are working at a low baud rate terminal or on a slow system. It takes less time to refresh the screen when you display fewer lines of text.

To change the VIEW\_RANGE display setting, you type SET VIEW\_RANGE and the number of lines you want SED to display before and after the current line. The number must be from 1 to 10. The maximum is 10. For example, if you want SED to display 5 lines before and 5 lines after the current line, type

```
*SET VIEW_RANGE 5 ;
```

Always use the SET command to change the VIEW\_RANGE display setting. If you type CLEAR VIEW\_RANGE, SED displays the following error message and redisplay the command:

```
CLEAR must be followed by a different argument,
correct the command:
*CLEAR VIEW_RANGE
```

## DISPLAY\_MODE

The DISPLAY\_MODE setting determines how SED displays the current line. SED provides four options, numbered from 0 to 3. Table 3-2 explains each DISPLAY\_MODE option.

**Table 3-1. Summary of SED Display Settings**

<b>Display Setting</b>	<b>Default Setting</b>	<b>SET/ON</b>	<b>CLEAR/OFF</b>
VIEW_RANGE n	10	SED displays the number of lines (n) you specify (from 1 to 10) before and after the current line.	You must use the SET command to change the VIEW_RANGE display setting; CLEAR displays an error message.
DISPLAY_MODE n	0	<p>SED displays the current line according to the option number (n) you specify. n may be 0, 1, 2, or 3 as follows:</p> <p>Option 0 displays the current line bright and the rest dim.</p> <p>Option 1 displays all lines bright.</p> <p>Option 2 displays all lines bright and places a dimmed asterisk before the current line.</p> <p>Option 3 displays the current line dim and the rest bright.</p>	You must use the SET command to change the DISPLAY_MODE display setting; CLEAR displays an error message.
TYPYR_MODE	ON	The cursor maintains its column position when you press the uparrow or downarrow key or any of the screen position keys.	The cursor moves to the leftmost column position when you press the uparrow or downarrow key or any of the screen position keys.
BLANK_MODE	OFF	SED pads all lines with trailing blanks, counting each as a character.	SED removes trailing blanks from all lines.
PRELOAD_OLD_LINE	ON	Saves a copy of the previous current line. CTRL-A and the rightarrow key repeat characters of the previous line.	Does not save a copy of the previous current line. CTRL-A moves the cursor to the end of the line. The rightarrow key moves the cursor one position to the right.
SKIP_WHITE_SPACE	ON	The cursor moves to the first nonblank character on the next line when you press the NEW LINE key.	The cursor moves to the leftmost position on the next line when you press the NEW LINE key.

(continues)

**Table 3-1. Summary of SED Display Settings**

Display Setting	Default Setting	SET/ON	CLEAR/OFF
UPPERCASE_SHIFT_CHARACTER x	OFF	(For users of D1 terminals) Typing the character you specify (x) causes the following character to be uppercase. Otherwise all characters you type are translated to lowercase.	Clears the character (x) you set to translate input to lowercase.
ESCAPE_CHARACTER x	OFF	Typing the character you specify (x) inserts an escape character in your file.	Clears the character (x) you set to insert escape characters in your file.
NEW_LINE_IS_VIEW	ON	Pressing NEW LINE when the cursor is on an empty command line refreshes the screen.	Pressing NEW LINE when the cursor is on an empty command line has no effect on the screen display.
ESCAPE_IS_VIEW	ON	Pressing ESC when the cursor is on an empty command line refreshes the screen.	Pressing ESC when the cursor is on an empty command line has no effect on the screen display.
LINE_NUMBER_DISPLAY	ON	SED displays your text with line numbers.	SED displays your text without line numbers.

(concluded)

**Table 3-2. DISPLAY\_MODE Options**

Option	Current Line Display
0 (default)	Displays the current line bright and the rest dim
1	Displays all lines bright
2	Displays all lines bright but places a dimmed asterisk before the current line
3	Displays the current line dim and the rest bright

To change the DISPLAY\_MODE setting, type SET DISPLAY\_MODE and the number of the option you want. For example, if you want SED to display all lines bright and place a dimmed asterisk before the current line, type SET DISPLAY\_MODE 2 ;

Like VIEW\_RANGE, you always change the DISPLAY\_MODE setting with the SET command. If you type CLEAR DISPLAY\_MODE, SED displays the following error message and redisplay the command:

*CLEAR must be followed by a different argument, correct the command:*  
\*CLEAR DISPLAY\_MODE

### TYPYR\_MODE

The TYPYR\_MODE setting determines the action of the cursor when you press the uparrow, downarrow, or the screen position function keys. TYPYR\_MODE can be ON, which is the default setting, or OFF. When ON, pressing the uparrow or downarrow key or any of the screen position function keys in keyboard input mode causes the cursor to move to the specified line while maintaining its column position. When OFF, the cursor moves to the leftmost column on the specified line.

If `TYPYR_MODE` is OFF and you want to set it ON, type

`SET TYPYR_MODE` ]

If `TYPYR_MODE` is ON and you want it OFF, type

`CLEAR TYPYR_MODE` ]

## **BLANK\_MODE**

The `BLANK_MODE` setting determines whether SED strips trailing blanks from lines of text or pads lines with trailing blanks. `BLANK_MODE` can be ON or OFF, which is the default setting. When `BLANK_MODE` is OFF and you press the NEW LINE key after editing a line of text, SED removes any blanks at the end of the line. When ON, SED leaves all blanks at the end of the line, counting each as a character. Setting `BLANK_MODE` ON allows you to create a file with lines of exactly the same length, such as a file of fixed-length records.

If `BLANK_MODE` is OFF and you want it ON, type

`SET BLANK_MODE` ]

If `BLANK_MODE` is ON and you want it OFF, type

`CLEAR BLANK_MODE` ]

## **PRELOAD\_OLD\_LINE**

The `PRELOAD_OLD_LINE` setting determines whether or not SED saves a copy of the previous line of text. This display setting affects the action of CTRL-A and the rightarrow key in keyboard input mode. When `PRELOAD_OLD_LINE` is ON (the default setting) and you enter keyboard input mode with the APPEND, INSERT, or REPLACE command, SED saves a copy of the previous current line. When you press CTRL-A, SED copies the previous current line onto the current line. If the current line is blank, CTRL-A copies the entire previous current line. If the current line is shorter than the previous current line, CTRL-A appends what will fit of the previous line to the current line. The previous line does not overwrite the current line.

Pressing the rightarrow key when `PRELOAD_OLD_LINE` is ON types out the previous current line, one character at a time.

When `PRELOAD_OLD_LINE` is OFF, pressing CTRL-A moves the cursor to position after the last character on the line. Pressing the rightarrow key moves the cursor one position to the right.

To change `PRELOAD_OLD_LINE` from OFF to ON, type

`*SET PRELOAD_OLD_LINE` ]

To change `PRELOAD_OLD_LINE` from ON to OFF, type

`*CLEAR PRELOAD_OLD_LINE` ]

## **SKIP\_WHITE\_SPACE**

The `SKIP_WHITE_SPACE` setting affects the position of the cursor after you press the NEW LINE key. `SKIP_WHITE_SPACE` can be ON, which is the default setting, or OFF. When ON, pressing NEW LINE causes the cursor to move to the first nonblank character on the next line. When OFF, pressing NEW LINE causes the cursor to move to the leftmost position on the next line.

To change `SKIP_WHITE_SPACE` from OFF to ON, type

`*SET SKIP_WHITE_SPACE` ]

To change `SKIP_WHITE_SPACE` from ON to OFF, type

`*CLEAR SKIP_WHITE_SPACE` ]

## **UPPERCASE\_SHIFT\_CHARACTER x**

If you are not using a Dasher D1 display terminal, skip this section.

The `UPPERCASE_SHIFT_CHARACTER` setting allows the user of the Dasher D1 terminal to translate alphabetic characters to lowercase. `UPPERCASE_SHIFT_CHARACTER` can be OFF, which is the default setting, or ON and set with x. x can be any displayable character on the main keypad, except a control character.

When ON and set with x, SED translates all alphabetic characters you type to lowercase. When you type x, SED translates the next character you type to uppercase. If you want x to appear in your text, type two in a row.

NOTE: To use the character in subsequent commands that you type on the command line, you must type two in a row. For example, if you set `UPPERCASE_SHIFT_CHARACTER` with 3 and you want to execute the command `POSITION 365`, you must type `POSITION 3365`. The character you set does not affect the function key commands.

If you are using a D1 terminal and want to translate the text you type to lowercase, type `SET UPPERCASE_SHIFT_CHARACTER` and a character. For example:

`*SET UPPERCASE_SHIFT_CHARACTER { }`

Then, SED translates each alphabetic character you type to lowercase. When you want an uppercase alphabetic



character in your text, type { before the character. If you want { to appear in your text, type {{.

If you execute the DISPLAY command, you will see that SED updates the Upper mode line as follows:

*Upper mode: ON Upper character: {*

To clear the UPPERCASE\_SHIFT\_CHARACTER setting of the character you set, type

**\*CLEAR UPPERCASE\_SHIFT\_CHARACTER )**

## ESCAPE\_CHARACTER x

The ESCAPE\_CHARACTER setting allows you to insert escape characters in your edit file. ESCAPE\_CHARACTER can be OFF, which is the default setting, or ON and set with x. x can be any displayable character on the main keypad, except a control character.

**NOTE:** Because SED reads every occurrence of the character you set as an escape character, choose a character that you won't use as itself in your text or in a SED command that you type on the command line. For example, you may choose to set ESCAPE\_CHARACTER with • or ' because they are not frequently used in text, and do not appear in any SED command.

To set a character that will insert escape characters in your file, type SET ESCAPE\_CHARACTER and a character. For example:

**\*SET ESCAPE\_CHARACTER \ )**

Then, each time you type \ in your text, SED displays \ but stores it as an escape character. If you execute the DISPLAY command, you will see that SED updates the Escape mode line as follows:

*Escape mode: ON Escape character: \*

To clear the character you set and turn ESCAPE\_CHARACTER OFF, type

**\*CLEAR ESCAPE\_CHARACTER )**

If you clear the ESCAPE\_CHARACTER setting after inserting escape characters in your file, SED translates each displayed escape character to ^]. If you modify a line containing ^], SED deletes ^] and all characters to the right of it on that line.

## NEW\_LINE\_IS\_VIEW

The NEW\_LINE\_IS\_VIEW setting determines whether or not SED refreshes the screen when you enter just NEW LINE on the command line. NEW\_LINE\_IS\_VIEW can be ON (the default setting), or OFF. When ON, pressing the NEW LINE key

on an empty command line defaults to a VIEW command; SED displays a range of lines around the current line. When OFF, pressing NEW LINE on an empty command line has no effect on your screen display.

To change NEW\_LINE\_IS\_VIEW from OFF to ON, type

**\*SET NEW\_LINE\_IS\_VIEW )**

To change NEW\_LINE\_IS\_VIEW from ON to OFF, type

**\*CLEAR NEW\_LINE\_IS\_VIEW )**

## ESCAPE\_IS\_VIEW

The ESCAPE\_IS\_VIEW setting determines whether or not SED refreshes the screen when you enter just ESC on the command line. ESCAPE\_IS\_VIEW can be ON (the default setting), or OFF. When ON, pressing the ESC key on an empty command line defaults to a VIEW command; SED displays a range of lines around the current line. When OFF, pressing ESC on an empty command line has no effect on your screen display.

To change ESCAPE\_IS\_VIEW from OFF to ON, type

**\*SET ESCAPE\_IS\_VIEW )**

To change ESCAPE\_IS\_VIEW from ON to OFF, type

**\*CLEAR ESCAPE\_IS\_VIEW )**

## LINE\_NUMBER\_DISPLAY

The LINE\_NUMBER\_DISPLAY setting determines SED whether displays your text with or without line numbers. By default, SED displays your text with line numbers.

To display your text with line numbers, type

**\*SET LINE\_NUMBER\_DISPLAY )**

To display your text without line numbers, type

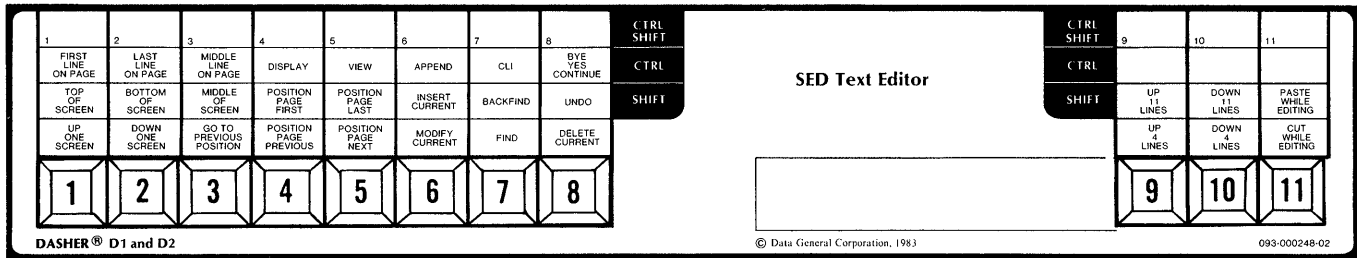
**\*CLEAR LINE\_NUMBER\_DISPLAY )**

When you clear your text display of line numbers, SED deletes the statement about line numbers from the file status display.

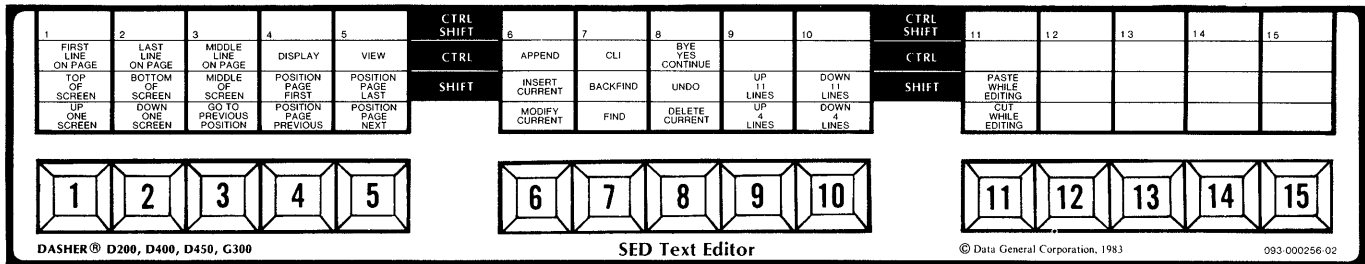
## Defining and Clearing Function Keys (SET and CLEAR)

With the SET command, you can define or redefine all of the function keys to execute SED commands. You can clear many function keys of their definitions with the CLEAR command. When you define or clear function keys during an editing session, they remain defined or cleared for that session only.

## D2 Terminal



## D200 Terminal



SD-02466

Figure 3-1. Function Keys and the SED Templates

If you want to define several function keys at the same time, you can create a separate file containing SED commands that define function keys. Then, you can execute the commands in that file by specifying its filename in the `/PROFILE=pathname` switch when you execute SED. Or, you can use the filename as an argument to SED's EXECUTE command, which we explain later in this chapter.

When you define or redefine function keys, you can write the name of the command in the proper box on the template. If you define commands for the blank boxes on the CTRL and CTRL-SHIFT rows, write the names in the blank boxes on the SED template. If you redefine the preset function keys, use a blank template to create your own customized template.

Figure 3-1 shows the function keys and the SED templates on the DASHER D200 and DASHER D2 terminal keyboards. In this figure, we numbered the function keys to make learning to define them easier. The numbers are for illustration only; they do not appear on your keyboard.

## SET

To define or redefine a function key, you type the SET command, the keyword FUNCTION, the number of the function key, and one or more SED commands in quotation marks.

**SET FUNCTION** number "string" [: "string"]

number is the number of the function key you want to define. It can be a number or a number preceded by SHIFT, CTRL, or CTRL SHIFT.

"string" is the SED command(s) that you want the function key to execute. You must enclose each command in quotation marks (" ") or apostrophes ( ' '). If "string" contains more than one command, you must separate them with semicolons.

If you want to redefine a function key on the bottom row, type SET FUNCTION, the number of the key, and the SED command in quotation marks or apostrophes. For example:

```
*SET FUNCTION 7 "HELP" ;
```

Then, when you press the seventh function key, SED executes the HELP command.

To redefine a function key on the SHIFT row, type SET FUNCTION SHIFT, the number of the function key, and the SED command in quotation marks or apostrophes. For example:

```
*SET FUNCTION CTRL 7 "SET LINE_NUMBER_
  DISPLAY" )
```

Then, when you press and hold a SHIFT key and press the seventh function key, SED clears your text display of line numbers.

If you want to define or redefine a function key on the CTRL row, type SET FUNCTION CTRL, the number of the function key, and the SED command enclosed in quotation marks or apostrophes:

```
*SET FUNCTION CTRL 7 "SET LINE_NUMBER_
  DISPLAY" )
```

Then, when you press and hold the CTRL key and press the seventh function key, SED displays your text with line numbers.

If you want to define a function key on the CTRL SHIFT row, type SET FUNCTION CTRL SHIFT, the number of the function key, and the SED command enclosed in quotation marks or apostrophes:

```
*SET FUNCTION CTRL SHIFT 7 "SAVE" )
```

Then, when you press and hold both the CTRL key and the SHIFT key and press the seventh function key, SED executes the SAVE command.

Remember that the function keys you set or clear during a SED session remain set or cleared for that session only. If you want to have the same function key definitions from one session to the next, you can create a separate file of SED commands that set and clear function keys. Then, you can execute those commands by specifying the filename in the /PROFILE=pathname switch when you execute SED, or by typing the filename as an argument to SED's EXECUTE command during an editing session.

NOTE: SED has a limited amount of space for storing function key definitions. If you exceed this space, SED displays the following error message:

*Function key string space exhausted, key not set*

When SED displays this message, you can either clear another function key and then set the function key you want to define, or choose not to define the function key and type the command on the command line.

## CLEAR

You can clear many function keys of the command definitions that SED established for you or that you established with the SET command. The following seven function keys are exceptions: UP ONE SCREEN, DOWN ONE SCREEN, GO TO PREVIOUS POSITION, TOP OF SCREEN, BOTTOM OF SCREEN, MIDDLE OF SCREEN and MIDDLE LINE ON PAGE. You can redefine them with the SET command, but when you clear them with the CLEAR command, they default to the original definitions that SED established.

To clear a function key of its command definition, type CLEAR, the keyword FUNCTION, and the number of the function key.

CLEAR FUNCTION number

number is the number of the function key that you want to clear. It can be a number or a number preceded by SHIFT, CTRL, or CTRL SHIFT.

Examples:

```
*CLEAR FUNCTION 8 )
```

This command removes the command definition stored in the ninth function key (bottom row).

```
*CLEAR FUNCTION SHIFT 10 )
```

This command clears the tenth function key on the SHIFT row of its command definition.

NOTE: If you press a cleared function key when you are in keyboard input mode, SED displays the following error message:

*Unknown command, correct the command:*

In addition to the error message, SED displays the command that placed you in keyboard input mode (APPEND, MODIFY, INSERT, or REPLACE) and positions the cursor at the first character of the command. To correct this error, you can do one of the following:

- Press NEW LINE and SED will execute the displayed command.
- Type another command.
- Erase the command by pressing ERASE EOL or CTRL-U, and then press a defined function key.

## Creating and Removing a Page Break (SPLIT and JOIN)

### SPLIT

The SPLIT command allows you to create a page break anywhere in your file. This is an easy way to manage a lot of text; it may also be necessary, since SED accepts a maximum of 1023 lines on any page.

**SPLIT** [*address*]

*address* can be

- a line number within the current page
- a single page argument
- the keywords CURRENT, PREVIOUS, NEXT, LAST

*address* is the line where you want the page break located. The split occurs at the line before this location so that *address* becomes line 1 of the new page. If you specify a page address, SED creates a blank page in front of the page you specified as the argument. If you omit the address, the current line becomes line 1 of the new page.

When SED splits a page, it automatically renumbers the lines on the new page, starting at line 1. SED sets the current line to line 1 on the new page. The cursor remains at the command line.

You can define a function key to execute the SPLIT command. If you do so, you must position the current line where you want the page break before pressing the SPLIT function key.

### Creating a Page Containing Text

You can use the SPLIT command to create a new page that contains some of the text already in your file. SED moves all text from the line you specify by address to the end of the current page.

To create a second page in your file, type

```
*SPLIT 6 ↓
```

```
SPLIT 6
```

```
*_ page 2 line 1
```

- ```
1 This will be line 1 of page 2.  
2 This is the second line.  
3 This line had three mistakes.
```

The lines that were 6, 7, and 8 on page 1 are now lines 1, 2, and 3 on page 2. Your current line is line 1 on page 2.

### Displaying Page Information

The DISPLAY command gives you page number and line number information. Press the DISPLAY function key, and SED will show you the updated file status information:

```
display
```

```
*_ page 2 line 1
```

```
Edit file name - :UDD:USERNAME:SAMPLE
```

```
Current page: 2 Last page: 2
```

```
Current line: 1 Last line: 3
```

```
View range: 10 0
```

```
Display mode:
```

```
·  
·  
·
```

The current page is 2 (We just created it); the current line is line 1 on page 2. The last page of the file is 2, and the last line in the current page is 3.

To redisplay the text on page 2, press NEW LINE, ESC, or the VIEW function key.

### Creating a Blank Page

You can insert a new, blank page in front of an existing page by typing SPLIT with a page argument. When SED has completed the split, it renumbers the pages following the blank page and displays the page following the blank page. You can then position to the blank page and add text to it with the APPEND command.

To create a blank page between page 1 and page 2 of your file, type

```
*SPLIT PAGE 2 ↓
```

```
SPLIT PAGE 2
```

```
*_ page 3 line 1
```

- ```
1 This will be line 1 of page 2.  
2 This is the second line.  
3 This line had three mistakes.
```

You now have 3 pages in your file. To take a look at the blank page you just created, press the POSITION PAGE PREVIOUS function key.

```
position page previous
```

```
*_ page 2 line 0
```

Let's add some text to this empty page. Press the APPEND function key and then type

This is text for the blank page. }  
 Later, we will cut this line. }  
 Then, we will paste these }  
 two lines. }



append

\*\_ page 2 line 4

1 This is text for the blank page.  
 2 Later, we will cut this line.  
 3 Then, we will paste these  
 4 two lines.

You now have four lines of text on page 2 of your file. The current line is line 4.

When you want to edit text on another page in your file, you can use the following function keys: POSITION PAGE PREVIOUS, POSITION PAGE NEXT, POSITION PAGE FIRST, and POSITION PAGE LAST. Or, you can type the POSITION command and a page argument on the command line. Once you are positioned on the page you want to edit, you can use any SED commands to modify that page.

SED allows a maximum of 255 pages in any one file. If you try to exceed this number with the SPLIT command, SED displays the following error message:

*File exceeded 255 page maximum, page 'SPLIT' cannot be done*

You can move some of the text onto a new file with the MOVE command, close the file you are currently editing, and edit the new file.

## JOIN

The JOIN command allows you to remove page breaks that you created with the SPLIT command. SED joins the current page or the page you specify to the previous page.

JOIN [address]

The address can be:

- line 1 of the current page
- a single page argument

You can omit the address, but the current line *must be* line 1 of the current page. Thus, you can join the current page to the previous page in one of three ways:

- You can type JOIN 1
- You can type JOIN PAGE and the number of the current page
- You can make line 1 the current line and type JOIN without an address

If you omit the address and the current line is not line 1, SED displays an error message and does not remove the page break. It redisplay the command which you can modify by typing 1 after the JOIN command.

When SED has joined a page, it updates the line numbers and page numbers and sets the current line to what was line 1 of the joined page.

To join page 2 to page 1 in your file, you have three options. You can

- Type JOIN 1 }
- Type JOIN PAGE 2 }
- Press the uparrow key three times to set the current line to line 1, and then type JOIN }

Because it's the fastest and easiest option, type

JOIN 1 }

JOIN 1

\*\_ page 1 line 6

1 Here is the first line of text.  
 2 This is the second line.  
 3 This line had three mistakes.  
 4 Now you have a fourth line.  
 5 We will move this fifth line.  
 6 This is text for the blank page.  
 7 Later, we will cut this line.  
 8 Then, we will paste these  
 9 two lines.

What was line 1 on page 2 is now line 6 on page 1. The current line is line 6.

If you want to remove the page break between pages in which the current line is not located, you can type the JOIN command with a page argument. The page argument is the second of the two pages you want to join. For example, if the current page is page 8 and you want to remove the page break between pages 5 and 6, type

\*JOIN PAGE 6 }

This command joins page 6 to the previous page. SED updates the line numbers on page 5 and the subsequent page numbers. SED sets the current line to what was line 1 on page 6.

## Cutting a Line and Merging Lines (CUT and PASTE)

### CUT

The CUT command allows you to cut a line into two lines. You may want to cut a line for the sake of neatness or to add text to that line. You may have to cut a line if the line length exceeds the line length of your printer. (An alternative is to use the /FOLDLONGLINES switch with the CLI QPRINT command. Consult your *Command Line Interpreter (CLI) (AOS and AOS/VS) User's Manual* for more information on this command and switch.)

There are two ways to cut a line. You can use the CUT WHILE EDITING function key or type the CUT command with an address and column number argument.

CUT address column

address can be

- a line number within the current page
- the keywords CURRENT, PREVIOUS, NEXT, LAST

A column number is a number from 0 (the beginning of the line) to 255 (the maximum line length). You specify the column number after which you want SED to cut the line. For example, in this line of text:

```
22 Now is the time to cut.
```

The *N* is in column 1, the *o* is in column 2, the *w* is in column 3, etc. To cut this line so that the *t* in *time* is the first character on the next line, you would type

```
*CUT 22 11 ↓  
722 Now is the  
23 time to cut.
```

When SED has completed the cut, it sets the current line to the line that was cut. The cursor remains at the command line.

Because it isn't easy to determine the column number where you want to cut a line, you will find it more convenient to use the CUT WHILE EDITING function key. You must be in keyboard input mode to cut a line with the function key. You position the cursor at the point where you want to cut the line and then press the CUT WHILE EDITING function key.

Let's cut line 7 on page 1 of your text. First, make line 7 the current line by pressing the downarrow key.

To enter keyboard input mode, press the MODIFY CURRENT function key. To position the cursor at the *c* in *cut*, press CTRL-F three times.

Now, press the CUT WHILE EDITING function key.

*cut\_line current at cursor*

\*

*page 1 line 7*

```
1 Here is the first line of text.  
2 This is the second line.  
3 This line had three mistakes.  
4 Now you have a fourth line.  
5 We will move this fifth line.  
6 This is text for the blank page.  
7 Later, we will _  
8 cut this line.  
9 Then, we will paste these  
10 two lines.
```

The cursor remains where you positioned it for the cut. SED rennumbers the lines, beginning with the line following the line that was cut.

### PASTE

The PASTE command allows you to merge a line with the following line or to merge a range of lines into one line.

There are two ways to paste lines. You can use the PASTE WHILE EDITING function key, or you can type the PASTE command followed by an address or a range argument.

```
PASTE [address  
range]
```

address can be

- a line number within the current page
- the keywords CURRENT, PREVIOUS, NEXT, LAST

range can be

- line numbers within the current page
- the keywords CURRENT, PREVIOUS, NEXT, LAST, ALL, REMAINING, FOR

If you want to paste two lines, specify the number of the first of the two lines. For example, to paste the following two lines

```
34 This is a short line.  
35 And so is this.
```

you would type

```
*PASTE 34 ↓
```

```
34 This is a short line.And so is this.
```

If you want to paste a range of lines, type the PASTE command followed by the first and last line numbers of the lines you want to merge. For example, to merge the following four lines

1 Here is  
2 an  
3 example  
4 of pasting.

you would type

**\*PASTE 1 4**

1 Here is an example of pasting.

Note that SED does not insert a space between the pasted lines. You must modify the line to insert and/or delete spaces.

Using the PASTE WHILE EDITING function key can be easier and faster. If you are in keyboard input mode and you position the cursor where you want the first character of the second of the two lines, you do not have to modify the line.

To see how this works, let's paste lines 9 and 10 of your text. Position the cursor to the end of line 9 by pressing the downarrow key twice and then CTRL-A. To leave a space between the last word on line 9 and the first word on line 10, press the space bar once. Now press the 6PASTE WHILE EDITING6 function key.

*paste current next at cursor*

\*

*page 1 line 9*

1 Here is the first line of text.  
2 This is the second line.  
3 This line had three mistakes.  
4 Now you have a fourth line.  
5 We will move this fifth line.  
6 This is text for the blank page.  
7 Later, we will  
8 cut this line.  
9 Then, we will paste these two lines.

The cursor remains where you positioned it. SED updates the line numbers, beginning with the line following the pasted line.

The next command we describe is the DO command. Since you must type this command on the command line, return to command mode by pressing the ESC key.

ESC

## Executing CLI Commands (DO and CLI)

The SED commands DO and CLI allow you to execute CLI commands during an editing session. They temporarily suspend the editing session to execute one or more CLI commands. The DO command allows you to execute one CLI command at a time, while SED's CLI command allows you to execute several CLI commands in succession.

**NOTE:** To use the CLI and DO commands, your user profile must allow you to create at least two son processes. If you get an error message from the DO or CLI command, see your system manager about your profile.

### DO

The DO command allows you to leave the SED utility temporarily to execute a CLI command. You do not have to terminate the editing session; you type the DO command with a CLI command as an argument.

DO CLI-command

DO invokes a subordinate CLI process to execute the command, then terminates this CLI process. The subordinate CLI process has the same environment as the SED process.

If the information displayed on your screen is more than 21 lines, the command line scrolls off the top of your screen and then returns at the end of the display. The CLI display remains on your screen until you press NEW LINE, ESC, ERASE PAGE, or the VIEW function key to refresh the screen with text. When your text display returns to your screen, the cursor will be at the command line, and you can continue editing.

Three of the CLI commands that you will find useful are FILESTATUS, TYPE, and XEQ SED. We will give examples of these three commands.

### DO FILESTATUS

The DO FILESTATUS command without an argument displays the pathname of your working directory and the names of all files in it. This display includes the file you're currently editing and its auxiliary and temporary files. You may want to use the DO FILESTATUS command to check the name of a file you want to append or insert.

\*DO FILESTATUS )

DO FILESTATUS

\* \_

page 1 line 9

DIRECTORY :UDD:USERNAME

SAMPLE	SAMPLE.ED	MEMOS
MO_REPORT	SAMPLE.SC	PASCAL
SAMPLE.S2	.....	.....

If you have subordinate directories, they will appear as files in this directory.

With the DO FILESTATUS command, you can use CLI filename templates to list files with specific names. For example, the command DO FILESTATUS +.BU lists only those files that end in .BU. You can also use command switches with the DO FILESTATUS command. The /SORT switch, for example, lists the filenames alphabetically. For more information on filename templates and command switches, see the *Command Line Interpreter (CLI) (AOS and AOS/VS) User's Manual*.

### DO TYPE

The DO TYPE command followed by a pathname displays the text of another file. This command is useful if you want to check the contents of a file before appending or inserting it in the file you are currently editing.

\*DO TYPE :UDD:KW:FORMS:TABLE1 )

### DO XEQ SED

With the DO command, you can also instruct the CLI to create another SED process. This allows you to edit another file in your working directory. When you type BYE to terminate the second editing session, you return to the first SED session instead of the CLI. For example:

```
*DO XEQ SED ANOTHER_FILE )
Do you want ANOTHER_FILE to be created? Y )
*APPEND )
/ There will be only 1 line in this file. )
```



```
*BYE )
Output file - :UDD:USERNAME:ANOTHER_FILE
*_
```

To refresh your screen after typing the DO command, press NEW LINE, ESC, ERASE PAGE, or the VIEW function key.

The DO command actually creates a CLI process *subordinate* to the SED process; it doesn't return you to the parent process. So when you issue one of the following CLI commands via the SED DO command, the command affects only the subordinate CLI process, not the process that created SED.

ACL	CHARACTERISTICS
CLASS1	CLASS2
DATAFILE	DEFACL
DIRECTORY	LEVEL
LISTFILE	LOGFILE
PERMANENCE	PROMPT
SCREENEDIT	SEARCHLIST
SQUEEZE	STRING
SUPERPROCESS	SUPERUSER
TRACE (AOS/VS only)	

For example, if you want to change the listfile for SED, you must exit from SED, change the listfile using a CLI command, and then reexecute SED as follows:

```
*BYE )
) LISTFILE FOO )
) XEQ SED FILE3 )
*
```

### CLI

SED's CLI command allows you to temporarily suspend your editing session to execute as many CLI commands as you wish. You do not have to terminate the editing session. You can press the CLI function key or type the CLI command on the command line.

### CLI

Like DO, CLI creates a subordinate CLI process, but it does not take an argument. Instead, when you enter SED's CLI command, it displays two lines similar to the following:

```
AOS/VS CLI REV 01.50 6-JUL-82 15:02:01
Son of Sed) _
```

The second line contains the prompt (*Son of SED* and a right parenthesis) followed by the cursor. You can then type any CLI command you want to execute. After executing each CLI command, the prompt returns. When you type BYE to terminate the process, you resume your editing session. For example:

```
*CLI )
```



AOS/VS CLI REV 01.50 10-JUL-82 12:36:17  
Son of Sed) FILESTATUS/SORT )

DIRECTORY :UDD:USERNAME

COBOL	DATASHEET	FORMS
KEYS	PROJECT	SAMPLE
SAMPLE.ED	SAMPLE.S2	SAMPLE.SC
.....	.....	.....

Son of Sed) TYPE KEYS )

SET FUNCTION CTRL 9 "SAVE";  
SET FUNCTION CTRL 10 "SET LINE\_NUMBER\_  
DISPLAY";  
SET FUNCTION CTRL 11 "CLEAR LINE\_NUM-  
BER\_DISPLAY"

Son of Sed) BYE )

AOS/VS CLI TERMINATING 10-JUL-82  
12:38:07

\*\_

During this session with the subordinate CLI process, it displayed an alphabetical listing of the files in the working directory and the contents of the file KEYS. The BYE command terminated the subordinate CLI process and returned the cursor to the SED command line.

Some CLI commands change the environment of a process and its subordinate processes, but not its parent process. When you execute one of the following CLI commands via the SED DO command, it affects only the subordinate process, not the process that created SED.

ACL	CHARACTERISTICS
CLASS1	CLASS2
DATAFILE	DEFACL
DIRECTORY	LEVEL
LISTFILE	LOGFILE
PERMANENCE	PROMPT
SCREENEDIT	SEARCHLIST
SQUEEZE	STRING
SUPERPROCESS	SUPERUSER
TRACE (AOS/VS only)	

## Changing Your Working Directory (DIRECTORY)

SED's DIRECTORY command allows you to change your working directory without terminating the editing session. You might do this for two reasons:

- If you want to insert or append files from another directory, you often have to type a long pathname. Using SED's DIRECTORY command, you need to type only a simple filename.

- You can also use the DIRECTORY command to change your working directory so that you can finish editing a file in one directory and then begin editing another file in a second directory. (See the BYE CONTINUE sequence at the end of Chapter 2.)

DIRECTORY pathname

Let's say you want to insert text from a file in a directory whose pathname is very long (:UDD:USERNAME:MEMOS:PROBLEMS) and you are working in your user directory. You could type

```
*INSERT FROM :UDD:USERNAME:MEMOS:  
PROBLEMS:TAXES 3 45 )
```

or you could change your working directory and then insert the range of lines from the file

```
*DIRECTORY :UDD:USERNAME:MEMOS:  
PROBLEMS; INSERT FROM TAXES 3 45 )
```

Obviously, if you want to insert or append from only one or two files, you hardly save any time. But if you want to edit a file in one directory and insert or append several files from another directory, SED's DIRECTORY command does save time and effort.

When you terminate the editing session, you will be in the directory in which you began the session.

Don't confuse the SED DIRECTORY command with the CLI command. The SED DIRECTORY command changes your working directory during a SED session only.

## Executing a File (EXECUTE)

You can use SED's EXECUTE command during an editing session to execute a file that contains SED commands. If there are a number of SED commands you frequently want to execute together, create a separate file containing them. Any time during an editing session, you can execute the commands all at once by typing EXECUTE and the filename as an argument. This saves you time because you do not have to type out all of the commands.

EXECUTE pathname

Let's say you have created a file named START that contains the following SED commands: SET DISPLAY\_MODE 2; CLEAR TYPED\_MODE; CLEAR PRELOAD\_OLD\_LINE. Then, when you want to execute these commands while editing another file, you type

```
*EXECUTE START )
```

SED executes all of the commands contained in the file START; the DISPLAY\_MODE setting is 2 and

TYPED\_MODE and PRELOAD\_OLD\_LINE are OFF. SED displays all of the commands on the command line as it executes them.

Don't confuse the SED EXECUTE command with the CLI EXECUTE command. The SED EXECUTE command executes a file of SED commands only.

## Printing Text on @LIST (PRINT)

The PRINT command allows you to duplicate up to one page of text onto the generic listfile @LIST. A *generic listfile* is a file you designate with the CLI LISTFILE command. It can be a file like a text file, or it can be a line printer. Users often designate a line printer as their listfile so that they can print the line or range of text they specify.

NOTE: You must set the listfile from the CLI *before* you execute SED. You cannot set the listfile with SED'S DO or CLI commands. You must have Read and Write access to the file to which you set the listfile, including the line printer. See your system manager for information on the ACL for the line printer.

PRINT [*range*]

*range* can be

- line numbers within the current page
- a single page argument
- the keywords CURRENT, PREVIOUS, NEXT, LAST, ALL, REMAINING, FOR

If you omit the range, SED prints the current line on @LIST. The PRINT command does not change the current line.

## Setting the Listfile

To set the listfile to the line printer, type the LISTFILE command and @LPT next to the CLI prompt

```
) LISTFILE @LPT )
```

To set the listfile to a text file, type the LISTFILE command and the pathname of the file. For example:

```
) LISTFILE :UDD:JAN:SUMMARY )
```

This command sets the listfile to the file SUMMARY. If SUMMARY does not exist, LISTFILE creates it. If it does exist, LISTFILE appends the subsequent text you specify with SED's PRINT command. See the *Command Line Interpreter (CLI) (AOS and AOS/VS) User's Manual* for more information on the LISTFILE command.

## Printing a Range of Lines

To print a range of lines in the current page on @LIST, type the PRINT command and a range argument.

Examples:

```
*PRINT ALL )
```

Print all lines on the current page.

```
*PRINT REMAINING )
```

Print all lines from the line following the current line to the last line on the current page.

```
*PRINT 27 85 )
```

Print lines 27 through 85 on the current page.

## Printing a Single Page

To print all lines on any page in your file, type the PRINT command and a single page argument.

Examples:

```
*PRINT PAGE LAST )
```

Print all lines on the last page of the file.

```
*PRINT PAGE NEXT )
```

Print all lines on the page after the current page.

```
*PRINT PAGE 2 )
```

Print all lines on page 2 of the file.

Unless you have cleared your text display of line numbers, SED prints each line with its number and a tab space. For example, if SED is displaying line numbers and you type

```
*PRINT 1 3 )
```

the printout would look like this:

```
1 This is line 1.
```

```
2 This is line 2.
```

```
3 This is line 3.
```

SED's PRINT command can print only a single page of text at a time. If you want to print out all the pages in the file, return to the CLI and type the CLI QPRINT command. (The CLI QPRINT command does not print line numbers.)

You can type the PRINT command any number of times during an editing session. If the listfile is set to a text file, SED inserts a blank line before appending the text you specify with each subsequent PRINT command.

## Getting Spelling Assistance (SPELL)

The SPELL command allows you to check the spelling of a word by specifying its beginning letter or letters. SED searches its dictionary for all words that start with the letter(s) you specify, and lists those words on your terminal screen.

**SPELL string**

**string** is the beginning of the word whose spelling you want to check. You can type **string** in upper- or lowercase.

SED displays the words from its dictionary that begin with **string**. SED lists one word per line, in lowercase and in alphabetical order. After the last word in the list, SED displays **-NO MORE-**. If there are more words than your screen display holds, the word list will scroll on your screen. Use CTRL-S to freeze the display and CTRL-Q to continue the display. You can interrupt the display of words by pressing CTRL-C CTRL-A; SED stops listing words and displays **\*INTERRUPTED\***, following the last word on the screen. If SED can't find any words that begin with the letters you typed, it displays **-NO MATCHES-**.

Let's say you want to check the spelling of *surgically*. You know it begins with *surg*, so you type

```
*SPELL surg )
```

```
SPELL surg
```

```
*_
```

```
page 1 line 9
```

```
surge  
surged  
surgeon  
surgeoncy  
surgeons  
surgery  
surgical  
surgically  
surging  
- No More -
```

SED displays all the words in its dictionary that begin with *surg*. *surgically* is next to the last word in the list. When you have finished checking the list, press one of the keys that refresh the screen.

### Adding to SED's Dictionary

You or your system manager can add words to SED's dictionary, which is a file named SED.DICTIONARY. You may want to include nonstandard words that are commonly used by people on your system.

The words you add to SED.DICTIONARY must be placed in alphabetical order within the file. Type each

word in lowercase on a line by itself. After typing each word, press the NEW LINE key.

SED.DICTIONARY may be a link to another file.

## Terminating This Editing Session (BYE and ABANDON)

### BYE

In Chapter 2, you learned how to terminate a SED editing session with the BYE command. Let's terminate this session, asking SED to save the original copy of SAMPLE as a backup file. You can then compare the two versions of the file to see the changes you made during this second editing session. Type

```
BYE )
```

```
Do you want to save the original file as a backup  
file? Y )
```

```
Output file - :UDD:USERNAME:SAMPLE
```

```
Backup file - :UDD:USERNAME:SAMPLE.BU
```

```
)
```

You are now back in the CLI where you can use the QPRINT command to print SAMPLE and SAMPLE.BU.

### ABANDON

If you make an error in your edit file that you cannot UNDO or modify, you can abort the entire editing session by typing the ABANDON command. SED returns you to the CLI and ignores any changes you made to the file during the session.

Because ABANDON does not update the file (and thus throws away all editing changes), SED requires that you confirm this command to make sure that you did not accidentally type ABANDON. For example:

```
*ABANDON )
```

```
SED clears the text from your screen and displays the  
following question below the command line:
```

```
Do you really want to discard your edits of this file?
```

(If you haven't made any changes during the editing session, SED doesn't ask this question. It terminates the editing session and returns you to the CLI.)

To abort the editing session, type Y). Then the CLI prompt reappears.

```
*ABANDON )
```

```
Do you really want to discard your edits of this file? Y )  
)
```

If the file existed before the editing session, it contains the text that was in it before you began the session in which you typed ABANDON. If you created the file at this session, ABANDON removes the appended text, but leaves the empty file in the directory.

To tell SED to ignore the ABANDON command, type **N** or just **).** SED returns the cursor to the command line and redisplay the text that was on your screen before you typed ABANDON. You can continue editing the file as if you had never typed ABANDON.

```
*ABANDON )
```

```
Do you really want to discard your edits of this file? N )
```

```
*_
```

Although ABANDON tells SED to ignore all changes you made to *this* file during the editing session, it does not affect other files. If, during the editing session, you created another file with the MOVE, DUPLICATE, DO, or CLI commands, abandoning your editing session will not affect these files. Any changes made to them will still exist. The same is true if these files already existed and you used the MOVE or DUPLICATE command to append text to them.

## Continuing with SED

You can abort the current editing session and begin editing another file by typing ABANDON with the argument CONTINUE on the same line.

```
*ABANDON CONTINUE )
```

```
Do you really want to discard your edits of this file?
```

If you answer **Y** to SED's confirmation question, SED aborts the current session and asks for the name of the next file to edit. If you answer **N**, SED returns the cursor to the command line and ignores the argument CONTINUE.

## Once Again, Congratulations!

You have just completed your second editing session with SED. In this chapter, you learned about SED's advanced commands and features. You read about the switches you can use with the XEQ SED command. You learned about SED's display settings, how to change them, and how to set and clear function keys. With the SPLIT and JOIN commands, you created and removed page breaks in your file. With the CUT and PASTE commands, you cut a line into two lines and merged lines into one line. You learned that, *during* an editing session, SED allows you to execute CLI commands, change your working directory, execute a file of SED commands, and duplicate a range of text onto @LIST.

That's a lot of learning! You may not feel comfortable with all of SED's commands and features yet. But with each SED session, you will remember more of the details.

During future editing sessions with SED, you can consult Chapter 4, SED Commands and Arguments, for a summary of each SED command and argument.

End of Chapter

# Chapter 4

## SED Commands and Arguments

You'll find SED commands listed two ways in this chapter: by category and alphabetically. Table 4-1 groups commands by category or function, for example, adding text and deleting text. The rest of the chapter is a reference section that lists SED commands and arguments alphabetically. The description for each command or argument gives the format, rules for using the command or argument, and examples.

Listed first, in both Table 4-1 and the reference section, is the CLI EXECUTE SED command.

**Table 4-1. SED Command Summary**

Category	Command or Function Key	Description
Opening a File	EXECUTE SED <i>[SWITCHES]</i> <i>[pathname]</i>	Opens a file for editing and controls how you execute SED.
Adding Text	APPEND <i>[FROM pathname]</i> <i>[range]</i>  APPEND function key  DUPLICATE <i>[range]</i> destination  INSERT <i>[address]</i> <i>[FROM pathname]</i> <i>[range]</i>  INSERT CURRENT function key  UNDO  UNDO function key	Allows you to add text to the end of the page by typing it at your terminal keyboard, or by specifying a source file.  Allows you to add text to the end of the page by typing it at your terminal keyboard.  Copies text from one location in the page to another, or onto another file.  Inserts text before a location in the page. Text comes from the terminal keyboard or from a source file you name.  Inserts text before the current line. You type the text at your terminal keyboard.  Restores the text most recently deleted with the DELETE command.  Same as the UNDO command.
Changing Text	MODIFY <i>[range]</i>  MODIFY CURRENT function key  REPLACE <i>[range]</i>  SUBSTITUTE "searchstring 1" <i>[FOR]</i> "searchstring 2" <i>[range]</i>  SPLIT <i>[address]</i>  CUT address column  CUT WHILE EDITING function key  PASTE range <i>[OR]</i> address  PASTE WHILE EDITING function key	Allows you to revise lines of text that you retype, using control characters and screen position function keys.  Allows you to revise lines of text that you retype, beginning with the current line.  Deletes text and replaces it with text you type in at your terminal keyboard.  Substitutes a word or phrase for another word or phrase throughout a range of text.  Sets a page break in the file.  Splits a line into two lines.  Splits a line into two lines at the current cursor position.  Merges a range of lines into a single line.  Merges the current line with the following line at the current cursor position.
Deleting Text	DELETE <i>[range]</i>  DELETE CURRENT function key  MOVE <i>[range]</i> destination  JOIN <i>[address]</i>	Removes a range of text from the file.  Removes the current line from the file.  Moves text from one location in the page, or onto another file.  Removes a page break in the file.

(continues)

**Table 4-1. SED Command Summary**

Category	Command or Function Key	Description
Listings	<p>LIST <i>[range]</i></p> <p>VIEW</p> <p>VIEW function key</p> <p>PRINT <i>[range]</i></p>	<p>Lists a range of text on the terminal screen without changing the current line.</p> <p>Lists a range of lines around the current line.</p> <p>Same as the VIEW command.</p> <p>Prints a range of text on @LIST (usually the line printer).</p>
Display Information	<p>DISPLAY</p> <p>DISPLAY function key</p> <p>DISPLAY FUNCTION</p> <p>HELP <i>[command keyword]</i></p>	<p>Displays file status information: edit file-name, current page and line, last page, last line on the current page, each display setting.</p> <p>Same as the DISPLAY command.</p> <p>Displays the current command definitions for all function keys, except those that cannot be cleared.</p> <p>Displays information about commands and keywords on the terminal screen.</p>
Positioning	<p>POSITION address</p> <p>GO TO PREVIOUS POSITION function key</p> <p>UP 4 LINES function key</p> <p>UP 11 LINES function key</p> <p>DOWN 4 LINES function key</p> <p>DOWN 11 LINES function key</p> <p>TOP OF SCREEN function key</p> <p>MIDDLE OF SCREEN function key</p> <p>BOTTOM OF SCREEN function key</p> <p>UP ONE SCREEN function key</p> <p>DOWN ONE SCREEN function key</p>	<p>Changes the current line to an address in the file.</p> <p>Sets the current line to the previous current line. This key stores the previous 15 line positions.</p> <p>Sets the current line to the fourth line back.</p> <p>Sets the current line to the eleventh line back.</p> <p>Sets the current line to the fourth line ahead.</p> <p>Sets the current line to the eleventh line ahead.</p> <p>Sets the current line to the first line on the screen.</p> <p>Sets the current line to the middle line on the screen.</p> <p>Sets the current line to the last line on the screen.</p> <p>Displays the previous screen of text and sets the current line to the middle line on the screen.</p> <p>Displays the next screen of text and sets the current line to the middle line on the screen.</p>

(continued)

**Table 4-1. SED Command Summary**

Category	Command or Function Key	Description
Positioning (Cont.)	<p>FIRST LINE ON PAGE function key</p> <p>MIDDLE LINE ON PAGE function key</p> <p>LAST LINE ON PAGE function key</p> <p>POSITION PAGE PREVIOUS function key</p> <p>POSITION PAGE FIRST function key</p> <p>POSITION PAGE NEXT function key</p> <p>POSITION PAGE LAST function key</p> <p>FIND "searchstring" <i>[range]</i></p> <p>FIND function key</p> <p>BACKFIND "searchstring"</p> <p>BACKFIND function key</p>	<p>Sets the current line to the first line on the page.</p> <p>Sets the current line to the middle line on the page.</p> <p>Sets the current line to the last line on the page.</p> <p>Sets the current line to the first line on the previous page.</p> <p>Sets the current line to the first line on the first page.</p> <p>Sets the current line to the first line on the next page.</p> <p>Sets the current line to the first line on the last page.</p> <p>Finds a word or phrase in a range of text, beginning at any line and working forward.</p> <p>Finds the next occurrence of the searchstring you typed on the command line with either the FIND or BACKFIND command.</p> <p>Finds a word or phrase in a page of text, beginning at the line before the current line and working backward.</p> <p>Finds the next previous occurrence of the searchstring you typed on the command line with either the FIND or BACKFIND command.</p>
Exiting	<p>BYE <math>\begin{matrix} \text{YES} \\ \text{NO} \end{matrix}</math> <i>[CONTINUE]</i></p> <p>BYE YES CONTINUE function key</p> <p>ABANDON</p> <p>DO CLI-command</p> <p>CLI</p> <p>CLI function key</p>	<p>Concludes the editing session and updates the file with the changes. Optionally, you may create a backup file and then edit another file.</p> <p>Concludes the editing session, updates the file, creates a backup file, and prompts you for the next file to edit.</p> <p>Aborts the editing session without updating the file with new changes.</p> <p>Executes a CLI command and then returns you to the editing session.</p> <p>Allows you to suspend the editing session and then execute as many CLI commands as you wish before returning to the editing session.</p> <p>Same as the CLI command.</p>

(continued)



**Table 4-1. SED Command Summary**

Category	Command or Function Key	Description
Changing Display Settings	SET BLANK_MODE	Allows padding of lines with trailing blanks.
	CLEAR BLANK_MODE	Removes the ability to pad lines with trailing blanks.
	SET DISPLAY_MODE n	Sets the way SED displays the current line.
	SET ESCAPE_CHARACTER x	Sets a character that allows you to insert escape characters in your file.
	CLEAR ESCAPE_CHARACTER	Removes the definition of the character you set to insert escape characters in your file.
	SET ESCAPE_IS_VIEW	Sets the ESC key to refresh the screen when you press it on an empty (blank) command line.
	CLEAR ESCAPE_IS_VIEW	Clears the ESC key so that it does not refresh the screen when you press it on an empty (blank) command line.
	SET LINE_NUMBER_DISPLAY	Displays your text with line numbers.
	CLEAR LINE_NUMBER_DISPLAY	Clears your text display of line numbers.
	SET NEW_LINE_IS_VIEW	Sets the NEW LINE key to refresh the screen when you press it on an empty (blank) command line.
	CLEAR NEW_LINE_IS_VIEW	Clears the NEW LINE key so that it does not refresh the screen when you press it on an empty (blank) command line.
	SET PRELOAD_OLD_LINE	In keyboard input mode, SED saves the previous current line.
	CLEAR PRELOAD_OLD_LINE	In keyboard input mode, SED doesn't save the previous current line.
	SET SKIP_WHITE_SPACE	Causes the NEW LINE key to move the cursor to the first nonblank character on the next line of text.
	CLEAR SKIP_WHITE_SPACE	Causes the NEW LINE key to move the cursor to the leftmost position on the next line of text.
	SET TYPED_MODE	Causes the up- and downarrow keys to maintain their column position.
	CLEAR TYPED_MODE	Causes the up- and downarrow keys to move the cursor to the leftmost position.
	SET UPPERCASE_SHIFT_CHARACTER x	Converts keyboard input to lowercase, unless preceded by 6x6.
CLEAR UPPERCASE_SHIFT_CHARACTER	Removes the definition of the character you set to translate uppercase input to lowercase.	
SET VIEW_RANGE n	Sets the number of lines SED displays before and after the current line.	

(continued)

**Table 4-1. SED Command Summary**

<b>Category</b>	<b>Command or Function Key</b>	<b>Description</b>
Changing Function Keys	SET FUNCTION number "string" [;"string"]	Sets a function key to execute one or more SED commands.
	CLEAR FUNCTION number	Removes the definition of a function key.
Miscellaneous Commands	EXECUTE pathname	Executes SED commands from a file you created previously.
	DIRECTORY pathname	Changes the working directory to pathname.
	SAVE	Updates a copy of the file, including changes, without terminating the session.

(concluded)

---

## EXECUTE SED

Executes SED from the CLI.

---

### Command Format

EXECUTE SED *[/SWITCHES] [pathname]*

### Command Switches

<code>/ED=directory</code>	Stores the .ED file in the directory specified
<code>/NO_ED</code>	Executes SED without creating a .ED file
<code>/PROFILE=pathname</code>	Executes a file containing SED commands
<code>/WORK=directory</code>	Stores the temporary .SC file in the specified directory
<code>/NO_SCREEN</code>	SED does not update the screen during your editing session
<code>/NO_FORM_FEED</code>	SED removes all page breaks (form feeds) when you terminate the session

This command executes the SED utility from the CLI.

### Rules

You can EXECUTE SED with or without a pathname; pathname can be the name of an existing file or of a file you want to create. If you EXECUTE SED without a pathname argument, SED will ask for a filename to edit. If you provide a pathname argument, and SED has enough information to begin, it displays the \* prompt and lets you begin the editing session. If SED cannot find the file in the directory, SED asks if you want to create the file in your working directory. If you do, type Y). If you don't, perhaps because you want to look elsewhere for the file, or because you misspelled the name of the file, type N). SED asks if you want to start over again. You cannot edit a file that is in a directory to which you do not have Write access; SED won't let you.

The abbreviation for EXECUTE is XEQ or X.

### Examples

```
) XEQ SED )  
Name of file to edit: PROBLEMS )  
Do you want PROBLEMS to be created? Y )
```

```
view  
*_  
page 1 line 1
```

---

```
) X SED PROBLEMS )
```

```
view  
*_  
page 3 line 47
```

---

```
) X `SED/PROFILE=FOO/ED=EDITS )  
Name of file to edit:
```

---

## ABANDON

**Aborts the editing session.**

---

### Command Format

ABANDON

This command aborts the editing session without updating the file with changes.

### Rules

- If you've made any changes during the current editing session, ABANDON requires a confirmation before SED will abort the editing session. If you answer N) to the confirmation request, SED returns the cursor to the command line so that you can continue the editing session. If you answer Y), SED restores the edit file to its original form. If you created the file during the editing session, ABANDON removes its contents, but leaves an empty file.
- Any files edited with the DO or CLI commands during the current editing session retain any changes made to them. ABANDON affects only the current edit file. Any files affected by MOVE or DUPLICATE, other than the current edit file, retain the changes made to them.
- You can abort the current editing session and begin editing another file by typing ABANDON CONTINUE. When you answer Y) to SED's confirmation question, SED discards the edits of the current file and asks you for the name of the next file to edit.

### Examples

```
*ABANDON )
Do you really want to discard your edits of this file? Y )
)

*ABANDON )
Do you really want to discard your edits of this file? N )
*_

*ABANDON CONTINUE )
Do you really want to discard your edits of this file? Y)
Name of file to edit:
```

---

## address

**Specifies a line or page.**

---

### Argument Format

$$\left\{ \begin{array}{l} \text{PAGE} \\ \text{[LINE]} \end{array} \right\} \left[ \begin{array}{l} [+ ] \\ [- ] \end{array} \right] \begin{array}{l} \text{number} \\ \text{CURRENT} \\ \text{PREVIOUS} \\ \text{NEXT} \\ \text{LAST} \end{array}$$

This argument specifies a particular line or page in the file.

### Rules

- If the address is a line in the file, you may omit the word LINE. If the address is a page in the file, you must specify the PAGE. The modifier following LINE or PAGE can be a number, a number preceded by a plus sign, a number preceded by a minus sign, CURRENT, PREVIOUS, NEXT, or LAST. If you omit a modifier, SED assumes CURRENT.
- If you use a + or - sign with *number*, it means the address is relative to the current page or line. If you use *number* without a + or - sign, the address is the page or line specified by *number*.

### Examples

```
*POSITION 1 )
*JOIN PAGE 2 )
*DELETE LAST )
```

---

## APPEND

Appends text to the current page.

---

### Function Key Use

The APPEND function key allows you to add text typed at your terminal keyboard to the end of the current page.

### Command Format

APPEND [*FROM pathname*] [*range*]

This command adds text to the end of the current page. You can type the text in at your terminal keyboard or tell SED to add text from a different file. You may specify a complete file or a *range* of lines from a file.

### Rules

- Use APPEND to add text to a newly created file or to the end of the current page in an existing file.
- If you type APPEND with no argument or press the APPEND function key, the text must come from your terminal keyboard.
- SED displays line numbers before each line of new text. If you do not want line numbers displayed, you must use the CLEAR LINE\_NUMBER\_DISPLAY command.
- To return to command mode after appending text from your terminal keyboard, press the ESC key.
- If you specify a *pathname* argument with APPEND, the text comes from the file specified in *pathname*.
- If the file from which you are appending is a page long, you may APPEND a *range* of lines.
- You may use the DIRECTORY command to change your working directory during a SED session. This permits you to append files from another directory, specifying a filename instead of a long *pathname* each time.
- The current line is set to the last line of text added.

### Examples

\*APPEND ↵

```
1 At the beginning of this ↵  
2 session, you learned... ↵
```

```
·  
·  
·
```

ESC

\*APPEND FROM TABLE1 ↵

\*APPEND FROM SOURCE 13 34 ↵

---

## BACKFIND

**Finds a text string before current line.**

---

### Function Key Use

The BACKFIND function key locates the next previous occurrence of the last searchstring typed on the command line with either the BACKFIND or the FIND command.

### Command Format

BACKFIND "searchstring"

This command finds a word or phrase in a page of text beginning at the line before the current line and working backwards in the file.

### Rules

- You must enclose searchstring in quotation marks (" ") or apostrophes ( ' ') if it contains spaces or symbols other than a dollar sign or underscore.
- A searchstring enclosed in quotation marks or apostrophes causes a case-sensitive search. If you don't enclose the searchstring, SED finds the string that matches it, regardless of case.
- The search begins at the line before the current line and works backwards in the current page. You can BACKFIND on the current page only.
- SED sets the current line to the line containing the first previous occurrence of searchstring.
- If you want to keep looking backwards, press the BACKFIND function key since it stores the searchstring.
- If SED cannot find the searchstring, it displays an error message and does not change the current line.

### Examples

```
*BACKFIND ANIMALCULES )
```

```
345 He was perturbed to discover animalcules in the water.
```

```
*BACKFIND "hereditary" )  
(Initial position is line 500.)
```

```
457 was the disease hereditary or was it caused  
458 by micro-organisms? Hereditary theorists  
459 felt they were right, of course.  
(SED sets the current line to line 457.)
```

---

## BYE

**Terminates editing normally.**

---

### Function Key Use

The BYE YES CONTINUE function key terminates the editing session, updates the file, creates a backup file, and prompts you for the name of the next file to edit.

### Command Format

BYE

This command terminates the editing session, updates the file with changes, and returns you to the CLI.

### Rules

- If you created the file during this session, SED displays the output file name followed by the CLI prompt on the next line.
- If the file existed before this editing session and you made no changes to it during the session, SED displays no message. The CLI prompt returns.
- If the file existed before this session and you changed the file during the session, SED asks if you want to create a backup of the original file. If you answer NO ), SED displays the output file message and the CLI prompt returns. If you answer YES ), SED displays the output file message and a backup file message and the CLI prompt returns. SED names the backup file filename.BU. If a backup file already exists for this file, SED deletes it and creates a new one.
- If you want to continue editing with SED, you can save some time by typing BYE, either YES or NO, and CONTINUE all on the same line or by pressing the BYE YES CONTINUE function key. If you do, SED asks for the name of the next file you want to edit.
- If you have made the edit file or the backup file permanent with the CLI PERMANENCE command, you must type the argument PERM\_OFF on the same line with the BYE command.

### Examples

```
*BYE )
```

```
Do you want to save the original file as a backup file? Y )
```

```
Output file - :UDD:BARRY:MO_REPORT  
Backup file - :UDD:BARRY:MO_REPORT.BU  
)
```

```
*BYE NO CONTINUE )
```

```
Output file - :UDD:JOHN:EXAMPLE
```

Name of file to edit:

---

## CLEAR

**Clears display settings and function keys.**

---

### Command Format

CLEAR {  
BLANK\_MODE  
ESCAPE\_CHARACTER  
ESCAPE\_IS\_VIEW  
LINE\_NUMBER\_DISPLAY  
NEW\_LINE\_IS\_VIEW  
PRELOAD\_OLD\_LINE  
SKIP\_WHITE\_SPACE  
TYPER\_MODE  
UPPERCASE\_SHIFT\_CHARACTER  
FUNCTION number

This command changes display settings from ON to OFF and removes command definitions from function keys.

### Rules

- CLEAR BLANK\_MODE strips any trailing blanks from a line when you press the NEW LINE key during editing.
- CLEAR ESCAPE\_CHARACTER removes the character that you SET to insert escape characters in your file.
- CLEAR ESCAPE\_IS\_VIEW causes ESC typed on an empty command line to have no affect on your text display.
- CLEAR LINE\_NUMBER\_DISPLAY clears the text display of line numbers.
- CLEAR NEW\_LINE\_IS\_VIEW causes NEW LINE typed on an empty command line to have no affect on your text display.
- CLEAR PRELOAD\_OLD\_LINE instructs SED not to save the previous line of text. This display setting affects the action of CTRL-A and the rightarrow key. For details, see the section on Display Settings in Chapter 3.
- CLEAR SKIP\_WHITE\_SPACE causes the cursor to move to the first position on the next line when you press the NEW LINE key.
- CLEAR TYPER\_MODE causes the uparrow and downarrow keys to position the cursor at the beginning of the previous or next line of text.

- CLEAR UPPERCASE\_SHIFT\_CHARACTER clears the character you defined to translate uppercase input to lowercase. This display setting is for users of D1 terminals.
- CLEAR FUNCTION followed by the number of the function key removes the command definition that SED predefined or that you established with the SET command. The number can be a number or a number preceded by SHIFT, CTRL, or CTRL SHIFT. The following seven function keys are exceptions: UP ONE SCREEN, DOWN ONE SCREEN, GO TO PREVIOUS POSITION, TOP OF SCREEN, BOTTOM OF SCREEN, and MIDDLE OF SCREEN. When you clear these function keys, they revert to the original definitions that SED established.

### Examples

- \*CLEAR BLANK\_MODE )
- \*CLEAR TYPER\_MODE; CLEAR  
PRELOAD\_OLD\_LINE )
- \*CLEAR FUNCTION 7 )
- \*CLEAR FUNCTION CTRL 9 )

---

## CLI

**Executes CLI commands.**

---

### Function Key Use

The CLI function key temporarily suspends the editing session, allowing you to execute as many CLI commands as you wish.

### Command Format

CLI

This command performs the same action as the CLI function key.

### Rules

- This command clears your screen of text and displays a prompt [*Son of Sed*] after which you can type any CLI command. After execution of each CLI command, the prompt returns until you type BYE to terminate the process. The cursor then returns to the SED command line where you can continue editing.
- In order to use the SED CLI command, your user profile must allow you to create at least two son processes.
- The following CLI commands affect only the CLI process you created with SED's CLI command. They have no affect on the CLI process that created SED.

ACL	CHARACTERISTICS
CLASS1	CLASS2
DATAFILE	DEFACL
DIRECTORY	LEVEL
LISTFILE	LOGFILE
PERMANENCE	PROMPT
SCREENEDIT	SEARCHLIST
SQUEEZE	STRING
SUPERPROCESS	SUPERUSER
TRACE (AOS/VS only)	

### Example

Press the CLI function key.

```
AOS/VS CLI REV 01.50 10-JUN-82 16-45-57
Son of SED) FILESTATUS )
DIRECTORY :UDD:MYDIR
PROJECT FORMS PLANS2
PLANS1 SAMPLE FORMS.ED
Son of SED) QPRINT FORMS )
QUEUED, SEQ= 1576, QPRI= 127
Son of SED) BYE )
```

```
AOS/VS CLI TERMINATING 10-JUN-82 16-46-46
```

---

## CUT

**Splits a line into two lines.**

---

### Function Key Use

The CUT WHILE EDITING function key cuts a line at the current cursor position.

### Command Format

CUT address column

This command splits the line in the address at the column specified.

### Rules

- You can type the CUT command on the command line or press the CUT WHILE EDITING function key when you are appending, inserting, or modifying.
- If you type the CUT command on the command line, you must specify both address and column.
- The column must be a number between 0 (the beginning of the line) and the length of the line.
- In keyboard input mode (APPEND, INSERT, MODIFY, or REPLACE), you can cut a line by moving the cursor to the column where you want the line cut and pressing the CUT WHILE EDITING function key.

### Examples

```
120 Here is a line I want to cut.
```

```
*CUT 120 15 )
```

```
120 Here is a line
```

```
121 I want to cut.
```

```
386 This line contains the cursor.
```

Press the CUT WHILE EDITING function key.

```
386 This line contains _
```

```
387 the cursor.
```



---

## DELETE

**Deletes a line or lines of text.**

---

### Function Key Use

The DELETE CURRENT function key deletes the current line.

### Command Format

DELETE [*range*]

This command removes one or more lines of text from the file.

### Rules

- You can type the DELETE command on the command line or press the DELETE CURRENT function key.
- If you type DELETE on the command line and omit the *range*, SED deletes the current line.
- The *range* must be a line or lines on the current page or a single page argument. The DELETE command does not allow you to delete more than one page at a time.
- SED sets the current line to the line immediately after the last line deleted. If the last line deleted is also the last line on the page, SED sets the current line to the line immediately before the first line deleted. If the deleted text is an entire page, SED sets the current line to the last line of the page immediately before the page deleted.
- You can restore the most recently deleted text with the UNDO command.

### Examples

```
*DELETE 1 6 ;  
*DELETE PAGE 5 ;  
*DELETE REMAINING ;
```

---

## destination

**Specifies a destination address or file.**

---

### Argument Format

BEFORE address  
AFTER address  
ONTO pathname

This argument specifies a new location for a block of text.

### Rules

- If the destination is BEFORE address or AFTER address, the address must be in the current page. If the destination is BEFORE address, SED places the text before the specified address. If the destination is AFTER address, SED places the text after the specified address.
- If the destination is a file specified by pathname, SED appends the text to the current contents of the file. If the file does not already exist, SED creates it and displays a message.

### Examples

```
*MOVE 16 BEFORE 85 ;  
*DUPLICATE 11 16 AFTER LAST ;  
*DUPLICATE 1 25 ONTO SUBDIR:YOURFILE ;
```

---

## DIRECTORY

**Changes the working directory during a SED session.**

---

### Command Format

DIRECTORY pathname

This command changes your working directory during a SED session. It does not change the directory in which the file you are editing was created.

### Rules

- You must have read and execute access to the subdirectory in the pathname.
- Use this command to save typing a long pathname when you want to append or insert files from another directory or when you want to close the file you are currently editing and begin editing a file in another directory.
- Do not confuse this DIRECTORY command with the CLI DIRECTORY command.

### Examples

```
*INSERT FROM :UDD:WIN:SPECS:NEWWORK:PLANS  
3 54 }
```

```
*APPEND FROM :UDD:WIN:SPECS:NEWWORK:  
PROJECT }
```

OR

```
*DIRECTORY :UDD:WIN:SPECS:NEWWORK }
```

```
*INSERT FROM PLANS 3 54 }
```

```
*APPEND FROM PROJECT }
```

---

## DISPLAY

**Displays statistics on the current file.**

---

### Function Key Use

The DISPLAY function key displays file status information: edit filename, current page and line, last page, last line on the current page, and the current setting for each display setting.

### Command Format

DISPLAY

This command performs the same action as the DISPLAY function key.

### Rule

To display file status information, you can either type the DISPLAY command on the command line or press the DISPLAY function key.

### Example

Press the DISPLAY function key.

*Edit file name - :UDD:USERNAME:NOTES*

*Current page: 5            Last page: 7  
Current line: 378        Last line: 433*

*View range: 10  
Display mode: 0*

*Typewriter mode: ON  
Blank mode: OFF  
Preload old line: ON  
Skip white space: ON*

*Upper mode: OFF  
Escape mode: OFF*

*New\_line is view: ON  
Escape is view: ON*

*Line numbers being displayed*

---

## DISPLAY FUNCTION

Displays the current function key settings.

---

### Command Format

DISPLAY FUNCTION

This command displays the current command definitions for most of the function keys.

### Rule

To display the function key definitions, type DISPLAY FUNCTION on the command line. (Using the SET command, you can define a function key to execute the DISPLAY FUNCTION command.) The DISPLAY FUNCTION command lists the command definitions for all of the function keys that are currently defined, except for: UP ONE SCREEN, DOWN ONE SCREEN, GO TO PREVIOUS POSITION, TOP OF SCREEN, BOTTOM OF SCREEN, MIDDLE OF SCREEN, and MIDDLE LINE ON PAGE.

### Example

```
*DISPLAY FUNCTION ↓
```

*Function keys currently defined*

	4	"position page previous"
	5	"position page next"
	6	"modify current"
	7	"find"
	8	"delete current"
	9	"position -4"
	10	"position +4"
	11	"cut_line current at cursor"
Shift	4	"position page 1"
Shift	5	"position page last"
Shift	6	"insert current"
Shift	7	"backfind"
Shift	8	"undo"
Shift	9	"position -11"
Shift	10	"position +11"
Shift	11	"paste current next at cursor"
Ctrl	1	"position 1"
Ctrl	2	"position last"
Ctrl	4	"display"
Ctrl	5	"view"
Ctrl	6	"append"
Ctrl	7	"cli"
Ctrl	8	"bye yes continue"

---

## DO

Executes a CLI command.

---

### Command Format

DO CLI-command

This command temporarily suspends the editing session to execute a CLI command. When execution is complete, the cursor returns to the SED command line.

### Rules

- The CLI-command argument can be any CLI command, including XEQ SED to another file. If the command is XEQ SED, when you type BYE to terminate that editing session, another SED prompt appears instead of the CLI prompt; this returns you to the first editing session.
- In order to use DO, your user profile must allow you to create at least two son processes.
- The CLI commands affect only the CLI process created to execute the CLI command you type after DO. These commands have no affect on the CLI process that created SED.

ACL	CHARACTERISTICS
CLASS1	CLASS2
DATAFILE	DEFACL
DIRECTORY	LEVEL
LISTFILE	LOGFILE
PERMANENCE	PROMPT
SCREENEDIT	SEARCHLIST
SQUEEZE	STRING
SUPERPROCESS	SUPERUSER
TRACE (AOS/VS only)	

### Examples

```
*DO FILESTATUS ↓  
DIRECTORY :UDD:MYDIR  
  ALPHA GAMMA EXAMPLE3.ED  
  EXAMPLE.SC EXAMPLE3 BETA  
  EXAMPLE.S2  
*DO XEQ SED GAMMA ↓
```

---

## DUPLICATE

**Copies text to the current page or another file.**

---

### Command Format

```
DUPLICATE [range] { BEFORE address }  
                  { AFTER address  }  
                  { ONTO pathname  }
```

This command copies text from one location in the current page to another or onto another file.

### Rules

- If you omit the *range*, SED duplicates the current line.
- The range must be lines on the current page or a single page argument.
- The destination can be BEFORE an address on the current page or AFTER an address on the current page.
- The destination can also be ONTO a pathname. If the file doesn't already exist, SED creates it, displays a create message, and adds the text to the new file. If the file does exist, SED appends the duplicated text to the file's contents.
- SED sets the current line to the last line to be duplicated.

### Examples

```
*DUPLICATE 118 130 BEFORE 57 }  
*DUPLICATE 1 22 AFTER LAST }  
*DUPLICATE 3 36 ONTO TABLE2 }  
File TABLE2 created
```

---

## EXECUTE

**Executes a file containing SED commands.**

---

### Command Format

```
EXECUTE pathname
```

This command executes *pathname*, a file you have created that contains SED commands.

### Rules

- Use this command during a SED session. Do not confuse it with the CLI EXECUTE command.
- *pathname* may contain any number of SED commands. You may choose to create files with lists of commands that you want to perform often. This is a convenient way to change display settings and to define function keys. See the SET and CLEAR commands in this chapter.

### Example

```
) TYPE START )  
SET LINE_NUMBER_DISPLAY;  
SET DISPLAY_MODE 2;  
CLEAR PRELOAD_OLD_LINE;  
SET FUNCTION CTRL 9 "SAVE"  
  
) XEQ SED EXAMPLE )  
(You edit EXAMPLE.)  
*EXECUTE START }
```

---

## FIND

**Finds a text string.**

---

### Function Key Use

The FIND function key locates the next occurrence of the last searchstring typed on the command line with either the FIND or BACKFIND command.

### Command Format

FIND "searchstring" [range]

This command finds a word or phrase in a range of text.

### Rules

- You must enclose the searchstring in quotation marks (" ") or apostrophes ( ' ') if it contains spaces or symbols other than a dollar sign or underscore.
- A searchstring enclosed in quotation marks or apostrophes causes a case-sensitive search. If you don't enclose the searchstring, SED finds the first searchstring that matches it regardless of case.
- The *range* can be lines within the current page, a single page argument, or a series of pages.
- If you do not specify a range, the search begins at the line following the current line and continues to the end of the current page.
- SED sets the current line to the line containing the first occurrence of the searchstring. To find the same searchstring further down the page, press the FIND function key since it stores the searchstring. To find the same searchstring on another page in the file, you must retype the FIND command with a page argument.
- If SED cannot find the searchstring, it displays an error message and does not change the current line.

### Examples

```
*FIND program )
23  The Program Introduction consists of
*FIND "summary of" ALL
1   This book contains a summary of the
*FIND "87%" PAGE 1 )
344 reasearch shows that 87% of the students
```

---

## HELP

**Explains a command or keyword.**

---

### Command Format

HELP [ *command* ]  
          [ *keyword* ]

This command displays information about SED commands and keywords.

### Rules

- HELP, without an argument, displays a list of SED commands and keywords by category.
- HELP, with one of the listed commands or keywords as an argument, displays information about that particular command or keyword.

### Examples

```
*HELP SAVE )
```

```
SAVE_COMMAND
```

*WRITES A COPY OF THE FILE, INCLUDING ALL CHANGES MADE IN THE CURRENT EDITING SESSION, TO A "SAVE FILE". THE NAME OF THE SAVE FILE IS THE PATHNAME OF THE FILE BEING EDITED, WITH THE .SV EXTENSION. IF MULTIPLE SAVE COMMANDS ARE ISSUED DURING A SESSION, PREVIOUS .SV FILES ARE OVERWRITTEN.*

---

## INSERT

**Inserts text between existing lines.**

---

### Function Key Use

The INSERT CURRENT function key inserts text before the current line. The source of the text is the terminal keyboard.

### Command Format

INSERT [*address*] [*FROM pathname*] [*range*]

This command inserts text before an address in the file. The source of the text can be the terminal keyboard or another file.

### Rules

- If you press the INSERT CURRENT function key, SED opens a space before the current line on your screen. You then type the text you want to insert.
- If you type INSERT on the command line and omit the address, SED inserts the text before the current line.
- If you omit the *FROM pathname* argument, you must type the text at your terminal keyboard. If you include the *FROM pathname* argument, the text comes from the file specified by the pathname.
- Before inserting text from another file, you can use the DIRECTORY command to change your working directory. This permits you to use a filename instead of a long pathname when you type the INSERT command.
- You can specify a *range* of lines if the file in *pathname* is one page long.
- If you are displaying line numbers, SED renumbers the lines to include the inserted text.
- To terminate the INSERT command and return to command mode after typing text at your terminal keyboard, press the ESC key.
- SED sets the current line to the last line of inserted text.

### Examples

```
1 One, two buckle my shoe.
2 Five, six pick up sticks.
3 Seven, eight lay them straight.
```

The current line is line 2.  
Press the INSERT CURRENT function key.

```
1 One, two buckle my shoe.
2 —
* Five, six pick up sticks.
* Seven, eight lay them straight.
```

\*INSERT FROM :UDD:ROBIN:FORMS:TABLE\_3.2 ↓

---

## JOIN

Removes a page break.

---

### Command Format

JOIN [*address*]

This command removes a page break in your file.

### Rules

- If you omit the address, the current line must be line 1.
- If you include an address, it must be either line 1 of the current page or a single page argument.
- SED sets the current line to what used to be line 1 of the second page.
- SED updates the line numbers and page numbers.
- If you try to join two pages that would produce a page longer than the 1023-line maximum, SED displays an error message and does not join the pages.

### Examples

```
*JOIN )
```

```
*JOIN 1 )
```

```
*JOIN PAGE 4 )
```

---

## LIST

Displays text without changing the current line.

---

### Command Format

LIST [*range*]

This command lists a range of text on the terminal screen without changing the current line.

### Rules

- If you omit the *range*, SED lists the current line.
- The *range* must be a line or lines within the current page or a single page argument.
- To redisplay a range of lines around the current line, press NEW LINE, ESC, ERASE PAGE, or the VIEW function key.

### Examples

```
*LIST 1 3
1 C THIS IS A FORTRAN PROGRAM
2 C THAT COMPUTES PRINCIPAL
3 C AND INTEREST.
```

```
*LIST ALL )
1 C THIS IS A FORTRAN PROGRAM
2 C THAT COMPUTES PRINCIPAL
```

```
.
.
38 END
```

---

## MODIFY

Edits a line or range of text.

---

### Function Key Use

The MODIFY CURRENT function key allows you to edit lines of text, beginning with the current line.

### Command Format

MODIFY  $\left[ \begin{array}{l} \text{address} \\ \text{range} \end{array} \right]$

This command allows you to make corrections to:

- lines within the current page, beginning at the current line or the line specified by *address*
- a range of lines within the current page, beginning at the first line specified in *range*
- lines on another page in the file

You can use the control characters, the rightarrow and leftarrow keys, and the screen position keys to make changes.

### Rules

- If you press the MODIFY CURRENT function key or type the MODIFY command without an *address* or a *range*, SED moves the cursor to the current line for editing.
- The *address* must be a line on the current page or a single page argument.
- If you specify a *range*, it must be lines within the current page or a single page argument.
- Pressing the NEW LINE key enters the revised line and moves the cursor to the next line for modification.
- To terminate the MODIFY command and return to command mode, press the ESC key.
- SED automatically returns you to command mode when you press NEW LINE after modifying the last line in the current page or the last line in the range you specified.

### Examples

```
*MODIFY 2 3 |
```

```
1 This is line 1.
```

```
2 This is line 2.
```

```
3 This is lin 3.
```

Press the rightarrow key four times, to position the cursor at the fifth character, and press CTRL-E to open a space in the line. Line 2 then looks like this:

```
2 This _is line 2.
```

Press the space bar once to insert a space, then press CTRL-E again to close up the line. Now press NEW LINE to enter the modified line and move the cursor to the beginning of line 3.

```
3 This is lin 3.
```

To correct line 3, it's easier to work from the end of the line. Press CTRL-A to move the cursor to the end of the line; then press the leftarrow key three times to backspace to the proper position. Press CTRL-E, then type *e*, then CTRL-E, then NEW LINE. SED automatically returns you to command mode because you finished modifying the last line in the range. Your text display now looks like this:

```
1 This is line 1.
```

```
2 This is line 2.
```

```
3 This is line 3.
```

Of course, you could have retyped the lines instead of using the control character and arrows. We show the control characters because they speed up editing long lines. You could also have positioned the current line to line 2 and then pressed the MODIFY CURRENT function key or typed MODIFY 2 on the command line. The only difference is that SED would not automatically return you to command mode; you would have to press the ESC key.



---

## MOVE

Moves text from one location to another.

---

### Command Format

```
MOVE [range] { BEFORE address }  
              { AFTER address  }  
              { ONTO pathname  }
```

This command moves text from one location in the page to another or onto another file. It deletes the text from its original location and inserts it at the destination.

### Rules

- If you omit the range, SED moves the current line.
- The *range* can be lines within the current page or a single page argument.
- The destination can be BEFORE an address on the current page or AFTER an address on the current page.
- The destination can also be ONTO a pathname. In this case, SED creates the file if it doesn't already exist, displays a create message, and adds the text. If the file exists, SED appends the moved text to the current contents.
- SED sets the current line to the line following the range of text that was moved. If the last line moved was the last line on the page, SED sets the current line to the line *before* the range of text that was moved.

### Examples

```
*MOVE 243 259 BEFORE 12 \  
*MOVE 1 10 AFTER LAST \  
*MOVE 4 47 ONTO PROJECT2 \  

```

---

## PASTE

Merges a range of lines into a single line.

---

### Function Key Use

The PASTE WHILE EDITING function key merges the current line with the following line.

### Command Format

```
PASTE [address ]  
      [range  ]
```

This command merges *address* with the following line, or merges a *range* of lines into one line.

### Rules

- If you omit both *address* and *range*, SED pastes the line following the current line to the current line.
- If you specify an *address*, it must be a line within the current page.
- If you specify a *range*, it must be lines within the current page.
- If the merged line would exceed the 255-character maximum, SED displays an error message and does not merge the lines.
- When you enter keyboard input mode with the MODIFY command, you can merge the current line and the following line by using the PASTE WHILE EDITING function key. You position the cursor where you want the next line pasted and then press the function key.
- You may find when you PASTE lines that you have more or fewer spaces between words than you want. Edit the line to add or delete spaces.

### Examples

```
1 This is a  
2 n examle of pas  
3 ting.  
*PASTE 1 3 \  
1 This is an example of pasting.
```

---

## POSITION

**Changes the current line position.**

---

### Function Key Use

The **BOTTOM OF SCREEN** function key sets the current line to the last line on the screen.

The **DOWN 4 LINES** function key sets the current line to the fourth line ahead.

The **DOWN 11 LINES** function key sets the current line to the eleventh line ahead.

The **DOWN ONE SCREEN** function key displays the next screen of text and sets the current line to the middle line on the screen.

The **FIRST LINE ON PAGE** function key sets the current line to the first line on the page.

The **GO TO PREVIOUS POSITION** function key sets the current line to the previous current line. This key stores the previous fifteen line positions.

The **LAST LINE ON PAGE** function key sets the current line to the last line on the page.

The **MIDDLE LINE ON PAGE** function key sets the current line to the middle line on the page.

The **MIDDLE OF SCREEN** function key sets the current line to the middle line on the screen.

The **POSITION PAGE FIRST** function key sets the current line to the first line on the first page.

The **POSITION PAGE LAST** function key sets the current line to the first line on the last page.

The **POSITION PAGE NEXT** function key sets the current line to the first line on the next page.

The **POSITION PAGE PREVIOUS** function key sets the current line to the first line on the previous page.

The **TOP OF SCREEN** function key sets the current line to the first line on the screen.

The **UP 4 LINES** function key sets the current line to the fourth line back.

The **UP 11 LINES** function key sets the current line to the eleventh line back.

The **UP ONE SCREEN** function key displays the previous screen of text and sets the current line to the middle line on the screen.

### Command Format

**POSITION** address

This command moves the current line position to a different address in the file.

### Rules

- The address must be a line within the current page or a single page argument.
- If you give a line address, that line becomes the current line. If you give a page address, the first line on that page becomes the current line.
- You can use the screen position keys and the uparrow and downarrow keys to change the current line in both command mode and keyboard input mode. See Chapter 2.
- The **DISPLAY** command lists information about current page and line positions.

### Examples

\*POSITION 86 ↓

\*POSITION PAGE 4; POSITION 22 ↓

---

## PRINT

Prints text on @LIST.

---

### Command Format

PRINT [*range*]

This command sends text to the generic CLI file @LIST.

### Rules

- You must set the @LIST file from the CLI before executing SED, or SED will display an error message.
- If you set the listfile to @LTP (the line printer), you must have Read and Write access to the line printer.
- If you omit *range*, SED prints the current line.
- *range* can be lines within the current page or a single page argument.
- If you are displaying line numbers, they will appear in the line printer listing. If you do not want line numbers to appear, use the CLEAR LINE\_NUMBER\_DISPLAY command before you print your text.
- The PRINT command does not change the current line.
- If yours is a multiuser system, your print request will be spooled for output; thus it may not be printed immediately.

### Examples

```
*PRINT 1 10 ;
*PRINT ALL ;
*PRINT PAGE 19 ;
```

---

## range

Specifies a block of text.

---

### Argument Format

$\left. \begin{array}{l} \text{ALL } [PAGES] \\ \text{REMAINING } [PAGES] \\ \text{address} \\ \text{address } [TO] \text{ address} \\ \text{address FOR number} \end{array} \right\}$

This argument specifies a particular group of lines or pages.

### Rules

- In all formats, if you omit *PAGES*, SED assumes lines in the current page. To specify pages, you must include the keyword *PAGE*.
- *address* must be lines within the current page or a single page argument.
- You can omit the keyword *TO*.
- The number following FOR must be a number, not a relative position like FOR or LAST.

### Examples

```
*FIND "source" ALL PAGES ;
*LIST REMAINING ;
*DUPLICATE 4 18 AFTER LAST ;
*REPLACE 1 FOR 3 ;
```

---

## REPLACE

**Replaces existing lines with text typed at the terminal keyboard.**

---

### Command Format

REPLACE [*range*]

This command deletes text and replaces it with new text. The new text comes from the terminal keyboard.

### Rules

- If you omit *range*, SED deletes the current line, and you replace it with one or more lines you type at the terminal keyboard.
- *range* can be lines within the current page or a single page argument.
- SED displays a line number before you type in the replacement text on each line. If you have executed the `CLEAR LINE_NUMBER_DISPLAY` command, SED does not display line numbers, and you type in the text on a blank line.
- The number of lines deleted does not have to equal the number of lines replaced. SED renumbers the lines to take up the slack or excess.
- To terminate the REPLACE command and return to command mode, press the ESC key.
- SED sets the current line to the last line of replacement text.

### Example

```
1 This is line 1.
2 This is line 2.
(The current line is line 2.)
```

```
*REPLACE ↵
2 Replacement line. ↵
```

ESC

```
1 This is line 1.
2 Replacement line.
```

```
*REPLACE 1 FOR 3 ↵
1 This replaces lines 1 through 3. ↵
```

ESC

```
1 This replaces lines 1 through 3.
2 This is line 4.
3 This is line 5.
```

---

## SAVE

Creates a SAVE file without terminating the session.

---

### Command Format

#### SAVE

This command creates a copy of the file, including all changes made in the current editing session. The save file has the name of the file plus the .SV extension.

Each time you execute the SAVE command during a session, SED overwrites the existing .SV file.

### Example

```
*SAVE )
```

---

## searchstring

Specifies a character string.

---

### Argument Format

```
["]searchstring["]
```

This argument specifies a particular word or phrase in the BACKFIND, FIND, and SUBSTITUTE commands.

### Rules

- You must enclose the searchstring in quotation marks or apostrophes if it contains blank spaces or symbols other than a dollar sign or underscore.
- If you enclose the searchstring in quotation marks or apostrophes, SED matches the strings exactly. If you don't, SED matches it with any string, regardless of case.
- If apostrophes are part of the searchstring, enclose the string in quotation marks; if quotation marks are part of the searchstring enclose it in apostrophes.

### Examples

```
*FIND "Brown" PAGE 5 )
```

```
*SUBSTITUTE 'June' 'May' ALL PAGES )
```

---

## SET

**Sets display settings or defines function keys.**

---

### Command Format

SET {  
BLANK\_MODE  
DISPLAY\_MODE n  
ESCAPE\_CHARACTER x  
ESCAPE\_IS\_VIEW  
LINE\_NUMBER\_DISPLAY  
NEW\_LINE\_IS\_VIEW  
PRELOAD\_OLD\_LINE  
SKIP\_WHITE\_SPACE  
TYPER\_MODE  
UPPERCASE\_SHIFT\_CHARACTER x  
VIEW\_RANGE n  
FUNCTION number "string" [; "string"]

This command changes some display settings from OFF to ON and sets other display settings with a number or a character. It also defines function keys to execute SED commands.

SED record the current display settings, including any changes you make while editing, in a file which has the .ED extension. If you're editing file MUMBLE, SED records its display settings in MUMBLE.ED. This information remains in MUMBLE.ED from one editing session to the next, even though you may log off and on again.

The function keys you define remain defined for the current editing session only.

### Rules

- You can display the current display settings with the DISPLAY command, and you can display the current function key definitions with the DISPLAY FUNCTION command. You can clear most of the function keys and all of the display settings except for DISPLAY\_MODE and VIEW\_RANGE with the CLEAR command. See the DISPLAY and CLEAR commands in this chapter.
- The SET BLANK\_MODE command allows you to pad lines of text with trailing blanks.
- The SET DISPLAY\_MODE command followed by a number from 0 to 3 sets the display mode option. Option 0, the default, displays the current line bright and the rest dim. Option 1 displays all lines bright. Option 2 displays all lines bright but places a dimmed \* before

the current line. Option 3 displays the current line dim and the rest bright.

- The SET ESCAPE\_CHARACTER command followed by a character allows you to insert escape characters in your file by typing the specified character in your text.
- SET ESCAPE\_IS\_VIEW causes ESC typed on an empty command line to default to the VIEW command; SED displays a range of lines before and after the current line.
- SET LINE\_NUMBER\_DISPLAY instructs SED to display a line number before each line of text.
- SET NEW\_LINE\_IS\_VIEW causes NEW LINE typed on an empty command line to default to the VIEW command; SED displays a range of lines before and after the current line.
- SET PRELOAD\_OLD\_LINE instructs SED to save a copy of the previous current line. Then, when you are appending, inserting, or replacing text and you press CTRL-A or the rightarrow key, SED repeats characters of the previous line.
- SET SKIP\_WHITE\_SPACE causes the cursor to move to the first nonblank character on the next line when you press the NEW LINE key.
- SET TYPER\_MODE causes the cursor to maintain its column position when you press the uparrow or downarrow key or any of the screen position function keys.
- The SET UPPERCASE\_SHIFT\_CHARACTER followed by a character instructs SED to translate all characters typed on the terminal keyboard to lowercase unless preceded by the character you set. This display setting is for users of D1 terminals.
- The SET VIEW\_RANGE command followed by a number from 1 to 10 sets the number of lines SED displays before and after the current line. The default is 10 lines before and 10 lines after the current line.
- The SET FUNCTION command followed by the number of a function key and one or more SED commands enclosed in quotation marks or apostrophes defines that function key to execute the SED command(s). The number of the function key is a number or a number preceded by SHIFT, CTRL, or CTRL SHIFT.

### Examples

```
*SET BLANK_MODE )  
*SET DISPLAY_MODE 1; SET VIEW_RANGE 5 )  
*SET FUNCTION CTRL SHIFT 8 "HELP" )
```

---

## source

Specifies a source for text.

---

### Argument Format

*[FROM source]*

This argument specifies where text is to come from.

### Rules

- If you omit a source, the text comes from the terminal keyboard.
- If you include a source, it must be the pathname of an existing file.

### Examples

```
*INSERT 22 FROM TABLE.2 }
```

```
*APPEND FROM :UDD:BARRY:PLANS:PROJECT3 }
```

---

## SPELL

Lists words for spelling aid.

---

### Command Format

SPELL string

This command lists words that begin with the letters you specify.

### Rules

- string is the beginning letters of the word you want to check.
- SED lists the words beginning with string in lowercase and in alphabetical order.
- You or your system manager can add words to SED's dictionary (a file named SED.DICTIONARY). The words you add must be in lowercase and placed in alphabetical order within SED.DICTIONARY. Type one word per line, pressing NEW LINE after each word.

### Example

```
*SPELL atten }
```

```
attend  
attendance  
attendances  
attendant  
attendants  
attended  
attendee  
attender  
attenders  
attending  
attends  
attention  
attentional  
attentions  
attentive  
attentively  
attentiveness  
attenuate  
attenuates  
attenuation
```

---

## SPLIT

Creates a page break.

---

### Command Format

SPLIT [*address*]

This command makes a page break in the file.

### Rules

- If you omit *address*, SED creates the page break before the current line. The current line then becomes line 1 on the new page, and SED appropriately renumbers all subsequent lines.
- If you include *address*, it must be a line within the current page or a single page argument. If the address is a line, SED splits the page before that line. That line becomes line 1 on the new page, and SED appropriately renumbers all subsequent lines.
- If *address* is a page, SED creates a blank page in front of the page you specified as the argument. The current line is line 1 of the page following the blank page. SED renumbers the pages to reflect the split.
- The maximum number of lines on a page is 1023, and the maximum number of pages in any one file is 255. If you try to exceed these numbers, SED displays an error message.

### Examples

```
*SPLIT )  
*SPLIT 374 )  
*SPLIT PAGE 3 )
```

---

## SUBSTITUTE

Replaces a string with another string.

---

### Command Format

SUBSTITUTE "searchstring 1" [*FOR*]  
"searchstring 2" [*range*]

This command substitutes a word or phrase for another word or phrase throughout a range of text.

### Rules

- You must enclose the searchstring in quotation marks or apostrophes if it contains spaces or symbols other than a dollar sign.
- "searchstring 1" replaces "searchstring 2".
- If you do not enclose the first searchstring in quotation marks or apostrophes, SED translates it to uppercase. If you do not enclose the second searchstring, the substitution is case insensitive. That is, the first searchstring replaces the second searchstring regardless of the case of the letters in the second searchstring.
- If you omit *range*, SED makes the substitution beginning at the current line to the end of the page.
- *range* can be lines within the current page, a single page argument, or a series of pages.
- SED displays each line in which it made a substitution. If it does not find the second searchstring within the range, SED displays an error message.
- SED sets the current line to the last line in which it made a substitution.
- To display a range of lines around the current line, press NEW LINE, ESC, or the VIEW function key.

### Example

```
*SUBSTITUTE '1977' '1976' ALL PAGES )  
64 All proposals for 1977 must be  
189 Because fiscal 1977 ends in May  
197 Our plans for 1977 include upgrading
```



---

## UNDO

**Restores last deletion.**

---

### Command Format

UNDO

This command restores the most recently deleted text in the file.

### Rules

- You can press the UNDO function key or type the UNDO command on the command line.
- The UNDO command restores only the most recent deletion from the current editing session. Using several UNDO commands in a row will not restore several blocks of deleted text.
- If there is no deleted text to restore, SED displays a message to that effect.
- SED renumbers the lines to reflect the restoration and sets the current line to the first line restored.
- If the last block of text deleted was a page, the UNDO command restores the text by inserting it at the beginning of the next page. Use the SPLIT command to restore the page break.

### Example

```
1 This is line 1.
2 This is line 2.
3 This is line 3.
*DELETE 1 )
1 This is line 2.
2 This is line 3.
*UNDO )
1 This is line 1.
2 This is line 2.
3 This is line 3.
```

---

## VIEW

**Displays a range of lines around the current line.**

---

### Function Key Use

The VIEW function key displays a range of text before and after the current line.

### Command Format

VIEW

This command performs the same action as the VIEW function key.

### Rules

- You can press the VIEW function key or type the VIEW command on the command line.
- You can use the VIEW command to refresh your screen with text after executing the DISPLAY, DISPLAY FUNCTION, CLI, DO, HELP, LIST, SPELL, or SUBSTITUTE commands.
- SED displays 10 lines before the current line, the current line, and 10 lines after the current line.
- You can decrease the number of lines SED displays around the current line with the SET VIEW\_RANGE command.
- If the VIEW command asks for more lines around the current line than are actually available, SED displays only the available lines.
- The VIEW command does not change the current line.

## VIEW (continued)

### Example

Press the DISPLAY function key.

*Edit file name - :UDD:USERNAME:NOTES*

<i>Current page:</i>	<i>5</i>	<i>Last page:</i>	<i>7</i>
<i>Current line:</i>	<i>378</i>	<i>Last line:</i>	<i>433</i>
<i>View range:</i>	<i>10</i>		
<i>Display mode:</i>	<i>0</i>		
<i>Typewriter mode:</i>	<i>ON</i>		
<i>Blank mode:</i>	<i>OFF</i>		
<i>Preload old line:</i>	<i>ON</i>		
<i>Skip white space:</i>	<i>ON</i>		
<i>Upper mode:</i>	<i>OFF</i>		
<i>Escape mode:</i>	<i>OFF</i>		
<i>New_line is view:</i>	<i>ON</i>		
<i>Escape is view:</i>	<i>ON</i>		

*Line numbers being displayed*

Press the VIEW function key.

*288 All data processing involves*  
*289 three basic considerations:*  
*290*  
*291 INPUT or source data entering*

*.  
. .  
.*

End of Chapter

# Appendix A

## Terminal Differences

This manual assumes that you're using a DASHER D2 or D200 display terminal. If you're not, this appendix explains the differences.

### The DASHER D1 Terminal

The D1 keyboard is similar to the D2 keyboard. However, the D1 is an uppercase only terminal. To create and edit a SED file with both upper- and lowercase characters, you must use the SET UPPER\_CASE\_SHIFT\_CHARACTER command. For the details of this command, see the section on Display Settings in Chapter 3.

The following function keys are not available on the D1 keyboard: UP 4 LINES, UP 11 LINES, DOWN 4 LINES, DOWN 11 LINES, CUT WHILE EDITING, and PASTE WHILE EDITING. You can define other function keys to execute these commands, or type the POSITION, CUT, and PASTE commands on the command line. For more information on making these modifications, see the section on Defining and Clearing Function Keys in Chapter 3 or the specific command you want to use.

### The DASHER D400 and D450 Terminals

Figure A-1 shows the D400/450 terminal keyboard and template.

The D400 and D450 keyboards are similar to the D200 keyboard. You use the same template for these terminals as you do for the D200. However, there are five additional function key on the far right of the D400 and D450 keyboards. They are called *local function keys*. You cannot redefine these function keys. Table A-1 describes the each local function key.

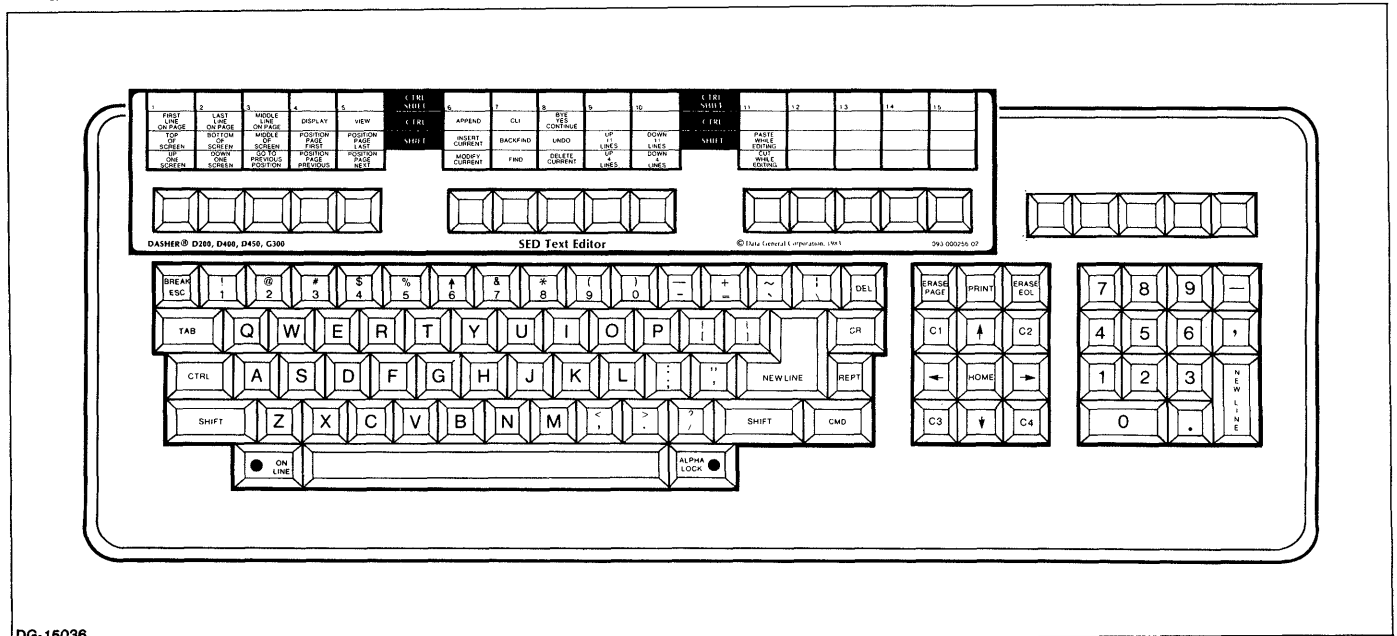


Figure A-1. DASHER D400/D450 Terminal Keyboard and Template

**Table A-1. Local Function Keys on the D400 and D450 Keyboards**

Local Function Key	What It Does
CURSOR TYPE	Changes the appearance of the cursor on the terminal screen.
NORM COMP	Changes from normal display (which has approximately 80 characters per line) to compressed display (which has approximately 135 characters per line), and back again.
LOCAL PRINT	Not applicable.
SCROLL RATE	Adjusts the rate at which text scrolls onto the terminal screen.
HOLD	Freezes text on the terminal screen. Pressed a second time, resumes scrolling text on the terminal screen.

On the D400 and D450 terminals, you can display text of approximately 80 characters per line or approximately 135 characters per line. See the NORM COMP function key in Table A-1.

## The DASHER G300 Terminal

Figure A-2 shows the G300 terminal keyboard and template.

The G300 keyboard is similar to the D200 keyboard. You use the same template for this terminal as you do for the D200 terminal. However, the G300 has five additional function keys on the far right of the keyboard. These five function keys and the three keys to their left are *local function keys*. You cannot redefine them. For more information on using the G300 terminal, see the *DASHER G300 Display Terminal User's Manual*.

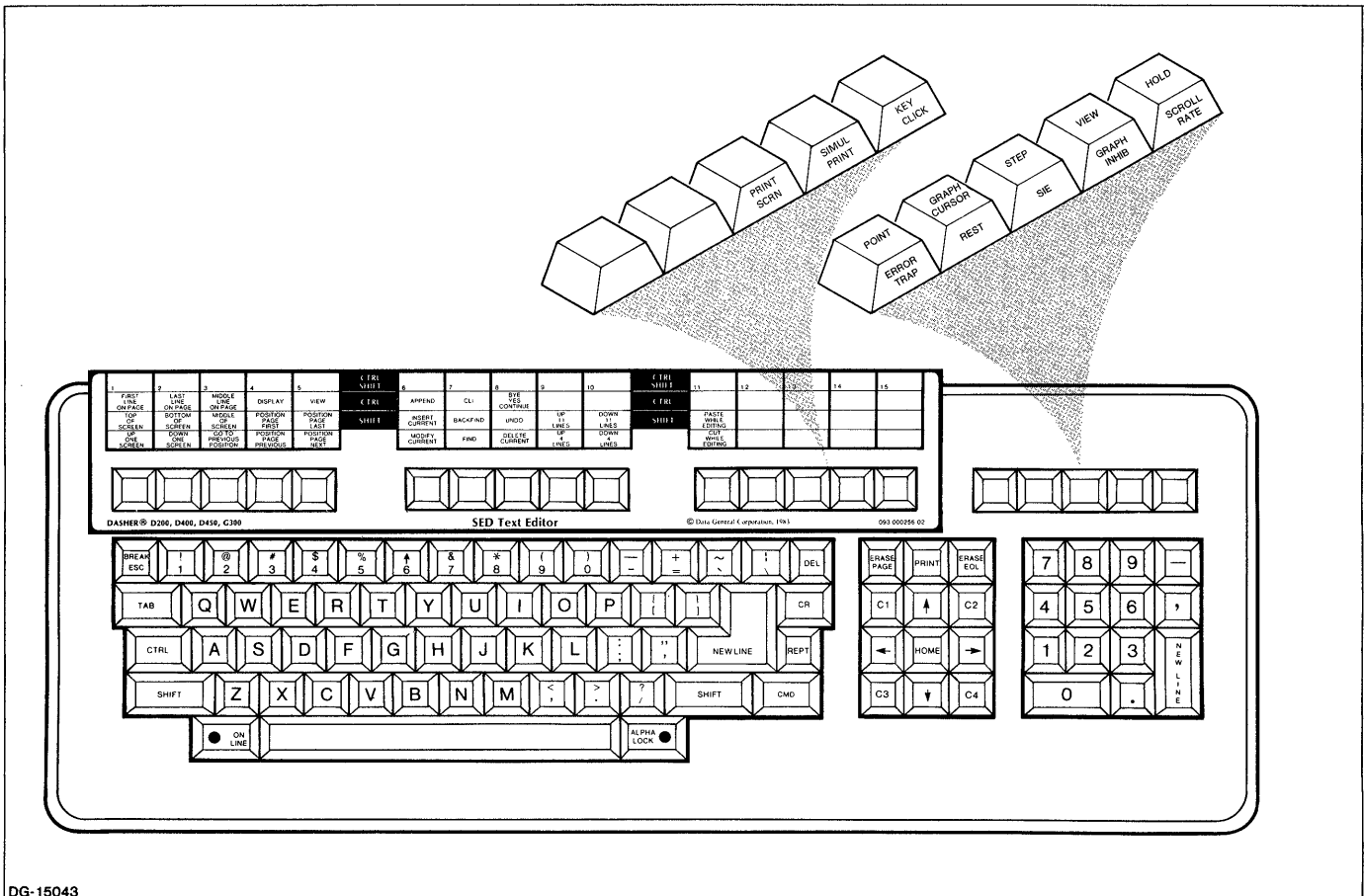


Figure A-2. DASHER G300 Display Terminal Keyboard and Template

End of Appendix

# Appendix B

## SED Messages

Most messages that SED displays are error messages, but some are simply informative. This appendix lists *all* messages in alphabetical order. We explain why SED displays each message and how to correct the error.

### *Ambiguous abbreviation for 'PAGE' or 'LINE'*

PAGE and PREVIOUS both begin with P, so the abbreviation for PAGE requires two letters to identify it uniquely. LINE and LAST both begin with L, so the abbreviation for LINE requires two letters to identify it uniquely. Correct the abbreviation to two letters and re-enter the command line.

### *Ambiguous object with HELP*

You typed an ambiguous abbreviation for a HELP item. Retype the item.

### *A number must follow the plus or minus sign*

If you use a plus or minus sign in an address, you can't follow it with CURRENT, NEXT, PREVIOUS, or LAST. Edit the command to include a number.

### *Backup file has permanence set*

SED displays this message when permanence is *ON* the backup file and you terminate the editing session with the BYE command and ask SED to save a backup file. With permanence *ON*, SED cannot delete the old backup file and give its name to the new backup file. Use the argument PERM\_OFF with the BYE command.

### *Cannot access list file*

You receive this message when you try to use the PRINT command without having set the listfile with the CLI LISTFILE command before executing SED. You must terminate the editing session and return to the CLI, set the listfile, and then return to SED in order to use the PRINT command. If you have set the listfile to @LPT (the line printer) and you receive this message, it means that the ACL for the line printer does not allow you access. See your system manager.

### *Cannot execute 'DO' command*

You need two son processes in your user profile to execute the DO command. See your system manager.

### *'CLEAR' must be followed by a different argument*

You typed an invalid argument to the CLEAR command. See HELP CLEAR and try again.

### *CUT not performed*

You did not specify a column with the CUT command. Specify a column and enter the command again, or use the CUT WHILE EDITING function key to cut the line.

### *Edit file has permanence set*

SED displays this message when permanence is *ON* the edit file and you try to terminate the session with BYE NO. With permanence *ON*, SED cannot delete the original edit file and give its name to the updated file. Type the argument PERM\_OFF with the BYE command.

### *Extra characters on command*

You receive this message when you type invalid characters after a SED command. Edit the command, and try again.

### *File exceeded 255 page maximum, page 'SPLIT' cannot be done*

SED has a limit of 255 pages per file. Any attempt to create page 256 with the SPLIT command generates this message. The file remains intact, but it has only 255 pages.

### *File xxxxxxxx created*

SED displays this message when the MOVE ONTO pathname or DUPLICATE ONTO pathname commands specify a file that doesn't already exist. This is not an error message, just a note that SED created a new file to which it added the text you specified.

### *Function key number out of range*

You tried to define a function key with the SET command, specifying a number greater than the number of function keys on your terminal keyboard. Edit the command to include a valid function key number.

*Function key string space exhausted, key not set*

SED displays this message when you define a function key with the SET command, and SED has run out of space to store that function key definition. You can CLEAR another function key to make room for the one you want to define.

*Identifier expected*

This message occurs when you type SET ESCAPE\_CHARACTER or SET UPPERCASE\_SHIFT\_CHARACTER without specifying a character. Edit the command to include a single character following the name of the display setting. This message also occurs when you type the SPELL command without specifying a string. Edit the command to include the beginning letter(s) of the word you want to check.

*Illegal function key used*

You accidentally pressed a key that does not work in SED. Press CTRL-U to clear the command line, and then press another key or type a command.

*Illegal pathname*

This message is generated if the pathname following ONTO or FROM doesn't conform to the CLI file naming conventions. Just fix the command line and enter it again.

*Improper destination clause, use 'ONTO', 'BEFORE', or 'AFTER'*

SED displays this message when you omit all or part of the destination clause following the MOVE or DUPLICATE command. ONTO must precede a legal pathname. BEFORE or AFTER must precede an address. Correct the command and enter it again.

*Invalid extra arguments in command*

SED displays this message if you add extra options or characters to a command that doesn't allow them. Just correct the command line and try again.

*Invalid syntax, type HELP SET or HELP CLEAR*

This message occurs when you type the SET or CLEAR command with an argument that SED doesn't recognize. Retype the command after you check the HELP file.

*'JOIN' requires the first line of a page*

The JOIN command requires that you set the current line to line 1 before executing the command or that you type a 1 or a single page argument after JOIN. Correct the command and try again.

*Line does not exist on this page*

You receive this message when the lines addressed by the command are greater than the number of lines on the

page. You also receive this message when you try to position the current line before line 1 on the page. Correct the command line or press a different screen position key.

*Line too long*

You see this message when you enter keyboard input mode with the MODIFY command only. It means that you tried to edit a file with lines longer than the 255 characters, the maximum SED permits. Press ESC, and you will not shorten any lines in the file. If you press NEW LINE, you will shorten the line. Use another text editor on the file.

*Lines too long to paste together*

SED allows a maximum line length of 255 characters. If you try to PASTE two or more lines and the line produced would exceed 255 characters, SED does not paste them.

*Maximum VIEW exceeded*

SED displays this message when you attempt to SET the VIEW\_RANGE with a number greater than 10. SED can display a maximum of 21 lines of text at a time, 10 lines before and 10 lines after the current line. Correct the command and enter it again.

*Missing number following 'FOR'*

This message appears when you forget to follow the keyword FOR in the command line with a number. This error also appears if you try to follow FOR with a relative position like: CURRENT, PREVIOUS, or NEXT. Correct the command line and enter it again.

*No deleted text to 'UNDO'*

You can use the UNDO command only once after a DELETE. SED displays this message if you haven't deleted any text during the session or if you attempt to use several UNDO commands in a row.

*Only 'FROM' is allowed after 'APPEND'*

You see this message if you try to follow the APPEND command with anything other than the FROM pathname argument. Correct the command line and try again.

*Page exceeded 1023 line limit, line not accepted on this page*

SED allows only 1023 lines per page in a file. Any attempt to add line 1024 generates this message. It doesn't matter which command tried to add the line: INSERT, APPEND, MOVE, DUPLICATE, or REPLACE. You should split the page into two or more pages, as the message suggests, or split the file into two or more files (via MOVE) if you don't want the extra pages.

*'PAGE' must be followed by location*

SED displays this message when you fail to follow the word PAGE, or its abbreviation, by a number or location

(like NEXT or LAST). Correct the command line and try again.

*Page number greater than last page in file*

You receive this message when you follow the word PAGE, or its abbreviation, by a number or location (like NEXT or LAST) which exceeds the number of pages in the file. Correct the command line and try again or press a different screen position key.

*PASTE\_LINES cannot be done on last line*

The PASTE command needs at least two lines. You only have one if the current line is the last line and you pressed the PASTE WHILE EDITING function key, or if you specified a line after the last line when you typed the PASTE command.

*Previous backup file deleted*

This message occurs after you tupe BYE and request a backup file, when a backup file already exists for this file. SED deletes the old backup file and gives its name to the new backup file.

*'SET' must be followed by a different argument*

You typed an invalid argument to the SET command. See the HELP SET file and retype the command.

*SET VIEW must be followed by a number*

You typed the SET VIEW\_RANGE command without specifying the number of lines you want SED to display. Edit the command to include a number from 1 through 10.

*Single character expected*

SED displays this message when you SET the display settings ESCAPE\_CHARACTER or UPPER-CASE\_SHIFT\_CHARACTER with more than one character. Edit the command, specifying only one character after the display setting.

*String must be enclosed in quotes*

This message occurs when the searchstring you typed following the FIND or SUBSTITUTE commands contains blanks or symbols other than a dollar sign or underscore. Enclose the searchstring in quotation marks or apostrophes.

*String not found*

You receive this message when SED can't locate the searchstring within the range you specified with the FIND, BACKFIND, or SUBSTITUTE command. Remember, if you omit the range, SED searches the remainder of the current page with the FIND and SUBSTITUTE commands and all previous lines on the

current page for the BACKFIND command. The current line may be beyond or previous to the line containing the searchstring.

*The end of the range is beyond the end of the page*

This message occurs if the range of lines specified exceeds the number of lines on the current page. Correct the command, and try again.

*The range for this command must be in one page*

All commands, except FIND and SUBSTITUTE, must limit their range to lines within the current page or a single page argument. Correct the command, and try again.

*There are no lines in this range*

This message occurs if you attempt to specify an empty or negative range, like LIST 2 TO 1. Correct the command, and try again.

*This file cannot be edited because it exceeds the maximum of 255 pages*

SED displays this message when you first enter SED. You immediately return to the CLI since SED cannot edit any file that exceeds 255 pages.

*'TO' must be followed by a location*

The word TO must be followed by a line number or relative location, like NEXT or LAST. Edit the command to include a location.

*Too many lines on page xxx, this page was SPLIT*

This message occurs after an XEQ SED command, as SED scans the file. Another utility or program could have created the excessive page length. SED's maximum page length is 1023 lines per page. When a page exceeds the maximum, SED breaks it into two pages in order to continue the editing session.

*Unable to open SED.DICTIONARY*

SED displays this message when you use the SPELL command and you do not have access to SED's dictionary file. See your system manager.

*Unknown command*

You receive this message when you type a command that SED does not recognize. Erase the command, and check the HELP list.

*Unknown object with HELP, type HELP New Line*

You typed an item that is not on the HELP list. Check the list, and try again.

*Unrecognized argument to ABANDON*

CONTINUE is the only argument SED accepts with the ABANDON command. If you type YES, NO, or PERM\_OFF following ABANDON, SED displays this message.

*WRITE access to file or directory denied*

OR

*WRITE ACCESS DENIED*

You receive this message when you try to change the contents of a file or directory to which you do not have WRITE access. You may receive this message when you try to edit such a file or when you use the DUPLICATE ONTO or MOVE ONTO command to add text to such a file. You must change the ACL for the file or directory by using the ACL command from the CLI.

End of Appendix



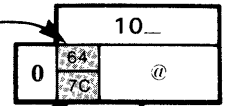
# Appendix C ASCII Chart

## ASCII Character Set

To find the *octal* value of a character, locate the character, and combine the first two digits at the top of the character's column with the third digit in the far left column.

### Legend:

Character code in decimal  
EBCDIC equivalent hexadecimal code  
Character



Octal	00_	01_	02_	03_	04_	05_	06_	07_
0	0 00 NUL	8 16 BS (BACK-SPACE)	16 10 DLE  P	24 18 CAN  X	32 40 SPACE	40 4D (	48 F0 0	56 F8 8
1	1 01 SOH  A	9 05 HT (TAB)	17 11 DC1  Q	25 19 EM  Y	33 5A !	41 5D )	49 F1 1	57 F9 9
2	2 02 STX  B	10 15 NL (NEW LINE)	18 12 DC2  R	26 3F SUB  Z	34 7F (QUOTE)	42 5C *	50 F2 2	58 7A :
3	3 03 ETX  C	11 0B VT (VERT TAB)	19 13 DC3  S	27 27 ESC (ESCAPE)	35 7B #	43 4E +	51 F3 3	59 5E :
4	4 37 EOT  D	12 0C FF (FORM FEED)	20 3C DC4  T	28 10 FS  \ 5B	36 5B \$	44 6B (COMMA)	52 F4 4	60 4C <
5	5 2D ENQ  E	13 0D RT (RETURN)	21 3D NAK  U	29 1D GS  }	37 6C %	45 60 -	53 F5 5	61 7E =
6	6 2E ACK  F	14 0E SO  N	22 32 SYN  V	30 1E RS	38 50 &	46 4B (PERIOD)	54 F6 6	62 6E >
7	7 2F BEL  G	15 0F SI  O	23 26 ETB  W	31 1F US  —	39 7D (APOS)	47 61 /	55 F7 7	63 6F ?

Octal	10_	11_	12_	13_	14_	15_	16_	17_
0	64 7C @	72 C8 H	80 D7 P	88 E7 X	96 79 (GRAVE)	104 88 h	112 97 p	120 A7 x
1	65 C1 A	73 C9 I	81 D8 Q	89 E8 Y	97 81 a	105 89 i	113 98 q	121 AB y
2	66 C2 B	74 D1 J	82 D9 R	90 E9 Z	98 82 b	106 91 j	114 99 r	122 A9 z
3	67 C3 C	75 D2 K	83 E2 S	91 8D [	99 83 c	107 92 k	115 A2 s	123 C0 }
4	68 C4 D	76 D3 L	84 E3 T	92 E0 \	100 84 d	108 93 l	116 A3 t	124 4F
5	69 C5 E	77 D4 M	85 E4 U	93 9D ]	101 85 e	109 94 m	117 A4 u	125 D0 }
6	70 C6 F	78 D5 N	86 E5 V	94 5F   or ^	102 86 f	110 95 n	118 A5 v	126 A1 ~ (TILDE)
7	71 C7 G	79 D6 O	87 E6 W	95 6D — or -	103 87 g	111 96 o	119 A6 w	127 07 DEL (RUBOUT)

SD-00217

Character code in octal at top and left of charts.

↑ means CONTROL

Figure C-1. ASCII Character Set

End of Appendix



# Index

Within this index, “f” or “ff” after a page number means “and the following page” (or “pages”). In addition, primary page references for each topic are listed first. Commands, calls, and acronyms are in uppercase letters (e.g., CREATE); all others are lowercase.

## A

ABANDON command 4-8, 3-19f, 4-4 (table)

abbreviations

acceptable 2-2

in /PROFILE 3-2

abort editing session 3-19f, 4-4 (table), 4-8

add text 2-2f

see also APPEND command

add to SED dictionary 3-19

address argument 4-8, 1-8f

advanced editing 3-1ff

aid, spelling 3-19, 1-7, 4-27

ALPHA LOCK key 1-2

AOS iii

AOS/VS iii

APPEND

command 4-9, 2-2f, 4-2 (table)

function key 2-2f, 4-2 (table), 4-9

append text 2-2f, 4-2 (table), 4-9

arguments to commands

address 1-8f, 4-8

definition of 1-8

destination 1-9, 4-13

range 1-9, 4-23

searchstring 1-9f, 4-25

source 1-9, 4-27

ASCII character set C-1 (fig.)

assistance

spelling 3-19, 1-7, 4-27

with SED, see HELP command

asterisk 3-2

see also SED prompt

auxiliary file (.ED) 3-1

## B

BACKFIND

command 4-10, 2-13f, 4-4 (table)

function key 2-13f, 4-4 (table), 4-10

backup file 2-15f, 3-1

BLANK\_MODE, see display settings

blanks, trailing 3-8

block of text, see range

BOTTOM OF SCREEN function key 4-3 (table), 2-5, 4-22

BREAK ESC, see escape (ESC) key

BYE command 4-10, 2-15f, 3-19, 4-4 (table)

BYE YES CONTINUE function key 4-4 (table), 2-16, 4-10

## C

carriage return (CR) 1-2, 2-7

case 2-2, 1-2, 3-8

change

current line position 2-4ff, 4-22

display settings 3-4ff, 4-11, 4-26

lines of text, see MODIFY command

working directory 3-17, 4-14

character 1-1

position 1-1

set, ASCII C-1 (fig.)

string

specify (searchstring) 4-25

substitute 4-28

characters, control (CTRL) 2-6f

clear

display settings 3-4f, 4-11

function keys 3-9ff, 4-5f (table), 4-11

CLEAR command 4-11, 3-4f, 4-5f (table)

see also display settings

CLEAR FUNCTION 3-11, 4-6 (table), 4-11

CLI (Command Line Interpreter)

command (SED) 4-12, 1-6f, 3-16f, 4-4 (table)

commands (AOS)

EXECUTE SED 4-7

executing from SED 3-15ff

FILESTATUS 3-15f

LISTFILE 3-18

PERMANENCE 2-16

QPRINT 3-18

TYPE 3-16

XEQ SED 3-16

display 1-6f

executing SED from 4-7

function key 3-16, 4-4 (table), 4-12

- command
  - arguments 1-8ff
  - format conventions iii
  - line 1-5
    - interpreter, see CLI
    - mistakes 1-10
    - multiple 2-2
  - mode 1-1
  - switches
    - description 3-2
      - /ED 3-2, 4-7
      - /NO\_ED 3-2, 4-7
      - /NO\_FORM\_FEEDS 3-3, 4-7
      - /NO\_SCREEN 3-3, 4-7
      - /PROFILE 3-2, 4-7
      - /WORK 3-2f, 4-7
    - use with EXECUTE SED 3-2, 4-7
  - terminator, see NEW LINE key
- commands
  - CLI 3-15f
    - see also CLI commands
  - SED 4-1, 4-2ff (table)
    - see also SED commands
- CONTINUE argument 2-16, 3-20
  - see also BYE YES CONTINUE function key
- control characters 2-6f
- control (CTRL) key 1-4f
- conventions
  - command format iii
  - file naming 2-1
- copy text 4-16
  - see also DUPLICATE command
- CR, see carriage return
- create
  - blank page 3-12f
  - page break 4-28
  - page containing text 3-12
  - SAVE file 4-25
- CTRL-A 2-6, 3-8
- CTRL-B 2-6
- CTRL-C CTRL-A 2-6
- CTRL-E 2-6
- CTRL-F 2-6
- CTRL-H 2-7
- CTRL-I 2-7
- CTRL-J 2-7
- CTRL-M 2-7
- CTRL-Q 2-6
- CTRL-S 2-6
- CTRL-U 2-6
- CTRL-X 2-7
- CTRL-Y 2-7

- current line
  - change position 2-4ff, 4-22
  - description 1-2
  - display 3-5ff, 3-7 (table)
  - modify 2-7f
- cursor
  - action 3-7
  - control
    - keypad 1-4
    - keys 2-6f
  - definition 1-5
  - position on new line 3-8
  - representation in manual 1-5
- CUT command 4-12, 3-14, 4-2 (table)
- CUT WHILE EDITING function key 3-14, 4-2 (table), 4-12

**D**

- DASHER terminals
  - D1 A-1, 1-2
  - D2 1-2
  - D200 1-2
  - D400 A-1f, 1-2
  - D450 A-1f, 1-2
  - G300 A-2, 1-2
- default display settings 3-4ff
- define function keys 3-10f
  - see also SET command
- DEL, see delete (DEL) key
- delete (DEL) key 1-2
- DELETE command 4-13, 2-11f, 4-2 (table)
- DELETE CURRENT function key 2-12, 4-2 (table), 4-13
- deletion, restore last 4-29
- destination argument 4-13, 1-9
- dictionary, SED 3-19
- directory, change working 3-17, 4-6 (table), 4-14
- DIRECTORY command 4-14, 3-17, 4-6 (table)
- discard edits 3-19f
  - see also ABANDON command
- display
  - another page 2-3f
  - CLI 1-6f
  - current line 3-5ff
  - file status 1-6, 3-4, 3-12, 4-14
  - function key 1-6
  - HELP 1-7
  - line range 4-29f
  - lines (VIEW) 2-4

display (cont.)  
 lines in current page 2-3  
 page information 3-12  
 range of text 2-3f  
 SED screen 1-5ff  
 settings, see display settings  
 SPELL 1-7  
 text 1-6

**DISPLAY**  
 command 4-14, 3-4, 4-3 (table)  
 function key 4-14, 3-4, 3-12, 4-3 (table)

**DISPLAY FUNCTION** command 4-15, 1-6, 4-3

display settings  
 BLANK\_MODE 3-8, 3-6 (table)  
 change 3-4f, 4-11  
 clear 4-11  
 default 3-4ff, 3-6f (table)  
 definition 3-4  
 DISPLAY\_MODE 3-5ff  
 ESCAPE\_CHARACTER 3-9, 3-7 (table)  
 ESCAPE\_IS\_VIEW 3-9, 3-7 (table)  
 LINE\_NUMBER\_DISPLAY 3-9, 3-7 (table)  
 NEW\_LINE\_IS\_VIEW 3-9, 3-7 (table)  
 PRELOAD\_OLD\_LINE 3-8, 3-6 (table)  
 set 4-26  
 SKIP\_WHITE\_SPACE 3-8, 3-6 (table)  
 summary 3-6f (table)  
 TYPER\_MODE 3-7f, 3-6 (table)  
 UPPERCASE\_SHIFT\_CHARACTER 3-8f, 3-7 (table)  
 VIEW\_RANGE 3-5, 3-6 (table)

DISPLAY\_MODE, see display settings

DO CLI command 4-4 (table), 4-15

DO command 4-15, 1-6f, 3-15ff, 4-4 (table)

DO FILESTATUS command 3-15f

DO TYPE command 3-16

DO XEQ SED command 3-16

DOWN 11 LINES function key 4-3 (table), 2-5 (table), 4-22

DOWN 4 LINES function key 4-3 (table), 2-5 (table), 4-22

DOWN ONE SCREEN function key 4-3 (table), 2-5 (table), 4-22

downarrow 1-4, 3-7

DUPLICATE command 4-16, 2-10f, 4-2 (table)

duplicate text  
 onto another file 2-11  
 within current page 2-10f

## E

/ED command switch 3-2, 4-7

edit  
 control characters 2-7  
 existing file 3-3  
 file 1-2  
 text 4-20  
 see also MODIFY command

editing  
 advanced 3-1ff  
 elementary 2-1ff  
 session  
 abort 3-19f, 4-4 (table), 4-8  
 advanced, example 3-1ff  
 elementary, example 2-1ff  
 suspend 3-15ff, 4-12  
 terminate 2-15, 3-19f, 4-10

edits, discard 3-19f  
 see also ABANDON command

elementary editing 2-1ff

enter SED commands 2-2

ERASE EOL key 1-4

error messages B-1ff

ESC, see escape (ESC) key

escape (ESC) key 1-2

ESCAPE\_CHARACTER, see display settings

ESCAPE\_IS\_VIEW, see display settings

examples, how to use 2-1

EXECUTE command 4-16, 3-5, 3-17f, 4-6 (table)

EXECUTE SED  
 CLI command 2-1f, 4-7  
 command switches 3-2f

## F

file  
 auxiliary 3-1  
 backup 3-1  
 edit 1-2  
 edit existing 3-3  
 execute 3-17f  
 length, maximum 3-13  
 naming conventions 2-1  
 protect 2-16  
 save backup 2-15f  
 SED 3-1ff  
 status display 1-6, 3-4, 3-12, 4-14  
 temporary 3-1f  
 update 2-11, 2-15, 4-10

filename templates 3-16

FILESTATUS, CLI command 3-15f

## FIND

- command 4-17, 2-12f, 4-4 (table)
- function key 2-13, 4-4 (table), 4-17
- FIRST LINE ON PAGE function key 4-4 (table), 2-5 (table), 4-22
- fixed head disk 3-2
- form feeds, remove 3-3, 4-7
- format conventions, command iii
- function key
  - define and clear 3-9ff, 1-4, 4-26
  - display 1-6
  - mistakes 1-10
  - redefine 3-10f
  - templates 3-10 (fig.)
  - use 1-4f
- function keys
  - APPEND 2-2f, 4-2 (table), 4-9
  - BACKFIND 2-13f, 4-4 (table), 4-10
  - BOTTOM OF SCREEN 4-3 (table), 2-5 (table), 4-22
  - BYE YES CONTINUE 4-4 (table), 2-16, 4-10
  - CLI 4-4 (table), 4-12
  - CUT WHILE EDITING 3-14, 4-2 (table), 4-12
  - DELETE CURRENT 2-12, 4-2 (table), 4-13
  - DISPLAY 4-14, 3-4, 3-12, 4-3 (table)
  - DOWN 11 LINES 4-3 (table), 2-5 (table), 4-22
  - DOWN 4 LINES 4-3 (table), 2-5 (table), 4-22
  - DOWN ONE SCREEN 4-3 (table), 2-5 (table), 4-22
  - FIND 2-13, 4-4 (table), 4-17
  - FIRST LINE ON PAGE 4-4 (table), 2-5 (table), 4-22
  - GO TO PREVIOUS POSITION 4-3 (table), 2-5 (table), 4-22
  - INSERT CURRENT 2-8, 4-2 (table), 4-18
  - LAST LINE ON PAGE 4-4 (table), 2-5 (table), 4-22
  - local A-1f
  - MIDDLE LINE ON PAGE 4-4 (table), 2-5 (table), 4-22
  - MIDDLE OF SCREEN 4-3 (table), 2-5 (table), 4-22
  - MODIFY CURRENT 2-7, 4-2 (table), 4-20
  - PASTE WHILE EDITING 3-14f, 4-2 (table), 4-21
  - POSITION PAGE FIRST 4-4 (table), 2-5 (table), 4-22
  - POSITION PAGE LAST 4-4 (table), 2-5 (table), 4-22
  - POSITION PAGE NEXT 4-4 (table), 2-5 (table), 4-22
  - POSITION PAGE PREVIOUS 4-4 (table), 2-5 (table), 4-22
  - predefined 2-4f
  - screen position 2-4f, 3-7
  - TOP OF SCREEN 4-3 (table), 2-5 (table), 4-22
  - UNDO 2-12, 4-2 (table), 4-29
  - UP 11 LINES 4-3 (table), 2-5 (table), 4-22
  - UP 4 LINES 4-3 (table), 2-5 (table), 4-22
  - UP ONE SCREEN 4-3 (table), 2-5 (table), 4-22
  - VIEW 4-3 (table), 4-29f

## G

- generic listfile, definition 3-18
- GO TO PREVIOUS POSITION function key 4-3 (table), 4-22

## H

- HELP
  - command 4-17, 1-7f, 2-4, 4-3 (table)
  - display 1-7, 1-8 (fig.)
- HOME key 1-4, 2-7

## I

- informative messages B-1ff
- input mode, keyboard 1-1
- INSERT command 4-18, 2-8f, 4-2 (table)
- INSERT CURRENT function key 2-8f, 4-2 (table), 4-18
- insert text 2-8f

## J

- JOIN command 4-19, 3-13, 4-2 (table)

## K

- keyboard 1-2ff, A-1f
  - input mode 1-1
- keypad
  - cursor control 1-4
  - main 1-2ff
  - numeric 1-4
- keys
  - define function 3-9ff, 1-4, 4-26
  - function 1-4f
  - local function A-1f
  - screen position 2-4f, 1-4

## L

- last deletion, restore 4-29
- LAST LINE ON PAGE function key 4-4 (table), 2-5 (table), 4-22
- leftarrow 1-4, 2-7
- length
  - file 1-2, 3-13
  - line 1-1
  - page 1-2, 2-3
- line
  - change current 2-4ff, 4-22
  - current 1-2
  - definition 1-1
  - display current 3-5ff
  - display range 4-29f
  - length 1-1
  - numbers 1-1, 3-9
  - replace 2-14f, 4-2 (table), 4-24

LINE\_NUMBER\_DISPLAY, see display settings  
lines

- merge (PASTE) 3-14f, 4-21
- multiple command 2-2
- split (CUT) 3-14, 4-12

LIST command 4-19, 2-3f, 4-3 (table)

@LIST, printing text on 3-18, 4-23

listfile

- generic 3-18
- setting the 3-18

LISTFILE, CLI command 3-18

local function keys A-1f

locate text 2-13

- see also FIND and BACKFIND

lowercase 3-8

- see also case

## M

main keypad 1-2ff

merge lines 3-14f, 4-21

messages

- error B-1ff
- informative B-1ff

MIDDLE LINE ON PAGE function key 4-4 (table),  
2-5 (table), 4-22

MIDDLE OF SCREEN function key 4-3 (table), 2-5  
(table), 4-22

misspelling 3-3

mistakes

- command line 1-10
- function key 1-10
- spelling 3-3
- typing 1-10

mode

- command 1-1
- keyboard input 1-1

modify

- current line 2-7f
- line 4-20
- range 2-8, 4-20

MODIFY command 4-20, 2-7ff, 4-2 (table)

MODIFY CURRENT function key 2-7, 4-2 (table), 4-20

MOVE command 4-21, 2-9f, 4-2 (table)

move text 2-10

- see also MOVE command

multiple command lines 2-2

## N

NEW LINE key 1-2, iv

NEW\_LINE\_IS\_VIEW, see display settings

/NO\_ED command switch 3-2, 4-7

/NO\_FORM\_FEEDS command switch 3-3, 4-7

/NO\_SCREEN command switch 3-3, 4-7

numeric keypad 1-4

## O

organization of manual iii

## P

page

break

create 3-12f, 4-28

remove 3-3, 3-13, 4-19

definition 1-2

information display 3-12

length 1-2, 2-3

print a single 3-18

PASTE command 4-21, 3-14f, 4-2 (table)

PASTE WHILE EDITING function key 3-14f, 4-2  
(table), 4-21

PERMANENCE

CLI command 2-16

remove 2-16

position

changing current line 2-4ff, 4-22

character 1-1

keys, screen 2-4f, 1-4

POSITION command 4-22, 2-4ff, 4-3 (table)

POSITION PAGE FIRST function key 4-4 (table), 2-5  
(table), 4-22

POSITION PAGE LAST function key 4-4 (table), 2-5  
(table), 4-22

POSITION PAGE NEXT function key 4-4 (table), 2-5  
(table), 4-22

POSITION PAGE PREVIOUS function key 4-4 (table),  
2-5 (table), 4-22

PRELOAD\_OLD\_LINE, see display settings

print

range of lines 3-18

single page 3-18

text on @LIST 3-18, 4-23

PRINT command 4-23, 3-18, 4-3 (table)

/PROFILE command switch 3-2, 4-7

prompt

SED 1-5, 2-1

Son of SED 3-16f, 4-12

protect files 2-16

## Q

QPRINT, CLI command 3-18

## R

### range

- argument format 4-23
  - definition 1-9
  - delete 2-12
  - display 4-29f
  - examples 4-23
  - list 4-19
  - modify 2-8
  - print 3-18
  - rules 4-23
  - VIEW\_RANGE display setting 3-5f
- redefine function keys 3-10f
- redisplay screen 1-7
- refresh screen 1-7, 3-9
- remove PERMANENCE 2-16
- repeat (REPT) key 1-4
- replace
- lines 4-24
  - text 2-14f
- REPLACE command 4-24, 2-14f, 4-2 (table)
- REPT, see repeat (REPT) key
- restore
- last deletion 4-29
  - text 2-11
- rightarrow 1-4, 2-7

## S

- sample editing session 2-1ff, 3-1ff
- save backup file 2-15f
- SAVE
- command 4-25, 2-11, 3-1, 4-6
  - file, create 4-25
- screen
- displays, SED 1-5ff
  - position function keys 1-4, 2-4f, 3-7
  - refresh 1-7, 3-9
  - terminal 1-2
- searchstring argument 4-25, 1-9f
- SED
- arguments to commands 1-8ff, 4-2ff (table)
  - capabilities 1-1
  - continue with 2-16, 3-20
  - definition 1-1
  - display settings 3-6f (table)
  - entering commands 2-2
  - error messages B-1ff
  - execute from CLI 2-1f, 4-7
  - execute with command switches 3-2f
  - informative messages B-1ff
  - messages B-1ff
  - prompt 1-5, 2-1
  - requirements 1-1
  - screen displays 1-5ff
  - templates 1-4, iii, 3-10 (fig.)
- SED command summary
- adding text 4-2 (table)
  - changing display settings 4-5 (table)
  - changing function keys 4-6 (table)
  - changing text 4-2 (table)
  - deleting text 4-2 (table)
  - display information 4-3 (table)
  - exiting 4-4 (table)
  - listings 4-3 (table)
  - miscellaneous commands 4-6 (table)
  - opening a file 4-2 (table)
  - positioning 4-3f (table)
- SED commands
- entering 2-2
  - explanation, see HELP command
- SED dictionary, add to 3-19
- SED files
- .BU file 3-1
  - .ED file 3-1
  - edit file 3-1
  - .SC file 3-1
  - SED.DICTIONARY file 3-19
  - .SV file 3-1
  - .S2 file 3-1f
- session, sample editing 2-1ff, 3-1ff
- set
- display settings 4-26
  - function keys 3-9ff, 4-5f (table)
  - listfile 3-18
- SET command 4-26, 3-4f, 4-5f (table)
- see also display settings
- SET FUNCTION command 4-6 (table), 4-26
- settings, display 3-4ff
- see also display settings
- SHIFT key 1-4f
- SKIP\_WHITE\_SPACE, see display settings
- Son of SED prompt 3-16f, 4-12
- son processes 1-6, 3-15
- source argument 4-27, 1-9
- specify
- block of text 4-23
  - character string 4-25
- SPELL
- command 4-27, 1-7, 3-19
  - display 1-7
- spelling
- assistance 3-19, 1-7, 4-27
  - mistakes 3-3



- SPLIT command 4-28, 2-15, 3-12f, 4-2 (table)
- split lines 4-12
- string
  - specify 4-25
  - substitute 4-28
- subordinate process 3-15ff
- substitute
  - character strings 4-28
  - text across page boundaries 2-14
  - text within current page 2-14
- SUBSTITUTE command 4-28, 2-14, 4-2 (table)
- suspend editing session 3-15ff, 4-12
- switches, command 3-2f
- symbols, see command format conventions

## T

- TAB key 1-4, 2-7
- template
  - filename 3-16
  - SED iii, 1-4, 3-10 (fig.), A-1f
- temporary files 3-1f
- terminal
  - DASHER D1 A-1, 1-2
  - DASHER D2 1-2
  - DASHER D200 1-2
  - DASHER D400 A-1f, 1-2
  - DASHER D450 A-1f, 1-2
  - DASHER G300 A-2, 1-2
  - differences A-1f
  - screen 1-2
- terminate editing session 2-15, 3-19f, 4-10
- text
  - add 2-2f, 4-9
  - append 2-2f, 4-9
  - change 2-7ff, 4-20
  - copy 4-16
  - delete 2-11f, 4-13
  - display 1-6
  - duplicate 2-10f, 4-16
  - edit 2-7ff, 4-20
  - insert 2-8f, 4-18
  - locate 2-12ff, 4-10, 4-17
  - modify 2-7ff, 4-20
  - move 2-9f, 4-21
  - print on @LIST 3-18, 4-23
  - replace 4-24
  - restore 2-11
  - source 4-27
  - substitute 2-14
  - units of 1-1f

- TOP OF SCREEN function key 4-3 (table), 2-5 (table), 4-22
- TYPE, CLI command 3-16
- TYPED\_MODE, see display settings

## U

- UNDO
  - command 4-29, 2-12, 4-2 (table)
  - function key 2-12, 4-2 (table), 4-29
- units of text 1-1f
- UP 11 LINES function key 4-3 (table), 2-5 (table), 4-22
- UP 4 LINES function key 4-3 (table), 2-5 (table), 4-22
- UP ONE SCREEN function key 4-3 (table), 2-5 (table), 4-22
- uparrow 1-4, 3-7
- update
  - file
    - BYE command 2-15, 4-10
    - SAVE command 2-11
    - screen, not automatic (/NO\_SCREEN switch) 3-3, 4-7
- upper mode line 3-9
- uppercase 3-8
  - see also case
- uppercase only terminal A-1
- UPPERCASE\_SHIFT\_CHARACTER, see display settings
- user profile 1-6, 3-15

## V

- VIEW
  - command 4-29f, 2-4, 4-3 (table)
  - function key 2-4, 4-3 (table), 4-29f
- VIEW\_RANGE, see display settings

## W

- window 2-8
- /WORK command switch 3-2f, 4-7
- working directory, change 3-17, 4-14

## X

- XEQ, see EXECUTE command
- XEQ SED, see EXECUTE SED



# Data General Users group

## Installation Membership Form

Name \_\_\_\_\_ Position \_\_\_\_\_ Date \_\_\_\_\_

Company, Organization or School \_\_\_\_\_

Address \_\_\_\_\_ City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_

Telephone: Area Code \_\_\_\_\_ No. \_\_\_\_\_ Ext. \_\_\_\_\_

### 1. Account Category

- OEM  
 End User  
 System House  
 Government

### 5. Mode of Operation

- Batch (Central)  
 Batch (Via RJE)  
 On-Line Interactive

### 2. Hardware

M/600  
 MV/Series ECLIPSE®  
 Commercial ECLIPSE  
 Scientific ECLIPSE  
 Array Processors  
 CS Series  
 NOVA®4 Family  
 Other NOVAs  
 microNOVA® Family  
 MPT Family

Qty. Installed	Qty. On Order
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

Other \_\_\_\_\_  
 (Specify) \_\_\_\_\_

### 6. Communication

- HASP       X.25  
 HASP II     SAM  
 RJE80      CAM  
 RCX 70     XODIAC™  
 RSTCP      DG/SNA  
 4025       3270  
 Other

Specify \_\_\_\_\_

### 3. Software

- AOS       RDOS  
 AOS/VS    DOS  
 AOS/RT32  RTOS  
 MP/OS     Other  
 MP/AOS

Specify \_\_\_\_\_

### 7. Application Description

○ \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_

### 4. Languages

- ALGOL     BASIC  
 DG/L      Assembler  
 COBOL    FORTRAN 77  
 Interactive COBOL  FORTRAN 5  
 PASCAL    RPG II  
 Business BASIC    PL/1  
 APL  
 Other

Specify \_\_\_\_\_

### 8. Purchase

From whom was your machine(s) purchased?

- Data General Corp.  
 Other \_\_\_\_\_  
 \_\_\_\_\_

### 9. Users Group

Are you interested in joining a special interest or regional Data General Users Group?

○ \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_

CUT ALONG DOTTED LINE

FOLD

FOLD

TAPE

TAPE

FOLD

FOLD



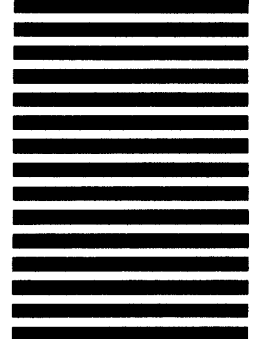
NO POSTAGE  
NECESSARY  
IF MAILED  
IN THE  
UNITED STATES

**BUSINESS REPLY MAIL**  
FIRST CLASS PERMIT NO. 26 SOUTHBORO, MA. 01772

Postage will be paid by addressee:

 **Data General**

ATTN: Users Group Coordinator (C-228)  
4400 Computer Drive  
Westboro, MA 01581





**DATA GENERAL CORPORATION  
TECHNICAL INFORMATION AND PUBLICATIONS SERVICE  
TERMS AND CONDITIONS**

Data General Corporation ("DGC") provides its Technical Information and Publications Service (TIPS) solely in accordance with the following terms and conditions and more specifically to the Customer signing the Educational Services TIPS Order Form shown on the reverse hereof which is accepted by DGC.

**1. PRICES**

Prices for DGC publications will be as stated in the Educational Services Literature Catalog in effect at the time DGC accepts Buyer's order or as specified on an authorized DGC quotation in force at the time of receipt by DGC of the Order Form shown on the reverse hereof. Prices are exclusive of all excise, sales, use or similar taxes and, therefore are subject to an increase equal in amount to any tax DGC may be required to collect or pay on the sale, license or delivery of the materials provided hereunder.

**2. PAYMENT**

Terms are net cash on or prior to delivery except where satisfactory open account credit is established, in which case terms are net thirty (30) days from date of invoice.

**3. SHIPMENT**

Shipment will be made F.O.B. Point of Origin. DGC normally ships either by UPS or U.S. Mail or other appropriate method depending upon weight, unless Customer designates a specific method and/or carrier on the Order Form. In any case, DGC assumes no liability with regard to loss, damage or delay during shipment.

**4. TERM**

Upon execution by Buyer and acceptance by DGC, this agreement shall continue to remain in effect until terminated by either party upon thirty (30) days prior written notice. It is the intent of the parties to leave this Agreement in effect so that all subsequent orders for DGC publications will be governed by the terms and conditions of this Agreement.

**5. CUSTOMER CERTIFICATION**

Customer hereby certifies that it is the owner or lessee of the DGC equipment and/or licensee/sub-licensee of the software which is the subject matter of the publication(s) ordered hereunder.

**6. DATA AND PROPRIETARY RIGHTS**

Portions of the publications and materials supplied under this Agreement are proprietary and will be so marked. Customer shall abide by such markings. DGC retains for itself exclusively all proprietary rights (including manufacturing rights) in and to all designs, engineering details and other data pertaining to the products described in such publication. Licensed software materials are provided pursuant to the terms and conditions of the Program License Agreement (PLA) between the Customer and DGC and such PLA is made a part of and incorporated into this Agreement by reference. A copyright notice on any data by itself does not constitute or evidence a publication or public disclosure.

**7. DISCLAIMER OF WARRANTY**

DGC MAKES NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR PARTICULAR PURPOSE ON ANY OF THE PUBLICATIONS SUPPLIED HEREUNDER.

**8. LIMITATIONS OF LIABILITY**

IN NO EVENT SHALL DGC BE LIABLE FOR (I) ANY COSTS, DAMAGES OR EXPENSES ARISING OUT OF OR IN CONNECTION WITH ANY CLAIM BY ANY PERSON THAT USE OF THE PUBLICATION OF INFORMATION CONTAINED THEREIN INFRINGES ANY COPYRIGHT OR TRADE SECRET RIGHT OR (II) ANY INCIDENTAL, SPECIAL, DIRECT OR CONSEQUENTIAL DAMAGES WHATSOEVER, INCLUDING BUT NOT LIMITED TO LOSS OF DATA, PROGRAMS OR LOST PROFITS.

**9. GENERAL**

A valid contract binding upon DGC will come into being only at the time of DGC's acceptance of the referenced Educational Services Order Form. Such contract is governed by the laws of the Commonwealth of Massachusetts. Such contract is not assignable. These terms and conditions constitute the entire agreement between the parties with respect to the subject matter hereof and supersedes all prior oral or written communications, agreements and understandings. These terms and conditions shall prevail notwithstanding any different, conflicting or additional terms and conditions which may appear on any order submitted by Customer.

**DISCOUNT SCHEDULES**

**DISCOUNTS APPLY TO MAIL ORDERS ONLY.**

**LINE ITEM DISCOUNT**

5-14 manuals of the same part number - 20% 15 or more manuals of the same part number - 30%
--

**DISCOUNTS APPLY TO PRICES SHOWN IN THE CURRENT TIPS CATALOG ONLY.**



# DATA GENERAL CORPORATION TECHNICAL INFORMATION AND PUBLICATIONS SERVICE TERMS AND CONDITIONS

Data General Corporation ("DGC") provides its Technical Information and Publications Service (TIPS) solely in accordance with the following terms and conditions and more specifically to the Customer signing the Educational Services TIPS Order Form shown on the reverse hereof which is accepted by DGC.

**1. PRICES**

Prices for DGC publications will be as stated in the Educational Services Literature Catalog in effect at the time DGC accepts Buyer's order or as specified on an authorized DGC quotation in force at the time of receipt by DGC of the Order Form shown on the reverse hereof. Prices are exclusive of all excise, sales, use or similar taxes and, therefore are subject to an increase equal in amount to any tax DGC may be required to collect or pay on the sale, license or delivery of the materials provided hereunder.

**2. PAYMENT**

Terms are net cash on or prior to delivery except where satisfactory open account credit is established, in which case terms are net thirty (30) days from date of invoice.

**3. SHIPMENT**

Shipment will be made F.O.B. Point of Origin. DGC normally ships either by UPS or U.S. Mail or other appropriate method depending upon weight, unless Customer designates a specific method and/or carrier on the Order Form. In any case, DGC assumes no liability with regard to loss, damage or delay during shipment.

**4. TERM**

Upon execution by Buyer and acceptance by DGC, this agreement shall continue to remain in effect until terminated by either party upon thirty (30) days prior written notice. It is the intent of the parties to leave this Agreement in effect so that all subsequent orders for DGC publications will be governed by the terms and conditions of this Agreement.

**5. CUSTOMER CERTIFICATION**

Customer hereby certifies that it is the owner or lessee of the DGC equipment and/or licensee/sub-licensee of the software which is the subject matter of the publication(s) ordered hereunder.

**6. DATA AND PROPRIETARY RIGHTS**

Portions of the publications and materials supplied under this Agreement are proprietary and will be so marked. Customer shall abide by such markings. DGC retains for itself exclusively all proprietary rights (including manufacturing rights) in and to all designs, engineering details and other data pertaining to the products described in such publication. Licensed software materials are provided pursuant to the terms and conditions of the Program License Agreement (PLA) between the Customer and DGC and such PLA is made a part of and incorporated into this Agreement by reference. A copyright notice on any data by itself does not constitute or evidence a publication or public disclosure.

**7. DISCLAIMER OF WARRANTY**

DGC MAKES NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR PARTICULAR PURPOSE ON ANY OF THE PUBLICATIONS SUPPLIED HEREUNDER.

**8. LIMITATIONS OF LIABILITY**

IN NO EVENT SHALL DGC BE LIABLE FOR (I) ANY COSTS, DAMAGES OR EXPENSES ARISING OUT OF OR IN CONNECTION WITH ANY CLAIM BY ANY PERSON THAT USE OF THE PUBLICATION OF INFORMATION CONTAINED THEREIN INFRINGES ANY COPYRIGHT OR TRADE SECRET RIGHT OR (II) ANY INCIDENTAL, SPECIAL, DIRECT OR CONSEQUENTIAL DAMAGES WHATSOEVER, INCLUDING BUT NOT LIMITED TO LOSS OF DATA, PROGRAMS OR LOST PROFITS.

**9. GENERAL**

A valid contract binding upon DGC will come into being only at the time of DGC's acceptance of the referenced Educational Services Order Form. Such contract is governed by the laws of the Commonwealth of Massachusetts. Such contract is not assignable. These terms and conditions constitute the entire agreement between the parties with respect to the subject matter hereof and supersedes all prior oral or written communications, agreements and understandings. These terms and conditions shall prevail notwithstanding any different, conflicting or additional terms and conditions which may appear on any order submitted by Customer.

## DISCOUNT SCHEDULES

**DISCOUNTS APPLY TO MAIL ORDERS ONLY.**

### LINE ITEM DISCOUNT

5-14 manuals of the same part number - 20% 15 or more manuals of the same part number - 30%
--

**DISCOUNTS APPLY TO PRICES SHOWN IN THE CURRENT TIPS CATALOG ONLY.**





## **TIPS ORDERING PROCEDURE:**

Technical literature may be ordered through the Customer Education Service's Technical Information and Publications Service (TIPS).

1. Turn to the TIPS Order Form.
2. Fill in the requested information. If you need more space to list the items you are ordering, use an additional form. Transfer the subtotal from any additional sheet to the space marked "subtotal" on the form.
3. Do not forget to include your MAIL ORDER ONLY discount. (See discount schedules on the back of the TIPS Order Form.)
4. Total your order. (MINIMUM ORDER/CHARGE after discounts of \$50.00.)

If your order totals less than 100.00, enclose a certified check or money order for the total (include sales tax, or your tax exempt number, if applicable) plus \$5.00 for shipping and handling.

5. Please indicate on the Order Form if you have any special shipping requirements. Unless specified, orders are normally shipped U.P.S.
6. Read carefully the terms and conditions of the TIPS program on the reverse side of the Order Form.
7. Sign on the line provided on the form and enclose with payment. Mail to:

TIPS  
Educational Services - M.S. F019  
Data General Corporation  
4400 Computer Drive  
Westboro, MA 01580

8. We'll take care of the rest!





# User Documentation Remarks Form

Your Name \_\_\_\_\_ Your Title \_\_\_\_\_  
 Company \_\_\_\_\_  
 Street \_\_\_\_\_  
 City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_

We wrote this book for you, and we made certain assumptions about who you are and how you would use it. Your comments will help us correct our assumptions and improve the manual. Please take a few minutes to respond. Thank you.

Manual Title \_\_\_\_\_ Manual No. \_\_\_\_\_

Who are you?  EDP Manager  Analyst/Programmer  Other \_\_\_\_\_  
 Senior Systems Analyst  Operator \_\_\_\_\_

What programming language(s) do you use? \_\_\_\_\_

How do you use this manual? (List in order: 1 = Primary Use) \_\_\_\_\_

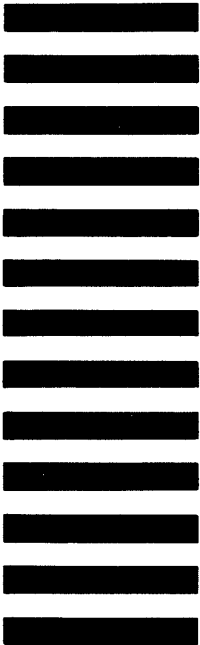
Introduction to the product  Tutorial Text  Other \_\_\_\_\_  
 Reference  Operating Guide \_\_\_\_\_

About the manual:		Yes	Somewhat	No
Is it easy to read?		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Is it easy to understand?		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Are the topics logically organized?		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Is the technical information accurate?		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Can you easily find what you want?		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Does it tell you everything you need to know?		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Do the illustrations help you?		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

If you have any comments on the software itself, please contact Data General Systems Engineering.  
 If you wish to order manuals, use the enclosed TIPS Order Form (USA only).

Remarks:

Date



User Documentation, M.S. E-111  
4400 Computer Drive  
Westborough, Massachusetts 01581



POSTAGE WILL BE PAID BY ADDRESSEE

**BUSINESS REPLY MAIL**  
FIRST CLASS PERMIT NO. 26 SOUTHBORO, MA. 01772

NO POSTAGE  
NECESSARY  
IF MAILED  
IN THE  
UNITED STATES





**Data General Corporation, Westboro, MA 01580**



093-000249-02