

codex
A Subsidiary of  MOTOROLA INC.

DOC

Document Processor

User's Guide

Document No. 72745a



c 1980 Codex Corporation
Intelligent Terminal Systems
All rights reserved

First Edition
October 31, 1980

Preface

This user's guide is a complete course of instruction, intended for the beginning as well as experienced operator, in the Codex Document Processor (DOC). After reading this guide, the operator should be able to utilize all DOC commands, including page format, line format, and input/output commands. This guide also includes samples of formatted text using DOC commands.

Other Codex publications that may be of interest include the Codex Disk Operating System (CODOS) User's Guide, the CODOS Reference Manual, the EDITOR User's Guide, and the Hardware Reference Manuals appropriate to the user's system configuration.

This manual is valid for Versions 4.0 through 5.1 of the Document Processor only.

Table of Contents	Page
CHAPTER 1. INTRODUCTION	1-01
Hardware Support Required	1-01
Optional Hardware Supported	1-02
Software Support Required	1-02
Software Installation	1-02
DOC Features	1-02
CHAPTER 2. GENERAL CONCEPTS	2-01
Spaces and Punctuation	2-03
Default Values	2-04
DOC Command Format	2-04
CHAPTER 3. PAGE FORMAT COMMANDS	3-01
Heading	3-01
HL - Specify Heading Length	3-02
BH - Begin Heading Text	3-02
Body	3-03
BL - Specify Body Length	3-03
Footing	3-03
FL - Specify Footing Length	3-03
BF - Begin Footing Text	3-04
Page Numbers	3-04
AL - Set Absolute Left Margin for Page Numbers	3-05
AR - Set Absolute Right Margin for Page Numbers	3-05
PB - Position Page Numbers at Bottom of Page	3-06
PT - Position Page Numbers at Top of Page	3-07
PE - Page Number End on This Page	3-08
RN - Use Roman Numerals for Page Numbers	3-08
OP - Optional "Page," Print Only Numbers	3-09
RP - Resume Printing of "Page"	3-09
Page Format Defaults Summary	3-10
CHAPTER 4. LINE FORMAT COMMANDS	4-01
Horizontal Spacing Control	4-02
LM - Set Left Margin	4-02
RM - Set Right Margin	4-02
PL - Purge Current Output Line	4-04
SP - Insert Spaces in Line	4-05
TC - Clear Tab Stops	4-05
TS - Set Tab Stop	4-05
TT - Tab to Logical Tab Stop	4-07
TB - Tab to Absolute Column	4-07
LT - Set Left Margin to Logical Tab Stop	4-08
RT - Set Right Margin to Logical Tab Stop	4-08
TF - Tab with Filler	4-09
LD - Define Leader Characters	4-10
Vertical Control	4-11
NL - New Line	4-11
NE - New Line, Even Margins	4-12
SL - Skip Lines	4-12

Table of Contents	Page
PP - Form Paragraph	4-13
IS - Number of Spaces for Paragraph Indentation	4-14
PG - Skip to Top of Next Page	4-14
PS - Conditional Skip to Top of Next Page	4-14
Line Justification	4-15
LJ - Left Justify Text	4-15
RJ - Right Justify Text	4-16
CN - Center Subsequent Text	4-17
Em - Form Even Margins	4-18
 CHAPTER 5. AS-IS TEXT COMMANDS	 5-01
AS - Begin Subsequent Text As-Is	5-01
// - End As-Is Text	5-02
 CHAPTER 6. INPUT/OUTPUT CONTROL COMMANDS	 6-01
KY - Switch Input Stream to Console Keyboard	6-01
TY - Switch Output Stream to Console Typewriter	6-02
LP - Switch Output Stream to Line Printer	6-02
QP - Switch Output Stream to Text Quality Printer	6-02
OS - Return Control to CODOS, Terminating DOC	6-03
CM - Set Printing Mode to Continuous	6-03
NC - Set Printing Mode to Non-Continuous	6-03
Fi - Assign Diskette File to Logical Input File i	6-03
Di - Switch Input Stream to Logical Input File i	6-04
Ri - Rewind Logical Input File i	6-05
Ai - Print or Display Logical Input File i As-Is	6-05
FR - Return Input Stream to Logical Input File	6-06
** - Display Comment on Console	6-06
PR - Restart Printing at Specified Page Number	6-06
Multiple File Processing	6-07
File Recursion	6-08
 CHAPTER 7. TABLE OF CONTENTS COMMAND	 7-01
F0 - Open Table of Contents File	7-01
C0 - Close Table of Contents File	7-02
BE - Begin Table of Contents Entry	7-03
P0 - Insert As-Is Text to Format Table of Contents	7-03
Creating a Table of Contents	7-04
 CHAPTER 8. MESSAGES AND COMMAND ERRORS	 8-01
Command Errors	8-02
 APPENDIX A. DOC COMMAND SET	 A-01
Command Alphabetic Listing	A-01
Command Listing by Category	A-02
Page Format Commands	A-02
Line Format Commands	A-03
Horizontal Spacing	A-03
Vertical Spacing	A-03

Table of Contents

Page

Line Justification	A-03
As-is Text	A-03
Input/Output	A-04
Table of Contents	A-04
APPENDIX B. DOC DEFAULT CONDITIONS	B-01
APPENDIX C. DOC EXAMPLES	C-01
APPENDIX D. GLOSSARY	D-01

CHAPTER 1. INTRODUCTION

The Codex Document Processor (DOC), in conjunction with the EDITOR and the Disk Operating System, provides the CDX-68 Display System with a text processing capability. This processing capability arranges the contents of a text file in a format designed by the user. The operator uses the EDITOR to create the input text files. Embedded in the text files are the DOC commands that arrange the text for display and/or printing. The input text files are stored on a disk or diskette and can be recalled upon request.

DOC permits users to gain substantial savings in both the time and effort required to produce business documentation. Text is entered only once and stored on disk or diskette. As many copies as required can then be produced on the system printer. When changes are required, only the corrections are typed to produce an updated document. In addition, commands permit specified text (standard introductions, proposal boilerplate, and so on) to be inserted into documents at any point, further speeding operations. Extensive operator prompting capabilities permit these insertions quickly and easily by even relatively untrained operators.

In a typical text processing application, the operator uses the EDITOR text editing program to compose the document. Text is entered through the typewriter-style keyboard of the Codex Intelligent Terminal System and viewed on the display screen, where corrections and revisions may be made. DOC commands are entered along with the text to serve as instructions for formatting the document. A simple command instructs DOC to process the text file, interpret the format commands, and send a completely formatted document to the printer, disk, or terminal display screen.

DOC also has a multiple file feature. This allows the user to print standard form letters as originals, one for every name and address supplied by another file.

This document assumes that the user is familiar with the EDITOR's methods of file building, saving, and loading.

Hardware Support Required

The minimum hardware configuration required to support DOC consists of:

- CDX-68 Basic Display Terminal with the appropriate firmware options
- 32K bytes of user memory (RAM)

- Microcomputer Module D
- System Self-Test firmware package
- 10 Mb Disk or 1 Mb Diskette Storage

Optional Hardware Supported

DOC also supports a variety of printers, including Matrix and Character printers (the Codex SP Series).

Software Support Required

No additional software is required to run DOC as it comes shipped on the system disk.

Software Installation

There is no software installation that need be performed. All DOC software is on the disk containing the selected software package. This disk contains DOC as well as the operating system and all programming languages and other software in the selected software package.

DOC Features

DOC provides the following features:

- . Joining of files together (i.e., chapters of a book)
- . Multiple file input capability (ie., form letters/address file)
- . Automatic table of contents file
- . Automatic printing of multiple copies
- . Restart printing capability
- . Heading/footering text specification
- . Automatic page numbering at top or bottom of page
- . Roman numeral page numbers for introductory material
- . Optional left or right justification of text
- . Optional centering of text
- . Specification of tab stops

CHAPTER 2. GENERAL CONCEPTS

DOC is a mechanism for formatting and then producing a finished document. The document is entered prior to using DOC and is stored on a diskette or disk. In order to understand how DOC works, it is helpful to compare the method of preparing a document using a typewriter to that of preparing a document using the EDITOR and DOC.

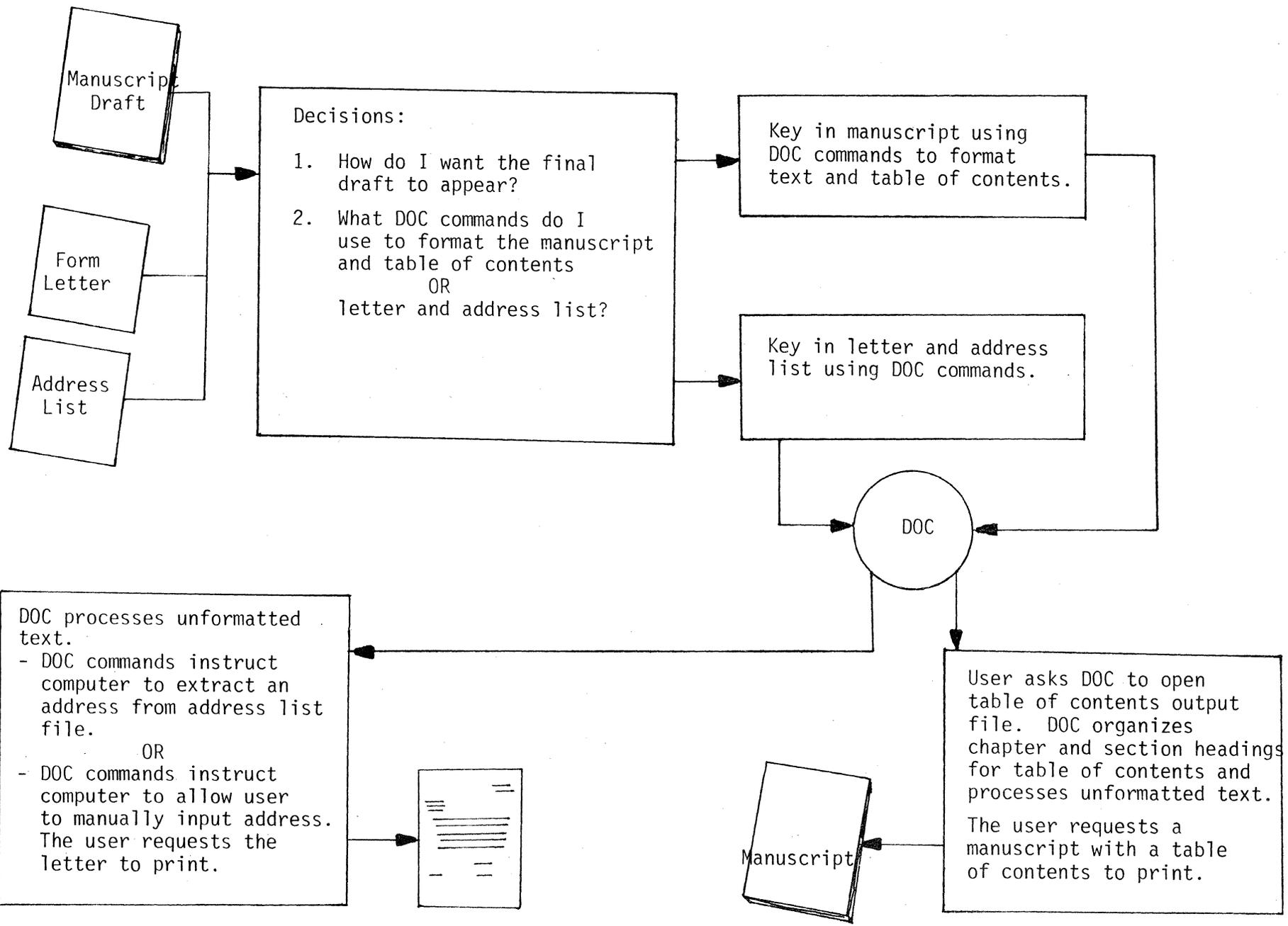
When a document is to be manually typed on a conventional typewriter, the operator reviews the document and sets the margins and tab stops. If the text involves tables, lists, and diagrams, the typist must take even more time to arrange the text.

With DOC, the operator also reviews the document and decides on the format. When the operator enters the document into the terminal, specific commands are placed into the text of the document. These commands tell DOC how to arrange the contents of the document. For example, the operator does not have to manually line space because once a command is entered in the text, DOC automatically arranges the desired spacing. With DOC, the operator is able to produce a final document quickly and efficiently; the operator's time is now devoted to entering the document, not arranging it.

When all of the text is entered and stored on disk or diskette, DOC is called into use. The DOC commands entered along with the text, are then executed by DOC, and the text file is formatted into a finished document.

When preparing the text file using the EDITOR, DOC commands are entered in the places within the text where they are to take effect. Since DOC commands are effective immediately, (except for margin commands), the placement of the DOC commands is crucial to the text formatting process. When executed, a DOC command acts immediately upon the line of text where it is placed. Some DOC commands are effective until terminated by an identical command with new variables, by a termination command, or by a command with new conditions. Other DOC commands are effective only once and must be re-entered each time that particular command is needed. DOC commands do not appear in the final printed version, unless specially requested (see As-Is).

In order to expedite the use of DOC, the user should scan the draft of the document and determine what DOC commands are needed to format the document. This includes chapter and section headings, whether a table of contents is required, text headings and footings, and spacing and tabbing requirements. It is helpful to do some advance planning before you start to key in the text.



Appendix A lists the DOC commands alphabetically and categorically. Examples of the use of DOC commands are contained in Appendix C. This manual frequently uses the terms, "input text" and "output text" in examples showing the use and the consequences of certain DOC commands. Input text refers to the text entered by the operator; output text refers to the text after it has been arranged by DOC.

Spaces and Punctuation

If several spaces (two or more consecutive spaces) are entered into a line of input text, DOC treats the several spaces as if they were a single space. These are the exceptions: spaces following a period (.), colon (:), question mark (?), or exclamation mark (!).

When the above punctuation characters are used, a space character is inserted into the input text following the character. If the user wishes to have two spaces follow a punctuation character, a second space must be entered. If more than two spaces follow a punctuation character, DOC treats them as exactly two spaces. For example.

Input Text:

Turn on power. Insert diskette.

Output Text:

Turn on power. Insert diskette.

Input Text:

First.Second.
 First. Second.
 First. Second.
 First. Second.
 First. Second.

Output Text:

First.Second.
 First. Second.
 First. Second.
 First. Second.
 First. Second.

The carriage return, (the key labelled "RETURN,") when used at the end of each line, is treated as a single space.

Note that if multiple spaces are to appear in a line of output text, they must be specified by the "SP" command or by tab specification commands. These commands are explained later on in this guide.

Default Values

If you do not use certain DOC commands when entering your text, DOC uses certain default values and formats your document accordingly. The following list summarizes the most important default conditions.

<u>Format Aspect</u>	<u>Default Value</u>
Left Margin	Column 10
Right Margin	Column 70
Line Justification	Left
Heading Length	7 lines
Body Length	54 lines
Footing Length	5 lines
Paragraph Indentation	5 columns
Page Printing	Continuous

Appendix B lists all default conditions. If a DOC command is not used for a particular format aspect, the default value, as listed in Appendix B is assumed.

DOC Command Format

A DOC command consists of two characters preceded by a reverse slash "\" sign, with no embedded spaces. For example:

```
\pp (form paragraph)
```

The majority of the DOC command characters are alphabetic; a few use numeric and special characters. It is important to note that the alphabetic characters may be entered in either upper or lower case. All DOC commands are listed in Appendix A. The operator should note that CODOS has a "=" prompt, while DOC has a "?" prompt.

Several of the DOC commands require arguments. Arguments are the numbers that define the command. The argument is entered with no space between the command characters and the argument. A typical example is: \ppnnn.

A space must follow the argument if the argument is shorter than the allowed maximum, or if the subsequent text character could be mistaken for another argument character.

If a "\" sign is to be entered into the input text as part of the text, it must be entered as "\\." DOC interprets the double reverse slash as a single reverse slash.

The following example letter has DOC commands in place. When this letter is printed, the DOC processor formats the letter according to the given commands and produces a finished letter. All of the DOC commands are explained further on in this Guide. This example is just to give you an idea of how an unformatted document with DOC commands appears.

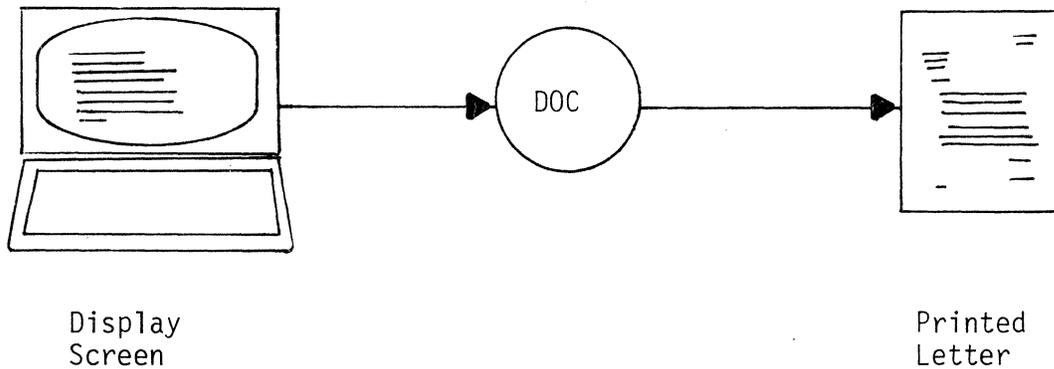
Input Text:

```

\lm10\rm70\s13
Converging Lines, Inc.
\nl
Ms. Teri Herbert
\nl
11938 East Rhode Island School of Design Terrace
\s12
Dear Ms. Herbert
\s12
Thank you for your interest in WorldWide Rulers' line of
measuring devices. We offer the eighteen-inch rulers you
inquired about in several different materials:
\s12\tc\tl5\tl25\lm25\pl\ttl
Wood
\tt2
- certainly the most natural material, and a favorite
from long ago
\s12\ttl
Plastic
\tt2
- the most popular material today, and available
in a wide variety of colors
\s12\ttl
Metal
\tt2 - the most precise and durable material chosen by
most professionals
\s11\lm10\s11
We can also customize your order with the imprinting of
your company's logo or other design.
\s12
But the most important reason for ordering from WorldWide is
our prompt, friendly service. We hope to, dare we say it,
rule the world, and your continued business is the only way
we can achieve our goal.
\s12
Thanks again for your interest, and we hope to be hearing
from you soon.
\s12\sp20
Sincerely yours,
\s15\sp20
Raoul Linear
\nl\sp20
Customer Service Manager

```

Several things should be noted about the example. The input text portion is unformatted. No attention is placed on the number of spaces between words, lines, columns. Since the actual formatting is done only when DOC processes the text file, the operator is free to enter the text in any manner. It is the presence of the DOC commands that indicate where the lines end, where columns begin, and so on.



CHAPTER 3. PAGE FORMAT COMMANDS

Before a final copy of a page can be produced, the user must be aware of the requirements of the page. The user must plan to size the heading, the body, and the footing of the page. If page numbers are to be included, they must be considered also. Note that the heading and footing are blocks of lines that may or may not have explanatory text; the body is a block of lines available for text.

If DOC commands for sizing those page components are omitted, default lengths are automatically assigned.

Page numbering is omitted unless specifically requested using a DOC command. If page numbering is specified, the positioning of page numbers can be controlled by DOC commands or by default values.

The list of DOC page format commands is as follows:

Command -----	Function -----
HLnnn	Specify heading length
BHtext	Begin heading text
BLnnn	Specify body length
FLnnn	Specify footing length
BFtext	Begin footing text
ALnnn	Set absolute left margin (page number)
ARnnn	Set absolute right margin (page number)
PTxxxxx	Position page number at top of page
PBxxxxx	Position page number at bottom of page
PE	Page numbers end on this page
OP	Optional "Page," print only numbers
RP	Resume printing "Page"
RNnnn	Use roman numerals for page numbers

NOTE: The alphabetic characters used in DOC commands may be in either upper or lower case.

Heading

The heading of a page consists of a block of lines at the top of the page. If heading text is inserted in the heading block, it appears on the fourth line of the heading. If the heading block is less than four lines, a heading text cannot be inserted. Note that page numbers, if specified for the top of the page, appear in the heading. Refer to the "PT" command.

HL - Specify Heading Length

Format: \HLnnn

Where: "nnn" is a number specifying the number of lines in the heading.

Note: An error occurs if the number is missing. A value of zero is valid.

The default heading length is 7 lines.

BH - Begin Heading Text

Format: \BHtext

Where: "text" is the data to be inserted in heading.

Note: Text appears as-is, starting in column 1 of the fourth line of the heading block. If you wish to align the heading with whatever the left margin is set at, insert spaces in front of your heading text. For example, if your left margin is at 10, insert 9 spaces in front of your heading text in order to begin heading in column 10.

Text is terminated by either a "\" sign or a carriage return.

DOC commands or "\" signs cannot be inserted in the heading text.

Whenever another "BH" command and its text is encountered, the current heading text is changed to the new text.

If a "BH" command is given with text consisting only of spaces or a carriage return, the effect is to turn off the insertion of text in the headings of subsequent pages.

Initially, the heading block is blank, i.e., if no "BH" command and accompanying text is encountered, no text appears in the heading block until the first "BH" command with its text is encountered.

Body

The body consists of the actual text of the document. The default body size is 54 lines. If the default heading length is used, the eighth line on the page is the first line of the body. When the last line of the body prints, the footing automatically prints, and the formatted output advances to the top of a new page. Chapter 4 details the commands that control the format for the text within the body of a page.

BL - Specify Body Length

Format: `\BLnnn`

Where: "nnn" is a number specifying the number of lines in the body.

Note: An error occurs if the number is missing or is given a value of zero.

If the heading and footing are set to zero lines, continuous text is produced with no intervening page numbers, headings, or footings appearing on pages.

The default body length is 54 lines.

Footing

The footing consists of a block of lines at the bottom of each page. If footing text is inserted in the footing block, it appears on the third line of the footing. If the footing block is made to be less than three lines, footing text cannot be inserted. Note that page numbers, if specified for the bottom of the page, appear in the footing. Refer to the "PB" command in this chapter.

FL - Specify Footing Length

Format: `\FLnnn`

Where: "nnn" is a number specifying the number of lines in the footing.

Note: An error occurs if the number is missing. A value of zero is valid.

The default footing length is 5 lines.

BF - Begin Footing Text

Format: \BF text

Where: "text" is the data to be inserted in footing.

Note: Text appears as-is, starting in column 1 in the third line of the footing block. If you wish to align the footing with whatever the left margin is set at, insert spaces in front of the footing text. For example, if your left margin is at 10, insert 9 spaces in front of your footing text in order to begin in column 10.

Text is terminated by either a "\" sign or a carriage return.

DOC commands or "\" signs cannot be inserted in the footing text.

Whenever another "BF" command and its text is encountered, the current footing text is changed to the new text.

If a "BF" command is given with text consisting only of spaces or a carriage return, the effect is to turn off the insertion of text in the footings of subsequent pages.

Initially, the footing block is blank; i.e., if no "BF" command and accompanying text is encountered, no text appears in the footing block until the first "BF" command with its text is encountered.

Page Numbers

DOC has six commands that affect page numbers. DOC suppresses page numbers unless a page numbering command is given. Page numbers can be positioned at the top of the page or at the bottom of the page. A DOC command also exists for the positioning of lower case roman numerals which can only be positioned at the bottom of the page.

DOC has two commands that position a page number in either the heading or the footing. An absolute left and absolute right margin position a page number in either the heading or footing text line (if page numbers are to appear). These margins do not affect the placement or format of text in the heading, body, or footing.

The page numbers are placed in the appropriate line after the heading or footing text is moved into the line. Thus, it is possible for the page number to overlay part of the heading or footing text, if positioning is carelessly done. The page numbers are always placed so that they are centered between the current values of the absolute left and right margins. Note that the word "Page" always accompanies the page number, unless deleted by a "OP" command.

AL - Set Absolute Left Margin for Page Numbers

Format: `\ALnnn`

Where: "nnn" is a number specifying the column position of the left margin for page numbers.

Note: The value of "nnn" must be greater than 0, less than 133, and less than the current column position of the absolute right margin for page numbers.

An error occurs if "nnn" is missing or is invalid.

The default value for the absolute left margin for page numbers is column 10.

Page numbers, if specified, are centered between the absolute left and right margins.

AR - Set Absolute Right Margin for Page Numbers

Format: `\ARnnn`

Where: "nnn" is a number specifying the column position of the right margin for page numbers.

Note: The value of "nnn" must be greater than 0, less than 133, and greater than the current column position of the absolute left margin for page numbers.

An error occurs if "nnn" is missing or is invalid.

The default value for the absolute right margin for page numbers is column 70.

Page numbers, if specified, are centered between the absolute left and right margins.

Examples:

If absolute margins are not specified and page numbers are to appear, the page numbers center between columns 10 and 70.

If the command "\AL60" is given and an absolute right margin command is not specified, page numbers appear right justified on the top or bottom of the page, dependent upon the page position command. Note that "Page xxxxx" consists of 10 column positions - 4 for "Page," 1 for a space, and 5 for the page numbers.

PB - Position Page Numbers at Bottom of Page

Format: \PBxxxxx

Where: "xxxxx" specifies the page number.

Note: Page number "xxxxx" must be exactly five characters, followed by either a "\" sign or a carriage return. The fifth character must not be a "\" sign or carriage return.

When the "PB" command is encountered, page numbers of the form:

Page xxxxx

appear at the bottom of each page, centered between the absolute left and right margins, and starting with the current page.

Page numbers increment by one for successive pages.

Spaces in "xxxxx" are treated as zeros.

Page numbering continues until the end of the file unless turned off by a "PE" command.

If a new page numbering command is encountered, the numbering of subsequent pages is based on the value of "xxxxx."

Refer to the "PT" command for page numbering examples.

PT - Position Page Numbers at Top of Page

Format: \PTxxxxx

Where: "xxxxx" specifies the page number.

Note: Page number "xxxxx" must be exactly five characters, followed by either a "\" sign or a carriage return. The fifth character must not be a "\" sign or a carriage return.

When the "PT" command is encountered, page numbers of the form:

Page xxxxx

appear at the top of each page, centered between the absolute left and right margins, starting with the subsequent page.

Page numbers are incremented by one for successive pages.

Spaces in "xxxxx" are treated as zeros.

Page numbering continues until the end of the file unless turned off by a "PE" command.

If a new page numbering command is encountered, the numbering of subsequent pages is based on the new value of "xxxxx."

Examples:

The following examples illustrate the sequence of page numbers that generate for the specified initial values:

Example 1	Example 2	Example 3	Example 4
\PB 1	\PB01-01	\PB00001	\PB****1
1	01-01	00001	****1
2	01-02	00002	****2
3	01-03	00003	****3
4	01-04	00004	****4
5	01-05	00005	****5
6	01-06	00006	****6
7	01-07	00007	****7
8	01-08	00008	****8
9	01-09	00009	****9
10	01-10	00010	
etc.	etc.	etc.	

The page numbers in Example 2 should not be used if page numbers are to exceed 99. Page numbers in Example 4 should

not be used if page numbers are to exceed 9. In both cases, exceeding these limits cause the generation of unrecognizable page numbers.

PE - Page Number End on This Page

Format: \PE

Note: No argument is required.

Page numbering is suppressed, beginning with the current page, until a page numbering command is encountered, or the end of the file is reached.

RN - Use Roman Numerals for Page Numbers

Format: \RNnnn

Where: "nnn" is a number indicating the roman numeral page number to be positioned at the bottom of the page.

Note: The value of "nnn" must be in the range from 1 to 20.

Roman numerals increment by one for successive pages until page 20 (xx) is reached.

Roman numerals are positioned at the bottom of each page, starting with the current page, and are centered between the left and right margins.

A value of zero, no "nnn," or the "PE" command can be used to suppress roman numeral page numbering.

If another page numbering command is encountered, the new page numbering command takes effect.

Example:

The following table illustrates the roman numerals that are available with the use of the "\RN" command:

<u>Command</u>	<u>Number</u>	<u>Command</u>	<u>Number</u>	<u>Command</u>	<u>Number</u>
\rn1	i	\rn8	viii	\rn15	xv
\rn2	ii	\rn9	xix	\rn16	xvi
\rn3	iii	\rn10	x	\rn17	xvii
\rn4	iv	\rn11	xi	\rn18	xviii
\rn5	v	\rn12	xii	\rn19	xix
\rn6	vi	\rn13	xiii	\rn20	xx
\rn7	vii	\rn14	xiv		

OP - Optional "Page," Print Only Numbers

Format: \OP

Note: All of the page numbering commands described so far, cause the word, "Page" to print before the page number. This command deletes the word "Page."

No argument is required.

The page numbers are positioned as specified by the previously described page commands.

RP - Resume Printing of "Page"

Format: \RP

Note: This command causes the word "Page" to resume printing.

No argument is required.

This command is only necessary when an "OP" command is used, and the operator wishes to resume printing of the word "Page."

Page Format Defaults Summary

If no page format commands are entered, a total page length of 66 lines is allotted in the following proportions:

Heading	7 lines
Body	54 lines
Footing	5 lines

Page numbers are not inserted. If a page top or page bottom command is given, the default values are 10 for absolute left margin and 70 for absolute right margins. The page number is centered between columns 10 and 70. Note that the word, "Page," always accompanies the page numbers except when the "OP" command is used.

The following is an example of the use of some DOC commands for formatting an initial page of a chapter.

```
\h16
\bhCHAPTER 1\
\b150
\f14
\bf*See Chapter 4\
\al60
\pb 1
```

- . Six lines are allotted to a heading block.
- . "CHAPTER 1" is to be inserted in the fourth line of the heading, beginning in column 1.
- . The body is to accommodate up to 50 lines of text.
- . Four lines are allotted to a footing block.
- . "*See Chapter 4" is to be inserted in the third line of the footing, beginning in column 1.
- . "Page 1" is to be inserted at the bottom of the page, right justified (assuming a default of "AR70"). Subsequent pages are automatically numbered, with the page count incremented by one for each page.

CHAPTER 4. LINE FORMAT COMMANDS

The body of a page contains text that fills its allotted space partially or completely. DOC commands permit the user to control lines of text within the body in regards to their horizontal spacing, vertical spacing, and line justification.

The following is a list of DOC line format commands.

Command -----	Function -----
Horizontal Spacing	
LMnnn	Set left margin
RMnnn	Set right margin
PL	Purge current output line
SPnnn	Insert spaces
TC	Clear tab stops
TSnnn	Set tab stops
TTnnn	Tab to logical tab stop
LTnnn	Set left margin to logical tab stop
RTnnn	Set right margin to logical tab stop
TBnnn	Tab to absolute column
TFnnn	Tab with filler
LDxx	Define leader characters
Vertical Spacing	
NL	New line
NE	New line, even margins
SLnnn	Skip lines
PP	Form paragraph
PPnnn	Form paragraph, skipping lines
ISnnn	Specify spaces for paragraph indentation
PG	Skip to top of next page
PSnnn	Conditional skip to top of next page
Line Justification	
LJ	Left justify text
RJ	Right justify text
CN	Center subsequent text
EM	Form even margins

NOTE: The alphabetic characters used in DOC commands may be in either upper or lower case.

Horizontal Spacing Control

Lines of text in the body need to be bound by left and right margins. Margin settings for text are not to be confused with margins settings used for controlling the positioning of page numbers (see Chapter 3).

Tabbing (the setting of tab stops) is a useful device for setting up columns of text or creating tables. Once the tab stops are set, text can be directed to the desired positions on the page. In addition, right and left margins can be reset by tab stops at times when margins need to be changed only temporarily from their original settings. Once the tab stops are cleared, the original margin settings are returned.

LM - Set Left Margin

Format: `\LMnnn`

Where: "nnn" is a number that specifies a column position for the left text margin.

Note: The value of "nnn" must be greater than 0, less than 133, and less than the current column position of the right margin.

An error occurs if "nnn" is missing or is invalid.

The default value for the left margin is column 10.

If left and right margins are set to make an exceedingly narrow body, long words may not fit on a single line. See the example with the "RM" command.

RM - Set Right Margin

Format: `\RMnnn`

Where: "nnn" is a number that specifies a column position for the right text margin.

Note: The value of "nnn" must be greater than 0, less than 133, and greater than the current position of the left margin.

An error occurs if "nnn" is missing or is invalid.

The default value for the right margin is column 70.

If left and right margins are set to make an exceedingly narrow body, a single word or a hyphenated word may not fit on a line, even if it is the only word on the line. In such cases, the line begins at the left margin and as many characters as can be accommodated are inserted. Then, whatever characters remain are placed on the following line, beginning at the left margin. The following illustrates this problem.

Input Text:

```
\lm35\rm45\pl
Long words like "undifferentiated" and
hyphenated words like "new-line-even-margins"
have a hard time being placed into
narrow-margined output.
```

Output Text:

```
Long words
like
"undifferen
tiated" and
hyphenated
words like
"new-line-e
ven-margins
" have a
hard time
being
placed into
narrow-marg
ined
output.
```

PL - Purge Current Output Line

Format: \PL

Note: No argument is required.

The margins of the body remain fixed until a margin change command is received. The current line being formatted has margins fixed by the old margin command. To cause a margin change to take place with the current line, the "PL" command must be given along with the new margin command. The effect is to purge the current line and reset margins to values supplied by the new margin commands.

The following two command sequences have the same effect in regards to the output text.

Change margins after this line.\nl\lm5\rm75\pl
But before this line.

Change margins after this line.\lm5\rm75\nl
But before this line.

In the first case, the margin commands are encountered after the "\nl" command. If the margin commands are to be immediately effective (before more text is placed into the output line), the "\pl" command is used to purge the line. If any text were encountered before the "\pl" command, and before the line was filled (or printed), that text is purged as well.

In the second case, the margin commands are encountered before the current line prints. Since the margins are not changed until after the line prints, the result is that the left and right margins are columns 5 and 75, respectively, after the "\nl" command is processed.

SP - Insert Spaces in Line

Format: \SPnnn

Where: "nnn" is a number specifying the spaces to be inserted in a line.

Note: Since multiple spaces (two or more) in a line are always removed and replaced by a single space in the output text, the "SP" command must be used if multiple spaces are desired.

An error occurs if the value of "nnn" is 0, greater than 132, or if "nnn" is missing.

Example:

Input Text:

```
This\sp3line\sp4has\sp5wide\sp6spaces.\nl
This      line      does      not.\nl
```

Output Text:

```
This      line      has      wide      spaces.
This line does not.
```

TC - Clear Tab Stops

Format: \TC

Note: No argument is required.

Commands that refer to a defined tab stop cause errors if used after tab stops are cleared.

Initially, all tab stops are cleared, i.e., no tab stops exist until a tab stop command is given.

All current tab stops are cleared when the "TC" command is encountered.

TS - Set Tab Stop

Format: \TSnnn

Where: "nnn" is a number specifying a column position for a tab stop.

Note: The value of "nnn" must be greater than 0 and less than 133.

An error occurs if "nnn" is missing or is invalid.

Up to 20 tab stops can be set.

All existing tab stops should be cleared with the "TC" command before new tab stops are set.

Tab stops need not be set in any particular ascending order, i.e., "\ts20\ts10" is valid.

Tab stops can be referenced by other tab commands, using logical tab stop numbers. Refer to the "TT," "LT," and "RT" commands.

Example: The following DOC commands set 10 tab stops.

```
\tc\ts10\ts20\ts30\ts40\ts50
  \ts60\ts70\ts15\ts55\ts25
```

The following correspondence exists between logical tab stop numbers and column positions.

Logical Tab Stop Number	Corresponding Column Position
-----	-----
1	10
2	20
3	30
4	40
5	50
6	60
7	70
8	15
9	55
10	25

Note that the logical tab stop numbers are in a logically sequenced order while column positions are not necessarily in any order. That is, tab stop 10 corresponds to column 25, while tab stop 9 corresponds to column 55.

TT - Tab to Logical Tab Stop

Format: `\TTnnn text`

Where: "nnn" is a number specifying a tab stop (defined previously by a "TS" command) by its logical tab stop number. Text is data to be inserted beginning at the tab stop.

Note: The value "nnn" must be greater than 0, less than 21, and must be specified.

An error occurs if the logical tab stop number has not been defined.

If the "TT" command causes the column position to be greater than the right margin, the tabbed text is placed on the next line, beginning at the left margin.

Example:

Input Text:

```
\tc
\ts10\ts14\ts18\ts22\ts26
\ts30\ts34\ts38\ts42\ts46
\ts50\ts54\ts58\ts62\ts66
\ts70
\tt14 fourth. \tt3 First, \tt10 third, \tt6 second,
\tt16 \nl
```

Output Text:

```
First,          second,          third,          fourth.
```

TB - Tab to Absolute Column

Format: `\TBnnn text`

Where: "nnn" is a number that specifies a column position for a tab stop. "text" is data to be inserted at the tab stop.

Note: An error occurs if "nnn" is missing or if its value is 0 or is greater than 132.

The "TB" command causes tabbing to a specific column position, regardless of whether or not previous tab stops have been set.

If the "TB" command causes the column position to be greater than the right margin, the tabbed text is placed on the next line, beginning at

the left margin.

The effect of the "TB" command is that input text following the command is inserted beginning at the specified column position.

LT - Set Left Margin to Logical Tab Stop

Format: `\LTnnn`

Where: "nnn" is a number specifying a tab stop (defined previously by a "TS" command) by its logical tab stop number.

Note: The value "nnn" must be greater than 0, less than 21, and must be specified.

The column specified by the logical tab stop number must be less than the current column position of the right margin.

If the left margin is to be varied, it is recommended that the left margin be set by the "LT" command rather than the "LM" command. Thus, when the left margin is to be changed only the "nnn" value of the "LT" command need be changed, not the "LM" command.

Example:

```
\tc\ts30\ts20\ts50\ts40\lt2\pl
```

This causes the left margin to be set at column 20.

RT - Set Right Margin to Logical Tab Stop

Format: `\RTnnn`

Where: "nnn" is a number specifying a tab stop (defined previously by a "TS" command) by its logical tab stop number.

Note: The value "nnn" must be greater than 0, less than 21, and must be specified.

The column specified by the logical tab stop number must be greater than the current column position of the left margin.

If the right margin is to be varied, it is recommended that the right margin be set by the "RT" command rather than by the "RM" command.

Example:

```
\tc\ts30\ts20\ts50\ts40\rt4\pl
```

This causes the right margin to be set at column 40.

TF - Tab with Filler

Format: \TFnnn text

Where: "nnn" is a number specifying a tab stop (defined previously by a "TS" command) by its logical tab stop number (refer to "TS" command). "text" is data to be inserted at the tab stop.

Note: The value "nnn" must be greater than 0, less than 21, and must be specified.

An error occurs if the logical tab stop number has not been defined by a "TS" command.

The tab stop specified must have a column position greater than the current column position, i.e., tabbing can only be done to the right of the current column position.

The "TF" command can be used anywhere a leader is required to guide the eye of the reader, e.g., in a table of contents, between entries in a table.

The two-character leader is defined by use of the "LD" command.

The effect of the "TF" command is for text following the command to be inserted at the specified column position. The columns between the current column position, and the specified column position are filled with a leader defined by the "LD" command.

If the "TF" command causes the column position to be greater than the right margin, the tabbed text is placed on the next line, beginning at the left margin.

Example: The assumption is made here that no "LD" command has been encountered. The default leader characters are used (space, period).

Input Text:

```

\tc
\tsl0\tsl4\tsl8\tss22\tss26\tss30\tss34\tss38
\tss42\tss46\tss50\tss54\tss58\tss62\tss66\tss70
\tt3 Item 1 \tf8 Item 2 \tf13 Item 3\nl
\tt3 Item 4 \tf8 Item 5 \tf13 Item 6\nl
\tt3 Item 7 \tf8 Item 8 \tf13 Item 9\nl

```

Output Text:

```

Item 1 . . . . . Item 2 . . . . . Item 3
Item 4 . . . . . Item 5 . . . . . Item 6
Item 7 . . . . . Item 8 . . . . . Item 9

```

LD - Define Leader Characters

Format: \LDxx

Where: "xx" are any displayable characters other than the "\" sign.

Note: The leader characters fill the columns between the current column position and the column position specified by a "TF" command.

The default characters for the leader are the space and period (.).

Example: Using three sets of leader characters, the following results:

Input Text:

```

\tc
\tsl0\tsl4\tsl8\tss22\tss26\tss30\tss34\tss38
\tss42\tss46\tss50\tss54\tss58\tss62\tss66\tss70
\ld==
\tt3 Item 1 \tf8 Item 2 \tf13 Item 3\nl
\ld..
\tt3 Item 4 \tf8 Item 5 \tf13 Item 6\nl
\ld -
\tt3 Item 7 \tf8 Item 8 \tf13 Item 9\nl

```

Output Text:

```

Item 1 =====Item 2 =====Item 3
Item 4 .....Item 5 .....Item 6
Item 7 - - - - - Item 8 - - - - - Item 9

```

Vertical Control

The text formatted for output must be bounded not only on the left and right by the margins of the body, but also by the top and the bottom limits of the body. All lines formatted in the body are printed with single spacing between lines. Commands exist for skipping lines, skipping lines between blocks of text, for beginning new paragraphs, for beginning new pages, and for conditionally starting at the top of a new page.

NL - New Line

Format: \NL

Note: No argument is required.

The "NL" command forces justification of the current output line and causes the subsequent text to begin at the left margin of the new line.

If left and right margins are specified, the "NL" command is required only at the end of a block of text, not at the end of each line.

Example:

Input Text:

\lm30\rm50\pl

The new line command is only required at the end of a block of text. Normally, lines from the input file are automatically formatted by

DOC.\nl

This is the beginning of a new line.\nl

Output Text:

The new line command is only required at the end of a block of text. Normally, lines from the input file are automatically formatted by DOC. This is the beginning of a new line.

NE - New Line, Even Margins

Format: `\NE`

Note: No argument is required.

The "NE" command forces text of a line to begin at the left margin and end at the right margin.

The proper number of spaces is inserted in the formatted line to force the line to the required length.

Note that the "NE" command affects one line and is not to be confused with the "EM" commands that affects all lines.

Example:**Input Text:**

```
\lm25\rm55\pl
```

The new line even margins command is used in rare cases where the last line of a block of text must be printed with even margins as well as the rest of the next.\ne

Output Text:

```
The new line even margins
command is used in rare cases
where the last line of a block
of text must be printed with
even margins as well as the
rest of the text.
```

SL - Skip Lines

Format: `\SLnnn`

Where: "nnn" is a number specifying the number of lines to be skipped.

Note: An error occurs if "nnn" is zero or is missing.

If the number of lines to be skipped exceeds the number of lines remaining in the body of the page, the output text is positioned at the top of the following page.

A command of "`\SL1`" is the same as a "`\NL`" command.

The "SL" command causes execution of the

specified skip and beginning of a new line for the subsequent text.

Example:

Input Text:

```
This is the first line.\sl1
This is the second line.\sl2
This is the third line.\sl3
This is the fourth line.\sl4
Get the point?\nl
```

Output Text:

```
This is the first line.
This is the second line.

This is the third line.

This is the fourth line.
```

Get the point?

PP - Form Paragraph

Format: \PP or \PPnnn

Where: "nnn" is optional and specifies the number of lines to be skipped to form a paragraph.

Note: If "nnn" is zero or is missing, the current line is terminated as if a "NL" command is encountered. The next line is then indented to the column position specified by an "IS" command.

If "nnn" is specified with a value greater than zero, the effect is as if a "SL" command is encountered. The next line is indented to the column position specified by the "IS" command.

If a "IS" command is not encountered, the indentation of the new paragraph is 5 columns, the default condition.

IS - Number of Spaces for Paragraph Indentation

Format: \ISnnn

Where: "nnn" is a number specifying the number of columns of indentation for a new paragraph.

Note: An error occurs if "nnn" is zero or is missing.

The default value for the "IS" command is 5 columns.

PG - Skip to Top of Next Page

Format: \PG

Note: No argument is required.

Normally, pages are filled so that the body contains as much text as possible. However, if the user wishes not to fill the entire page, but to force the beginning of new text to the top of the next page, the "PG" command is used.

The "PG" command is useful at such times when the body does not have enough remaining lines to accommodate an entire list or table, or if only one or two lines of a new paragraph can be accommodated.

PS - Conditional Skip to Top of Next Page

Format: \PSnnn

Where: "nnn" is a number specifying the number of lines that must remain in the body for subsequent lines to still fit on the current page.

Note: If fewer than "nnn" lines remain in the body, a new page is formed as if a "PG" command is encountered.

An error occurs if "nnn" is zero or is missing.

The "PS" command is used to ensure that tables, lists, or diagrams (items that have fixed lengths) are not split between two pages.

Line Justification

Lines of text in the body can be justified between left and right margins by one of the following modes:

- . left justification
- . right justification
- . centering
- . even margins

LJ - Left Justify Text

Format: `\LJ`

Note: No argument is required.

Regardless of the input text, the output text has the first character of each line of text in the column of the left margin.

The right margin is ragged, depending on the number of words that can fit on each line.

If no line justification command is encountered, the default justification is left justification.

The left justification mode remains in effect until another line justification mode is encountered.

Example:

Input Text:

```
\lj
\lm25\rm55\pl
Learning to use DOC is like learning
a new language. Practice is the
key to success. Once the basics
are learned, then practice permits
the user to make everwidening use of
the DOC command set.
```

Output Text:

```
Learning to use DOC is like
learning a new language.
Practice is the key to success.
Once the basics are learned,
then practice permits the user
to make everwidening use of the
DOC command set.
```

RJ - Right Justify Text

Format: \RJ

Notes: No argument is required.

Regardless of the input text, the output text has the last character of each line of text in the column of the right margin.

The left margin is ragged, depending on the number of words that can fit on each line.

The right justification mode remains in effect until another line justification mode command is encountered.

If any parts of a line are formatted with tabbed text, only the text to the right of the tabbed text is right justified. Text to the left of the tabbed text is not justified.

Example: Using the same example of input text as for the "LJ" command, but replacing the "LJ" command with a "RJ" command, the following is obtained:

Output Text:

```

      Learning to use DOC is like
      learning a new language.
Practice is the key to success.
      Once the basics are learned,
      then practice permits the user
to make everwidening use of the
      DOC command set.
```

Input Text:

```

\lm25\rm55\pl
\rj
xxxx \tb40 yyyy\nl
xxx \tb40 yyy\nl
xx \tb40 yy\nl
x \tb40 y \nl
xx \tb40 yy \nl
xxx \tb40 yyy\nl
xxxx \tb40 yyyy\nl
```

Output Text:

xxxx	YYYY
xxx	YYY
xx	YY
x	Y
xx	YY
xxx	YYY
xxxx	YYYY

CN - Center Subsequent Text

Format: \CN

Note: No argument is required.

When the "CN" command is encountered, output text is positioned so that it lies evenly between left and right margins.

The "CN" command is useful for centering such items as chapter headings, table titles, and list titles.

The centering mode remains in effect until a new line justification mode command is encountered. End the current line with a "NL" or similar command before giving a new line a justification command.

Example: Using the same example of input text as for the "LJ" command, but replacing the "LJ" command with a "CN" command, the following is obtained:

Output Text:

Learning to use DOC is like
learning a new language.
Practice is the key to success.
Once the basics are learned,
then practice permits the user
to make everwidening use of the
DOC command set.

Em - Form Even Margins

Format: \EM

Note: No argument is required.

If an "EM" command is encountered, even left and right margins are obtained.

Even margins are obtained by distribution of spaces between words in a line. Every other line has spaces inserted from left to right. The remaining lines have spaces inserted from right to left. This space distribution avoids giving the text a uneven appearance, ensuring a uniform appearance.

Example: Using the same example of input text as for the "LJ" command, but replacing the "LJ" command with an "EM" command, the following is obtained:

Output Text:

Learning to use DOC is like
learning a new language.
Practice is the key to success.
Once the basics are learned,
then practice permits the user
to make everwidening use of the
DOC command set.

CHAPTER 5. AS-IS TEXT COMMANDS

Sometimes it is easier to format a specific table or section of a document manually with the EDITOR than it would be to add the necessary DOC commands to arrive at the same format. The "as-is" command allows the suppression of DOC formatting insofar as word spacing, line spacing, and line justification are concerned. The text is still formatted so that the limits of left and right margin and body size are maintained. There are two commands for specifying the extent of "as-is" text.

AS - Begin Subsequent Text As-Is

Format: \AS

Note: No argument is required.

All text that follows the "AS" command is treated as "as-is" text.

The "AS" command remains in effect until a "//" command is encountered.

Any DOC commands following the "AS" command are treated as text and are not executed when the DOC processor is invoked.

The "AS" command is useful when items such as diagrams are to be inserted into the text. DOC commands then have no affect and diagrams are not distorted.

Margin commands given before the "AS" command is encountered are not suppressed. Text following the "AS" command is printed between the left and right margins of the body.

If left and right margins are set so that "as-is" text is longer than the distance between margins, each overly long line of text is truncated. The truncated text is placed on the next line, beginning at the left margin.

All commands in effect before the "AS" command is encountered are reinstated after the "//" command is given.

Refer to the "//" command for an example of "as-is" text.

// - End As-Is Text

Format: \//

Note: No argument is required.

The "//" command must be the first command on an input line. It must precede any other DOC command.

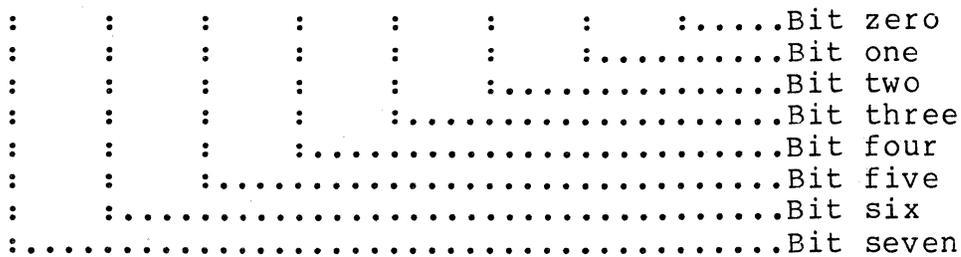
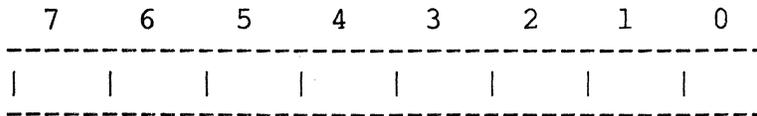
The input text following the "//" command is then formatted according to the DOC commands that were in effect before the "AS" command was given.

The "//" command is ignored if a previous "AS" command was not given.

Example:

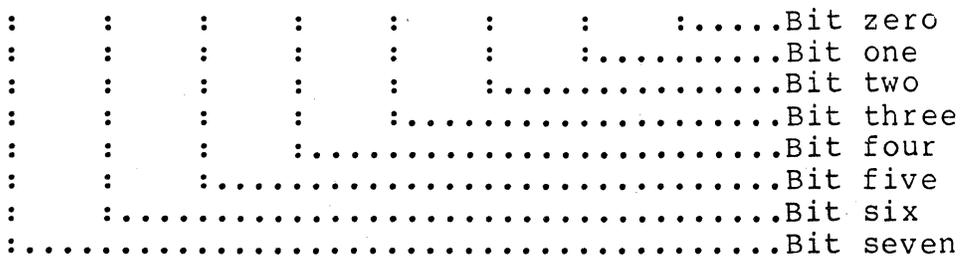
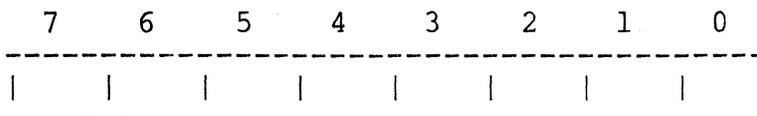
Input Text:

\as



\//

Output Text:



CHAPTER 6. INPUT/OUTPUT CONTROL COMMANDS

Normally, when DOC is first started up, the default input device is the console keyboard, and the default output device is the printer. Commands exist for reassigning either the input or the output device. These commands, as well as some miscellaneous commands related to input/output functions, are described in this chapter.

The DOC input/output commands are:

Command	Function
-----	-----
KY	Switch input stream to console keyboard
TY	Switch output stream to console typewriter
LP	Switch output stream to line printer
QP	Switch output stream to text quality printer
OS	Return control to CODOS, terminating DOC
CM	Set printing mode to continuous (default)
NC	Set printing mode to non-continuous
Fi	Assign diskette file to logical input file i (i = 1-3)
Di	Switch input stream to logical input file i (i = 1-3)
Ri	Rewind logical input file i (i = 1-3)
Ai	Print or display logical input file i (i = 1-3)
FR	Return input stream to logical input file
**	Display comment on console
PR	Restart printing at specified page number

NOTE: The alphabetic characters used in DOC commands may be in either upper or lower case.

KY - Switch Input Stream to Console Keyboard

Format: \KY

Note: No argument is required.

The "KY" command specifies that the console keyboard is to be the input device.

An input prompt, "?" displays when the "KY" command is encountered.

Usually, the "KY" command is given from one of the diskettes installed in the drive unit, permitting the insertion of a name, address, date, etc., at an appropriate place in the input text.

After the response to an input prompt, control usually returns to the line printer by means of the "LP" or "QP" command.

When DOC is invoked, the default input device is the console keyboard, indicated by the display of the input prompt "?."

WARNING: When using CHAIN file processing (see CODOS manual), the "KY" prompt cannot be used for operator communications.

TY - Switch Output Stream to Console Typewriter

Format: \TY

Note: No argument is required.

The "TY" command specifies that the console typewriter or display is to be the output device.

When DOC is invoked, the default output device is the console typewriter or display.

LP - Switch Output Stream to Line Printer

Format: \LP

Note: No argument is required.

The "LP" command specifies that the output device is to be the line printer.

QP - Switch Output Stream to Text Quality Printer

Format: \QP

Note: No argument is required.

The "QP" command specifies that the output device is to be the Text Quality Printer (SP5).

OS - Return Control to CODOS, Terminating DOC

Format: \OS

Note: No argument is required.

When the "OS" command is encountered, the table of contents file closes if open. The last page on the printer ejects, and control returns to CODOS.

The "OS" command is used to terminate the DOC processor.

CM - Set Printing Mode to Continuous

Format: \CM

Note: No argument is required.

The continuous printing mode is the default mode for all printing.

NC - Set Printing Mode to Non-Continuous

Format: \NC

Note: No argument is required.

The non-continuous mode allows the printing of one page at a time. This command is used when printing on letterhead or special size paper.

Fi - Assign Diskette File to Logical Input File i

Format: \F1 filename
\F2 filename
\F3 filename

Where: "filename" is in a standard file specification format as follows:

<filename> [.<suffix>] [:<logical unit number>]

Note: The default values are "SA" for suffix and "0" for the logical unit number.

The "F1" command assigns a specified file (by way of its filename) to logical input file 1. The "F2" command assigns a specified file to logical input file 2, and the "F3" command assigns a specified file to logical input file 3.

The "Fi" commands open specified files and cause previously specified files to be closed.

If filenames are not specified, file command errors occur.

The current input device is not affected by the encounter of a "Fi" command.

An error occurs if a specified file is not in the correct format.

Di - Switch Input Stream to Logical Input File i

Format: \D1
\D2
\D3

Note: No argument is required.

Once a file is defined, either with DOC invoking command or with an "Fi" command, the input stream can be switched with a "Di" command so that it comes from one of the logical input files.

The effect of a "Di" command is to terminate the fetching of text from the current input device. All subsequent text is fetched from a specified logical input file until another "Di" or "end-of-file" condition is encountered.

When an "end-of-file" condition is encountered, the console keyboard becomes the input device and default conditions are restored.

An error occurs if a diskette file is not assigned to the logical input file that the "Di" command specifies.

The "D1" command switches to logical input file 1. The "D2" command switches to logical input file 2; and the "D3" command switches to logical input file 3.

Ri - Rewind Logical Input File i

Format: \R1
\R2
\R3

Note: No argument is required.

"Ri" commands rewind logical input files so that they can be reused without the need to reassign them by file name.

The effect of a "Ri" command is to position the specified logical input file's pointer to the beginning of the file.

An error occurs if the diskette file has not been assigned to the specified logical input file by way of a "Fi" command.

The "R1" command rewinds logical input file 1.
"R2" command rewinds logical input file 2.
"R3" command rewinds logical input file 3.

Ai - Print or Display Logical Input File i As-Is

Format: \A1
\A2
\A3

Note: No argument is required.

"Ai" commands permit processing logical input files as-is, i.e., with EDITOR and DOC commands in place.

An "Ai" command causes input from a specified logical input file to print or display as-is until an "end-of-file" condition is met. The input device is then switched back to the console keyboard with the "KY" command.

An error occurs if a diskette file has not been assigned to a specified logical input file by an "Fi" command.

FR - Return Input Stream to Logical Input File

Format: \FR

Note: No argument is required.

The "FR" command permits return of the input stream to the last logical input file requesting input from the console keyboard by a "KY" command.

A "Di" command could be used, but with the "FR" command, the user does not have to remember which of the three logical input files is to be reassigned to the input stream.

** - Display Comment on Console

Format: **text

Where: "text" is any comment the user wishes to display.

Note: The text is terminated by a carriage return.

The "**" command causes the text to be displayed. Thus, messages in a logical input file can be directed to the user. These messages can be directions as to what input is to be supplied, or they can apply to monitoring information.

PR - Restart Printing at Specified Page Number

Format: \PRxxxxx

where: "xxxxx" specifies the beginning page number.

Note: The "PR" command permits partial printing of a file or reprinting of a portion of the file beginning somewhere else other than on the first page.

The page number "xxxxx" must be exactly five characters in length, followed by a "\" sign or a carriage return. The fifth character cannot be a "\" sign or a carriage return.

The use of the "PR" command is restricted to documents with page numbers.

If page number "xxxxx" is never encountered, printing does not take place.

Multiple File Processing

There may be times when, instead of processing and printing a series of files individually, it is desirable to link the files in order to process and print them as a single document. For example, if a series of input files have been prepared, each representing a chapter of a book, the user may wish to link them in order to obtain a printed book.

To prepare the files for linking, each input file must contain a "PG" command on its initial page. This forces the beginning of the file on a new page. A "D1" command at the end of the file is needed to switch back to logical input file 1. The end of the input file should also contain suppression commands (in front of the "D1" command) for heading and footing text and page numbers. This prevents the possibility of carrying over these numbers into the next linked file. Be sure that a new page (PG) command is inserted before the suppression commands in order to prevent the deletion of the footing on the last page of text.

Linking is accomplished by creating a new file, referred to as a "driver" file. The filenames of the files to be linked, are entered along with file specification and switching commands. For example, the following input files exist, and the user wishes to link them to obtain a printed book.

```
CHAPTER1
CHAPTER2
CHAPTER3
CHAPTER4
```

The "driver" file is created in the EDITOR with a filename of "Book."

```
=EDIT BOOK
```

The files are entered into "Book" with file specifications of "F2" and file switching of "D2." The "driver" file then appears as follows:

```
\f2CHAPTER1\d2
\f2CHAPTER2\d2
\f2CHAPTER3\d2
\f2CHAPTER4\d2
```

The book is then processed and printed by invoking DOC as follows:

```
=DOC BOOK
?\d1
?\lp
```

Each of the parts of the book is then reached via logical input file 2. When the "\D1" command executes in the chapter files, control returns to the "driver" file which invokes the next part to be printed. As long as the file switching commands are used, the format control conditions remain unchanged when control passes from one file to another. Only when an "end-of-file" condition is detected will the initial DOC default values be restored.

If a table of contents file exists with a filename of "CONTENTS," and the user wishes that file to be part of the printed book, the "driver" file then appears as follows:

```
\f2CONTENTS\d2
\f2CHAPTER1\d2
\f2CHAPTER2\d2
\f2CHAPTER3\d2
\f2CHAPTER4\d2
```

Refer to Chapter 7 for details on creating a table of contents file.

File Recursion

File recursion simply means to repeat a process. It is used when you have a form letter that's sent each time to a different address. DOC eliminates the need to manually type the letter each time it's used. All of this is done through the use of the file switching and file rewinding commands (see Di and Ri commands in this same chapter).

For example, a user has a rough draft of a letter. This letter is going to about 100 people. The user inputs the letter with all the necessary DOC commands. However, in the places where the name and address is to appear, the DOC command "\d2" is placed. "D2" tells the computer that the name and address is contained in logical input file 2. File 2 is an address list file already stored on disk. The letter just inputed is logical input file 1. The letter and address files are specified as logical input files 1 and 2 with the "Fi" command.

For example, if logical input file 1 contains a form letter (LETTER), and logical input file 2 contains names and addresses (ADDRESS), then the following input is required to invoke DOC and print multiple copies of the form letter.

Note: CODOS prompts the operator with "="; DOC prompts the operator with "?."

```
=DOC LETTER,ADDRESS
?\d1
```

The format of the "LETTER" file may be represented by the following. The sequence of periods (...) indicates text in the file. "Name" and "Address" are to be supplied by the "ADDRESS" file.

```
text....\d2 (Name) text....\d2 (Address)....\pg\r1\d1\nc
```

The format of the "ADDRESS" file may be represented by the following:

```
Name 1 \d1 Address 1 \d1
Name 2 \d1 Address 2 \d1
Name 3 \d1 Address 3 \d1
      Etc.
```

The "LETTER" file calls the "ADDRESS" file for two inputs per page. The first call is for the name. The second call is for the address. After the body of the letter prints, a "\pg" command ejects the page. The "LETTER" file then rewinds itself (\r1) and returns control to itself at the beginning (\d1). If the letter is printed on a non-continuous form (i.e., letterhead), the "\nc" command is used to allow the printing of one page at a time.

This process repeats until an "end-of-file" condition is detected in the "ADDRESS" file. Since the "ADDRESS" file is not rewound, the operator inputs the new names and addresses for each copy of the letter that prints. Appendix C contains a detailed example of this process.

CHAPTER 7. TABLE OF CONTENTS COMMAND

For large manuals, a table of contents is essential to the overall usefulness of the document. DOC provides the facility for the automatic generation of a table of contents file. Once the table of contents file is built, it can be incorporated into the normal sequence of printing that produces the completed document. Alternatively, the table of contents file can be printed separately and inserted in the front of the finished document.

In order to take advantage of the table of contents feature, the user must have an understanding of the DOC commands described in the previous chapters, especially the input/output control commands in Chapter 6.

The table of contents file is referred to as logical file "0." Unlike the other file commands described in the previous chapter, the table of contents file command creates an output file.

The DOC table of contents file commands are as follows:

Command -----	Function -----
F0 filename	Open table of contents file
C0	Close table of contents file
BE n nn text	Enter in table of contents file
P0as-is text	Insert as-is text to format table of contents file

NOTE: Similarity between the zero (0) and the letter O, could confuse the user and lead to the misspelling of the above commands. The zero is the correct character to use.

F0 - Open Table of Contents File

Format: \F0 filename

Where: "filename" is in standard file specification format as follows:
<filename>[.<suffix>][:<logical unit number>]

Note: To open a table of contents file, the DOC command must be given naming the file to be scanned, followed by the "F0" command.

It is recommended that the filename used with the "F0" command should be something like "TOC" or "CONTENTS" in order that it not be confused with other DOC commands.

The filename must not exceed eight characters. Its first character must be alphabetic, and it must be an upper case letter.

If not given, the value of the suffix defaults to "SA." The default value for the logical unit number is zero.

If the "F0" command is not given, all other table of contents file commands are ignored.

Refer to "Creating a Table of Contents" in this chapter.

NOTE: For the "F0" command, the zero (0) is the correct character to use.

C0 - Close Table of Contents File

Format: \C0

Note: No argument is required.

The "C0" command closes the table of contents file. Refer to "Creating a Table of Contents" in this chapter.

After the "C0" command is given, all other table of contents file commands are ignored.

If a "C0" command is not given to close the table of contents file, the file is closed automatically when control is returned to CODOS. It is recommended that the "C0" command always be used to close a table of contents file.

NOTE: For the "C0" command, the zero (0) is the correct character to use.

BE - Begin Table of Contents Entry

Format: \BEnnn text

Where: "nnn" is a number specifying one of the logical sequence of tab stops set up by the "F0" command.

"text" is normally a chapter or section heading selected as a table of contents entry.

Note: The "BE" command, placed before text in the input text file, results in a table of contents entry when the "F0" command is encountered.

Text is terminated by a carriage return or "\" sign. "\" signs should not be used in the text.

An error occurs if "nnn" is not a logical tab stop, or if it is missing.

Refer to "Creating a Table of Contents" in this chapter.

P0 - Insert As-Is Text to Format Table of Contents

Format: \P0as-is text

Where: "as-is text," entered into the input file, formats information for the table of contents file.

Note: The "P0" command is normally used to insert formatted information in the input file to affect the table of contents file. Items such as heading and footing text and roman numeral page numbers, not automatically supplied when the "F0" command is encountered, can be inserted with the "P0" command.

The "as-is text" is terminated by a carriage return.

Refer to "Creating a Table of Contents" in this chapter.

Example: The entry into the input file of:

```
\P0\BF User's Guide \RN2
```

would, when a "F0" command is encountered, cause "User's Guide" to appear in the footing in the table of contents, and page numbers would be roman numerals, beginning with "ii."

NOTE: For the "P0" command, the zero (0) is the correct character to use.

Creating a Table of Contents

When "BE" and "P0" commands are placed into an input file, a table of contents file (an output file) is created automatically when a "F0" command follows the DOC command. The "BE" command selects the chapter and section headings for entry into the table of contents. The "P0" command supplies the text for the table of contents file.

When the "F0" command is encountered, a table of contents file is opened with logical file zero assigned. Left and right margins of columns 10 and 70, respectively, are set. Sixteen tab stops are set between margins, for each page of the file. The width of each tab is four columns, thus providing fifteen tabs.

The "BE" command has a format of "\BEnnn text," where "nnn" is a number representing one of the logical tab stops. The larger the number, the the greater is the indentation of the text from the left margin. That is, a command of "\BE1 text" begins the text in column 10, a command of "\BE2 text" begins the text in column 14, and so on.

In addition, the "F0" command causes the heading, "TABLE OF CONTENTS," to be inserted at the top of each page of the table of contents file. It also causes the word "Page" to be inserted above the columns where page numbers for the entries are to be placed.

The input text is then scanned for "BE" and "P0" commands. When a "BE" command is found, the text is inserted in the table of contents file as an entry. The page number of the input file where the text is located, is inserted in the page number columns of the table of contents file. A filler, consisting of the leader characters, space and period, is inserted between the text entry and the page number. When a "P0" command is found, the specified format is carried out. When a page is filled, a page command forces continuation of the table of contents to a new page.

When a "C0" command is encountered, the heading, "TABLE OF CONTENTS," is suppressed, all tab stops are cleared, and insertion of page numbers is ended. Continuation of the input file processing is to a new page, and control is returned to logical input file 1.

The following example illustrates the embedding of "BE" and "P0" commands in the input text. Lines with periods represent intervening text between headings in the input file. It is important to note that when a "nl" command is

given with the "be" or "p0" commands, DOC incorporates them in the table of contents and also in the text of the document itself. Thus the command:

```
\be 1.0 INTRODUCTION\nl
\p0\nl
```

instructs DOC to skip 2 lines in the table of contents as well as in the document.

NOTE: For the "F0," "C0," and "P0" commands, the zero (0) is the proper character to use.

Input text:

```
\bel 1.0 INTRODUCTION\nl
\p0\nl
....
\be2 1.1 System Configuration\nl
....
\be2 1.2 DOC Features\nl
....
\be2 1.3 General Concepts\nl
....
\be3 1.3.1 DOC command format\nl
....
\be3 1.3.2 Format Assumptions\nl
....
\be3 1.3.3 Spaces and Punctuation\nl
....
\be2 1.4 Invoking the DOC Processor\nl
....
\be3 1.4.1 Multiple Copies\nl
....
\p0\nl
```

The following illustrates the portion of the table of contents file, discussed above, created when a "F0" command is encountered.

Output Text:

TABLE OF CONTENTS	Page
1.0 INTRODUCTION	1
1.1 System Configuration	1
1.2 DOC Features	2
1.3 General Concepts	2
1.3.1 DOC command format	2
1.3.2 Format Assumptions	4
1.3.3 Spaces and Punctuation	5
1.4 Invoking the DOC Processor	6
1.4.1 Multiple Copies	7

A table of contents file can be created for either a single file or for multiple files. For a single file, it is a matter of giving the DOC command and naming the input file. A "F0" command is given, naming the table of contents file. DOC then creates the table of contents file, with the processing visible on the display screen. When the processing stops, the "C0" command is given to close the table of contents file. For example, given the file, "MANUAL," a table of contents file, "CONTENTS," is created as follows:

```
=DOC MANUAL
?\F0CONTENTS\D1
(processing now takes place)
?\C0
```

For multiple files, a table of contents file can be created, treating the files individually or collectively (as a linked file). Using the example of treating files individually, the files must be specified with "Fi" commands and switched with "Di" commands. For example, the following files exist for which a table of contents is to be obtained:

```
CHAPTER1
CHAPTER2
CHAPTER3
CHAPTER4
```

The following sequence of commands can be used:

```
=DOC                                =DOC CHAPTER1
?\F0CONTENTS                        ?\F0CONTENTS\D1
.....                               .....
?\F1CHAPTER1\D1                    ?\F1CHAPTER2\D1
.....                               .....
?\F1CHAPTER2\D1                    ?\F1CHAPTER3\D1
.....                               .....
?\F1CHAPTER3\D1                    ?\F1CHAPTER4\D1
.....                               .....
?\F1CHAPTER4\D1                    ?\C0
.....                               ?\F1CONTENTS\QP
?\C0                                ?\D1
?\F1CONTENTS\QP                    (file prints)
?\D1                                ?\OS
(file prints)
?\OS
```

The first file is designated logical input file 1. The "F0" command is given to scan the first file and processing takes place (represented by the dotted line). When the first file finishes processing, the user specifies the second file as logical input file 1 and switches input to the second file. The second file is then processed. Files are specified and inputs switched until all files are processed.

The "C0" command closes the table of contents file. The operator can then instruct DOC to print the table of contents file. The "OS" command returns control to CODOS.

Multiple files can be treated as a single file when input files are linked as described under, "Multiple File Processing," Chapter 6. For example, given the linked file "BOOK," a table of contents file can be created for "BOOK."

```
=DOC
\F0CONTENTS\D1
(processing now takes place)
?\F1CONTENTS\QP
(the table of contents prints)
?\C0
?\OS
```

The table of contents should only be created when the manuscript is in its final form. If there are any changes to the manuscript that would affect the table of contents (i.e., resequencing of page numbers), a new table of contents file must be created using the same process as with the first. If there are just changes to the table of contents file itself, the user may invoke the EDITOR to make the changes. For example:

```
=EDIT CONTENTS
```

The changes can now be made to the table of contents file.

CHAPTER 8. MESSAGES AND COMMAND ERRORS

Messages

The following messages can be generated with the use of DOC commands.

DOCUMENT PROCESSOR V.R

This message displays as a sign-on message by the DOC command program. The letters "V" and "R" reflect the current version and revision numbers of DOC.

?

This is the input prompt that displays each time the input stream switches to the console keyboard. Initially, the keyboard is the input stream. Thus, the input prompt immediately follows the sign-on message. It is through console input that the operator gives commands to the DOC processor for assigning files, switching input and output streams, setting up initial conditions other than the default conditions, or for supplying additional text for the formatted document.

INVALID NUMBER OF COPIES

This message indicates that the parameter following the semicolon on the command line, invoking DOC is non-numeric or has a value of zero.

CANNOT USE MULTIPLE COPIES AND MULTIPLE FILE SPECIFICATIONS

This message indicates that the multiple copies option was requested on the command line invoking DOC, and that more than one file name was entered. Only a single file can be used with the multiple copies option. That file must be "<file1>."

** COMMAND ERROR --xx--

This message indicates that a DOC command (specified by the string "xx" in the message) is invalid and cannot be processed. Command errors can occur for a variety of reasons. The two most common reasons are a mistyped command and an invalid or missing command argument.

PRINTED BY DOCUMENT PROCESSOR V.R

This message prints at the top of the last page of a document. The presence of this message indicates that DOC read all input files until their "end-of-file" was encountered.

Command Errors

The two main reasons for command errors are mistyped DOC commands and missing or invalid command arguments. The following list contains those DOC commands that can cause an error if used incorrectly. The special notation used in the descriptions is:

Symbol -----	Meaning -----
nnn	A number; 1-3 digits.
xx	A two-character string.
xxxxx	A five-character string.
f.s:u	A standard CODOS file name specification.

Command -----	Cause of Error -----
A1	Logical file 1 is not open.
A2	Logical file 2 is not open.
A3	Logical file 3 is not open.
ALnnn	The number "nnn" is missing, not in the range 1-132, or greater than the current value of the absolute right margin.
ARnnn	The number "nnn" is missing, not in the range 1-132, or less than the current value of the absolute left margin.
BEnnn	The number "nnn" is missing, not in the range 1-16 (assuming standard table of contents tab stops), or refers to an undefined tab stop.
BLnnn	The number "nnn" is missing or has a value of zero.
D1	Logical file 1 is not open.
D2	Logical file 2 is not open.
D3	Logical file 3 is not open.

F0f.s:u	The table of contents file already exists and is either write protected or of the wrong format.
Flf.s:u	The file specified does not exist or is of the wrong format.
F2f.s:u	The file specified does not exist or is of the wrong format.
F3f.s:u	The file specified does not exist or is of the wrong format.
FLnnn	The number "nnn" is missing.
FR	The keyboard input stream was never given control from one of the three logical input files.
HLnnn	The number "nnn" is missing.
ISnnn	The number "nnn" is missing or has a value of zero.
LDxx	The leader characters "xx" were not exactly two characters. That is, a reverse slash sign or carriage return was encountered as either the first or the second character.
LMnnn	The number "nnn" is missing, not in the range 1-132, or greater than the current value of the right margin.
LTnnn	The number "nnn" is missing, not in the range 1-20, refers to an undefined tab stop, or the column associated with the specified tab stop is greater than the current value of the right margin.
PBxxxxx	The page number string "xxxxx" is missing or is shorter than five characters.
PBxxxxx	The page number string "xxxxx" for restarting is missing or is shorter than five characters.
PSnnn	The number "nnn" is missing or has a value of zero.
PTxxxxx	The page number string "xxxxx" is missing or is shorter than five characters.
Rl	Logical file l is not open.

R2	Logical file 2 is not open.
R3	Logical file 3 is not open.
RMnnn	The number "nnn" is missing, not in the range 1-132, or is less than the current value of the left margin.
RNnnn	The number "nnn" is not in the range 0-20.
RTnnn	The number "nnn" is missing, not in the range 1-20, refers to an undefined tab stop, or the column associated with the specified tab stop is less than the current value of the left margin.
SLnnn	The number "nnn" is missing or has a value of zero.
SPnnn	The number "nnn" is missing or is not in the range 1-132.
TBnnn	The number "nnn" is missing or not in the range 1-132.
TFnnn	The number "nnn" is missing, not in the range 1-20, refers to an undefined tab stop, or the column number associated with the specified tab stop is less than the current column position.
TSnnn	The number "nnn" is missing, not in the range 1-132, or 20 tab stops have already been defined.
TTnnn	The number "nnn" is missing, not in the range 1-20, or refers to an undefined tab stop.

APPENDIX A. DOC COMMAND SET

This appendix lists the DOC command set alphabetically and by category.

Command Alphabetic Listing

Command	Function
-----	-----
Ai	Print disk file i as-is (i = 1 to 3)
AL	Set absolute left margin for page numbers
AR	Set absolute right margin for page numbers
AS	Begin subsequent text as-is
BE	Begin table of contents file entry
BF	Begin footing text
BH	Begin heading text
BL	Specify body length
CO	Close table of contents file
CN	Center subsequent text
CM	Set printing mode to continuous (default)
Di	Switch input stream to diskette file i (i = 1 to 3)
EM	Form even margins
FO	Open table of contents file
Fi	Define diskette file i (i = 1 to 3)
FL	Specify footing length
FR	Return input stream to logical input
HL	Specify heading length
IS	Specify spaces in paragraph indentation
KY	Switch input stream to console keyboard
LD	Define leader characters
LJ	Left justify text
LM	Set left margin
LP	Switch output stream to line printer
LT	Set left margin to logical tab stop
NC	Set printing mode to non-continuous
NE	New line, even margins
NL	New line
OP	Optional "Page," print only numbers
OS	Return control to CODOS
PO	Insert as-is text in table of contents file
PB	Position page numbers at bottom of page
PE	Page numbers end on this page
PG	Skip to top of next page
PL	Purge current output line
PP	Form paragraph
PR	Restart printing at specified page
PS	Conditional skip to top of next page
PT	Position page numbers at top of page

```

QP      Switch output stream to text quality printer
Ri      Rewind diskette file i (i = 1 to 3)
RJ      Right justify text
RN      Use roman numeral page numbers
RM      Set right margin
RP      Resume printing of "Page"
RT      Set right margin to logical tab stop
SL      Skip lines
SP      Insert spaces in line
TB      Tab to absolute column
TC      Clear tab stops
TF      Tab with filler
TS      Set tab stop
TT      Tab to logical tab stop
TY      Switch output stream to console typewriter
//      End as-is text
**      Display comment on console

```

Command Listing by Category

Page Format Commands

Command	Function	Page
-----	-----	-----
HL	Specify heading length	3-02
BH	Begin heading text	3-02
BL	Specify body length	3-03
FL	Specify footing length	3-03
BF	Begin footing text	3-04
AL	Set absolute left margin for page numbers	3-05
AR	Set absolute right margin for page numbers	3-05
PT	Position page numbers at top of page	3-07
PB	Position page numbers at bottom of page	3-06
PE	Page numbers end on this page	3-08
RN	Use roman numeral page numbers	3-08
OP	Print only page numbers, delete "Page"	3-09
RP	Resume printing of word "Page"	3-09

Line Format Commands

Horizontal Spacing

Command	Function	Page
-----	-----	----
LM	Set left margin	4-02
RM	Set right margin	4-02
PL	Purge current output line	4-04
SP	Insert spaces in line	4-05
TC	Clear tab stops	4-05
TS	Set tab stop	4-05
TT	Tab to logical tab stop	4-07
LT	Set left margin to logical tab stop	4-08
RT	Set right margin to logical tab stop	4-08
TB	Tab to absolute column	4-07
TF	Tab with filler	4-09
LD	Define leader characters	4-10

Vertical Spacing

Command	Function	Page
-----	-----	----
NL	New line	4-11
NE	New line, even margins	4-12
SL	Skip line	4-13
PP	Form paragraph	4-14
IS	Specify spaces in paragraph indentation	4-14
PG	Skip to top of next page	4-15
PS	Conditional skip to top of next page	4-15

Line Justification

Command	Function	Page
-----	-----	----
LJ	Left justify text	4-16
RJ	Right justify text	4-17
CN	Center subsequent text	4-18
EM	Form even margins	4-18

As-is Text

Command	Function	Page
-----	-----	----
AS	Begin subsequent as-is text	5-01
//	End as-is text	5-02

Input/Output

Command	Function	Page
-----	-----	-----
KY	Switch input stream to console keyboard	6-01
TY	Switch output stream to console typewriter	6-02
LP	Switch output stream to line printer	6-02
QP	Switch output stream to text quality printer	6-02
OS	Return control to CODOS	6-03
CM	Set printing mode to continuous (default)	6-03
NC	Set printing mode to non-continuous	6-03
Fi	Assign diskette file to logical input file i	6-03
Di	Switch input stream to logical input file i	6-04
Ri	Rewind logical input file i	6-05
Ai	Print or display logical input file i as-is	6-05
FR	Return input stream to logical input file	6-06
**	Display comment on console	6-06
PR	Restart printing at specified page number	6-06

Table of Contents

Command	Function	Page
-----	-----	-----
F0	Open table of contents file	7-01
C0	Close table of contents file	7-02
BE	Begin table of contents entry	7-03
P0	Insert as-is text to format table of contents file	7-03

APPENDIX B. DOC DEFAULT CONDITIONS

The following table is a list of all default conditions that are in effect when DOC is first called up.

Formal Aspect -----	Default -----
Body length	54 lines
Heading length	7 lines
Footing length	5 lines
Absolute left margin	Column 10
Absolute right margin	Column 70
Left margin	Column 10
Right margin	Column 70
Paragraph indentation	5 columns
Page numbering	Off
Line justification	Left
Input stream	Console keyboard
Output stream	Console typewriter or display
Table of contents	Closed
File 1	Command line <file1> or closed
File 2	Command line <file2> or closed
File 3	Command line <file3> or closed
Tab stops	Cleared
Heading text	Spaces
Footing text	Spaces
Page number restart	Off
AS-is text mode	Off
Leader characters	" ." (space-period)
Page printing mode	Continuous

APPENDIX C. DOC EXAMPLES

This appendix is provided to illustrate several examples of formatted text that have been prepared using DOC. The first example illustrates how tables can be formatted with the tabulation commands. Please note that commands can either be in upper or lower case. The following commands are used in the example:

```
\cn, Center the heading
\lj, Left justify text (stop centering)
\nl, Begin a new line
\tc, Clear tab stops
\ts, Set tab stop
\tt, Tabulate to a tab stop
\sp, Place multiple spaces in the line
```

The following table is from the "Agriculture Handbook No. 8."

Input Text:

```
\cn
Standards for Enrichment:\nl
Minimum and Maximum Amounts of Required Nutrients Specified
for Foods Labeled "Enriched"\nl
(Milligrams per pound of product)\nl\nl\lj
\ts20\ts25\ts30\ts35\ts40\ts45\ts50\ts55
Item
\tt1 Thiamine Riboflavin\sp2Niacin\tt7\sp2\Iron
\nl
\tt1 Min\sp2Max\sp2min\sp2Max\sp2Min\sp2Max\sp2Min\sp2Max
\nl
\nl
Bread\tt1 1.1\tt2 1.8\tt3 0.7\tt4 1.6\tt5 10.0\tt6 15.0
\tt7 8.0 \tt8 12.5\nl
Cornmeal\tt1 2.0\tt2 3.0\tt3 1.2\tt4 1.8\tt5 16.0\tt6 24.0
\tt7 13.0 \tt8 26.0\nl
Farina\tt1 2.0\tt2 2.5\tt3 1.2\tt4 1.5\tt5 16.0\tt6 20.0
\tt7 13.0\tt8 [3]\nl
Flour\tt1 2.0\tt2 2.5\tt3 1.2\tt4 1.5\tt5 16.0\tt6 20.0
\tt7 13.0\tt8 16.5\nl
Noodles\tt1 4.0\tt2 5.0\tt3 1.7\tt4 2.2\tt5 27.0\tt6 34.0
\tt7 13.0\tt8 16.5\nl
Rice\tt1 2.0\tt2 4.0\tt3 1.2\tt4 2.4\tt5 16.0\tt6 32.0
\tt7 13.0\tt8 26.0\nl
```

Output Text:

Standards for Enrichment:
 Minimum and Maximum Amounts of Required Nutrients Specified
 for Foods Labeled "Enriched"
 (Milligrams per pound of product)

Item	Thiamine		Riboflavin		Niacin		Iron	
	Min	Max	Min	Max	Min	Max	Min	Max
Bread	1.1	1.8	0.7	1.6	10.0	15.0	8.0	12.5
Cornmeal	2.0	3.0	1.2	1.8	16.0	24.0	13.0	26.0
Farina	2.0	2.5	1.2	1.5	16.0	20.0	13.0	[3]
Flour	2.0	2.5	1.2	1.5	16.0	20.0	13.0	16.5
Noodles	4.0	5.0	1.7	2.2	27.0	34.0	13.0	16.5
Rice	2.0	4.0	1.2	2.4	16.0	32.0	13.0	26.0

The second example is a poem and utilizes the following commands:

```
\lm, To change the left margin
\cn, To center the heading
\lj, To left justify the text
\nl, To begin a new line
\is, To set paragraph indentation
\pp, To indent a paragraph
```

```
Input Text:
\lm20
\cn Jabberwocky
\sl2\lj\is2
'Twas brillig, and the slithy toves
\ppDid gyre and gimple in the wabe:
\nlAll mimsy were the borogoves,
\ppAnd the mome raths outgrabe.
\sl2
"Beware the jabberwock, my son!
\ppThe jaws that bite, the claws that catch!
\nlBeware the jubjub bird, and shun
\ppThe frumious bandersnatch!"
\sl2
He took his vorpal sword in hand:
\ppLong time the manxome foe he shought-
\nlSo rested he by the tumtum tree,
\ppAnd stood awhile in thought.
\sl2
And, as in uffish thought he stood,
\ppThe jabberwock, with eyes of flame,
\nlCame whiffling through the tulgey wood,
\ppAnd burred as it came!
\sl2
One, two! One, Two! And through and through
\ppThe vorpal blade went snicker-snack!
\nlHe left it dead, and with its head
\ppHe went galumphing back.
\sl2
"And hast thou slain the jabberwock?
\ppCome to my arms, my beamish boy!
\nlO frabjous day! Callooh! Callay!"
\ppHe chortled in his joy.
\sl2
'Twas brillig, and the slithy toves
\ppDid gyre and gimple in the wabe:
\nlAll mimsy were the borogoves,
\ppAnd the mome raths outgrabe.
```

Output Text:

Jabberwocky

'Twas brillig, and the slithy toves
 Did gyre and gimble in the wabe:
 All mimsy were the borogoves,
 And the mome raths outgrabe.

"Beware the jabberwock, my son!
 The jaws that bite, the claws that catch!
 Beware the jubjub bird, and shun
 The frumious bandersnatch!"

He took his vorpal sword in hand:
 Long time the manxome foe he shought-
 So rested he by the tumtum tree,
 And stood awhile in thought.

And, as in uffish thought he stood,
 The jabberwock, with eyes of flame,
 Came whiffling through the tulgey wood,
 And burbled as it came!

One, two! One, Two! And through and through
 The vorpal blade went snicker-snack!
 He left it dead, and with its head
 He went galumphing back.

"And hast thou slain the jabberwock?
 Come to my arms, my beamish boy!
 O frabjous day! Callooh! Callay!"
 He chortled in his joy.

'Twas brillig, and the slithy toves
 Did gyre and gimble in the wabe:
 All mimsy were the borogoves,
 And the mome raths outgrabe.

In the third example, a letter prints using supplemental operator input, the following commands are used:

```
\lm, To change the left margin
\ky, To keyin input from the console
\**, To communicate with the operator
\fr, To return control to the logical input file
\em, To produce even margins
\rj, To right justify text
\sl, To skip multiple lines
```

```

Input Text:
\cn
The ABC Co.\nl
10101 Binary Boulevard\nl
Small Stone, Arkansas
\sl2\rj
**Enter the date followed by \fr
\ky
\sl2\em
** Enter Name followed by \fr
\ky
\nl
** Enter Address followed by \fr
\ky
\nl
** Enter City, State followed by \fr
\ky
\sl2 Dear Sir:
\sl2 Many thanks for your recent order.
\sl2 Since we have not had the pleasure of serving you
previously, there will be a slight delay until the
necessary credit data is received through normal credit
channels.
\sl2 You have our assurance, that as soon as satisfactory
information is received, your order will be processed for
immediate shipment.
\sl2 Any additional information you can furnish to help us
would be appreciated.
\sl2 Please direct any written requests to:
\lm25\sl2
XYZ Credit Corporation\nl
14 West Northern Drive East\nl
Salt Flat City, Utah
\sl2\rj
Very Truly Yours,\sl3
John Q. Doe\nl
Credit Manager
\lm10\sl2\lj
JQD/rw\pg

```

The operator is prompted during the printing of this letter for the day's date, the name, address, city, and state. After each operator supplied input, the DOC command "\fr" must be entered in order to return control to the letter file. The following illustrates what the letter would look like if the operator supplied these inputs:

```

4/24/80\fr
Quincy E. D. Smith\fr
5 Fourth Avenue\fr
Anywhere, Arizona\fr

```

Output Text:

The ABC Co.
10101 Binary Boulevard
Small Stone, Arkansas

4/24/80

Quincy E. D. Smith
5 Fourth Avenue
Anywhere, Arizona

Dear Sir:

Many thanks for your recent order.

Since we have not had the pleasure of serving you previously, there will be a slight delay until the necessary credit data is received through normal credit channels.

You have our assurance, that as soon as satisfactory information is received, your order will be processed for immediate shipment.

Any additional information you can furnish to help us would be appreciated.

Please direct any written requests to:

XYZ Credit Corporation
14 West Northern Drive East
Salt Flat City, Utah

Very Truly Yours,

John Q. Doe
Credit Manager

JQD/rw

The previous example can be slightly modified to cause the names and addresses to be obtained from another logical input file, rather than from the operator. In addition, a third file can be used to obtain the date, if desired. For example, if the letter file (logical input file 1) had the following format:

Input Text:

```
\cn
The ABC Cp.\nl
10101 Binary Boulevard\nl
Small Stone, Arkansas
\s12\rj
\d2
\s12\em
\d3
\s12 Dear Sir:
\s12 Many thanks for your recent order.
\s12 Since we have not had the pleasure of serving you
previously, there will be a slight delay until the
necessary credit data is received through normal credit
channels.
\s12 You have our assurance, that as soon as satisfactory
information is received, your order will be processed for
immediate shipment.
\s12 Any additional information you can furnish to help us
would be appreciated.
\s12 Please direct any written requests to:
\lm25\s12
XYZ Credit Corporation\nl
14 West Northern Drive East\nl
Salt Flat City, Utah
\s12\rj
Very Truly Yours,\s13
John Q. Doe\nl
Credit Manager
\lm10\s12\lj
JQD/rw\pg
\r1\d1\nc
```

The command "\d2" causes the date to be obtained from logical input file 2. The date file contains the following information:

Date in any format\r2\d1

The "\r2" command causes the date file to be rewound so that a subsequent call for input returns the date again. The "\d1" command returns control to the letter file. The commands in this file must be all on the same physical input record. Otherwise, rewinding the file would prevent the "\d1" command from ever being detected.

The name and address file, logical input file three, is obtained from the letter file by the "\d3" command. In this example, the name, address, city, and state are assumed to have DOC commands to cause the appropriate line spacing. In the manually entered example, the line spacing commands are contained in the letter file so that the operator did not have to supply them. The name and address file entries must have the following format:

NAME \n1 ADDRESS \n1 CITY, STATE \n1\d1

As many letters print as there are entries in the name and address file. When the "end-of-file" condition is detected in logical input file 3, the DOC processor terminates the processing of all files, and returns control to the console keyboard.

APPENDIX D. GLOSSARY

Absolute	When used to describe a margin or column position, identifies a margin or column position set by a command that overrides positions previously set.
Argument	A number that defines a command; it can be a decimal value or a logical number. For example, in the command "\sp3," 3 is the argument.
Body	The block of lines available on a page for text.
Column	The space that a displayable character occupies on a page. For example, under default conditions, left and right margins are set to columns 10 and 70, respectively, thus making 60 columns available in a line on a page.
Console	Contains the display screen for displaying input or output text.
Current	When used to describe a column position, line, or page, it identifies a column position, line, or page being worked upon at the moment.
Default	A condition assumed by DOC when the user fails to make a selection. See Appendix B.
DOC	Codex Document Processor
DOC	The system that executes DOC commands placed in input text to produce formatted output text.
File	A body of information, identified by a filename, suffix, and logical unit number, and stored on a diskette.
Filler	The sequence of two leader characters that fill spaces between items on a line, serving as a guide to lead the eye from one item to another.

Footing	A block of lines at the foot of a page that may be left blank or may have information inserted in it.
Heading	A block of lines at the top of a page that may be left blank or may have information inserted in it.
Input Stream	A flow of information, serving as input text, whose source is controlled by the user by way of the console keyboard or logical input file.
Input Text	Text inserted on a page by the user by way of the console keyboard or logical input file.
Justification	Forcing text on a page to the left and/or right margin.
Leader	Any displayable character used to make up a filler.
Line Printer	The printer that produces output text.
Linked File	A single file produced by joining individual files.
Logical Input File	A file assigned, in logical order, when files are specified by filename by way of the DOC processor invoking command or by "Fi" command.
Logical Tab Stop	A tab stop identified by logical number.
Logical Unit Number	The number assigned by the user or by default to identify the drive unit the disk file resides on.
Output Stream	A flow of information, displayed or printed as output text by way of the console, console typewriter, or printer.
Output Text	Text displayed or printed, formatted by DOC commands.
Page	Space available for text. A page consists of a heading, body, and footing.
Recursion	To repeatedly get an input file to obtain input text.

Suffix	This identifier is assigned by the user or by default to refer to a particular filename in order to differentiate between files of the same name.
Table of Contents File	A file created automatically when table of contents commands are encountered. Refer to Chapter 7.

codex

A Subsidiary of  **MOTOROLA INC.**

CODEX CORPORATION

20 Cabot Boulevard
Mansfield, Massachusetts 02048

CODEX PHOENIX

INTELLIGENT TERMINAL SYSTEMS
2002 West 10th Place
Tempe, Arizona 85281
(602) 994-6580