New Features of CDL's Text Output Processor II

This document describes the features added to our TOP (Text Output Processor) to make the new version, TOP II. Both versions are based on DEC's "RUNOFF" and are very useful for anyone writing documentation.

With the new version, you have the following enhancements:

1. Ability to embed control characters in the file. Useful for controlling printers (MX-80, etc.)

2. Ability to print "Page at a time". Useful for printing on letter-head paper with a friction feed type printer.

3. Ability to print selected portions. Useful when the ribbon runs out on one page, and just that page needs reprinting.

4. Ability to "include" other files in a main file. Similar to the ".INSERT" feature of the C.D.L. macro assembler. Note that "include" files may not themselves "include" another file.

5. Form feed vs. CRLF selection between pages.

6. Final page ejection.

7. Instant start-up

Details and Syntax for new commands.

1. Embedded control characters.

There is a need to be able to output a string of hex characters, regardless of their value, to the printer. However TOP, as well as most text editors, get confused with a string of control characters. The solution is to use two of the relatively unused characters to enclose a string of hex values. The characters chosen are the "curly braces". Any characters enclosed are taken as literal HEX values, and sent to the printer. Care must be taken to not split the enclosed string over a page border, etc. The syntax is:

```
text1....text2...{1BH,0CH,00H}...text3...text4...
```

In this example, after the text at "text2", the byte values enclosed in the brackets are sent to the printer as 8 bit values. The text following the closing brace (text3, text4, etc.) is affected by the control character sequence. Both the braces and the hex values do not appear in the printed

text, but of course are visible during edit sessions.

2. Page control

There are two new commands, ".PG" and ".PS". They stand literally for "Page Go" and "Page Stop". You may put a "Page Stop" at the beginning of a file and, from then on, the printer pauses at the bottom of each page. This gives the operator time to insert a new page. The operator then hits the space bar on the console which causes the next page to print. It also appears on the console while printing.

Note that this is also useful to force the printing to stop at some point, wait for the operator to change print wheels, etc., and then resume normal printing with "Page Go".

3. Printing selected portions

There are often times when the paper or ribbon jams and one or two pages of a long document are lost. This command allows the printing of just the one or two pages. The command line syntax is:

```
A>TOP file.top [switches] [/ range]
```

The text enclosed in the brackets means "optional". The "switches" are the normal TOP switches (see section 7 below) documented in the TOP manual. The NEW feature is the "range". Range is defined in the following examples:

(Note that there is a SPACE following the "/")

```
/ 23-24   = Print pages 23 & 24 only
/ -24     = Print from first page thru 24
/ 23-     = Print from page 23 to the end
/ 23      = Print JUST page 23
```

4. Including other files.

This allows using a common block of text in many different text files. The command is:

```
.XX filename[.ext]
```

where the default ".ext" is ".TOP". The file begins with the next line on the page unless the first statement of the "included" file is a ".PA", which forces a new page. At present, a file may not be inserted within a line. This ability is planned for a future release.

5. Form feed vs. Line feed.

There was a time when printers just did what they were told. That was good. Then, due to some defective thinking,

"Smart"(?) printers were invented. These printers decide to append a line feed to a carriage return, or add another line feed at the end of the page (perforation skip). Both of these "Features" cause more trouble than they are worth. They are a hardware solution to a software problem. If a perforation skip is desired, it should be done with software. The same with appending a "LF" to a sent "CR". On many printers, it is impossible to do proper underlining or other effects because you can't return the printhead to the start of the line without the printer popping to the next page. Ugh. When "normal" software (such as TOP) is used with a printer like that, it causes double spacing and a general mess. To allow TOP to be used with such machines, TOP II adds the command ".LO" which stands for "Linefeed Off". After this command, TOP II sends only CR's to get to the next line.

When going to the next page, TOP usually sends the correct number of line feeds to get to the next page top. However, with the printers using the "Perforation skip" feature (?), this causes an extra line to be added. Thus, after a few pages, alignment is lost, and you have a rather messy printout. One solution to this is to use the "Form feed" command to select a new page. However, some printers do not honor the form feed. Since this is system dependent, the method of selecting between form feed and line feed is imbedded in the TOP II code. If location 104H is a ZERO, the usual multiple line feeds are sent. If it is non zero, form feeds are used. Simply ZDDT (or DDT) the TOP II file, change the byte to suit your system, and SAVE it back (refer to your DOS manual for instructions on SAVEing files).

6. Final page ejection.

In systems where the Formfeed method is used, there is also a page parity. This causes the printer to eject one or two pages at the end of the printout. The reason for 1 or 2 pages is to preserve the "fold" location. Usually, you start printing the document with the "in" fold at the printhead. When TOP II is done, you are ready to do another.

7. Instant start-up / batch mode.

The original version of TOP waits for the operator to line up the paper, and then hit RETURN to start printing. If a switch value of "Q" is given on the command line, TOP II will start right up, with no operator intervention. This allows using TOP several times from a SUBMIT file. For example:

```
A>TOP file1 Q
```

This starts the printing process right away.

CDL Z80 Text Output Processor

User's Manual

Revision 1.0

September 30, 1979

Introduction

Files to be processed by the CDL Text Output Processor are prepared by using any text editor. To invoke TOP under CP/M (tm-Digital Research) or CDL's TPM (tm-CDL) type the following command:

A>TOP <filename{.ext}>

where {.ext} is optional and defaults to ".TOP" The formatted output goes to the CP/M (TPM) list device. Control of the output is by the following three commands:

Control-S.....Stops output temporarily

Control-Q.....Starts output after a control-S stop

Control-C.....Aborts output and returns to CP/M (TPM)

Each line read from the file is inspected for a first character of "." (period), which identifies a format control word. Format control words are not printed, but are interpreted to specify changes in the current output format. Control words may be entered in either upper or lower case, and should be separated from their operands, if any, by one or more blanks.

Control words may appear at the beginning of any line in the file, with any changes in format taking place below the point at which they occur. No input data should be included on lines containing control words, since this data could in some cases be lost or interpreted as an operand of the control word.

Control Words:

The control words are listed below.

Control	Meaning
.BL	Blank Line
.BM	Bottom Margin
.BR	Break
.BK	Break Mode
.CE	Center
.CM	Comment
.CO	Concatenate
.CP	Conditional Page
.DS	Double Space
.EN	End
.FO	Format
.HE	Heading
.HM	Heading Margin
.IG	Ignore
.IB	Ignore Break
.IN	Indent
.JU	Justify
.LL	Line Length
.NB	No Break
.NC	No Concatenate
.NF	No Format
.NJ	No Justify
.NS	No Space
.OF	Offset
.PA	Page
.PL	Page Length
.PW	Page Width
.PI	Permanent Indentation
.SS	Single Space
.SC	Space Character
.SP	Space Line
.SH	Subheading
.TB	Tab Settings
.TM	Top Margin
.UN	Undent

Default Values for Control Words

When processing a file, many variables, such as line length and page length, are assumed to have certain values ("default" values) until specified by a control word. This fact can simplify the formatting of files; many control words need not be used. The following is a list of control words having default values which are assumed to be in every file.

```
.PW 60
.LL 60
.PL 66
.TM 5
.BM 3
.HM 1
.PI 15
.TB 5 10 15 20 25 30 35 40 45 50 55 60 65 70 75
.FO  (.CO and .JU)
.SS
.BK
.SC _
```

Errors

The processor prints all errors encountered in the specified file with self-descriptive error messages, and with an exact image of the line at fault. Processing continues with the next input line.

Notes on Certain Uses

1. Top-Of-Page Format

Setting up the top of a processed page is not complicated once you understand the rules.

"Top Margin" is the number of lines between the top of

the page and the first line of text. It includes the heading line, the subheading line, and the heading margin.

A heading line and a subheading line are always printed, even if they are blank.

"Heading Margin" is the number of lines between the subheading line (even if it is not visible) and the first line of text.

For example, given the two control words

```
.tm 7
.hm 3
```

the following top-of-page format would be printed:

```
1st line:  Top margin blank line
2nd line:  Top margin blank line
3rd line:  Heading blank line with   Page <number>
4th line:  Subheading blank line
5th line:  Heading margin blank line
6th line:     "       "     "     "
7th line:     "       "     "     "
8th line:  First line of text
```

Given the default top margin (5) and the default heading margin (1) and the heading:

```
.he User's Manual
```

the following format would be produced:

```
1st line:  Top margin blank line
2nd line:  Top margin blank line
3rd line:  User's Manual          Page 92
4th line:  Subheading blank line
5th line:  Heading margin blank line
6th line:  First line of text
```

2. Space Character

A space indicates where line breaks (the end of a line) and justification (the insertion of spaces to make an even right margin) can occur. Sometimes it is desirable that a space be printed, but that a line end or extra

space insertion does not occur at this point. This "non-space" space is provided by the use of the "space-character". Normally, this character is the "_" (but may be changed by the ".SC" command.) When encountered by the processor, it is treated as a non-blank character, but is printed as a space. For example:

".IN_0"

This phrase will appear as ".IN 0"; it will not be split across two lines, and no extra spaces will be inserted between the N and the 0.

3. Tabulation

Tabulation is also provided through the use of a special character <HT> (Control-I). When encountered in the input, the tab character causes a "typewriter type" tabulation to the next "tab stop". The default is a tab stop every five spaces, but can be set to any value by the ".TB" control word. If the output pointer is currently at a tab stop, it will be moved to the next one (as with a typewriter).

It should be noted that a tab character or a space as the first character of an input line normally causes a "break", but this can be suppressed by the .NB and .IB controls.

Text Output Processor Example

```
.tm 10
.fo
.ce
Text Output Processor Example
.sp 2
.ds
This example will demonstrate some of the capabilities of
the Text Output Processor.  The next two pages will
appear first as they would in a file with the control
words visible; then the same file will appear as it
would if processed.
This paragraph was double-spaced with the .DS control.
.ss
.sp
No Break was needed here, since the .SS (Single-space)
control acts as a break.  Although this is in Format
mode, tabular information can be included:
.sp
          SPACE          .SP    .sp
          SINGLE SPACE   .SS    .ss
          DOUBLE SPACE   .DS    .ds
.sp
The blanks beginning each line caused a Break each time,
and the lines were not concatenated.
.sp
Use of the Line Length control allows space to be left
within a page for figures or drawings.  It may take some
experimentation to find how many lines will fit alongside
a figure.
.ll 30
.sp
The new line length must take effect at a paragraph,
since it acts as a Break.  The switch back to standard
line length (60) is also a Break, and usually ends a
paragraph.
.ll 60
.nf
By switching out of Format Mode          CAPTION
and doing some justification
by eye, other effects can be obtained.  This also takes
some practice and experimentation.
.sp
.fo
PARAGRAPHS
.br
If no space follows a paragraph heading, and if the
paragraphs are not indented, a Break is necessary in
Format Mode to keep the heading line from being
justified.
     A few leading blanks are the easiest way to force a
break and separate paragraphs, as this one was done.
.sp
```

The Center control is handy for small figures included in the text. A .CE in front of each line of the figure is necessary. Note that leading blanks count as characters when the line is centered.

```
.sp
.ce
--------------------------
.ce
!  FORMAT  !  EXAMPLE  n  !
.ce
!          !  E          !
.ce
--------------------------
.ce
Figure A
```

Text Output Processor Example

This example will demonstrate some of the capabilities of the Text Output Processor. The next two pages will appear first as they would in a file with the control words visible; then the same file will appear as it would if output by the Text Output Processor. This paragraph was double-spaced with the .DS control.

No Break was needed here, since the .SS (Single Space) control acts as a break. Although this is in Format mode, tabular information can be included:

SPACE	.SP	.sp
SINGLE SPACE	.SS	.ss
DOUBLE SPACE	.DS	.ds

The leading blanks caused a Break each time, and the lines were not concatenated.

Use of the Line Length control allows space to be left within a page for figures or drawings. It may take some experimentation to find how many lines will fit alongside a figure.

The new line length must take effect at a paragraph, since it acts as a Break. The switch back to standard line length (50) is also a Break, and usually ends a paragraph.
By switching out of Format Mode and doing some justification by eye, other effects can be obtained. This also takes some practice and experimentation.

CAPTION

PARAGRAPHS
If no space follows a paragraph heading, and if the paragraphs are not indented, a Break is necessary in Format Mode to keep the heading line from being justified.

A few leading blanks are the easiest way to force a break and separate paragraphs, as this one was done.

The Center control is handy for small figures included in the text. A .CE in front of each line of the figure is necessary. Note that leading blanks count as characters when the line is centered.

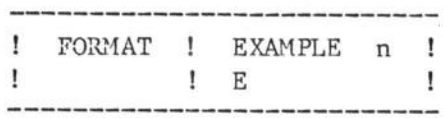

Figure A

BLANK LINE Control

Purpose:

The BLANK LINE control word generates a specified number of blank lines before the next printed line.

Format:

.BL <n>

<n> specifies the number of blank lines to be inserted in the output. If omitted, 1 is assumed.

Usage:

The BLANK LINE control word may be used anywhere in the file to generate blank lines. If the end of the page is reached during a BLANK LINE operation, the operation is terminated and a new page is started. If, after the specified number of blank lines are inserted in the output, there are less than two printable lines remaining on that current page, a new page is started.

Notes:

This control word acts as a BREAK.

The printing of blank lines is independent of the current spacing.

Examples:

.BL 3

Three blank lines are inserted in the output before the next printed line.

.BL

A single blank line is inserted in the output.

BOTTOM MARGIN Control

Purpose:

The BOTTOM MARGIN control word specifies the number of lines to be skipped at the bottom of output pages, overriding the standard value of three.

Format:

.BM <n>

<n> specifies the number of lines to be skipped at the bottom of output pages. If omitted, 1 is assumed.

Usage:

This control overrides the standard bottom margin size of three lines, and need not be included in the file if that value is satisfactory. It may be included anywhere in the file, and the most recent value set applies on any page.

Note:

The BOTTOM MARGIN control word also acts as a BREAK.

Example:

```
.BM 10
```

Ten lines will be left blank at the bottom of the current page, if possible, and on all subsequent pages.

BREAK Control

Purpose:

When CONCATENATE is in effect, BREAK causes the previous line to be typed without filling in words from the next line.

Format:

.BR

Usage:

BREAK is used to prevent concatenation of lines such as paragraph headings or the last line of a paragraph. It causes the preceding line to be typed as a short line; the next line will be printed on a new line.

Notes:

Many of the other control words have the effect of a BREAK. No BREAK is necessary when one of these is present.

A leading blank or tab character on a line has the effect of a BREAK.

Example:

Heading:

```
.BR
First line of the paragraph . . .
```

This part of a file will be printed by SCRIPT as:

Heading:

First line of the paragraph . . .

If the BREAK control word were not included, it would be typed:

Heading: First line of the paragraph . . .

BREAK MODE Control

Purpose:

The BREAK MODE control reestablishes the BREAK function of the tab and space characters when they are the first on a line.

Format:

.BK

Usage:

This command is provided to reestablish the normal BREAK function initiated by the occurrence of a tab or space character at the start of an input line. This would only be necessary if a .NB command had been given previously.

CENTER Control

Purpose:

The line following the CENTER control word will be centered over the specified column.

Format:

.CE <n>

<n> specifies the column over which the following line will be centered. If omitted, the line length divided by two is assumed.

Usage:

The line to be centered is entered on the line following the CENTER control word. It starts at the left margin, and leading blanks will be considered part of its length. The column over which the line is to be centered is independent of any currently in effect controls (e.g. indent, undent, etc.).

Notes:

The CENTER control acts as a BREAK.

If the line to be centered exceeds the current line length value, it is truncated.

Examples:

.CE

Other Methods

"Other Methods" will be centered over the column whose number is equal to the current line length divided by two.

.CE 27

Column Title

When this line of the file is typed, the title "Column Title" will be centered over column 27 of the output.

COMMENT Control

Purpose:

The COMMENT control word causes the remainder of the line to be ignored, allowing comments which are not printed when the file is processed.

Format:

.CM <comments>

Usage:

The .CM control word allows comments to be stored in a file for future reference. These comments can be seen when editing the file or when the file is listed. The comments may also be used to store unique identification that can be useful when attempting to locate a specific region of the file during editing.

Example:

.CM Remember to change the date.

The line above will be seen when examining an unformatted listing of the file and remind the user to update the date used in the text.

CONCATENATE Control

Purpose:

CONCATENATE cancels a previous NO CONCATENATE control word, causing output lines to be formed by concatenating input lines and truncating at the nearest word to the specified line length.

Format:

.CO

Usage:

The CONCATENATE control specifies that output lines are to be formed by shifting words to or from the next input line. The resulting line will be as close to the specified line length as possible without exceeding it or splitting a word. This resembles normal typing output. This is the normal mode of operation for the processor. CONCATENATE is only included to cancel a previous NO CONCATENATE control word.

Note:

This control word acts as a BREAK.

Example:

.CO

Output from this point on in the file will be formed to approach the right margin without exceeding it.

CONDITIONAL PAGE Control

Purpose:

The CONDITIONAL PAGE control word causes a new page to be started if space for less than the specified number of lines remain on the current page.

Format:

.CP <n>

<n> specifies the number of lines that must remain on the current page for additional lines to be printed on it.

Usage:

The .CP control word will cause printing to begin on a new page if "n" lines do not remain on the current page. This request is especially meaningful (1) before an .SP control word to guarantee that sufficient space remains on the current page for the number of spaces requested along with any titles, and (2) preceding a section heading to eliminate the possibility of a heading occurring as the last line of a page.

Note:

If no operand is specified with the .CP request, the request will be ignored.

Example:

```
.CP  10
```

If less than 10 lines remain on the current page, printing will begin on a new page. If 10 or more lines remain, printout will continue on the current page.

DOUBLE SPACE Control

Purpose:

The DOUBLE SPACE control word causes a line to be skipped between each line of printed output.

Format:

.DS

Usage:

DOUBLE SPACE may be included anywhere in the file to force double spaced output.

Notes:

This control word has the effect of a BREAK.

It affects all control words but BLANK LINE (.BL).

Example:

.DS

Blank lines will be inserted between output lines below this point in the file.

END Control

Purpose

The END control word is used to mark the end of the source file to the processor.

Format:

.EN

Usage:

The END control word must be the last line in the source file. When encountered, the output is spaced to the top of the next page, and the processor returns to the monitor.

Note:

This control word acts as a BREAK.

FORMAT Control

Purpose:

The FORMAT control word cancels a previous NO FORMAT control word (or NO CONCATENATE and/or NO JUSTIFY control word), causing concatenation and right justification of output lines to resume.

Format:

.FO

Usage:

The FORMAT control word is a shorthand way to specify the two control words: CONCATENATE and JUSTIFY. This control specifies that lines are to be formed by shifting words to or from the next line (concatenate) and padded with extra blanks to produce an even right margin (justify). Since this is the normal mode of operation for the processor, FORMAT is only included to cancel a previous NO FORMAT control word.

Notes:

This control word acts as a BREAK.

If a line without any blanks exceeds the current line length, it is truncated.

Example:

.FO

Output from this point on in the file will be formed to produce an even right margin on the output page.

HEADING Control

Purpose:

The HEADING control word specifies a heading line to be printed at the top of subsequent output pages.

Format:

.HE <line>

<line> specifies the heading to be printed at the top of subsequent pages.

Usage:

All of the line following the first blank after the HEADING control word is printed at the top of pages starting after the control word is encountered. No heading is printed on the first page of an output file. The heading is printed at the left margin. Its length must be at least 10 less than the output page width, to allow for a page number at the right margin. Leading blanks may be used to center the heading. The heading is printed in the line specified by the heading margin and top margin control words. Additional .HE control words may be included at any point in the file to change the heading on subsequent pages.

Note:

If a new heading is to be placed on a page forced

with the PAGE control word the HEADING control must precede the PAGE control.

Examples:

.HE ON-LINE EDITING SYSTEM

The characters "ON-LINE EDITING SYSTEM" will be printed at the left in the second-last line of the top margin on all pages started after this point in the file:

ON-LINE EDITING SYSTEM PAGE 7

.HE EDL

The heading blanks are considered part of the heading, so the characters "EDL" will be centered in the heading line:

EDL PAGE 8

HEADING MARGIN.Control

Purpose:

The HEADING MARGIN control word specifies the number of lines to be skipped between the two heading lines and the first line of text, overriding the standard value of one.

Format:

.HM <n>

<n> specifies the number of lines to be skipped after the heading lines.

Usage:

The heading lines will be placed a specified number of lines above the first line of text. If no HEADING MARGIN control word is included in the file, the default value is one. The HEADING MARGIN specified must always be less than or equal to the current TOP MARGIN minus the two heading lines.

Note:

This control word acts as a BREAK.

Examples:

.HM 3

Three lines will be left between the heading lines and the first line of text. If the default top

margin of 5 is in effect, the headings will occur at the top of the paper followed by three more blank lines (the heading margin) and then the text.

.HM 1

The standard heading margin of one is set.

IGNORE Control

Purpose:

The IGNORE control word allows a line beginning with a period (.) to be printed.

Format:

.IG

Usage:

The .IG control word specifies that the following line of the input is to be treated as text even if it begins with a period.

Note:

This control has no other effect than the above. It is not a BREAK.

Example:

```
.IG
. . . . and so forth.
```

The second line will be treated just as if it did not begin with a period.

IGNORE BREAK Control

Purpose:

IGNORE BREAK causes the next break initiated by a leading blank or tab character to be ignored.

Format:

.IB

Usage:

A line beginning with a space or a tab character causes a break; the line is printed on a new line. When the first character of a line is a space or a tab character, IGNORE BREAK causes the break to be ignored. The two-line string of characters (the preceeding and the current lines together) is printed as one line.

Note:

Ignore Break control will affect the next space- or tab character-initiated break; it need not be placed immediately before it.

INDENT Control.

Purpose:

The INDENT control word allows the left side of the printout to be indented.

Format:

.IN <n>

<n> specifies the number of spaces to be indented. If omitted, indentation will revert to the absolute margin.

Usage:

The .IN control word causes printout to be indented "n" spaces from the absolute left margin. This indentation remains in effect for all following lines, including new paragraphs and pages, until another .IN control word is encountered. ".IN 0" will cancel the indentation, and printout will continue at the absolute left margin.

Notes:

A) The .IN request acts as a BREAK.

B) The .IN request will reset the effective left margin, causing any .OF setting to be cleared. The .OF request may be used alone or in conjunction with .IN. When the latter is the case, .IN settings will

take precedence.

Examples:

.IN 5

All lines printed after this request will be indented 5 spaces from the absolute left margin. This indentation will continue until another .IN control word is encountered.

.IN 0

The effect of any current indentation will be canceled and printout will continue at the absolute left margin.

JUSTIFY Control

Purpose:

The JUSTIFY control word cancels a previous NO JUSTIFY control word (or part of a NO FORMAT control word), causing right justification of output lines to resume.

Format:

.JU

Usage:

This control word specifies that lines are to be justified (printed evenly on the right margin) by padding with extra blanks. If concatenate mode is in effect, the concatenation process occurs before justification. Since this is the normal mode of operation for the processor, JUSTIFY is only included to cancel a previous NO JUSTIFY control word or the NO JUSTIFY part of a NO FORMAT control word.

Notes:

A) This control acts as a BREAK.

B) If a line exceeds the current line length and CONCATENATE mode is not in effect, the line is printed as is.

C) This control word is seldom used without CONCATENATE mode. FORMAT should be used to enter both JUSTIFY and CONCATENATE modes.

Example:

.JU

Output from this point on in the file will be padded to produce an even right margin on the output page as long as the input lines do not exceed the line length.

LINE LENGTH Control

Purpose:

The LINE LENGTH control word specifies a line length that is to override the standard line length of 60 characters.

Format:

.LL <n>

<n> specifies output line length not greater than 132 characters.

Usage:

The LINE LENGTH control sets the length for output lines until the next LINE LENGTH control word is encountered. If no LINE LENGTH control is included in a file, the standard line length of 60 characters is used. In the JUSTIFY/NO CONCATENATE mode, lines shorter than line length are justified to length by blank padding. In the CONCATENATE mode, lines longer than line length are spilled into the following line. Shorter lines get words from previous or following lines to approach line length.

Note:

This control acts as a BREAK.

Example:

.LL 50

Succeeding lines will be no more than 50 characters in length.

NO BREAK Control

Purpose:

NO BREAK causes all subsequent space- or tab character-initiated breaks to be ignored.

Format:

.NB

Usage:

After the No Break control word, all breaks caused by a space or tab character as the first character on a line are ignored. Such lines will be printed where the previous line stopped.

Note:

NO BREAK will remain in effect until counteracted by .BK, Break Mode.

NO CONCATENATE.Control

Purpose:

The NO CONCATENATE control stops words from shifting to or from the next line.

Format:

.NC

Usage:

The NO CONCATENATE control word stops words from shifting to and from the next line to even out the line length. The printed lines will appear as they do in the source file. It is useful for sections of files containing tabular information or other special formats.

Note:

This control acts as a BREAK.

Example:

.NC

Concatenation will be completed for the preceding line or lines, but following lines will be printed without words being moved to and from lines.

NO FORMAT Control

Purpose:

The NO FORMAT control stops the CONCATENATE and JUSTIFY mode, causing lines to be printed just as they appear in the file.

Format:

.NF

Usage:

The NO FORMAT control is a short-hand way to specify the two control words: NO CONCATENATE and NO JUSTIFY. This stops line justification and concatenation until a FORMAT, JUSTIFY, or CONCATENATE control word is encountered. It is useful for sections of files containing tabular information or other special formats.

Note:

This control acts as a BREAK.

Example:

.NF

Justification and concatenation will be completed for the preceding line or lines, but following lines will be printed exactly as they appear in the file.

NO JUSTIFY Control

Purpose:

The NO JUSTIFY control stops padding lines to cause right justification of output lines.

Format:

.NJ

Usage:

The NO JUSTIFY control word stops the padding of lines with additional blanks to form even right margins. If CONCATENATE mode is in effect, lines will be formed that approach the current line length but will not be forced to the exact length. The resulting lines resemble the output usually produced by a typist.

Note:

This control acts as a BREAK.

Example:

.NJ

Justification will be completed for the preceding line or lines, but following lines will be printed without additional blanks inserted to pad the line.

NO SPACE Control

Purpose:

The NO SPACE control withholds the printing of a space after the last word on a line.

Format:

.NS

Usage:

A space is normally printed after the last character on a line as it appears in the source file. The NO SPACE control causes the next line of text to be printed without this last space. It is used in creating longer lines when this space is not wanted.

Note:

NO SPACE will affect the next line of text; it need not be placed immediately before it. It affects only that line.

OFFSET Control

Purpose:

The OFFSET control word provides a technique for indenting all but the first line of a section.

Format:

.OF <n>

<n> specifies the number of spaces to be indented after the next line is printed.

Usage:

The .OF control word may be used to indent the left side of the printout. Its effect does not take place until after next line is printed, and the indentation will remain in effect until a break or until another .OF word is encountered. The .OF control may be used within a section which is also indented with the .IN control. Note that .IN settings take precedence over .OF, however, and any .IN request will cause a previous offset to be cleared. If is desired to start a new section with the same offset as the previous section; it is necessary to repeat the ".OF n" request.

Notes:

A) This control acts as a BREAK.

B) Two.OFFSET control words without an intervening text line is considered an error condition.

Examples:

.OF 10

The line immediately following the .OF control word will be printed at the current left margin. All lines thereafter (until the next break or .OF request) will be indented 10 spaces from the current margin setting.

.OF

The effect of any previous .OF request will be canceled, and all printout after the next line will continue at the current left margin setting.

PAGE Control

Purpose:

PAGE causes a new page to be started.

Format:

.PA <n>

<n> specifies the page number of the next page. If "n" is not specified, sequential page numbering is assumed.

Usage:

Whenever a PAGE control word is encountered, the rest of the current page is skipped. The paper is advanced to the next page, the heading and page number are typed, and output resumes with the line following the PAGE control word.

Notes:

A) This control acts as a BREAK.

B) If the heading, line length, or other format parameters are to be different on the new page, the appropriate control words must appear before the PAGE word.

Examples:

.PA

The rest of the current page will be skipped. The heading and page number will be printed in the top margin of the next page, and output will resume.

.PA 5

Regardless of the number of the current page, the rest of that page will be skipped, the heading and page number 5 will be printed in the top margin of the next page, and output will resume.

PAGE LENGTH Control

Purpose:

The PAGE LENGTH control word specifies the length of output pages in lines. The value specified overrides the standard page length of 66 lines.

Format:

.PL <n>

<n> specifies the length of output pages in lines.

Usage:

The PAGE LENGTH control word allows varying paper sizes to be used for output. It should not be used to print additional lines on a page. Use the .BM control word. If no PAGE LENGTH control word is included in a file, 66 is the default value. This is the correct size of standard typewriter paper (at six lines per inch). Page length may be changed anywhere in a file, with the change effective on the current page if possible.

Note:

This control word acts as a BREAK.

Example:

.PL 51

Page length is set to 51 lines.

PAGE WIDTH Control

Purpose:

The PAGE WIDTH control specifies the width of the output page in characters. The value specified overrides the standard page width of 60 characters.

Format:

.PW <n>

<n> specifies the width of the output page in characters

Usage:

The PAGE WIDTH control word allows varying paper sizes to be used for output. It should not be used to control the length of the line printed on the page. Use the .LL control word. If no PAGE WIDTH control word is included in a file, 60 is the default value. This is the correct size of standard typewriter paper (at ten characters per inch) leaving a one and one-half inch margin at the left, and a one inch margin at the right. Page width may be changed anywhere in the file, with the change effective on the current page if possible.

Note:

This control word acts as a BREAK.

Example:

.PW 50

The page width is set to 50 characters.

PERMANENT INDENTATION Control

Purpose:

The PERMANENT INDENTATION control specifies the location of the left margin on the output page. The value specified overrides the standard margin of 15 characters (one and one-half inch).

Format:

.PI <n>

<n> specifies the size of the left margin in characters

Usage:

The PERMANENT INDENTATION control allows a fixed amount of space to be left at the left margin of the output page. If no PERMANENT INDENTATION control word is included in the file, a default value of 15 characters (one and one-half inch) is used. This is the normal margin for typewriter paper when the printer start at the extreme left of the paper.

Note:

This control word acts as a BREAK.

Example:

.PI 10

Permanent Indentation is set to 10 characters (one inch).

SINGLE SPACE Control

Purpose:

The SINGLE SPACE control word cancels a previous DOUBLE SPACE control word, and causes output to be single-spaced.

Format:

.SS

Usage:

Output following the SINGLE SPACE control word is single-spaced. Since this is the normal output format, SINGLE SPACE is only included in a file to cancel a previous DOUBLE SPACE control word.

Note:

This control word acts as a BREAK.

Example:

```
.SS
Single-spacing will resume below this point in the
file.
```

SPACE CHARACTER Control

Purpose:

Space Character control enables the user to change the space character from "_" to any other character.

Format:

.SC <character>

<character> specifies the next "space character".

Usage:

The Space Character control is used to change the space character "_" to any other character. This would be done when "_" should appear in the text. The space character is treated like a character in joining two words which the user wants printed on the same line, but it is printed as a space.

Example:

.SC !

This allows "_" to be printed out. "!" will be printed as a space.

SPACE LINE Control

Purpose:

The SPACE LINE control word generates a specified number of blank print lines before the next printed line.

Format:

.SP <n>

<n> specifies the number of blank print lines to be inserted in the output. If omitted, 1 is assumed.

Usage:

The SPACE LINE control word may be used anywhere in the file to generate blank print lines. If the end of the page is reached during a SPACE LINE operation, the operation is terminated and a new page is started. If after the specified number of blank printed lines are inserted in the output there are less than two printable lines remaining on that current page, a new page is started.

Notes:

A) This control acts as a BREAK.

B) The printing of blank lines is controlled by the current spacing, either single or double. In doublespace mode, ".SP" generated blank lines

alternate with ".DS" generated blank lines.

Examples:

.SP 3

Three blank print lines are inserted in the output before the next printed line.

.SP

A single blank print line is inserted in the output.

SUBHEADING Control

Purpose:

The SUBHEADING control word specifies a subheading line to be printed under the heading line on the top of subsequent output pages.

Format:

.SH <line>

<line> specifies the subheading to be printed under the heading line at the top of subsequent pages.

Usage:

All of the line following the first blank after the SUBHEADING control word is printed under the heading line of pages starting after the control word is encountered. No subheading is printed on the first page of an output file. The subheading is printed at the left margin. Leading blanks may be used to center the heading. The subheading is printed on the line specified by the heading margin and top margin control words. If no subheading is specified, a line of blanks is printed in its place. Additional .SH control words may be included at any point in the file to change the subheading on subsequent pages.

Note:

If a new subheading is to be placed on a page forced

with the PAGE control word, the SUBHEADING control must proceed the PAGE control.

Example:

```
.SH LINE EDITING COMMANDS
```

The characters "LINE EDITING COMMANDS" will be printed at the left margin underneath the heading line on the top of all pages started after this point in the file:

```
ON-LINE EDITING SYSTEM                    PAGE 7
LINE EDITING COMMANDS
```

TAB SETTING Control

Purpose:

The TAB SETTING control word specifies the tab stops to be assumed for the following lines when converting the TAB character into the appropriate number of spaces.

Format:

.TB <n(1) n(2) n(3) n(4) n(5)>

<n(i)> specifies the column location of the (i)th tab stop; the sequence must consist of values separated by one or more spaces.

Usage:

TAB characters entered into the file during EDIT file creation are expanded by the processor into one or more blanks to simulate the effect of a logical tab stop. The TAB SETTING control word specifies the locations of the logical tab stops. This overrides the default tab stops of 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75. A TAB SETTING control word without any tab stops specified, results in reversion to the default tab settings. This control word is useful for indenting the beginning of a paragraph (remember a TAB causes a paragraph BREAK) or for tabular information and diagrams.

Note:

This control word acts as a BREAK.

Examples:

.TB 10 20 30 40

Tab stops are interpreted as columns 10, 20, 30, and 40.

.TB

Tab stops will revert to default values of 5, 10, 15, etc.

TOP MARGIN Control

Purpose:

The TOP MARGIN control word specifies the number of lines between the text and the top of the page. This includes the heading line, the subheading line, and the heading margin.

Format:

.TM <n>

<n> specifies the number of lines to be skipped at the top of output pages. n must be two or greater.

Usage:

The specified number of lines will be left at the top of succeeding output pages before the first line of text. The page number and heading, if any, are placed within the top margin by the .HM control. If no TOP MARGIN control word is included in the file, the default value is five. The top margin specified must always be equal to or greater than the current heading margin plus two lines for the heading and subheading.

Notes:

To determine a top margin,

a) select a heading margin

b) add two lines (for the heading and subheading)

c) add the number of lines to be left blank above the heading line.

For example:

a) .hm 4

b) +2

c) +6 blank lines (1 inch)

= .tm 12

This control word acts as a BREAK.

Example:

.TM 3

Three lines will be left at the top of pages started after the current page. The heading and page number will be printed on the first line and the subheading on the second line under the default heading margin.

UNDENT Control

Purpose:

The UNDENT control word forces the immediately following line to start further left than the position indicated by the current indent.

Format:

.UN <n>

<n> specifies the number of spaces to be "undented" (negative indent) for the next line only; it must be less than or equal to the amount of indent currently in effect.

Usage:

The UNDENT control word serves the same purpose as the OFFSET control word but in a different manner. It is usually used to make the first line of a paragraph or section extend further to the left than the body of the paragraph. The choice between using UNDENT and OFFSET is usually a matter of personal preference. In general, UNDENT is more convenient once one becomes familiar with its usage.

Note:

This control acts as a BREAK.

Undent Control

Examples:

.UN 10

If an indentation of 10 is in effect, the next line will start at the left margin; all following lines will occur at normal indent position, 10 spaces from the left margin.

Appendix I
Text Output Processor Operation--Cassette Version

The cassett version of the CDL Text Output Processor requires 4K of memory, a CDL system monitor (either ZAP, ZAPPLE, or SMB), a reader device and a list device for proper operation. The reader MUST be under software control.

After loading the Output Processor by using the R command of the monitor, the previously prepared source file is readied in the reader device. The Processor is started by issuing the G command to the monitor. The Output Processor will sign on on the console, and then wait. At this time, adjust the paper in the output device so that printing will begin on the first line of the page. A CR (carriage return) is then entered to start the output.

Upon completion of the output, the Processor will TRAP to the monitor. If additional source is to be processed, the Processor may be restarted by simply issuing another G command to the monitor.

During operation, the output may be temporarily stopped by entering a Control-S on the console (provided it is not being used as the reader device). A Control-Q is entered to continue output. A Control-C will abort the Processor and TRAP back to the monitor. To resume output, use the monitor's G command.