



**Edinburgh
Regional
Computing
Centre**

User Note 48

(July 1984)

Title:

DPLIST: A Listing Utility for .DP Printers

Author:

John M. Murison

Contact:

Advisory Service

Software Support

Category: **C**

Synopsis

DPLIST is a simple text formatting program which allows users to exploit the features of the Philips GP300 dot matrix printer.

Keywords

Document formatting, DPLIST, text formatting

Introduction

Two high-quality Philips GP300 dot matrix printers have recently become available on the ERCC network. They have device names .DP15 and .DP25.

The GP300 printer prints onto A4 sheets of white paper. It can produce a variety of type faces, with underlining, bold printing, superscripts and subscripts, variable line spacing, etc. One can use the printer as a 'line printer substitute', but to take advantage of its capabilities one must use a text formatting program. DPLIST is a simple formatter which nonetheless enables users to exploit the printer's features. This document is an example of its output.

The emphasis when designing DPLIST was on ease of use. Thus, if you have a file called TEXT without any '@' or '~' characters in it, the command

Command:DPLIST(TEXT)

will print it in a reasonable format on the .DP15 printer.

DPLIST is accessible via directory CONLIB.GENERAL. It is therefore necessary to give the command

Command:OPTION(SEARCHDIR=CONLIB.GENERAL)

before calling it. This need only be done once.

The form of a call of DPLIST is as follows:

Command:DPLIST(input file, option settings)

'input file' and 'option settings' are described in detail below.

Input file

The input to DPLIST is a text file, in which all characters except '@' and '~' stand for themselves. '@' is used in the form @symbol(...text...), and this causes the text enclosed to be treated in some special way - underlined, bold printed, etc. - depending on the symbol after the '@'. The use of '~' is explained in the description of @U, below.

For example, the following input:

```
@h(This is a heading)
@s(This is a subheading)
```

Here is a short @b(paragraph), with a number of words@+(1) @u(underlined) or rendered in some @u(other @b(way)).

would produce the following output:

THIS IS A HEADING

This is a subheading

Here is a short paragraph, with a number of words¹ underlined or rendered in some other way.

Directives

The @x(....) form of command is known as a directive, and the complete set of directives is as follows:

@+(...)	Text is printed as a superscript.
@-(...)	Text is printed as a subscript.
@A(...)	Text is printed using an alternative font, which is selected before the file is processed. Details of the fonts available are given in User Note 50.
@B(...)	Text is printed in a bold face.
@H(...)	Text is treated as a main heading, and is centred on the line and printed in a different typeface, etc. (details are given below).
@I(...)	Text is printed in a Gothic Italic font. <i>Here is an example of an italic font; this one is known as 'gi12'.</i>
@N()	A new page is taken. The opening and closing brackets must be next to each other.
@P()	A new page is taken. The opening and closing brackets must be next to each other.
	Note that @N() or @P(), but <u>not</u> both, can be used in a single input file. @P() is provided for compatibility with the 'LIST' under UCSD' microcomputer utility.
@R(...)	Text is treated as a 'remark', and is not included in the output document.
@S(...)	Text is treated as a subheading. See below for details.
@U(...)	Text, including any spaces, is underlined. Alternatively, the symbol '~' can be used to switch underlining on and off. Thus: ~the cat sat~ becomes <u>the cat sat</u> .
@W(...)	Each word within the text is separately underlined. Spaces and punctuation are <u>not</u> underlined (cf. @U).

Notes on the directives

- The letter must follow '@' without intervening spaces; it can be in upper or lower case.

- The brackets can be any of the following pairs: (), [], {}, <>. The closing bracket must not appear within the enclosed text.
- It is possible to nest directives. Thus @b(@u(word)) produces word. @u(@b(word)) would have exactly the same effect. Note however that @H, @S and @N cannot be nested inside any other directives.
- A directive is automatically terminated at the end of a line, even if a closing bracket is not given. This is not treated as a fault.
- The ~...~ form of underlining is also terminated automatically at the end of a line if a closing '~' is not given.
- Underlined text is treated as a single word by DPLIST. For example, it is never broken across two lines in the output document, even when the text is being reformatted.
- Directives are not looked for within the text of a @R directive, which is terminated by the first appropriate close bracket, or by the end of the line, whichever comes first. Thus:
@R(I used @P() for newpage in this document) would not have the desired effect - the @R would finish after the ')' following '@P('.

The '@' symbol

As explained above, '@' is the only symbol, apart from '~', which does not stand for itself. If it is followed by a letter or a '+' or '-' then a directive is assumed, and a fault is reported if the letter is not one of those listed above.

@ can be used in conjunction with several non-alphabetic characters. If the symbol following '@' is '.' or '?' or '!' then DPLIST will not treat the punctuation as marking a sentence end (more space is normally inserted after a sentence). If the symbol is another '@' then a single '@' will appear in the output. If the '@' is at the end of a line then the line break will be preserved in the output even if DPLIST has been instructed to reformat the lines (this is described below). If the '@' is followed by a '~' then the '~' will appear in the output, rather than controlling underlining. If the '@' is followed by a closing bracket then the bracket will not be taken as marking the end of a directive. If the '@' is followed by any other symbol then the '@' is ignored.

How DPLIST handles text

Text in the input file is classed either as running text or as subtext. Lines which are off the left margin (i.e. start with space characters) and are followed either by blank lines or by more lines off the left margin are treated as subtext. Normally such lines are in tables or diagrams or indented sub-paragraphs. (See the 'Notes on the directives', above, for an example.) DPLIST never reformats subtext.

Lines which comprise paragraphs, i.e. have no margin of spaces at the start, apart possibly from the first line, are treated as running text. DPLIST normally reformats running text, but can be instructed not to.

DPLIST can use a different type face and different vertical line spacing for running text and for subtext. Some users prefer to have a proportional spacing font for running text with possibly one-and-a-half or double line spacing, while having subtext single spaced in a 10 characters/inch or 12 characters/inch font.

Options

As the description has implied, there are a number of user-settable options associated with the operation of DPLIST. These options can only be changed before the file is processed; they cannot be changed during processing. Each option has a pre-selected value, and if this value is satisfactory then no action regarding the option need be taken. Where the pre-selected value is not satisfactory, the user can change it by specifying the desired setting in the command invoking DPLIST, in the format

...,option-name=setting,...

As many options as one wishes can be changed in this way; the order of specification is immaterial and spaces are ignored.

The user can ask DPLIST to prompt him for option settings, instead of (or as well as) giving them in the command line. If the list of option settings is terminated by an asterisk, or consists solely of an asterisk, then DPLIST will prompt the user for further option settings. Each should be given on a separate line. The prompt will then be repeated. When all the desired option settings have been given the user should respond to the next prompt with END or GO, when DPLIST will proceed to generate the output file. The user can also reply with STOP, which causes DPLIST to terminate immediately, i.e. without generating an output file.

If the list of option settings given with the call of DPLIST is terminated by a question mark, or consists solely of a question mark, then DPLIST responds by printing out a complete list of options together with their current values. Thereafter it prompts the user for option settings, as described above. Valid calls of DPLIST:

```
Command:DPLIST(DPIN)
Command:DPLIST(CHAP3, PAGE NUMBER=27, OUTPUT=DPOUT)
Command:DPLIST(THESIS,?)
```

The normal reply to the prompt is of the form

option-name=setting

Instead of this the user can give a question mark. This causes DPLIST, as before, to print out details of the current values of all the options. Another possible reply from the user is of the form

option-name=?

In this case DPLIST responds with the current value of the option specified.

Defining one's own defaults

There are 20 options in total, and it is extremely tedious to change more than a few of them. For this reason a facility is provided whereby the user can define, for his own personal use of DPLIST, the preselected values of the options. He does this by changing the option values as he wishes, then replying to the next prompt with SAVE. This causes the current values of all the options to be stored (by use of the EMAS Profile Scheme, described in User Note 16). He can then proceed as before, i.e. giving further option settings or specifying END or GO or STOP. Thereafter, whenever he invokes DPLIST, the options will be set automatically to the values which were saved. He can later change his mind about the best initial values for the options, by using SAVE again. Only the latest set of saved values is available for use by DPLIST.

If prompting for input is not selected by the user, DPLIST starts processing the input file as soon as the command line has been analysed.

Initial values

If a user called DPLIST for the first time with the following command:

Command: DPLIST(INPUT,?)

DPLIST would check the availability of file INPUT, and then print:

<u>Option name</u>	<u>Current value</u>		<u>Option name</u>	<u>Current value</u>
output	.DP15		pagenumber	0
leftmargin	1.10		pagestarts	0
linesize	6.00		headingstyle	0
minrightmargin	0.20		subheadingstyle	3
topmargin	0.50		headingfont	ops
pagesize	10.00		subheadingfont	cps
minbottommargin	0.50		textfont	cps
textlines/inch	6		subtextfont	c12
subtextlines/inch	6		scriptfont	m15
format	2		altfont	sc12

Other valid replies to dplist:

save	Save current option values, for defaults henceforth.
go or end	Cause dplist to start processing the input.
stop	Cause dplist to stop immediately.

dplist:

This output gives the names of all the options and their initial values. The last line is the prompt for option settings from the user.

Option details

output Either an output file (for LISTing or SENDing later to a .DP printer) or a device name of the form .DPnn, where nn is the number of a .DP printer. If the .DPnn form is used, DPLIST sends the output directly to the printer. Default: .DP15

left margin The size in inches of the left margin. It cannot be set to less than 0.7". Default: 1.10

min right margin The minimum acceptable right margin, in inches. Any line which is not being reformatted by DPLIST and would cause the right margin to be smaller than this will be faulted. Default: 0.20 (effectively switched off)

line size The normal size of each line of text, in inches. This applies where running text is being line-filled (see format, below). The left margin plus line size plus min right margin cannot exceed 8.25". Default: 6.00

top margin The size in inches from the top of the A4 sheet to the first line of text. It cannot be set to less than 0.5". Default: 0.50

page size The vertical size, in inches, of the text on each page of output. This figure does not include the page number line. The top margin plus page size plus min bottom margin cannot exceed 11.66" (or 11.33" inches if there is a page number). Default: 10.00

min bottom margin The minimum acceptable margin, in inches, from the last line of text, or page number if there is one, to the bottom of the A4 sheet. Default: 0.50

text lines/inch The number of lines of running text per inch. Single line spacing corresponds to 6, one-and-a-half spacing to 4, and double spacing to 3. This option cannot be less than 1 or greater than 12. Default: 6

subtext lines/inch The number of subtext lines per inch ('subtext' is defined on page 4). The restrictions applying to text lines/inch apply to this option also. Default: 6

format A code specifying how DPLIST is to deal with running text:

- 0 - leave the line structure alone
- 1 - reformat the lines to fill them as much as the line size option allows
- 2 - reformat the lines and right justify them

DPLIST never reformats subtext. Default: 2 (fill and justify)

page number The page number to be printed at the bottom of the first output page. Subsequent pages are numbered correspondingly. If the value is 0 then the pages are not numbered. If the value is greater than 1000 then page numbering starts on the second output page. For example, a value of 1002 would cause the first page not to be numbered and the second page to be numbered '2'. Default: 0 (pages unnumbered)

page starts If this option is non-zero then DPLIST will print out (on the user's terminal) the first non-blank line of each output page. This facility is provided to enable the user to check that the pagination is satisfactory. Default: 0 (page starts not given)

heading style A code specifying how the text of centred headings (@H[...]) is to be printed:

0	- not bold, not underlined;	1	- bold, not underlined
2	- underlined, not bold;	3	- bold, underlined

Default: 0 (not bold, not underlined)

subheading style A code specifying how the text of subheadings (@S[...]) is to be printed. The meanings of the codes are as for heading style. Default: 3 (bold, underlined)

heading font A mnemonic code specifying the font to be used when printing centred headings (@H[...]). The code consists of one or two letters, capital or small, to indicate the face type, plus a number to indicate the characters per inch, or 'ps' to indicate 'proportional spacing'. The permissible codes are as follows; tables of the fonts are given in User Note 50:

g12	g10	gps	Gothic 10 chars/inch, 12 chars/inch, prop. sp.
gb12	gb10	gbps	Gothic Bold - 12, 10, ps
c12	c10	cps	Courier - 12, 10, ps
m12	m10	m15	Micro - 12, 10, 15
o12	o10	ops	Orator - 12, 10, ps (Letter o, not zero)
gi12	gi10	gips	Gothic Italic - 12, 10, ps
sc12	sc10	scps	Scientific - 12, 10, ps
gr12	gr10	grps	Greek - 12, 10, ps

Default for 'heading font': ops (Orator Proportional Spacing)

subheading font A code specifying the font to be used for subheadings (@S[...]). The valid codes are given above. Default: cps (Courier Proportional Spacing)

text font A code specifying the font to be used for running text. Default: cps

subtext font A code specifying the font to be used for subtext (tables, figures, sub-paragraphs, etc.). Margins and columns will not in general be preserved if this is a proportional spacing font. (The next section, 'General notes and hints', consists wholly of subtext.) Default: c12

script font A code specifying the font to be used for superscripts (@+[...]) and subscripts (@-[...]). Default: m15

alt font A code specifying which font is to be used when the @A(...) directive is used. Note that the Scientific fonts contain a number of mathematical and other symbols (e.g. *, ♦, •). See User Note 50 for details. Default: sc12

General notes and hints

- Do not put leading spaces in @H(...) text - they will be included in the centred 'text', thus making the words appear off-centre.
- Tables or lists should not start on the left margin, as they will then be treated as running text; the effect is predictable but surprising! This situation can arise with numbered lists - the number is sometimes placed on the left margin. In such lists, keep the lines quite short: a typical line length of 6" can only accommodate 60 characters at 10 characters/inch.
- Remember that subtext is never reformatted by DPLIST.
- It is better not to use a proportional spacing font for subtext.
- Avoid blank lines just before a newpage directive - they can cause a new page to be taken just before the @N() or @P() is encountered, resulting in a blank sheet.
- To prevent a very short paragraph from being treated as subtext, include at least one line break in it, with the second line starting on the left margin.
- If the 'page start' output indicates that the pagination is not what was intended, but 'output' was set to .DPnn (e.g. .DP25), then use the Subsystem command DOCUMENTS(.DPnn) to find the document identifier, and delete it using DELETEDOC. It is wasteful to let an incorrect document be printed.
- The '•' symbol is useful for marking items in lists (as in this list. It can be obtained by using @A(;) when option 'alt font' is set to 'sc12' (the default). Other useful graphics are also available; see User Note 50 for details.
- When tables or diagrams are being prepared, it is easier to type them in without directives, and then add the directives later once the text has been correctly positioned.

Currently available fonts

These are described in User Note 50.

Whenever any changes occur in the available fonts (or indeed in any aspect of DPLIST's use), DPLIST will give an explanatory message (once) when invoked.

Errors

The messages which DPLIST produces are intended to be self-explanatory. DPLIST does not generate an output document if it has detected any errors in the input.

Technical background

The format of DPLIST directives is loosely based on the format required by the text formatting program Scribe. Scribe is a large program with many facilities; it is not available on EMAS. The subset of Scribe facilities used in DPLIST approximates to those chosen by Mrs Ann Macintosh, ERCC, in designing the "LIST under UCSD" utility.

DPLIST reads the user's file once and from it generates a file called T£DPLAYIN, which is then automatically processed by DPLAY, a version of the text formatting program Layout. DPLAY generates a file in the format required by a .DP printer.

If the DPLIST option 'page starts' was non-zero, DPLAY also generates a file called T£LPLAYOUT, which is then scanned by DPLIST to find the first line on each output page. The user can examine T£LPLAYOUT, when DPLIST terminates, for details of the line and page structure of his final document. T£LPLAYOUT is suitable for listing to a line printer, although such things as font changes and use of subscripts, etc. will not be represented.

The two T£... files are deleted whenever the user logs off from EMAS.