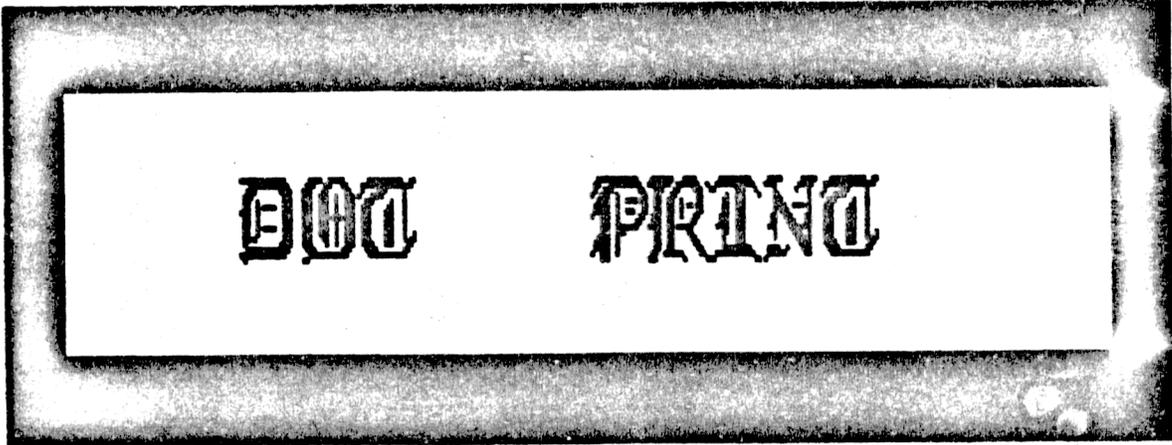


GEAR

GRAPHICS EDITOR AND PROGRAMMER



VERSION 1.5

REVISION DATE 09/10/82

ANOTHER GEAR EXPANSION MODULE FROM:



JF CONSULTING

74355 BUTTWOOD, PALM DESERT, CA. 92260 (714) 340-5471

COPYRIGHT 1982 BY WILLIAM K. MASON

GEAR DOT WRITER AND DOT PRINTER EXPANSION MODULE

DOTPRINT TABLE OF CONTENTS

D 1.0	DOT PRINT PROGRAM	1
D 1.0A	FILE MANAGEMENT	1
D 1.1	DOT COMMANDS	2
D 1.2	CREATING A TEXT FILE	3
D 1.3	SAMPLE TEXT FILE	4
	FILE	4
	PRINTOUT	5
	EXPLANATION	6
D 1.4	SUMMARY OF DOT COMMANDS	8
D 1.5	SUPPLIED (STANDARD) FONT FILENAMES	9
D 1.6	SAMPLE FONT DENSITIES	10
D 1.7	ASCII CHARACTER ORDER	12
D 1.8	SCRIPBIT AS AN EDITOR	14
APPENDIX A	- THE INITIAL PAGE	16
APPENDIX B	- SCRIPBIT AND DOTPRINT	17
APPENDIX C	- SUPPLIED FONT PARAMETERS	18
APPENDIX D	- ERROR MESSAGES	19
APPENDIX E	- UNDERLINING HINTS	21
APPENDIX F	- HINTS FOR LONG DOCUMENTS	24
COMMAND LIBRARY	- COMMANDS EXPLAINED	
DOT WRITE MANUAL		

DOT PRINT INSTRUCTIONS

INSTRUCTIONS FOR BEGINNERS

Although we have designed this package to operate with TRSDOS, we have also tested it with other systems. We have had little or no trouble with any of the systems tested. If you plan to use a DOS other than TRSDOS we suggest that you follow these simple guidelines.

1. Transfer the files from the supplied disk to your formatted working disk. Most of the problems we have encountered have been eliminated by changing all files, totally into the non-TRSDOS system. Write Protect the original disk before you make a back-up.
2. We do not support any DOS other than TRSDOS but, if you run into a problem, contact us and we will do our best to help.
3. If you use a system other than TRSDOS, please let us know how you make out. We have tested our programs under most of the current operating systems but we still appreciate any input from our users.

RUNNING DOTPRINT

To run GEAP or the Expansion Modules [EXPMODES or ?], follow the instructions contained in the appropriate manuals. To run DOTPRINT, simply BOOT the system; when DOS READY is displayed [or the equivalent], go to BASIC. BE SURE THAT YOU SELECT 4 FILES in whatever fashion your DOS requires. In TRSDOS this is done by answering the NUMBER OF FILES question with the value 4.

If you are using NewScript in conjunction with DOTPRINT, you may go directly from the NewScript menu to BASIC by selecting menu entry "0". NewScript has already set the files to 4.

You should now have BASIC READY [or equivalent] displayed on the monitor. Simply type - RUN "DOTPRINT" (ENTER) and DOTPRINT will load and run. DOTPRINT also loads some programs of its own as soon as it starts. When DOTPRINT is ready, it will display some self explanatory questions. Once those prompts have been answered, DOTPRINT will print your file but will follow all of the commands issued in order to create a customized printout. Look at the COMMAND LIBRARY at the end of this portion of the manual to become familiar with DOTPRINT's commands.

Finally, before you begin to set up your disk, files as required for your particular system, read sections W1.0 through 1.3. These sections explain the minimum file layout for operating the program in this package. The program cannot run properly if all of the necessary files are not on line.

D 1.0 INTRODUCTION TO THE DOTPRINT PROGRAM

DOTPRINT is designed to print the contents of a text file created by a word processor such as NewScript or, a file created by the GEAP "FG" command. Using "dot commands", the DOTPRINT program allows you to print the file using special fonts. You can use DOTPRINT with our fonts or, you can create your own fonts by using the second half of this program, EXPMOD8, in conjunction with GEAP 2.1

If you don't have NewScript you can still use DOTPRINT. It will work with any word processor that creates an ASCII file. You must, however, ignore the commands normally used by your word processor and instead, use the commands in the following sections.

The DOTPRINT process is very simple. An ASCII file is first created with GEAP or your word processor. Embedded "dot commands" (these commands are discussed later) are typed in along with your text. After the file is saved, go to BASIC and type: RUN"DOTPRINT" <ENTER>. Respond to the prompted questions and then enter name of the file to be printed. The text will be processed and printed by the DOTPRINT program.

One advantage to using our print time module is that all of the DOTPRINT character fonts may be used. Another advantage is that your own high resolution drawings or character fonts created with the GEAP 2.1 and EXPMOD8 can also be printed.

The DOTPRINT format for commands follow the NewScript format. We must, however, warn you that these commands are not always identical to those of NewScript, so NewScript users should be alert for some of the subtle differences. Please be sure to refer to the summary of commands to see which NewScript commands the Dot Printer module supports and what additional commands are made available.

D 1.0A FILE MANAGEMENT

In several places in this manual you will see references to file management. This is because, in order to get the best use of the program and your available disk space, you need to keep only those files on line that are necessary for the operation you are undertaking. If you have multiple drives this management becomes less critical however, limited drive space requires more diligent management.

Dot Writer contains numerous files. For this section of the manual we are concerned with the following:

1. DOTPRINT - This is the BASIC portion of the scripting program. It must be on line when you attempt to print a file. It can be loaded and then removed if operating space is that critical, but it is best to have it on line at all times.
2. WPML/CIM - Translated it stands for Word Processing Machine Language / Core Image. It must be on line any time DOTPRINT is run.
3. WP - This is a control program that must also be on line throughout

processing.

4. OE, SE, MB, BO, etc. are examples of Font Filenames. It is necessary to have only those fonts that you plan to use on line. If the file you are printing calls for a font that is not on line you will hear three beeps from the printer and an error message will appear on the screen. The printing will then continue with the plain font.
5. Finally, DOTREAD is also related. This is a utility program that reads and prints a bit image file, not an ASCII file. It is not necessary to have on any disk unless you are performing a special function that will be explained in more detail in the DOT WRITE portion of the manual. Also, the remaining programs on the diskette are related to Dot Write and their management is explained in sections W1.0 to W1.3 in the rear of the manual.

In summary, to use DOTPRINT you need only the ASCII file you wish to print, DOTPRINT, WPML/CIM, WP, and the Font files you plan to use. If you are using double density and NewScript you will find that you can fit NewScript's EDIT, FEDIT, NS/CMD, NSINIT and STARTUP/MIN, along with all of the DOTPRINT, required programs and all of the supplied fonts (and more) on one diskette. This makes for a very convenient processing station.

D 1.1 INTRODUCTION TO DOT COMMANDS

If you are a NewScript user, you will already be familiar with "dot commands". The name "Dot Command" has nothing to do with the name Dot Writer, but reflects the format in which the commands are placed in the text. The format of dot commands is as follows: a period or "dot" followed by a two letter instruction, followed by an option. Text placed on the same line as a dot command is ignored, except in special cases.

Example: .CE on

".CE on" is a dot command. For now don't worry about what this means, but just notice the format. A dot, followed by a two letter command, followed by an option.

How does DOTPRINT recognize a dot command? This is very simple. Any line that has a period at the very beginning is considered a command line. (Any text following a dot command on the same line is ignored unless the command requires text.)

The next section will give examples of how these "dot commands" are used in text. If you want, you can type in the example text and print it with DOTPRINT.

D 1.2 CREATING A TEXT FILE FOR DOT PRINT TO PRINT

The rules for creating a text file are very simple. First, the output file must be in ASCII. That is the only type of file DOT PRINT will recognize. Second, use the "dot commands" as they are described in the command summary section. If you are not familiar with NewScript format there are only a few things to remember. Don't worry about how the text looks on the screen, place each special command on its own line and have fun.

Most word processors adapt to creating DOTPRINT files very well but from time to time we get information that may be of help to the users of specific word processors. Scripsit is a good example. Though most Scripsit users will not have any problem, there are a few problems that have cropped up. Look at the appendices in the rear of this manual for special instructions on the use of DOTPRINT. If your word processor is not covered then no problems have been discovered.

The following pages contain a sample text file with the "dot commands" properly used. The lines are numbered for the sake of the discussion that follows the examples and the final printout and should not be included in the actual text.

D 1.3 SAMPLE TEXT AND DOT COMMAND FORMAT

Note that the following lines contain examples of some of the DOTPRINT commands. We intend to show you the FORMAT of commands and some of the output only. We are not covering all of the commands here. Also note that the line numbers are for reference only and are not used in the actual text file.

There are minimal rules to follow. First, multiple dot commands may be placed on a line as long as the first character on the line is a period (.) indicating a dot command line, the commands are separated by a semi colon (;) and, providing that commands that require a filename are the last command on any line.

```

1 .ls 2;. ad 3;. sd 3;. lh 8;. bf mb      (.LS, if used, MUST be 1st command)
2 .af sa
3 pr on;. fo off
4 RCM COMPUTERS
5 221 Hirschfield Dr.
6 Williamsville, N. Y. 14221
7 .sk2;. foon
8 !/J. F. CONSULTING
9 .br
10 74355 Buttonwood
11 .br
12 Palm Desert, Ca. 92260! ?
13 .bf pl
14 .pr on;. sd 3;. lh 5;. up 64;. sk2
15 .af mp-1
16 Dear Joe;
17 .pp
18 It looks like we have a great program here. Bill should
19 be proud of his work. I especially like the features.
20 !&underlining!%; !(double width!) and sub! /scrip! ?ting.
21 I'm sure that everyone with the Epson series printers
22 will want add this software package
23 to their collection.
24 .sk;. in25;. fooff
25 Sincerely,
26 .sk 4
27 Rick McGarvey
28 .en

```

This is the end of the present example. Continue on to see the results of this file and a line by line explanation.

RCM COMPUTERS
221 Hirschfield Dr.
Williamsville, N. Y. 14221

JCF CONSULTING
74355 BUTTERNWOOD
PALM DESERT, CA. 92260

Dear Joe;

It looks like we have a great program here. Bill should be proud of his work. I especially like the features, underlining, double width and sub ting. I'm sure that everyone with the Epson series printers will want add this software package to their collection.

Sincerely,

Rick McGarvey

Now lets take a line by line look at the text in the example. Again I stress that this is not a complete demonstration of the capabilities of DOTPRINT, just a sample to show you how to get started. To find out more about each command, and to learn the other commands, refer to the Command Library at the end of this manual.

1. Line #1 shows several commands on the same line. Note that the commands are separated by semi colons (;) and that the last command is the .BF command followed by the filename of a letterset. The .AD command sets the left margin, in this case to 5 tenths of an inch. The .LS command sets the LOGO SPACE feature to 2 lines. The next command sets the spaces between letters (.SD) to 3 dot spaces. The .LH command sets the width of the spaces between lines, in this case 8. And the final command is the BEGIN FONT command (.BF) which is used to load the type face to be used, in this case Medium Bold (mb).
2. Line #2 sets the alternate font to Small Enhanced (se) and again note that it is placed as the last command on the line. To follow it with another command would confuse the filename. You will see the alternate font used in line #8.
3. Now we turn on the proportional print and turn the format off. By issuing the .FO OFF command it makes it possible to keep individual lines on the left margin as in the address of RCM Computers.
4. Lines #4 through 6 are text lines. They are printed as is, because the format is off. If the format was not turned off, these would have been taken as a continuous line.
5. Line #7 skips two lines and turns the format back on. Note that the address for JF Consulting was done using control breaks (.BR) so it will print on the left margin just as if the format was off. Also, the address for JF Consulting will be printed in the alternate letterset (.af se) as was selected in line #2. The control codes issued before JF in line 8 and after the ZIP in line 12 turn the alternate font on and off.
6. Lines #9 and 11 are the control breaks that force DOTPRINT to look at the lines as if the format was off.
7. Line #14 resets some of the commands and adds a new one. The .UP command sets the pins to be used to underline. See the command library for more on that command and note its use in line #20.
8. Line #15 selects another alternate font, this time it is to be used as a subscript. See the .AF command in the Command Library for more information on how this is done.
9. The next lines contain text, paragraph commands (.PP) and some of the

special control codes. The codes are used in the words that describe their function. Line 24 offers a new command, .IN 25, causes the text that follows to be indented 25 spaces. This will continue until the .IN 0 is issued.

12 Line 28 is the next line of interest. Some word processors need an "end of file" indication or they print garbage at the end of the text. This is true of some issues of Scripsit and Electric Pencil. To avoid the problem the stop command (.st) can be issued at the end of text or the end command (.en) can be issued. These two commands cause DOTPRINT to stop working and prevent the addition of garbage to the file. The primary difference is that the stop command pauses the processor and waits for the <ENTER> before it resumes and the end command ends the file completely. We recommend that Scripsit and Pencil users issue the end (.EN) command at the end of each text file.

That is all there is to it. Obviously there are many more commands and a great deal more power but you get the idea. You are not required to layout the text. Simply type it in using the few rules we have explained and any extra rules found in the appendices that apply to your word processor and you are on your way!

D 1.4 SUMMARY OF DOT PRINT COMMANDS

With the GEAP DOT PRINT support module, the following commands are recognized.

Command	Default	
.AD n	5	Adjust width of left margin
.AF "filename"		Alternate Font
.AP "filename"		Append following "filename"
.BF "Name"	BF 1	Begin Font specified by name
.BF 1		Return to Regular Epson Font
.BM n	6	Bottom Margin
.BR		BReak, control break
.BT "title"		Bottom Title (like .TT)
.CE on, .CE off	off	Center Text/ normal
.CP n		Conditional page
.CW on, .CW off	off	Center/page as line center
.DA on, DA off	off	double strike/Dot Writer fonts
.DA 0,1,2,3	off	Darkness control/Epson fonts
.DS on,off	off	Double Spacing
.EM on,off	off	Emphasized print
.EN		End of Text File.
.ES "symbol"	!	Select Escape Code
.FM n	1	Footing Margin
.FO on, .FO off	on	Format on, Format off
.HL		Draw Horizontal Line
.HM n	1	Heading Margin
.IG on,off 1/2		Ignore text
.IM "filename"		Imbed file command
.IN n	0	Indent n number of spaces
.JU on, .JU off	on	Right justification on/off
.KE		Keyboard Entry
.LH	12	Line Height
.LL n	70	Set Line Length of n
.LS n	off	Set LOGO SPACE
.MF n	1	Magnify Font by factor of n
.MX 100, .MX 80		Select wide paper
.OF n		Offset command (hanging indent)
.PA		Page eject
.PL n		Select page length
.PN on,off		Page Numbering On or Off
.PP		Start new paragraph
.PR on,off		Proportional Print On/Off
.RE 0	on	Reverse Off
.RE 1	off	Reverse /blank spaces
.RE 2	off	Reverse /black spaces
.SD n	0	Dot Spaces Between Letters
.SK n		Skip n number of lines
.ST "message"		Stop, print "message"
☆.PS #	6	Redefine page number symbol

D 1.6 DOT WRITER FONT DENSITIES

When using Dot Print in the "Format On" (default) condition, there is no need to worry about the size of the letterset being used. That will be handled by Dot Printer and justification will be automatic. There are however, two conditions under which you will find it helpful to know the characters per inch in each font.

The first condition is when you select "Format Off". In this mode, justification is not automatic and you are responsible for determining the end of the line. If you are using the Olde English font for example, you will find that, even though you can fit any given line of text on the video display line, that same line may not fit on the printed page. That is because normal Epson font is 10 characters per inch which allows 65 characters on a 6.5 inch line. The Olde English font has a maximum of 6.8 characters per inch allowing only 44 characters per line. This can become even more complicated by inserting spaces between the letters which will, in turn, reduce the number of characters that will fit on a line.

The second condition is when you choose the magnify option and then attempt to print out in the "Format On" mode. In this case, the format is automatically turned off and you are again responsible for determining line length.

The following list contains the characters per inch values for 8 of the fonts that are presently available. Each font is considered three times; with 0 spaces between letters, 2 spaces between letters and finally, 4 spaces between letters. Use this list as a guide to help figure how many characters per inch you will get with any increase in dimensions such as increased spaces between letters or magnified fonts. In the case of increased spaces, a little guess work will be fine. When magnified fonts are used, the space will decline in proportion to the degree of magnification. If you select Olde English with 2 spaces between letters, you will get approximately 6 characters per inch. If you also select a 2x magnification you will get approximately one-half of that amount or 3 characters per inch.

FONT	#spaces between	Char/inch
Big Bold (BB)	SD= 0, 2, 4	7, 6, 5
Bold Enhanced (BO)	SD= 0, 2, 4	7, 6, 5
Mini-Cubes (MC)	SD= 0, 2, 4	7.8, 7, 6.2
Small Enhanced (SE)	SD= 0, 2, 4	12, 10, 9
Medium Bold X2 (MB2)	SD= 0, 2, 4	4, 3.7, 3.4
Microprint (MP)	SD= 0, 2, 4	20, 16, 12
True English (OE)	SD= 0, 2, 4	6.8, 6, 5.5
Medium Bold (MB)	SD= 0, 2, 4	7.6, 7, 6

D 1.7 KEYBOARD CHARACTER ORDER

When using GEAP Dot Writer, it is necessary to be aware of the order in which the keyboard characters are displayed. The lowest character is the SPACE at 32 decimal and the highest is the lowercase "z" at 122 decimal. There are additional values but they are not directly accessible from the keyboard so they are not of importance here.

When you set up a FRAME within which to draw a character or Hi-Res drawing, the computer will display the highest character that can be used to store the finished drawing in memory. The larger the frame is, the lower in value the largest storage location will be. This is a memory restriction - higher values ~~can~~ be stored on disk! Following is a list of the keyboard symbols in order from smallest to largest.

VALUE / CHARACTER	VALUE / CHARACTER	VALUE / CHARACTER
32 SPACE	62 >	92 NA
33 !	63 ?	93 NA
34 "	64 at sign	94 NA
35 #	65 A	95 NA
36 \$	66 B	96 NA
37 %	67 C	97 a
38 &	68 D	98 b
39 '	69 E	99 c
40 (70 F	100 d
41)	71 G	101 e
42 *	72 H	102 f
43 +	73 I	103 g
44 ,	74 J	104 h
45 -	75 K	105 i
46 .	76 L	106 j
47 /	77 M	107 k
48 0	78 N	108 l
49 1	79 O	109 m
50 2	80 P	110 n
51 3	81 Q	111 o
52 4	82 R	112 p
53 5	83 S	113 q
54 6	84 T	114 r
55 7	85 U	115 s
56 8	86 V	116 t
57 9	87 W	117 u
58 :	88 X	118 v
59 ;	89 Y	119 w
60 <	90 Z	120 x
61 =	91	121 y
		122 z

Your DOT WRITER package comes with 12 of our unique type fonts. In addition to those, we are constantly creating new fonts that will be made available at additional cost. Here are a few samples of the new fonts already available!

A B C D E F G 1 2 3 4 5 6 7 8 9

A B C D E F G 1 2 3 4 5 6 7 8 9

A B C D E F G 1 2 3 4 5 6 7 8 9

A B C D E F G 1 2 3 4 5 6 7 8 9

A B C D E F G 1 2 3 4 5 6 7 8 9

A B C D E F G 1 2 3 4 5 6 7 8 9

A B C D E F G H I J K L a b c d e f g h i j k l

A B C D E F G a b c d e f g 1 2 3 4 5 6 7 8 9 0

A B X Δ E Θ Γ α β γ δ ε θ γ

D 1.8 USING SCRIPSIT AS AN EDITOR

We have stressed using NewScript as a text editor throughout this manual and some of you may have come to think that it is the only word processor you can use. Obviously we are aware that many of you have word processors other than NewScript and they can be used as well. In fact, any program that creates an ASCII file can be used.

If you are a Scripsit user, there are some things you will have to know. The first and most important is always save the document with the comma A prefix. That will create the necessary ASCII file. For those of you who are not very familiar with Scripsit, that is done in the following format:

```
S,A filespec/ext. password: drive
```

You must also forget normal Scripsit formatting procedure. Your Scripsit file should look just like the example file shown earlier in the manual. Use the ENTER key to force a line end so that a dot command can be placed on a new line and also to force a line end after no more than 255 characters. You can force line ends as often as you wish but never less than every 255 characters. It is easiest to simply force a line end at the end of each line. Unlike in normal Scripsit, where the line end comes will have no effect on the way the lines are printed. If you are typing a paragraph and you wish to force a line end, you may do so.

So what it all boils down to is this:

1. Always save a file with the ",A" PREFIX
2. Disregard normal Scripsit format rules and follow the rules that we have explained in this manual.
3. Don't forget to force line ends - the lines will be joined together unless the format function is off or a control break (.pp for example) has been issued.
4. This is important. Due to an unusual way in which some Scripsit versions handle files, it is necessary to ALWAYS end a file with the .ST or .EN command. Failure to end the file with one of these commands may result in garbage being printed at the end of the file.

Finally, I will repeat, ANY ASCII FILE can be printed with Dot Printer as long as it has been set up in our format. Regardless of which word processor or text editor you use, it will work with Dot Printer as long as you can follow our rules.

APPENDIX A

THE INITIAL PAGE

The top of the initial page is treated differently than the tops of the other pages. The computer determines how to handle the top of the first page by going through the following steps:

- 1 . . . If the first command in the file was ".LS n" , then the computer assumes that the paper has been moved by hand "n" spaces from the top. The computer does not print a top title, but starts right in with printing text.
- 2 . . . If any line feeds have been given (by an .SK command, for example) before the first line of text, then the computer does not print a top title, but starts printing text.
- 3 . . . If no line feeds have been given before the first line of text, then the computer checks to see if it has received a top title (by the .TT command). If it has, then the top title is printed and printing of text begins at the top margin.
- 4 . . . If no line feeds have been given and no top title has been given before the first line of text, then the computer spaces down to the top margin and begins printing text. It does not print a top title.

APPENDIX B

USING DOTPRINT WITH SCRIPSIT

Although DOTPRINT is designed to be used with NEWSRIPT, you can use it with SCRIPSIT if you observe a few extra precautions:

First, do not type more than 255 characters without hitting <ENTER>. DOTPRINT reads the text in small chunks, each chunk being terminated with <ENTER>. If you go more than 255 characters without hitting <ENTER>, DOTPRINT may insert blanks into the middle of words. Remember that, with .FO ON, forcing a line by hitting <ENTER> has no effect on the appearance of the lines when they are printed. The appearance of the print is controlled by the dot commands.

Make sure you hit <ENTER> after each line of dot commands, so that the dot commands are separated from the text to be printed.

Make sure that the last entry in each file is the dot command:

.EN

Otherwise, DOTPRINT may print a string of "garbage" after the end of the text.

Do not insert the SCRIPSIT boundary markers for new paragraph or new page. DOTPRINT won't understand them.

Finally, save the text to be printed by using the command:

<BREAK> S, A "filename" <ENTER>

This will ensure that DOTPRINT will be able to "understand" the disk storage format.

APPENDIX C

SUPPLIED FONT PARAMETERS

It may be helpful for the user to know the frame size of the letters that he is using. The frame size is not the letter size but rather, the size of the frame in which the letterset was created. Each letter may or may not take up the entire area that was allowed. Obviously, the lowercase letters do not take as much room as the uppercase and a capital "T" will not be as wide as a capital "M". The frame size is expressed in print dots.

Also noted in the chart below, are the upper and lowercase availability of the font. Most fonts have both upper and lower case but some do not. "Y" in the column indicates that the font has the specified case and "N" means it does not.

File Name	Frame Dots Wide	Frame Dots High	#of Lines	Actual LETTER VERT. SIZE	LC
FF	9	16	2	9	N
MB	15	16	2	11	Y
MB2	31	32	3	22	N
BB	17	24	3	16	Y
BO	17	24	3	15	N
SE	9	16	2	9	Y
MP	5	8	1	5	N
MP1	5	8	1	5	N
OE	17	16	2	16	Y
PL	11	16	2	9	Y
MC	15	16	2	15	N
GR	11	16	2	9	Y

NOTE: DUE TO LIMITED DISK SPACE, MP1 AND FF
ARE NOT INCLUDED IN VERSION 1.5C

APPENDIX D
 ERROR MESSAGES

When the computer encounters a suspected error it will beep the printer three times and print an error message on the screen. It will then try to continue printing with a reasonable default value for the error. If there is no reasonable value, it will stop and ask for input from the operator.

The error messages are:

- LINE WON'T FIT AS ENTERED . . .** This message usually occurs with .FO OFF. The computer removes one character from the line and tries again to print the line. The computer keeps removing characters until the line will fit on the paper.
- LETTERSET NOT FOUND . . .** This message indicates that the .BF or .AF command specified a font that the computer couldn't find on any of the disks in the drives. The computer defaults to the ordinary Epson font.
- FILE NOT FOUND . . .** The computer couldn't find the specified text file. The computer pauses until the operator enters a new name for the text file.
- WORD TOO LONG . . .** This message occurs only with the format ON. A word is too long to be printed with the current line length. The computer will print as much of the word as will fit on the paper.
- USE POSITIVE NUMBER . . .** One of the dot commands was followed by a negative number or zero. The computer selects a value of +1.
- ERROR! ERROR CODE="NUMBER" . . .** The computer doesn't recognize the error. It prints out the code determined by the BASIC ERR/2+1 command, and pauses. You can look up the error code in your operating system manual.
- ILLEGAL DOT COMMAND . . .** The computer encountered a line consisting of a single dot with nothing after it. The computer ignores this line.
- FILE TOO BIG . . .** The computer tried to read a record number > 32767 from a letterset. The computer may print out some "garbage" after this message.
- LINE LENGTH TOO LONG . . .** The .LL command specified a line length > 255. The default value is .LL 70.
- MUST DEFINE REGULAR FONT BEFORE ALTERNATE FONT . . .** self-explanatory.
- MAGNIFYING FACTOR TOO BIG . . .** The computer selects a default value of "no magnification".
- ALTERNATE LETTERSET WON'T FIT WITH "FILENAME" . . .** The computer selects the smallest font, "MP", as the alternate font. The "MP" font fits with all other fonts.
- SUBSCRIPT STARTS TOO FAR DOWN . . .** The computer selects the last line of the regular letterset as the line to start subscripting.
- CAN'T INDENT THAT FAR AND KEEP RIGHT MARGIN . . .** The computer defaults to "no indentation".
- TOP OR BOTTOM MARGIN TOO BIG . . .** The sum of the top and bottom margins is bigger than the page length. The computer defaults to one inch for both margins.

TITLE TOO LONG . . . The top or bottom title won't fit on the paper. The computer defaults to no title.

In addition to the error messages, there is one warning the computer prints on the screen: **ALTERNATE FONT WON'T FIT WITH "FILENAME"**. This occurs after a .BF or .TF command. The current alternate font is too big to fit with the new font. You should define a new alternate font, or not use the current one.

APPENDIX E

UNDERLINING HINTS

On the following page is a diagram of the print head and how it is used in bit image printing. The following hints should give you a good idea of how the .UP and .HL commands work and, will undoubtedly give you a better perspective on the operation of bit image processing.

In order to fully understand this section you will have to look at the following diagram, the PIN DIAGRAM on the "Cheat Sheet", and the font parameters in Appendix C, page 18.

1. First, look at the "cheat sheet". The Print head layout shows the 8 out of 9 pins that we can use for bit image programming. The zero pin is not addressable since we only have 8 bits to work with. So, one print line is comprised of 8 dots arranged vertically. Each pin has a numerical value and by sending out that value, the pin is fired and a dot results. To combine pins, simply add their values: pin 8 is 128 and pin 1 is 1, so 129 will fire pins 1 and 8.
2. Now look at Appendix C. The fourth column, "# of lines" represents the number of print lines that are needed to create each font. Each line is comprised of 8 dots so, if a letter style uses 2 lines, it uses 16 dots. Since the minimum group of dots is 8, in order to use 9 dots it is necessary to use two lines, the first 8 dots of the first line and the 8th dot of the second line.
3. Now look at the following diagram. It is an example of a two line font, with a letter that uses 9 dots, all 8 of the first line and only one of the second line.
4. As you can see, dots 7 to 1 of the second line are available and that is how we use the underlining. If, in the example given, we want to underline the letter A, it can be done in many ways. By setting .UP to 64, the underline command will cause a line directly in contact with the bottom of the letter. Setting .UP to 32 would then underline with a dot space between the bottom of the letter and the line. Setting .UP to 40 (32+8), would double underline with a dot space below the letter and between the underlines.
5. Go back to Appendix C. The column "Actual LETTER VERT. SIZE" is just that. It is the number of dots that the letter extends from the top of the frame. This includes any spaces between the top of the letter and the top of the frame. So, now you can figure the .UP value easily. Let's do some examples.

MB is 2 print lines (like the diagram) and is 12 dots high. That means that the pins #4, 3, 2 and 1 of the second line are available for underlining. They can be set in any combination.

Now consider MC. Minicubes also uses 2 print lines but it is 15 dots high. that means that only dot 1 of the second print line is available for underlining. The underline will be in direct contact with the bottom of the letter.

That is all there is to it. The same procedure is followed for those fonts that use more than two print lines. The values are always the same for each relative dot.

Is this true? Here is an example:

```
.bf mc
```

```
.up 1
```

```
!$TEST!%
```

Will print as:

TEST

end.

```
.bf pl
```

```
.up 40 (32 + 8)
```

```
!$TEST!%
```

will print as:

TEST

APPENDIX F HINTS FOR THE LARGE DOCUMENT WRITER

Obviously, this manual can be considered a large document and by writing it I actually accomplished several goals. First, I was able to test the DOTPRINT program for bugs. Second, I was able to create a manual that will both instruct the user and demonstrate the program. And third, I learned a few tricks that make processing large documents much easier.

1. Processing large documents creates a great deal of room for the user to make mistakes. You might turn the center feature on, for example, but forget to turn it back off. It is really a terrible feeling to print out a large document and, after completing about half of it, find that a glaring error exists in your commands.
2. What about spelling errors or other similar mistakes. I use Electric Webster to check my spelling but that doesn't prevent me from leaving out a word or doubling a word. Again, this is a fatal error if you don't find it until the end of your printing process.

These are just some of the mistakes I have made and you may even spot a few of them in the manual. Even though we have added features to DOTPRINT that allow you to segment and reprint just portions of text with specified page numbers, it can still be a trial to get an error free document. So, based on my mistakes, here are some hints to make your work much easier.

1. Write your document in small sections. That allows for easier debugging. Since you can APPend or IMbed any number of files, there is no limit to the length of the overall document and page numbering will still be consecutive. Should you have to correct a section, you will only have to reprint small portion and you can use the "CURRENT PAGE NUMBER" option at print time to select the proper page sequence.
2. Use a "STANDARD PRINT SETUP" to keep your document uniform. For example: If most of your document will be printed in the PL letterset (as this document was) then create a file called STRD/PRT and include all of the setup commands like AD, I#, \$0, TM, BM, TT, BT, BF and AF etc. This "STANDARD PRINT" file can then be IMbedded at the top of each section or, IMbedded anywhere you lose track of what changes you have made. This will keep each segment uniform and give you a standard to fall back on in case you get lost. Be sure to include all of the NEUTRALIZING commands like CW OFF, IN 0, FO ON etc. so that you can clear any errors. If you are not sure how it is set, simply IMbed the STRD/PRT file and then reset the features you want.
3. Save your files frequently and keep an up to date backup. I have lost a great deal of hard work by crashing a one of a kind file, or disk!
4. Finally, use every feature your word processing station has; grammar and spelling checkers are great and any other special features can be useful. Remember that DOTPRINT only prints your file. You still have the full power of your word processor at your command!

DOT PRINT COMMAND LIBRARY

W - Command works only with Dot Writer Fonts

ST - Command works only with Standard Epson Fonts

Note that multiple dot commands may be placed on a single line, providing they are separated by a semi-colon (;) and provided each command which requires a filename, i.e. the .BF command, is on its own line. Also, for a dot command that takes a parameter, such as .LD, to work properly, it must be enclosed in double quotes. The proper format would be:

```
.bf pl  
"test" .ld  
TESTING HERE  
"test" .ld  
TESTING IN 0
```

LD "margin" LEFT MARGIN

This is your absolute left-hand margin in characters. It is specified by specifying a number. "0" would give you a left margin of zero characters. The right margin is specified by specifying a number. The default value is 5 which is equal to 5 characters.

ALF "name" ALTERNATE FONT

This command selects the letter set. When the regular letter set is selected with the .BF command, letters may be inserted among the regular letters. To begin printing with the alternate letter set, use the command .ALF "name". To return to printing with the regular letter set, use the command .BF "pl".

```
.bf pl  
"test" .ld  
testing !/ testing testing  
testing TESTING testing
```

The alternate letter set must have width less than or equal to the width of the regular letter set. You can, however, print the alternate letter set in double width mode. (See below for an explanation of the double width mode.) For example:

```
.bf pl  
"test" .ld  
testing !/ ! (testing !)? testing  
testing TESTING testing
```

You can use the alternate font for subscripting. To use the alternate letter set by "n" where n is the number of spaces down before printing. Note, however, that "n" must be less than the number of printer lines used by the regular letter set.

Dot Print Command Library

.bfpl
.bf ap-1
E=MC1/2?

will be printed as
E=MC
2

.bf /BZ
D/ MC-2
D/ OI?TP1/ OI?T
will be printed as

D, TP O O O

Font notes: Alternate characters always correspond to the regular font. If you wish to use the alternate font, you must specify the alternate font.

.AP "filename" APPEND FILE

Append is used to chain text files together. When encountered, the file specified by "filename" will be printed. Control of printing is also passed to this file.

.EF "name" BEGIN FONT ? .EF-1 CHANGE TO REGULAR EPSON FONT

This command is used to change from one character font to another. The character font specified by a dot file of "name" is used. The file name is a local disk file name. The file name should contain only characters from the character fonts or files created by the Dot Writer. If the file name contains characters not in the character fonts, resolution blocks may be printed using this command. Your own character fonts may be printed using this command. If the command is issued, the normal Epson font begins. NOTE: The .EF command must always be the last statement on a dot command line. We suggest that you always use .BF commands on individual lines.

EXAMPLE: .BF OE

This would specify that all following text be printed using Old English Character Font. "OE" is the file name that the Old English Character Font is stored under.

Suppose you had created a file by the name of "MYFILE" using the Dot Writer editor. To use it as a font you would specify:

.BF MYFILE

Dot Print Command Library

.BR BREAK

The **.BR** command causes text to be printed starting on the next line. Any unrecognized command that is encountered is treated as a **BREAK** command by Dot Printer.

.BM n BOTTOM MARGIN

This option allows you to specify the number of lines you wish to have on the bottom margin. Default value is **".BM 6"**.

.BT "title" BOTTOM TITLE

This command specifies the title to be printed at the bottom of each page. The default is no title. Otherwise this command works the same as **.TT** described above. Example **.bt pages** will cause the pages to be numbered at the bottom.

.CE on/off CENTER LINE

This is used to automatically center lines of text. The current line length and left margin is used to determine the line center. The contents on each line followed by the **".CE on"** command is centered until a **".CE off"** is encountered. Note that the **".FO on"** will also be temporarily turned to off. Also - the **CE** command should be used only when proportional (**.PR**) is off. Otherwise use **.CH**.

.CP n CONDITIONAL PAGE EJECT

With this command, a new page can be forced if the number of lines left on the page is less than the value specified by **"n"**. This is used to keep things, such as titles, from being printed "alone" on the bottom of the page.

.CW off/on CENTER using paper WIDTH *

The Center using paper width option is similar to the **".CE"** command except, the center of the paper is used. In the **".CE"** option the center is determined by your margins and line length. **.CW** only applies to the Graphic lettersets - not to the normal Epson font (**.BF 1**).

.DA on, off DARK PRINTING *

.DA 0, 1, 2, 3 DARK PRINTING **

The **".DA on"** command is actually the equivalent of the Epson double strike printing mode, except it is designed to be used with Dot Writer character fonts. You should note that for some letters such as the Microprint, or Minicubes, the **".DA off"** option may give better resolution.

The **"DA 0, 1, 2, 3"** options are used with the standard Epson font, and are defined as follows:

Dot Print Command Library

- .DA 0 Normal Epson print
- .DA 1 Emphasized mode
- .DA 2 Overstrike
- .DA 3 Double emphasized

.DS on, off DOUBLE SPACING

The double spacing option will automatically double space your text at print time.

.EM ON , .EM OFF EMPHASIZED PRINT :

This command applies only to the GEAP lettersets. It controls emphasized printing. Emphasized print is slightly wider than regular print. Using combinations of .EM and .DA, you can control the darkness of the print. For example:

```
.bf MB
.em off ; .ds off
testing
.em on
testing
.em off ; .ds on
testing
.em on
testing
```

will be printed as:

```
testing
testing
testing
testing
```

.EN END OF FILE

This is the END command. If the computer encounters .EN while processing an imbedded file (see the .IM command), it immediately returns to the main file. If the computer is processing the main file then .EN is the last command it will process. An .EN command should be put at the end of every SCRIPSIT or PENCIL file. Otherwise, the computer may print out a string of "garbage" at the end of the file.

.ES "symbol" ALTER ESCAPE CODE

The .ES command changes the symbol for the escape code. The default symbol is "!". You may want to print the "!" or other control symbols and thus need to change the escape code symbol to something else. Example:

```
.es #
test #test#/ test
will be printed out as:
test test test
```

The second symbol in the escape code sequences is, however, always the same. Assuming the escape code symbol is "!", then the escape

Dot-Print Command Library

sequences are:

!A . . . Begin alternate font.
!? . . . End alternate font and resume regular font.
!& . . . Begin underlining non-blank characters.
!\$. . . Begin underlining both blank and non-blank characters.
!% . . . End underlining.
!(. . . Begin double width mode.
!) . . . End double width mode.

These sequences can be intermixed. For example:

```
.bf pl  
.af se  
testing !/testing!$ testing !(testing !/  
testing !? testing !) testing.
```

will be printed as:

```
testing TESTING TESTING TESTING  
TESTING testing testing
```

.FO on/off FORMATTING CONTROL

The **FO** command will automatically format your text based on the line length that you have set. For example if you typed lines of text as follows:

```
I don't care what  
people say, Dot Writer fonts are  
here to stay.
```

They would appear exactly that way if the format was off and they would appear as below with the format turned on.

```
I don't care what people say. Dot Writer fonts are here to stay.
```

.FM n FOOTING MARGIN

The **FM** command sets the footing space to the integer **n**. When the text reaches the bottom margin on any page, the computer will move down "n" spaces before printing the bottom title. The default value is **.FM 1**.

.HL HORIZONTAL LINE #

This is the **HORIZONTAL LINE** command. It works only with the **GEAP** lettersets. When the computer encounters ".HL", it draws a horizontal line. The print head pin (or pins) that draw the line are determined by the **.UP** command. (See the section on the **.UP** command.) The length of the line is equal to the distance set by the **.LL** command. Example:

Dot Print Command Library

```
.lf 50  
.bf pl  
.hl  
.sk  
test  
.hl  
would be printed as:
```

```
test
```

.HM n HEADING MARGIN

The "HEADING MARGIN" is the space between the last Top Title line and the first line of the body of the document. One or two blank lines in this position will separate the Top Title from the body of the text to give a familiar format. An integer must be specified which is equal to the number of lines desired.

.IG ON , .IG OFF, .IG 1, .IG 2 IGNORE

The IGNORE command is used for "debugging" and preliminary drafts. The .IG ON command causes the computer to skip over all text and dot commands until it encounters an .IG OFF command. When the .IG OFF command is encountered, processing resumes as usual.

The .IG 1 command causes the computer to skip over text, but to continue processing dot commands until an .IG OFF command occurs, which resumes normal processing.

The .IG 2 command causes the computer to ignore .BF, .AF and .TF commands, but to process all other commands normally, until an .IG OFF command occurs. In other words, the .IG 2 command causes the computer to print all text in the standard Epson font. The Standard Epson font is processed faster than the graphics lettersets, so .IG 2 is useful for preliminary drafts and quick copies.

.IM "filename" IMBED FILE COMMAND

When Dot Print encounters the .IM command, the computer will open the file "filename" and begin printing text from that file. When all the text in "filename" has been printed, the computer will resume printing text from the original file. Example:

If file "A" has the text "test 1", and file "B" has the text "test 2", then the file:

Dot Print Command Library

```
.fo off  
start  
.im A  
.im B  
end
```

will be printed as

```
start  
test 1  
test 2  
end
```

Note that **.IM** commands cannot be nested.

.IN n INDENT LEFT MARGIN

This command will cause a relative indentation of left or right margin from the current position. The indentation is in tenths of an inch. All indent commands are additive. For example an indent command of 6 and an indent command of 5 given at a later time would give you a total indentation of 11. Specifying ".IN 0" will always restore the original left margin.

.JU on, off RIGHT JUSTIFY TEXT

With the JUsTify on, all text is right justified giving a smooth right hand column. Spaces are automatically inserted to achieve this. With the JUsTify off, a ragged right border will be printed.

.KE KEYBOARD ENTRY

This command requests keyboard input. When the computer encounters this command it pauses until a line is typed in from the keyboard. The line can contain dot commands or text to be printed. The computer processes the line as if it were read in from disk.

.LH n (where n = an integer)

To specify spacing between lines, use the .LH n, where n = a positive integer and specifies the number of dot widths to space down before beginning the next line. If n is small enough, the lines will overlap. This allows you to mix fonts on almost the same line. To restore spacing to normal specify .LH 12, which is the default. The .LH command works only with GEAP fonts.

Dot Print Command Library

Example 1:

```
.bf oe
.cw on
.lh 1
GEAP          GEAP
.bf mp
.sd 2
GEAP
```

Will print as:

GEAP **GEAP**

Example 2:

```
.fooff; .bf oe
.lh 1; .mf2
C
.bfl
chapter 1
```

Would be printed as:

Cchapter 1

You can use this technique to underline or create other special effects, but it will not work with all fonts due to differences in character sizes. Experiment for best results. Also, for underlining, the .UP command used with the underlining control codes, is a much easier approach.

.LL n LINE LENGTH

The .LL command is used to set the line length. In conjunction with the .AD command, it determines your right margin width. The default value is .LL 70 or 7 inches from the left margin.

.LS n Logo Space

This command works only on the first page. Also, it must be the very first command in a file if you select to use this option. You must position the paper to the proper starting point when using this command. The logo space command specifies the number of lines you wish to reserve for a logo (such as on a preprinted stationary). ".LS 15" would specify a 15 line border starting at the top of the page. If you want the minimum ".LS", you should specify 2 to keep from printing on the paper perforation.

.MF n MAGNIFICATION FACTOR

This command will allow you to Magnify any of the Dot Writer's character fonts by a factor specified by "n". Only whole numbers (ie. integers) are allowed. There is a maximum factor that can be specified, and this is determined by the particular font. If greater than the maximum allowed is specified, a default to no magnification occurs. "Format on" is temporarily disabled while this command is working. If too many characters are on the

Dot Print Command Library

line to be printed, the maximum number of characters that can be printed is calculated, and characters to the right of this are disregarded.

.MX 100 , .MX 80 MX 100 WIDE PAPER COMMAND

This command allows you to use DOTPRINT with the MX-100's wider paper. If you are using the MX 100 with 13.6" paper, then you should give the command **.MX 100** before the first line is printed. (The command may be preceded by other dot commands, like **.LS n**.)

.OF n OFFSET COMMAND

The **OFFSET** command is used only with the **GEAP** lettersets. It works somewhat like a "Reverse" paragraph command. The first line after the **.OF n** command is printed with normal length, but all following lines are indented by *n* tenths of an inch, until the next **.OF n** command is received. To restore normal printing, use the command **.OF 0** or the command **.IN 0**. For example:

.of 5

Now is the winter of our discontent made glorious summer
by this son of York. To be or not to be. That is the bare
bodkin.

.of5

Now is the winter of our discontent made glorious summer
by this son of York. To be or not to be. That is the bare
bodkin.

.of 0

Now is the winter of our discontent made glorious summer
by this son of York. To be or not to be. That is the bare
bodkin.

will be printed as

Now is the winter of our discontent made glorious summer by this son
of York. To be or not to be. That is the bare bodkin.

Now is the winter of our discontent made glorious summer by this son
of York. To be or not to be. That is the bare bodkin.

Now is the winter of our discontent made glorious summer by this son
of York. To be or not to be. That is the bare bodkin.

.PA PAGE EJECT

The **.PA** command causes the page to be automatically ejected, and printing will start on the next page.

.PL n PAGE LENGTH

The **PL n** command lets you use paper of different length than the standard 11" paper. Just give the command **.PL n** where *n* is the number of standard lines that will fit on the paper. (Standard Epson print is 6 lines per inch.)

Example: **.PL 60** would correspond to 10" paper.

Dot Print Command Library

.PN ON, .PN OFF PAGE NUMBER ON/OFF

This command turns page numbering on and off. The default is .PN ON. Each page, except the first, will be numbered at the top until either a .PN OFF or .TT "title" command is encountered.

.PP n NEW PARAGRAPH

The .PP creates a new paragraph with an indent of 5 tenths of an inch unless integer n is specified. If n is specified, the indent will be n tenths of an inch.

.PR ON, .PR OFF PROPORTIONAL PRINT ON/OFF ‡

This command turns proportional print on and off. It applies only to the GEAP lettersets.

If your operating system uses a lot of high memory, and the letterset is a large one (MB2, for example) then some of the letters may come out monospaced even with .PR ON

When using proportional print, it is a good idea to specify some spacing between letters with the .SD command. Also, to center proportional text, use the .CW command rather than the .CE command.

Example:

```
.bf pl
.pr off
.sd3
testing proportional print
.pr on
testing proportional print
```

will be printed as
testing proportional print
testing proportional print

.PS # REDEFINE PAGE NUMBER SYMBOL

The default PAGE NUMBER SYMBOL is the "\$". If you want to use this symbol in text, redefine the symbol with the .PS command. The symbol following the .PS will become the page number symbol. .PS # will change the symbol default to "#".

.RE 0, 1, 2 REVERSE OPTION ‡

This option is used mainly with high resolution drawings. It is similar to REVERSE video except it works on paper. "RE 1" will give blanks in between letters if Spaces Between letters "SD command" has been set greater than 0. "RE 2" will give print black spaces between letters if the "SD" command has been set greater than 0.

Dot Print Command Library

.SD n SPACE DOTS BETWEEN LETTERS *

The .SD command allows you to adjust the number of spaces between letters. Default value is zero. Due to limitations of disk storage space, some of the letters are packed on disk. One example is the Minicubes. If you print them with the default setting, they will print very close together. If you specify ".SD 6" and then print them, they will appear more attractive. This option also allows you to spread your text out to take up a given amount of space. There is a maximum SD factor for each letter, and will vary depending on the character font. The maximum is determined by the dot width of the character. After the maximum is specified, there is a default to the width of the letter in the current font. If you want more than this, simply use a space character when typing your text.

.SK n Skip n Lines Where n is any Integer

This is a straight forward command. SK means SKIP and n is any integer. .SK 5 will skip 5 lines. A control break is also generated by the .SK command.

.ST "message" STOP, Print message, await <ENTER>.

The .ST command is designed to allow you to stop the printing without losing continuity. Hitting <ENTER> resumes with whatever commands follow the .ST command, and page numbering, format, top title, bottom title, etc. remain in force. If you are printing a long document and the appended files run onto a disk that you cannot keep on line. The following procedure will allow uninterrupted printing:

```
text text text text
.st Message Here (Await <ENTER>)
.ap nextfile
```

The text will print until the .ST is hit. Then processing will stop and the message will be printed on the CRT only. After you have made the diskette swap, hit <ENTER> and "nextfile" will be appended - processing will continue as if it was never interrupted!

.SW x SPACE WIDTH *

This is a rather esoteric command. It applies when proportional print is in effect. It sets the minimum width for blanks. The number "x" is a decimal between 0 and 1, and determines the minimum width for blanks as a fraction of the maximum character width. The default value is .5. Example:

```
.pr on
Note spacing between words.
.sw .8
Note spacing between words.
```

will be printed as:

```
Note spacing between words
Note spacing between words.
```

Dot Print Command Library

.TM n TOP MARGIN

Use this command to adjust the top margin. Default top margin is ".TM 6". Top margin is specified in number of lines.

.TT "title" TOP TITLE

TOP TITLE specifies the title to be printed at the top of each page. If the "title" message contains the symbol "\$" then this symbol will be replaced by the page number. For example, **.TT MANUAL-PAGE\$** will cause **MANUAL-PAGE 2** to be printed at the top of page 2.

The default value for .TT is 60 blanks followed by "page\$". The top title will be printed on the first page only if a .TT command is received before any text has been printed.

The letterset in which the title is printed is determined by the .TF command, discussed below. The title can contain escape codes for underlining and double width. But the title may contain letters from an alternate font only if the alternate font is the same size or smaller than the title font.

.TF "name" TOP/BOTTOM TITLE FONT

This command determines the letterset in which the top and bottom titles will be printed. The default is standard Epson print. Formatting options such as .DA, .CW, etc. are those in effect when the .TF command is encountered. For example: **.PR ON ; .EM ON ; .TF OE** will cause the top and bottom titles to be printed in proportional, emphasized, Old English print.

Printing is much faster if the title font is the same as the font used for the main body of text. By the same, we mean the filenames of the fonts are identical, both capitalized the same, etc. If this is not done, .ie .bfpl and .tfPL will load the same file but will reload the file at the top of each page, wasting time! .BF PL and .TF PL will require only one disk access. They are identical

.UP n UNDERLINE PRINT

This command specifies which of the 8 print head pins is used for underlining. The bottom pin is 1, the next pin is 2, the next is 4, the next is 8, and so on until the top pin, which is 128. More than one pin can be fired at once by adding the pin numbers. The escape codes for underlining are:

Dot Print Command Library

"&" begin underlining non-blank characters,
"\$" begin underlining blank and non-blank
"% " to end underlining.

Examples:

```
.bf pl  
.up 64  
testing !$testing testing!% testing  
.sk  
.up 1  
testing !&testing testing!% testing  
.up 65  
.sk  
testing !$testing testing!% testing
```

will be printed as:

```
testing testing testing testing  
testing testing testing testing  
testing testing testing testing
```

The default value is .UP 1. Note the "CHEAT SHEET" for suggested UP values for each supplied font!

Miscellaneous

Block graphics can be used with DOTPRINT, even with GRAFTRAX-PLUS. The "PL" letterset contains the TRS-80 graphics blocks. To print using block graphics, draw the picture with GEAP, then save it to disk with GEAP's "FG" command. Suppose the filename of the saved picture is "pic". Then to print it with DOTPRINT use the following sequence:

```
.bf PL  
.fo off ; .pr off ; .lh 4  
.im Pic
```

and the picture will be printed. (You may also want to move the picture toward the center of the paper with the .IN command.)

The double-width mode, unlike the magnify mode, can be used with format on. Recall that the escape codes for double width are: "!" to begin double-width, and "!" to end double-width..

Dot Print Command Library

In past versions of DOTPRINT it has been necessary to specify font modifying commands (.ie .SD, .LH etc.) after the target font was loaded with the .BF command. That is no longer true. You may now specify font modifying commands at any time. Remember that modifying commands issued early in text will not cancel upon loading a new font, so be sure that you specify the modifications that you want for each font.

Any command that contains a filename or message must be the last command on a line; similarly for the .KE command.

Escape code commands only work with the GEAP lettersets. When using the standard Epson font, sentences containing escape code commands are printed "as is".

Finally, DOTPRINT now requires four open files to perform all its functions. This is the same number of open files that NewScript requires.

Finally, for clarification, in order to use the lock graphics that are available in the PL letterset, follow these instructions (duplicated in the Miscellaneous section on the last page of the Command Library). Draw the graphic display on the video using normal GEAP commands and save the completed work with the GEAP "FG" command. When you want the graphic display printed, simply imbed the file previously saved with the "FG" command, in your text, and DOTPRINT will load it and print the graphics. You must be in the PL letterset for this to work. If the "FG" saved file is called "PIC", the procedure would be as follows:

```
.BF PL  
.FO OFF;.PR OFF;.LH4  
.IM PIC  
.LH8;.PR ON;.FO ON  
.BL next font.
```