DOTWRITER 4.0 MODEL 4

for the TRS-80 Models I, III, and 4   DMP/OKI

INCLUDES THESE FOURTEEN LETTERSETS:

MB2 MEDIUM BOLD 2

MB Medium Bold

TR TypeRite

FF FLAT FACED

GR GREEK ГPEEK

OE Olde English

MP Micro Print

MC MiniCubes

PL Plain

SE Small Enhanced

BB Big Bold

BO Bold

MID Mid Night

SPL Small Plain

Copyright (c) 1984 by W. K. Mason and PROSOFT

Distributed by: PROSOFT

Box 560 / North Hollywood, CA 91603 / (818) 764-3131

This cover was printed, just as you see it, by DOTWRITER

DOTWRITER 4.0

for the TRS-80 Models I, III, and 4

This book explains how to use DOTWRITER (tm) on the TRS-80 (R) micro-computer. It applies to Version 4.0 and above. Additional Supplement sheets may be issued from time to time.

This edition is a minor revision of the DOTWRITER documentation.

Date: January, 1986   Printing: 15 14 13 12 11 10 9

P R O G R A M   L I C E N S E   A G R E E M E N T

THESE PROGRAMS ARE PROVIDED "AS-IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. PROSOFT, The Tesler Software Corporation, and/or the author shall have no responsibility or liability to the purchaser or any other person, persons, or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by use of these programs and/or this documentation. This expressly includes, but is not limited to, loss or invalidation of customer data, programs, files, or business opportunities.

Purchase of this software conveys to CUSTOMER title to the media (the diskette) but not title to the computer programs or files. CUSTOMER is granted a non-exclusive license to use this software on ONE computer.

Copies of SOFTWARE may be made only for BACKUP purposes. If you transfer any copy or modified copy of any portion of DOTWRITER to another party, your license is automatically terminated and all support, maintenance, and update privileges are lost. If these programs are to be used on more than one computer at a time in your company, a separate copy must be purchased for each machine. Please contact us for volume pricing.

MEDIA WARRANTY

PROSOFT warrants that the diskette on which you received DOTWRITER is free from mechanical or recording defects, and that we will replace any such defective diskettes within 90 days of the date of purchase. No other warranties are expressed or implied as to the operation, use or suitability of these programs or lettersets.

A C K N O W L E D G E M E N T S

TABLE OF CONTENTS

SECTION ONE
CHAPTER ONE
INTRODUCTION TO DOTWRITER

Welcome to DOTWRITER! This software package formats and prints text, like any other word processor; but it also lets you use a wide variety of fancy and large letters -- literally thousands of sizes and shapes that aren't built into your dot matrix printer. You can write your letters and headings with the "Editor" portion of your regular word processor, and then print the file using DOTWRITER.

The fancy letters are called "letter sets" or "fonts." Each set contains up to 96 characters, including a complete alphabet (usually lower-case as well as upper-case), numbers, and special symbols. You can choose whichever letterset you want to use at any particular moment, and you can use more than one of them at a time: in fact, you can use any two lettersets on the same print line, and use many of the lettersets on the same page, if you want to!

DOTWRITER can do a lot more than just print fancy shapes: it is a full-function text formatter, with the ability to print right-justified proportional text, centered headings, underlining, page titles, indentation, and vertical and horizontal lines. In other words, it's a wonderful supplement to your normal word processor. Needless to say, this manual was entirely printed by DOTWRITER.

HOW TO USE THIS BOOK

This book describes DOTWRITER 4.0. It contains two major sections. The first will help you get started with DOTWRITER, while the second explains every DOTWRITER command. We suggest you read the entire first section, install DOTWRITER as explained, and try the tutorial example. The second section is intended as a Reference Manual, so you probably won't need to read through it, but just skim it to get an idea of what you can do later on.

The Letterset Design System (LDS) is separate from DOTWRITER, and has its own documentation. The LDS book may be bound in at the back of this book, or may be supplied separately.

To get started with DOTWRITER, just follow these steps:

1. Install it as explained below;
2. Read the rest of this "TUTORIAL" section;
3. Write something with your word processor's editor;
4. Include formatting control commands;
5. Save the file to disk in "ASCII" format;
6. Make sure the needed lettersets are "on-line";
7. Print your file with the "DOT" command from DOS.

If you're using ALLWRITE or NEWSCRIPT, the files are already in "ASCII", and you can select "DOT" by soft key or from the Primary menu. If you don't have enough disk space to keep everything on-line, you can remove the "DOT/CMD" disk as soon as DOTWRITER starts asking you questions.

INSTALLATION

The way to install DOTWRITER depends on the computer and DOS you are using. The next few paragraphs cover the most common configurations. In all cases, you will need two disk drives. Installing optional "Font Disks" is a separate step, so please review those instructions also.

MODELS I and III

DOTWRITER for the Models I and III is distributed on a special disk that is mostly in single-density, Model I 35-track format. All popular DOS's can read this disk and copy files from it, but they cannot make direct "BACKUP" copies of the disk itself. Since you can copy the files to your own disks and then put our disk away, this recording method will not prevent you from making additional copies of the files for safety. What makes this disk unusual is that you can "boot" it on both a Model I _and_ a Model III, and then let it transfer its contents to your TRSDOS disks.

The DOTWRITER system includes too many files to fit on one Model I disk, so our distribution disk is in "flippy" format. A "flippy" disk is like two separate disks, not like a double-sided disk. To read the first side, insert the disk normally (label UP or facing LEFT). To read the second side, turn the disk over and insert it with the label DOWN or to the RIGHT. A double-sided drive cannot read the back of this disk, and with some unusual drives, you may have to cover the "left-hand" index hole (near the hub) temporarily. All popular DOS's on the Models I and III can read this disk: to read side two, just turn the disk over.

MODEL I

1. Use your DOS to "FORMAT" two data disks in drive 1.
   These must be SINGLE-DENSITY disks.
2. Remove your DOS disk from drive 0.
3. Insert Side 1 of the DOTWRITER master in drive 0.
4. Press "RESET" to boot our master.
5. When asked for the destination drive, reply "1", press ENTER.
6. Our Side 1 files will be copied to your data disk.
7. Put the other formatted data disk in drive 1.
8. Turn our master over and press "RESET" again.
9. Specify drive "1" again to copy the rest of the files.

Label both data disks, put your DOS back in drive 0, "RESET" once more, and then skip ahead in these instructions to "DOTWRITER COMPONENTS".

If you use a double-density DOS, and want to keep DOTWRITER on your DOS (drive 0) disk, then make a backup of your DOS, "KILL" as many files as you safely can, and then copy the appropriate "DOTxxx/CMD" and all the "xxx/PR" files from both sides of our disk to your DOS backup. See "COMPONENTS", below, to select the appropriate "DOTxxx/CMD". If you expect to use our Table of Contents or Index feature, also copy "TCONINX/CMD" to your disk.

MODEL III TRSDOS

1. Use TRSDOS to FORMAT a data disk in drive 1.
2. Remove the TRSDOS disk from drive 0.
3. Insert Side 1 of the DOTWRITER disk in drive 0.
4. Press the orange "RESET" button to boot our disk.
5. Specify "1" as the destination drive, press ENTER.
6. All side 1 files will be copied to your disk.
7. Turn our disk over to side 2.
8. Press "RESET" again.
9. Specify "1" again, to copy the rest of our files.

Now, skip ahead to "COMPONENTS" to complete the installation.

MODEL III DOSPLUS, LDOS, MULTIDOS, NEWDOS/80

These DOS's can read our single-density disk "as-is". With NEWDOS/80, you must change "PDRIVE" to recognize single-density, and this is explained in the first example of "PDRIVE" in your NEWDOS/80 manual.

READ THE NEXT PARAGRAPH CAREFULLY, BEFORE YOU ATTEMPT TO USE OUR DISKS!

Put your DOS in drive 0 and put side 1 of the DOTWRITER disk in drive 1. Use the "DIR :1" command to try to display our directory. If it does not work, use "CONVERT", "REPAIR", or "WRDIRP" to make our disk readable, and then try again to display the directory. (WARNING: If you are using a Model 4 manufactured after January 1st, 1984, some DOS's will permanently erase any single-density disk if you try to write to it or CONVERT or WRDIRP it. If this applies to you and you cannot read our disk "as-is", send it back to us before you erase it, and ask us to convert it for you.)

If you want to keep DOTWRITER on your DOS disk, make a backup of the DOS, "KILL" as many files as possible, and then copy the appropriate "DOTxxx/CMD" and all the "xxx/PR" files from both sides to your disk. If you want to keep DOTWRITER on a data disk or a hard drive, just copy the same files to the disk you want to use. Then, see "COMPONENTS", below, to complete the installation.

MODEL 4

The Model 4 DOTWRITER disk is a standard, single-sided data disk, so you can copy files from it or make BACKUP copies of it:

1. use your DOS to FORMAT a data disk in drive 1.
2. With TRSDOS, type: BACKUP :0 :1 (X)
    the DOTWRITER master must **NOT** be write-protected for this
    our password is either "PASSWORD" or nothing (just press ENTER)
    with DOSPLUS, type: BACKUP
3. Remove DOS from drive 0 and insert our disk in drive 0.
4. Answer your DOS's questions to complete backup.
5. Put your DOS back in drive 0 when done.

COMPONENTS OF DOTWRITER

Regardless of the computer or printer you are using, these are the components of the DOTWRITER system:

1. DOT/CMD -- the main printing program

   DOT/CMD does the actual text formatting and printing, and is the most important part of DOTWRITER. To match our program with your printer, we have given temporary names to the versions that support each printer. These names are listed below, and you should pick the one that works with your printer, then COPY or RENAME it to "DOT/CMD" on a disk of your own. We will refer to this program as "DOT/CMD", "DOT", or sometimes "DOTPRINT" throughout the documentation.

   Note: only one or two of these will be on your disk, and they will be on the FIRST side if you have a Model I or III version of DOTWRITER:

DOTEPS/CMD	Epson MX, RX, and FX; any IBM-compatible (or DMP in IBM mode)
DOTITO/CMD	C. Itoh 8510 and 1550
DOTOKI/CMD	Okidata Microline 84-II, 92, 93 (without Plug-n-Play)
DOTDMP/CMD	Radio Shack DMP 105 and up (in DMP, not IBM mode); CGP-220
DOTD21/CMD	Radio Shack DMP 2100 or 2100P

   Regardless of which printer you are using, you should do this "RENAME" so that you can use "DOT" as a valid DOS command.

   If you have two of these printers, and the disk you bought only supports one of them, you can order the second version directly from us for $25.00 (this price is available only to registered owners). The optional font disks contain versions for all supported printers, so you will not need additional copies of them.

   If you have a Gemini 10X or 15X, a Delta 10, a C. ITOH 8600B, or other supported printer not on the above list, please see the "SUPPLEMENTAL INSTALLATION INSTRUCTIONS" at the end of this chapter.

2. TCONINX/CMD -- table of contents, index preparation

   This processes the Table of Contents and Index, if you use those features. The same program works with all the printers. It may be on Side 1 or Side 2.

3. DOTPRINT -- a BASIC program (Models I, III only)

   This provides an interface from NEWSCRIPT to "DOT/CMD". If you're using NEWSCRIPT, copy this file to your working DOTWRITER disk; otherwise, you won't need it. It is not the "DOTPRINT" that is a synonym for "DOT/CMD".

4. Lettersets (on Side 2 of Model I/III disk)

    These files contain character patterns. They are on Side 1 of the Model 4 version, and on the flip side of the Model I/III version. There may be one or two of them on the first side of the I/III disk, so please check both sides. We've included fourteen useful lettersets with DOTWRITER. If you have an EPSON or C. ITOH printer, these lettersets are set up for use with 8-bit printers; if you have one of the DMP or Microline printers, the lettersets are set up for use with 7-bit printers.

    The 8 and 7-bit lettersets are <u>NOT</u> interchangeable. This won't matter to you if you just use the standard ones, but when you buy additional letterset disks from us, they will contain both forms of lettersets: the 8-bit ones will be on the front, and the 7-bit ones will be on the flip side. If you try to use the wrong ones, you'll know it instantly. (It won't hurt your printer, but it sure looks funny.) Solution? Turn the disk over and try again.

5. DOTGEM/BAS, DOTLQ/BAS, DOT8600B/BAS

    These are used to modify DOTWRITER for use with other printers. Please see the instructions at the end of this chapter if you need to use any of them. They are only on the EPSON/ITOH version of DOTWRITER.

6. Optional Font Disks (not included)

    You can buy additional lettersets from us at any time. As we're updating this edition of this book (Winter, 1985), there are 46 disks available, and several more in preparation. Each disk contains between three and sixteen lettersets, and there are over 300 such fonts altogether. When you decide you want more variety, just look over the samples we've sent you, or check with us for new releases.

INSTALLING OPTIONAL FONT DISKS

The font disks are in a "universal" format that allows them to work with all supported TRS-80 models and DOS's, and with all supported printers. The disks are in "flippy", single-density, 35-track format, which is different from a double-sided diskette. You can think of each side as a separate disk, so to use the reverse side, just turn the disk upside down before putting it into a disk drive.

Side 1 has the 8-bit versions of the fonts, and Side 2 has the 7-bit versions. To copy the fonts from these disks to your own disks, please see the instructions that come with the disks themselves. In a some cases, there are some 7-bit fonts on the 8-bit side, because the 7-bit fonts take more disk space. (These fonts have a "7" at the ends of their names.) You cannot make a direct "BACKUP" of the disks, but you can, and should, COPY all the files to your own disks, then make as many backups as you need of your own copies.

PRINT SAMPLES OF FONTS

Each font disk contains a sample file called "FONTx", where 'x' is the disk number: FONT1, FONT15, etc. We suggest you use DOTWRITER to print each sample file soon after receiving the font disk. This will verify that all the fonts are intact, and will also give you a full printout of the character sets.

Many of the FONTx files also show the recommended "spacing" values for the fonts. This won't mean anything to you until you have learned DOTWRITER, but the combination of the print samples and the spacing values is intended to give you all the information you may need to use each font. We offer a "Letterset Reference Catalog" that does the same thing as these sample files, but there are too many fonts now for us to keep that catalog up-to-date. By printing the FONTx files as you receive the disks, you can develop your own catalog, with print samples that match your printer.

Letterset Design System
(runs in Model I and Model III modes ONLY)

The Letterset Design System includes "TGEAP", which is a specialized drawing program, and thirteen "Letterset Manipulation Utilities." This optional system is available for the Models I and III (and Model III mode of the Model 4), but it cannot function at all in native Model 4 mode (if you have a Model 4, you can run LDS under a Model III DOS, such as LDOS or DOSPLUS; we recommend against using TRSDOS 1.3 in this case, because its disk formats are different from those of TRSDOS 6.x).

If you just want to use DOTWRITER as an ultra-fancy graphics text formatter, you already have everything you need. However, if you want to design lettersets of your own, or modify ours, including the ability to change their sizes, combine them, make them suitable for proportional printing, or anything else along those lines, then we recommend that you learn the Letterset Design System. LDS comes with its own manual, but if your copy of DOTWRITER includes LDS, the back of this book may contain the LDS documentation.

FINISHING THE INSTALLATION

After you've copied, killed, and renamed files, you should end up with a "WORKING" disk (or two working disks if you have single-density drives). The working disk may contain part of your DOS and be used from drive 0; or it may be just a "data" disk that goes in drives 1, 2, or 3. The working disk will also contain "DOT/CMD", "TCONINX/CMD", and fourteen "font" files, all of which have the extension "/PR".

If you have a hard drive, you can transfer everything to it, and DOTWRITER will run somewhat faster than it can from floppy disks.

While on the subject of optional hardware, DOTWRITER sends information directly to the printer, so it doesn't work with a software "spooler". An external hardware print buffer may or may not work with DOTWRITER: some buffers modify the data stream and thereby destroy the bit-image graphics commands. If your buffer can just pass the data along "as-is", DOTWRITER will run somewhat faster. You should also be aware that one page of bit-image graphics will fill a 64K buffer, so DOTWRITER won't be able to get several pages ahead of the printer, the way a normal word processor can.

The distribution disk also contains a file called "SAMPLE/F0" (that is a zero, not an 'oh'), and you can use it right now with DOTWRITER to verify that everything is working properly. Make sure you have a DOS in drive 0, and "SAMPLE/F0" on a disk that won't have to be removed if you have to swap disks (this is only a consideration with a two-drive, single-density Model I). Then, with a disk containing "DOT/CMD" in one of your drives, type this from DOS: DOT (and press the "ENTER" key).

After a few seconds, a large "DOTWRITER" logo will appear on the screen. A few seconds later, a message indicating which version you're using will appear. The key letters are "DM" for "DMP"; "EP" for "EPSON", "CI" for "C.ITOH", etc. Then, DOTWRITER will ask you four questions. For now, just press "ENTER" for the first three. The fourth question will ask for the name of the file to be printed. Reply "SAMPLE/F0" (no quote marks, and that is a zero, not an 'oh'), and press "ENTER". If your printer is ready, DOTWRITER will print several pages showing you the complete variety of character shapes in the fourteen standard font files.

If your printer isn't ready, DOTWRITER will ask you to ready it, and then will start printing automatically. If you don't get legible results, it probably means you are using the wrong version of DOTWRITER. If you get legible printing, but the words seem to be split in half and upside-down, it means you are using the right version of DOTWRITER, but the wrong version of the fonts. (This is unlikely now, but can happen when you start to use the optional font disks later on. If it happens, just use the other side of the optional font disk.)

If you don't get good results, please re-check your installation steps. If every-thing seems correct, please get in touch with us for assistance. You may have an incorrect version of DOTWRITER, a defective disk, or you may just have skipped a step in installation. In any case, we're here to help you if we can.

We also urge you to make a couple of extra BACKUP copies of your working disk(s), and to put our master away for safekeeping. You will need our master to get updates from us, and you will want those backups to protect yourself against equip-ment, power, software, or even human error.

ADDITIONAL INSTALLATION INSTRUCTIONS -- all printers

DMP, IBM, and Okidata Considerations

Many newer printers have an "IBM-compatibility" switch or option. When running in IBM mode, these printers look like an EPSON to Dotwriter (you must use DOTEPS/CMD). All the new Radio Shack "DMP" printers have this capability: to use it, turn on DIP switch 1-1. Since the TRS-80 expects the printer to supply automatic line feeds, and Dotwriter follows this convention, you should make sure the Auto-Line Feed switch of your printer will provide a line feed for each Carriage Return. (The TRS-80 standard is different from the IBM standard in this respect). If you have a Microline with Plug-n-Play, you must use our "DOTEPS/CMD", and not "DOTOKI". If you ordered the "OKI" version of Dotwriter, just return it to us and ask for an "EPSON" replacement.

The "DMP" version of Dotwriter will operate properly on the DMP 130 only if you specify ".PI 0" (that is a zero, not an "oh") at the beginning of each of your documents. The DMP 130 and 2100 print much wider than the other printers, and software cannot overcome this in DMP mode. If you want to operate your DMP 130 or DMP 2100P in "IBM" mode when using Dotwriter, we will be happy to replace your Dotwriter and LDS "DMP" disks by "EPSON" versions -- just send them to us and ask for the other version. The DMP 2100 (no "P") does not have an IBM or Epson mode.

The DMP 430 can operate properly with DOTDMP/CMD, but you may have to specify ".PL 65" (65 lines per page) because of the way that printer handles end-of-page. If you want to operate the DMP 430 in IBM-mode, we will replace your DOTDMP disk with the DOTEPS (EPSON) version.

GEMINI, LQ-1500, and C. ITOH 8600B Considerations

This modification step should ONLY be done on backup disks, NEVER on the original disk we sent you.

If you have a GEMINI 10X/15X or a DELTA, use "DOTEPS/CMD" and "DOTGEM/BAS". With both of them "on-line", get into BASIC from your DOS, with at least THREE files available. Then, just run "DOTGEM/BAS". It will tell you what it's doing, and will make some changes to "DOTEPS/CMD". When it's done, you should return to DOS (CMD"S") and RENAME DOTEPS/CMD to DOT/CMD. Then, test your results as described earlier. Please note that DOTWRITER will not work properly with the Gemini 10 or 15, only with the "X" series.

If you have an EPSON LQ-1500, use "DOTEPS/CMD" and "DOTLQ/BAS", and follow the instructions in the preceding paragraph. Be sure to use "DOTLQ/BAS", not "DOTGEM/BAS", of course.

If you have a C.ITOH 8600B, use "DOTITO/CMD" and "DOT8600B/BAS", and follow the instructions in the earlier paragraph. Be sure to use these names, not the names for the Epson or Gemini.

PRINT QUALITY GUIDELINES

Print quality is affected by paper, ribbons, and the printers themselves. A thin white line (streak) through a line of text is caused by slight paper mis-feeding. You usually can eliminate or minimize this by setting the "Friction/Tractor" lever on the printer to "Friction," but that may cause paper jams on some printers. The plastic gears on printers wear out with use, resulting in progressively worse paper alignment and movement as the years go by. There is no cure for the resulting random streaks except to replace the printer.

Streaks close to the top or bottom of the paper are due to the folds between the sheets: the springs on some paper bails (the bar that presses rollers against the paper) are too weak to keep the paper flat when the folds come through . Dotwriter has a command to print things darker (".DA ON"). This will improve print quality by overstriking everything, but it doubles the print time and increases wear and tear. An "almost" new ribbon is the best solution of all: a brand new one may leave smudges, and we all know the excuses for not replacing old, worn out ribbons.

CHAPTER TWO
DOTWRITER: An Overview

DOTWRITER is a graphics text formatter. That means it uses the dot graphics capabilities of certain printers to print text you've previously written with a regular Word Processor. This section of the manual will teach you how to create those text files. You'll also learn how the DOTWRITER function works, so that you can more easily adapt it to your needs. It is the program you renamed to "DOT/CMD" during installation, and when we say "DOTWRITER" here, we will always be referring to the file you renamed.

USING YOUR WORD PROCESSOR

DOTWRITER is the "printing" half of a word processor, and has no text "Editor" of it own. So, you must create a TEXT FILE with your Word Processing Editor BEFORE you run DOTWRITER. The text file must be saved in ASCII format, and most of the popular word processors can do this. These files must include the text formatting (layout) "dot" commands that are recognized by DOTWRITER. Most of the rest of this book will show you what those commands are and how to use them.

Here is a summary of what happens. First, you must create a document with your word processor, just as you normally would. This can be a letter, a book, a poster, a newsletter, or any text file you want to create. You will include special "dot" commands to tell DOTWRITER which lettersets you want to use and how things should be printed.

There are over 60 individual format commands for this. Each one starts with a period, and is a two-letter "mnemonic" that reminds you of an English word or term. For example, ".PP" stands for "PARAGRAPH". We will call these "dot commands" because they start with a dot (a period). Many of these commands can be followed by numbers or words that select something or turn a feature on or off. You can leave a space between the command and its "parameters", or omit the space; DOTWRITER will accept either method. In this book, we've generally omitted the spaces, since that saves keystrokes and is the way most people use the program. Usually, you can put more than one dot command on the same screen line, and may separate them with semi-colons: ".SK;.CEON" tells DOTWRITER to "skip" a line (leave a blank line) and then "center" the next lines of text. The period is always needed, since it's part of the control word. The semi-colon is optional (it was required in earlier versions).

Next, you save the file in ASCII format <u>by following the instructions included with the EDITOR or WORD PROCESSOR you are using.</u> Some programs use ASCII as their storage and others must be told to save the text file in ASCII. Be certain that your chosen editor or processor has the capability to save the file in ASCII format.

<u>NOTE:</u> If you are using "ALLWRITE" or "NEWSCRIPT" (these word processors are available from PROSOFT), files will always be in "ASCII" format. Also, you do not have to exit to DOS to run DOTWRITER: with ALLWRITE, just press Control Pound-Sign; with NEWSCRIPT, return to the PRIMARY MENU and select DOTWRITER.

After saving your text back to disk, exit from your word processor to DOS, type "DOT" to run DOTWRITER, and answer the questions it asks you. Usually, you can just press "ENTER" three times to bypass the first three questions. Reply to the fourth question by entering the name of the file you want printed (usually, the one you just created or updated). DOTWRITER will read through the file you specify, and print the text by following the formatting (dot) commands you placed in that file.

Text Files

DOTWRITER can print in many different formats. Not only can you mix and match the lettersets, but you can print multiple columns (with certain printers), center text, adjust left, right, top and bottom margins, include top and bottom titles, print with true proportion, control concatenation, fully justify text, integrate graphics with text, imbed frequently-used text (such as return address in letters), append text files to link numerous short files into one long file at print time - and much much more.

You do not have to type in all of the necessary format commands each time you create a file. The essential commands have default values and can be left out. Many simple documents, such as letters, often can be printed with just the default values. More complex text files and special effects will require the use of additional dot commands, and this is easily done. To make it really easy to set up your own defaults, we'll show you a short cut that lets you put your preferences into a STANDARD PRINT FILE. But for now, let's look at the actual creation of a file!

Contents of a DOTWRITER File

In a normal word processor, you have two kinds of controls: those that let you enter and modify the text (editing commands that move the cursor or text around); and those that tell the word processor how to print the text.

The most important thing to remember when making a DOTWRITER file is to for-get the formatting commands normally used by your Word Processor - only use DOTWRITER commands! DOTWRITER recognizes ONLY the commands in its command library (see the second section of this book). For example: if your word processor requires that you hit CONTROL plus LEFT ARROW to indicate a paragraph, you cannot use that sequence with DOTWRITER. The command for a paragraph in DOTWRITER is ".PP" and that is the ONLY "paragraph" command DOTWRITER will recognize. IF YOU ARE CREATING A FILE FOR DOTWRITER, USE ONLY DOTWRITER COMMANDS - REGARDLESS OF THE EDITOR YOU ARE USING!

DOT Commands

That brings us to the next point. DOTWRITER recognizes ONLY DOT commands. A DOT command is a command used by DOTWRITER, and each DOT command is preceded by a PERIOD or DOT. Dot commands can be mixed on line in most cases. Exceptions will be noted later. Also, the use of dot commands will become clear momentarily. (If you want to use a semi-colon instead of a period, for compatibility with ALLWRITE, please see the ".CW" command in The "LIBRARY" chapter later on.)

Regardless of the editor or word processor you use, DOTWRITER will format the text based on the DOT commands, not the screen display. For example, if the text screen displays the following lines:

The quick brown
fox jumps over
the
lazy
dog.

It will be printed as follows:

The quick brown fox jumps over the lazy dog.

Proportional Spacing

Line lengths on the screen seldom reflect line lengths on the printed page, because the screen only shows one size of "little" characters, while DOTWRITER can print many sizes of large, fancy characters. After a while, you will get used to letting DOTWRITER worry about fitting the text into nice lines for you, and you'll only need to tell it about new lines when setting up headings or bulletins.

When using proportional spacing, it's very hard to predict how many characters per inch (CPI) will be printed, because letters are placed varying distances apart, depending on how big they are. For example:

This line was printed with proportional OFF!
This line was printed with proportional ON!
And this line was printed in a bigger typeface!

Note the difference in the length of the lines and the spacing of the letters. In the first line, proportional spacing is off and the letters are monospaced. That means that the letter "i" takes up as much room in the line as the letter "M". In the second line, the "i" is printed in only that amount of space needed. Here is another example: the word "MAKE" is longer than the word "MIKE" because the "I" in MIKE is thinner than the "A" in MAKE. That is what proportional printing is all about!

Kerning

Version 4.0 has another command that can affect the spacing in a line or word. It is the KERNING command and, though it is explained in the command library, it's worth mentioning here as well. "KERNING" is the term used by typesetters to describe the overlapping of letters to save space and improve readability:

KERNING OFF
1) ToTal   2) LT
KERNING ON
1) ToTal   2) LT

Notice how the "o" in "ToTal" is tucked under the cross-bars of the "T"s in the second example, but not the first. When KERNING is off (.KROFF), spacing is based on the minimum width of each letter. That means that the "o" cannot start until the "T" is completely finished. When KERNING is on (.KRON), DOTWRITER will overlap letters in certain cases. The amount of kerning depends on the style and size of the letters in any given letterset, but it is completely automatic once the proper commands have been issued. Kerning improves the appearance of text, but slows down processing.

CHAPTER THREE
Creating A Text File

Now, let's get into the actual creation of a text file. First, start up your text editor. If you use NEWSCRIPT or ALLWRITE, you'll find they are very similar to DOTWRITER (since DOTWRITER is based on NEWSCRIPT). However, if you do not have either of those word processors, any word processor that can handle ASCII files will work just fine with DOTWRITER. (For more information on this, see Appendix "B".)

You should now have a blank screen in front of you (or whatever your editor displays as a beginning screen). In this example we will write a short letter, which normally would use mostly default values. However, to show you how DOTWRITER works, we will specify some of the commands. As you follow the examples in this book, if you don't understand what a particular dot command is doing, please look it up in the "Command Library" section of this book. We will summarize the commands here, but our real purpose is to help you understand how DOTWRITER works.

We sometimes refer to a DOTWRITER file as a "script", because it does the same things that the script for a stage play does: it contains not just the words, but also instructions for how those words should be presented. All DOT commands must begin with a period (unless you've changed this with the ".CW" command). The period must be immediately followed by two letters (the command itself). Many commands may be followed, or must be followed, by additional numbers or words, called "parameters". A parameter just makes the command more specific. In English, "WALK" is a command, but to make it more specific you might say "WALK FASTER".

DOT commands must go on screen lines of their own. Usually, there can be more than one dot command on a screen line, and we will note the exceptions in the "COMMAND LIBRARY" section. A command line must end with an "ENTER" (ASCII 13, Hex '0D'), however your Editor handles it. Also, the line just before each command line must end with an "ENTER". (If the first line of the file is a dot command, there would be no "ENTER" ahead of it, of course ... or it wouldn't be the first line.)

Text cannot go on the same lines as commands. Usually, text lines can be longer than screen size, and as we showed earlier, DOTWRITER can combine short lines into long ones under certain conditions. In the example we will show next, the command lines all ended with "ENTER", but the paragraphs of text only had "ENTER"s at the end of the paragraphs.

The next two pages show a DOTWRITER "script" as we typed it into our Editor, and then the final result as DOTWRITER printed the script:

```
.cm THIS IS A SAMPLE LETTER FOR THE DOTWRITER TUTORIAL
.PR ON .LM5 .LL 70 .TM6 .BM6 .SD3
.BF MB/PR
.AF TR/PR
.CE ON
John Smith
123 Main Street
West Town, NJ 07142
(291) 987-0000
January 1st, 1984
.CE OFF .SK .FO OFF
Bob Brown
456 Seventh Street
Los Angeles, CA 90000
.SK .FO ON
Dear Bob;
.PP
```

I was very happy to receive a copy of the !(NEW!) 4.0 version of DOTWRITER. It is by
far the best release yet. I am amazed at the speed and the additional commands.

```
.UP 2 .pp
```

As in the old versions !/the alternate letterset is easily turned on and off!? with con-
trol codes. It is even possible to substitute the alternate letterset for !/0!?ne letter
in a word. The !$Underlining feature is also just as good as in!% past versions and
!&the ability to control underlining has!% always been important to me.

```
.PP
```

Although the added features are great, the best feature by far is the greatly
increased speed. The printer never hesitates - it just prints on as quickly as possi-
ble. The increased speed has made it more practical for me to use the various letter-
sets you now offer. I used to try to stay with small lettersets because they printed
faster, but now I mix and match all 15 disks of lettersets. I hope you have many more
coming.

```
.PP
```

Thanks again for a great program and for so many great new features and lettersets.

```
.FO OFF .IN 45 .SK
Sincerely,
.SK 4
John Smith
.EN
```

John Smith
123 Main Street
West Town, NJ 07142
(291) 987-0000
January 1st, 1984

Bob Brown
456 Seventh Street
Los Angeles, CA 90000

Dear Bob;

I was very happy to receive a copy of the **NEW** 4.0 version of DOTWRITER. It is by far the best release yet. I am amazed at the speed and the additional commands.

As in the old versions the alternate letterset is easily turned on and off with control codes. It is even possible to substitute the alternate letterset for One letter in a word. The Underlining feature is also just as good as in past versions and the ability to control underlining has always been important to me.

Although the added features are great, the best feature by far is the greatly increased speed. The printer never hesitates – it just prints on as quickly as possible. The increased speed has made it more practical for me to use the various lettersets you now offer. I used to try to stay with small lettersets because they printed faster, but now I mix and match all 15 disks of lettersets. I hope you have many more coming.

Thanks again for a great program and for so many great new features and lettersets.

Sincerely,

John Smith

Explanation of Sample Letter

To relieve your anxiety, you probably won't use this many dot commands when you're first starting out. Some of them are defaults anyway, others would be in your standard setup file, and some are just there so we can show off a bit.

.CM stands for "COMMENT", and is a convenient way of making notes to yourself in the file. Comments are never printed, and must be the last thing on the line, since everything through the next "ENTER" will be ignored.

.PR ON means "PROPORTIONAL printing ON". This allows DOTWRITER to fit more on a line, and to eliminate unnecessary gaps between letters. You should use it most of the time, and turn it "OFF" only when setting up tables that require straight columns.

.LM 5 defines the "LEFT MARGIN" to be 5/10 of an inch wide. We put several commands on the same line to save space and to make it easier to distinguish between text and commands. Since the default left margin is "5", we could have left this one out if we weren't trying to show you how DOTWRITER works. By the way, ".LM5", with no space between the command and the parameter, is equally acceptable. We included the spaces for readability, but they're not required.

LL 70 sets the line length to 70/10 of an inch, which is seven inches. The right margin is whatever is left over. To print something where the sum of ".LM" plus ".LL" exceeds "80" (8 inches), see the ".WP" and ".MX" commands. "70" is the default, and you can omit ".LL70" if that's what you want to use.

.TM 6 sets the top margin to be six print lines. The top margin will contain the page number and top title (if you use one), and the rest of the space will be left blank. "6" is the default, by the way.

.BM 6 sets the bottom margin, etc.

.SD 3 sets the dot spacing between letters. It stands for "Spacing Definition". If omitted or set to zero, the letters will touch. This is desirable for certain fonts, but for normal text, you generally should make ".SD" equal to the number of printer lines used by the font. "2" or "3" are good choices for small fonts.

.BF MB/PR is important. ".BF" stands for "BEGIN FONT" or "BASE FONT", and is the way you select the typeface DOTWRITER will use to print your text. If you forget to specify ".BF", then the hardware character set of your printer will be used, and most of DOTWRITER's features will be suppressed. "MB/PR" is the name of one of the fonts on the standard disk we sent you. It's full name is "Medium Bold", but "MB/PR" is the file name, and that's what you have to use here. If you want to use a different font, just specify its name instead. You are not limited to using one font throughout a document, but can issue new ".BF" commands from line to line. ".BF" must be the last command on a line.

.AF TR/PR introduces a very nice, but somewhat advanced feature. It stands for "ALTERNATE FONT", and lets you select a <u>second</u> typeface to be used at the same time as the BASE font (".BF"). "TR/PR" is the alternate font we selected. The only thing to remember here is that the alternate font cannot be BIGGER than the BASE font; but it can be the same size or smaller.

.CE ON turns "CENTERING" ON. All text lines from now on will be centered within the current line length. Sooner or later, we will want to turn centering off, and you probably can guess how to do that. Dotwriter temporarily turns formatting off (.FO OFF) during centering.

The next five lines are text (finally). They will be printed in "MB/PR" typeface, and will be centered near the top of the page.

.CEOFF turns off centering. If formatting was on before centering began, it will be on again after centering is turned off.

.SK "SKIPS" one line on the printer. The size of the line depends on the font you are using and the current Line Height (".LH"), so larger fonts give you slightly bigger skips. If you want to skip 5 lines, you would tell Dotwriter: ".SK 5".

.FO OFF is important. It turns "FORMATTING" OFF. Normally, DOTWRITER formats text so that it can fill up each print line and print straight margins. (This is called "left and right justification.") That's fine for the body of a letter, but if you want to print the salutation of the letter, each line should be by itself. When you set ".FO OFF", you tell DOTWRITER to print each line "as-is" (except for the typeface and proportional spacing, that is). If you place too much text on a screen line when formatting is OFF, DOTWRITER will have to throw away what doesn't fit on the printed page. You will know about this quickly, because DOTWRITER also will tell you "LINE DOES NOT FIT" while it is discarding your wonderful words. Actually, what it's doing is catching an error, and showing you what <u>will</u> fit for your next try.

.SK leaves another blank line.

.FO ON turns formatting back on. From now on, DOTWRITER will shift text between screen lines as it fits words onto paper.

.PP starts a new paragraph. DOTWRITER leaves a blank line before the paragraph, and indents 5 spaces (by default) on the first line. If you want a different indentation, you can put the appropriate number after the ".PP".

.UP 2 stands for "UNDERLINE PIN", and lets you control the placement and thickness of underlining. There are seven or eight "pins" in the printhead of your printer (DMP 2100 and LQ 1500 printers adapt to this when controlled by DOTWRITER), and we number them in binary, from "1" to "64" or "128". ".UP 2" selects the

second pin from the bottom, and if you look at the printed letter, you'll see how it looks with the "MB/PR" typeface. If we used a much bigger font, we might have set ".UP 3", which would have made pins 1 and 2 fire together, resulting in a thicker line.

!(is not a dot command, but something stuck inside the text line that begins "I was very happy". We call this an "ESCAPE SEQUENCE", because that's what your printer manual calls this sort of thing. This particular escape sequence tells DOTWRITER to switch to double-width. Escape sequences are two characters in length, and by default, they begin with an exclamation mark (for pre-historical reasons we won't discuss here). The second character selects the specific feature. In this case, the left parenthesis, "(", starts double-width. To switch back to single-width, use the right-parenthesis, ")". (These conventions are compatible with ALLWRITE.) Their effect in this letter is to print "NEW" twice as wide as normal.

!/ is an escape sequence that selects the alternate font ("TR/PR", in this case). To return to the base font, use "!?".

!$ is the escape sequence that turns on underlining (to make use of that ".UP 2" we set a bit earlier). To turn underlining off, we use "!%".

.IN 45 stands for "INDENT", and it tells DOTWRITER to indent the left margin a specified number of tenths of an inch (45/10, or 4 and 1/2 inches, in this case). Indents are cumulative in DOTWRITER, so if we had an ".IN 10" later on, it would mean we've indented 5.5 inches. To get rid of all indentation, specify ".IN 0" (that is a zero).

.EN stands for "END". You don't have to use it most of the time, but if your word processor is in the habit of including some extra characters at the end of a file, this is the way to get past the problem.

     This completes the explanation of the sample letter. We covered several dot commands:

     CM   PR   LM   LL   TM   BM   BF   AF   CE   SD   SK   FO   PP   UP   IN   EN

and a couple of Escape Sequences:

         !(!)   !/   !?   !$!%

Now, there are about 50 more commands available if you need them, and The "COMMAND LIBRARY" chapter explains all of them in alphabetical order. All DOT commands are based on English words, so once you learn them, they should be easy to remember, and usually pretty obvious. As you gain experience in using them, you will discover how to produce special effects that can be eye-catching, startling, and, hopefully, pleasing.

The best way to gain this experience, of course, is by practice: make up some signs, write some simple letters, and remember that you only have to keep the good results.

CHAPTER FOUR
<u>Tricks Of The Trade</u>

This chapter will cover some DOTWRITER capabilities that may not be apparent from the descriptions of the individual commands. When you encounter seemingly inexplicable problems in fancy formatting techniques, come back here and see whether there's an answer.

Creating a DOTWRITER "script" is really pretty easy, once you get the hang of it. However, if you have looked at the command library, you know that we took some short cuts. For example, the letter we just wrote DID NOT specify the height of the space between lines or the minimum width of the space between words. It also didn't use too many lettersets, multiple columns, vertical tabbing, table of contents, index, imbedded or appended files and much more! And, although it specified the margins, it didn't have to, since they matched the defaults.

Once you have learned how to use the DOTWRITER program and how to create a file that will print properly, the rest is a simple matter of adding new commands to your DOTWRITER vocabulary. In fact, many of the features we have developed for DOTWRITER are the result of experimenting with various commands already included. We would discover that a particular group of commands created a special effect and we would then invent a single command to improve the use of the effect.

Experimentation is the name of the game. If you want to know how a command works, read the library and then try the command. You'll be amazed at how easily the commands fit into your current files. A simple letter like our example, can be made into a real class piece of work with only a little imagination.

Now, we won't drop you here with nothing more than the single example. We will show you some of the tricks we have developed (intentionally or accidentally) to create special effects in the printout. Be sure that you try these tricks yourself because by testing them, you'll find out just how easy it is to implement your own!

The Imbedded File

This is just a short note to cut down on your typing time. DOTWRITER has a wonderful command called IMbed. It allows you to indicate a place in a text file where DOTWRITER should momentarily stop, <u>READ A DIFFERENT FILE FROM DISK, PRINT IT,</u> and then return to the original file and continue printing.

Your most common use of ".IM" will be to make use of your own defaults file. We will show how to do this a couple of pages later on. Another common use of ".IM" occurs when writing a return address or a letter closing. Suppose that you always use the same letter head on all of your correspondence. Do you have to type it every time you write a letter? No, just type it once, save it to disk under a file name you can remember, then use the IM command to call it up any time you want it printed in the letter. The closing of the letter - Sincerely, etc, can be done in the same fashion.

The only limitation on IM files is that one IM file cannot contain a reference to another IM file. A regular file can call as many IM files as necessary though. In fact, if you write several sections of text that you use a great deal, it is possible that there will be times that your main file is simply a group of IM files. It might look something like this:

```
.IM start
.IM Part1
.IM Part2
.IM ENDFILE
```

The Appended File

The APpend command (AP) is similar to the IM file with one main difference. APpend tacks one file on the end of another, rather that bringing it into the middle as does the IM command. It lets you write documents of unlimited length, one small piece connecting to another. This book, for example, is composed of about 15 fairly small files. Keeping them small makes it quicker and safer to edit them, while still letting us print either a few pages at a time, or the whole book at once.

For this book, the first file is called "DOT1". The second is called "DOT1A", and the third is "DOT2". As the names suggest, we had to stick a lot of extra text into the middle of the book, and with APPEND, it was very easy to do. There are more files in the sequence, but you get the idea of what we did.

When editing, we only had to load and edit a small section, but at print time, the entire manual was printed in one piece. Another hint that will help is that when we printed the manual for the first time, we preceed each ".AP" command with a stop command (.ST). This stops the printing at the end of the file to let us check the section just printed for errors before continuing. After completing the check, ENTER lets printing continue as if it never stopped! While it is always best to end a file at a paragraph or chapter end, it isn't necessary. A file can end in the middle of a paragraph as easily as anywhere else and when the next file in line begins, the paragraph will continue as if uninterrupted.

If you already have a large file and you want to convert it to a DOTWRITER file, just add the necessary commands and APpend commands. Then save the file back to disk in blocks with each block ending at an AP command. The ability to split up a file depends on your editor - it can only be done if your editor allows saving blocks of text under different filenames. Most of the good editors allow this function.

The Standard Print File

DOTWRITER will let you control printing in an incredible variety of ways. You can specify margins, the number of columns, the space between letters and lines and even words, the type style, top and bottom titles, proportion, density of print, format, magnification, and much more. This versatility carries with it a small problem - keeping track of what commands you have specified. Also, it can become tedious to have to type all of the commands every time you want to create a file, and to remember them all from one letter to the next.

There are several ways around these problems but the easiest is the STANDARD PRINT file. Create a file that has all of the necessary commands set to your requirements. Call it by a name that you can remember easily - for example, if you are creating a file for a letter, call it LETTER/FMT for letter format. Be sure that the file contains such things as DAOFF, CEOFF, INO etc. It should also contain top and bottom titles, margins, line heights and spaces between letters, the name of your primary letterset and the like. We haven't covered some of these features in the tutorial, but as you discover them in the LIBRARY and then decide on your preferences, you can add them to your standard setups.

Once created, it is easy to IMbed this file in the very beginning of every letter. This way, you need type it only once. And, if you are typing and forget what changes you may have made, simply IMbed the file again and all of the commands will reset to the STANDARD FILE parameters which you already know. I would never be able to type out and print an entire manual if I didn't use this method. In fact, here is the file we used as standard print for this manual:

```
.DAO;.EMOFF;.MFI;.PI3;.PRon;.REO;.SD3;.SW5;.PNon;.Ceoff;.LM5
.LL70;.INO;.COoff;.JUALL;.UPI;.ESI;.HY£ON
.PS$;.WPOFF;.L5;.TM6;.HM2;.BM6;.FM2
.KRON;.KS1
.TFnost2/pr
.TR 35,35,92
.TT # Dot#Writer#4.0 Page#$
.BT # Copyright#1984#by#W.K.#Mason #
.coon .VT6
.TRO
.FOON;.UP32;.ceoff;.PL66;.CBl;.LH7;.DSOFF;.VLOFF;.JUON
```

Most of these commands are here as a precaution: just in case the preceding file in an "APPEND" chain turned something on, the standard file will make sure it's back off. You may find the following a perfectly satisfactory defaults file:

```
.LM7.LL65.PRON.KRON.KS1.SW3.UP1.SD2
```

Copyright 1986 by PROSOFT and W.K. Mason

That sets Left Margin, Line Length, Proportional printing, Kerning, Space Width, the Underline Pin, and inter-character micro-spacing.

We suggest you try using this method, and that you eventually establish several defaults files: one for letters, one for posters, etc. You'll find that they can save you a lot of time and effort. Even though most of the commands have default values, you find that the more you use DOTWRITER, the more you'll want to manipulate text. The more special features you use, the more you'll appreciate the STANDARD PRINT FILE!

Top And Bottom Titles

If you're making posters, banners, or one page letters, you won't have much use for page titles. However, to make longer letters or manuscripts more attractive, you will want to take advantage of DOTWRITER's capabilities in this area. Once you do so, you will find situations in which you will want a top title to be centered or right justified, rather than left-justified. Or, you may want the title centered but the page number flush right! Note the top title on this page and throughout the manual. The title is centered and the page number is flush right. The bottom title is centered. Also, both top and bottom titles are in a different letterset than the body of the manual! We will now show you how to manipulate the titles.

The first thing to remember - and the most important thing - is that THE COMMANDS IN EFFECT WHEN THE TITLE FONT IS SPECIFIED WILL AFFECT THE TOP AND BOTTOM TITLES EVERY TIME THEY ARE PRINTED - EVEN IF THEY ARE TURNED OFF IN THE THE BODY OF THE FILE! That means that by turning CEON BEFORE the ".TF" (Title Font) command is specified, you will have centered titles even when the TF command is followed by the CEOFF command.

Here is an important note - If you want to have the top and bottom title appear on the first page, they must be set before ANY text is printed. This is best done with a standard print file similar to the one I used in the last section. If you want the top title to start on the second page, specify it AFTER some other text has been printed. By default, titles are not used on the first page, because they don't belong on the first page of a letter, or the only page of a poster. Titles only belong on subsequent pages, most of the time.

That is all you need to know in order to select the functions needed for the top titles. Now, let's re-examine the specifications used to print this book:

1) .DA0;.EMOFF;.MF1;.PI3;.PRon;.RED;.SD3
2) .SW5;.PNon;.Ceoff;.LM5
3) .LL70;.IN0;.COoff;.JUALL;.UP1;.ES!;.HY£ON
4) .PS$;.WPOFF;.LS;.TM6;.HM2;.BM6;.FM2
5) .KRON;.KS1
6) .TFnost2/pr
7) .TR 35,35,92
8) .TT # Dot#Writer#4.0 Page#$
9) .BT # Copyright#1984#by#W.K.#Mason #
10).coon .VT6
11).TR0
12).F0ON;.UP32;.ceoff;.PL66;.CB1;.LH7;.DS0FF;.VLOFF;.JUON

The first thing to note is where the Title Font is selected. Look at line number 6. The command is TF NOST2/PR. This selects the title font as Engraved Nostalgia 2 line proportional. As I mentioned above, ALL OF THE COMMANDS THAT APPEAR BEFORE THE TF COMMAND WILL BE IN EFFECT EVERY TIME A TOP OR BOTTOM TITLE IS PRINTED.

If you look at lines 10 through 12, you'll note that I have reset many of the commands used BEFORE the TF command. Example - in line 3, the concatenation is turned off (COOFF) and in line 10, it is turned back on (COON). Though I want the concatenation off for the title (in order to put the page number flush right) I want it on for the body of the text. Note also that I have centering turned off (CEOFF) in line 2 and off again in line 12. This is because I don't want to use the centering in the title (I'll center with concatenation - which I'll explain momentarily) and I also don't want the centering on for the body of the text.

I could have used only one CEOFF command, but this STANDARD PRINT FILE was edited from another file where the centering was used. Also, if I wanted to edit this one to use for another application, I might want to turn on centering before the TF command and it is now there to remind me. It costs only a split second to use the double CEOFF command so I'll always use it in this fashion.

CHARACTER TRANSLATION and CHARACTER SPACING

How did I get the title centered and the page flush right? That is with the use of two commands that work well together. The first is the TRanslate command used to create a HARD SPACE. Most of the lettersets we sell have a hard space in ASCII 127 location in the letterset. If you're using NEWSCRIPT, you can use SHIFT-CLEAR-equals sign to enter a hard space: if you're using ALLWRITE, you can do it with CLEAR-equals sign: and if you're using any other Word Processor, you can tell DOTWRITER to translate another character to a hard space. Here is how it is done.

Character Translation

First look at the TR command. It is easy to make any keyboard character into the ASCII 127. Just TRanslate a known value - The # character is ASCII 35, the difference between 35 and 127 is 92 so all we have to do is add 92 to 35. TR 35,35,92 means - take every ASCII value between 35 and 35 and add 92 to it - then print that value instead of the original 35. The 35 values indicate one character but they could easily specify a range. If you wanted to change upper case to lower case or lower case to upper you could simply specify the range of ASCII characters like this - TR 65,90,30 or TR 90,65,-30. In the first case we have taken the range from uppercase A to uppercase Z and added 30 to each. When the uppercase A is found, it will be printed as 95 instead of 65. 95 is a lowercase a. This will continue for each character from ASCII 65 to ASCII 90, effectively changing upper case to lower case.

What is the significance of all this? The HARD SPACE is a special form of the blank character: it prints as empty space. However, if a blank falls at the end of a print line, it is discarded and the next word prints on the next line. By contrast, the HARD SPACE is treated like a visible character, and prevents the two words from being printed on separate lines. That's useful when you don't want "Mr." and "Smith" to be on two different lines.

Size of a Blank Character

Spacing between characters is controlled by several things:

* width of a blank (".SW")
* presence of a "hard space" (ASCII 127; plus-minus on Model 3/4)
* inter-character micro-spacing (".SD")
* kerning (".KR ON") and kerning space (".KS 1")
* proportional spacing (".PR ON" or ".PR OFF")

Spacing between lines of text is controlled by three things:

* the number of vertical dots in the font
* the line height (".LH")
* intentional blank lines (".SK" and ".PP")

How much space you want depends on the size of the letters, whether they are supposed to be connected (banner fonts), and the general effect you're trying to achieve. That's why there are so many controls available: as soon as you get past the "novice" stage with DOTWRITER, you will want to get fancy and produce novel, beautiful works of art. And these features will let you do it.

If you print something with the hardware characters of your printer, the blanks between the words will be the same size as the letters. If your printer has a "proportional" character set, the blank will be narrower than most letters, and that is also true in magazines and regular books. By default, DOTWRITER will print a blank (or hard space) at half the width of the widest character in the current font. Often, that is too wide, and you can change it with the the SW command. ".SW" is followed by a number from 1 to 9, and indicates the relative size of a space. ".SW 3" will make the spaces between words 30% the size of the widest character.

Having gone this far in the discussion, we had better point out that "Kerning" (".KRON") can partially override ".SW". DOTWRITER does its kerning after deciding what will be printed on the line. During kerning, excess micro-spaces between letters are removed. If left and right justification is in effect (".FO ON"), then, like any word processor, DOTWRITER will have to add some space back in, someplace, to get that right margin where it belongs. And, of course, it adds extra spaces between the words, thereby overriding all the good intentions of ".SW".

CONCATENATION and JUSTIFICATION

"Concatenation" is a fancy word for "connecting". It just means that a word processor can glue two screen lines together, and then chop the result back up in a different place when it prints the words. Normally CONCATENATION works with JUSTIFICATION. JUSTIFICATION is the process that makes words fit the entire line so that the last word ends exactly at the right margin. This allows for flush margins on the right side, as you see in this manual. CONCATENATION is the process that puts as many words on a justified line as possible. The two together cause a full, justified line. If justification is ON and concatenation is OFF, the lines will still be flush with the right margin, but there will be no more words on the printed line than there are on the computer line. Here is an example of text that is printed first with ".JU ON:.CO ON", then with ".JU ON:.CO OFF", and finally with ".JU ALL:.CO OFF":

Fourscore and seven years ago, our fathers brought forth upon this continent, a new nation conceived in liberty and dedicated to the proposition that all men are created equal.

Fourscore and seven years ago, our fathers
brought forth upon this continent, a new
nation conceived in liberty and dedicated to the
proposition that all men are created equal.

Fourscore	and	seven	years	ago,	our		fathers
brought	forth	upon	this	continent,		a	new
nation	conceived	in	liberty	and	dedicated	to	the
proposition	that	all	men	are	created		equal.

Note that in the first sample, the words are packed on the line as closely as possible, but in the second, words are not carried up from the bottom lines to fill the top lines. That is how concatenation works. It tries to put as many words on a line as possible. In the third example, words weren't moved up, but they were spread out by ".JU ALL". Please note that, when concatenation is ON, ".JU ON" and ".JU ALL" do the same thing. (This is a little different from ALLWRITE.)

OK, now we will see how to use that feature to print a page number flush-right. Look at these lines from the previous STANDARD PRINT FILE example:

```
1)  .DA0;.EMOFF;.MF1;.PI3;.PRon;.RE0;.SD3
2)  .SW5;.PNon;.Ceoff;.LM5
3)  .LL70;.IN0;.COoff;.JUALL;.UP1;.ES!;.HY£ON
4)  .PS$;.WPOFF;.LS;.TM6;.HM2;.BM6;.FM2
5)  .KRON;.KS1
6)  .TFnost2/pr
7)  .TR 35,35,92
8)  .TT # Dot#Writer#4.0 Page#$
9)  .BT # Copyright#1984#by#W.K.#Mason #
10).coon .VT6
11).TR0
12).FOON;.UP32;.ceoff;.PL66;.CB1;.LH7;.DSOFF;.VLOFF;.JUON
```

Now let's see how the technique is used. First, look at line 3 where concatenation is turned off (COOFF). This means that the text will be justified, but it will be justified with only what is in each line. Lines will not be added together to make a fuller line. In line 7, we translated the # character to a hardspace. It will look like a space but will actually be a blank and will be counted as a character. (If your word processor lets you enter an ASCII 127 directly onto the screen, as ALLWRITE and NEWSCRIPT do, then you don't need to use translation: just put hard spaces right into the text.) Finally, line 7 uses these features to create a centered title and flush right page number. Let's look at line 8 to see how this was done.

Actually, we have broken the page into 3 columns. The first column (left) is only a hard space character. Though the program will act as if something was printed, it will actually print nothing. The second column is the title "Dot Writer 4.0". Note that I have used hard space characters between the words in line 8 so that the phrase won't split up. Finally, the third column is the page number. This is comprised of the word Page and the symbol for the page number (see the PS command). I have joined these two words with a hardspace so that they will be in the same column but will be flush right in the third column. Note that the normal space character is used as the separation point for the columns.

Look at the top of any page in this manual to see how this works. This is the actual command format I used to create this manual and if you copy it, imbed it into your own file, it will work for you also.

Let's take a look at line 9. This is the bottom title, and if you look at the bottom of this page you will see how it worked. The page is again broken into 3 columns. The first is blank, the second is the phrase copyright notice and the third is blank. The result is that the bottom title is centered. Of course, we could just turn the centering on BEFORE the TF is selected, but that would center the entire top and bottom title and we want the page number in the top title moved into flush right.

This same technique can be used to create any number of effects. Also, though I used 3 columns, you could conceivably use more. Four columns would work, but you would not have a centered column - five columns would work and would be centered. The only limitation is the length of the line. You must make sure that your columns will fit on the page. You could use 80 columns as long as no more than 1 character was put in a column. Also, a hardspace is a character.

This technique can also be used to create columns in text. It is the closest you can get to tabbed data with DOTWRITER. Remember, the columns are printed from the right margin to the left. This means that the left edge of a column will not be justified.

In order to use this technique, the format must be on (F00N), the justification must be on (.JUALL) and the concatenation must be off (.C00FF). Also, commands such as centering (.CEON or .CEW) will interfere with the normal operation of the other commands because they automatically turn format off! If you try this technique EXACTLY as demonstrated above, it will work. You can run into trouble though, if you use JUON and JUALL together. If you use either one alone, all will work OK. If you use both, you may find that the technique stops working - if that happens just use the command JUALL!

SIDEWAYS BANNERS

Most fonts are meant to be printed across the page (like the text you are reading now). However, fonts with the extension, "/PRB" are "banner" fonts, and they print sideways: the bottoms of the letters face to the left. This means you can print banners that are several feet long. This only works with "/PRB" fonts that have been "rotated" in advance for the purpose.

Here are some considerations for printing banners:

1. Turn off the margins: .TM0.HM0.BM0.FM0.PN0FF.LM0.LL80.PL200
2. Type only ONE letter per line (remember, they go sideways)
3. If the letters are too squashed, use double-width or a lower pitch
4. Adjust the Line Height (".LH") to space or connect the letters
5. Use Magnification (".MF") to enlarge the letters
6. Use Wide Paper (".WP") if it applies to your printer and banner.

The normal spacing controls in DOTWRITER are intended for regular printing, so you will want to ignore most of them for banners. Line Height is the major exception to this: you will want to use it.

You can use LDS to rotate any normal font into banner format. This takes some time, but will save you the cost of buying the "/PRB" fonts.

CHAPTER FIVE
RUNNING DOTWRITER

So far, we've mostly just shown you how to create a file for DOTWRITER. We haven't covered all the commands you can use, but the next section of the book, "LIBRARY" will do that in detail. We've also given you hasty instructions for getting DOTWRITER to print for you, but really didn't explain how to operate it. Since this tutorial section is drawing to a close, it's high time we showed you how to actually run the program!

First, you must have RENAMEd the DOTxxx/CMD file for your printer so that its name is DOT/CMD. Then, you must write something with your word processor and save it to disk in ASCII format. (ALLWRITE! and NEWSCRIPT always do that.) Then, you must issue the DOS command, "DOT" to get DOTWRITER running. If you're using ALLWRITE!, you can just press Soft Key Capital 3, since the two systems are fully integrated.

Regardless of how you get it going, DOT will display a logo and prompt you for information. The first prompt asks for the current page number. If you are printing a page with a number OTHER THAN 1, enter the number. The first page will be printed with that number. EVEN IF THE PAGE NUMBER IS NOT PRINTED, IT WILL BE USED TO COUNT THE PAGE SO THAT, IF PAGE NUMBERING STARTS ON THE THIRD PAGE, IT WILL BE IN PROPER SEQUENCE. If you are starting at page 1, just hit ENTER. (That's what you normally will do.)

If you are using single ("cut") sheets and want to stop after each page so you can change paper; or if you are using a printer that isn't too good at reverse feeding the paper when you want to print multiple columns, the next prompt is for you. It asks you if you want to stop at the end of each PAGE OR COLUMN. Obviously, if you are feeding single sheets, you'll have to answer yes to this prompt. If you are printing multiple columns, we advise you to answer "yes" because you'll get the best column line up if you keep a little tension on the paper as it backfeeds for the next column. This is optional, though. The DEFAULT is NO STOP, so if your're using continuous form paper, just hit ENTER. (Usually, you will just press ENTER.)

If you want more than one printed copy at a time, answer the next prompt with the number of copies you want, and the print will repeat until that number has been printed. If you hit ENTER, you'll get just one copy.

NOTE: We will explain under "RUN TIME OPTIONS" how to cancel printing if you don't like the results. However, only the current copy is cancelled, so if you specified "5" copies, the next one will start immediately. You probably won't ask for multiple copies until you're sure they will be O.K., but if you do get into this situation, we suggest you hit the "RESET" key on the computer and then start over.

The fourth, final, and most important question prompts you for the name of the file you want to print. Just type the name and hit ENTER - the rest is automatic! (If you selected DOTWRITER from ALLWRITE, the name of your current file will appear on the DOTWRITER screen as the default file. You can accept it by just pressing "ENTER", or type another name.)

Version 4.0 has two new features that are useful at run-time. You can use the ".PG" command (see Section 2) to specify a starting page in the middle of your document, and DOTWRITER will format your text, but <u>NOT PRINT IT</u>, until it reaches that page. This can save you several hours if you need to re-do page 25 or pages 37-42 of a big document. It's a way of really taking advantage of the "machine-language" speed of this version. If you want to use ".PG" only at run-time, see "RUN-TIME" options, below.

The other useful feature is that, if a font, imbedded file, or appended file isn't found, you'll be prompted to type its name correctly. You can do that, or you can insert the correct disk (without removing the one being read for the primary file, of course) and just press "ENTER" to tell DOTWRITER to try to find the same file again.

VERY IMPORTANT

With some printers (the C.Itoh Prowriter for example) the very last line is not printed until the TOP OF FORM prompt is answered. It doesn't matter how you answer it, just be sure that you answer it before you move the paper. If you are printing top and bottom titles, the last line will fully print BEFORE you answer the TOP OF FORM prompt. Otherwise - answer it!

After you answer the TOP OF FORM prompt, the end menu will be displayed. For many users, the only options usable will be PRINT ANOTHER FILE and END. The PRINT ANOTHER FILE option will allow you to enter another file or DEFAULT to the last file you printed. The END option returns you to DOS. The remaining options are self-explanatory to those who can use them.

If you are using ALLWRITE's Editor, we've simplified a lot of this for you: you can use a soft key to start DOT/CMD, and other soft keys to define paragraphs, pages, underlining, etc. Also, since the filename is carried back and forth between ALLWRITE and DOTWRITER, you won't have to type it in over and over again. Finally, of course, the control words used in ALLWRITE (or NEWSCRIPT) are very similar to those used in DOTWRITER. If you want the two Formatters to be even more compatible, put a ".CW;" command at the beginning of your DOTWRITER files, and use the semi-colon instead of the period. (Or put a ";CW." in your ALLWRITE files.)

RUN-TIME OPTIONS

Run-time options are Dot commands you enter from the keyboard when DOTWRITER is just about to start printing. ".PG" is a good example of this, since you might want to select a page range for a specific print run, but wouldn't want to select those pages every time.

To enter a run-time option, hold down the BREAK key right after you type in the file name. When asked to enter a command, type a line of dot commands, each starting with the period. Then, press "ENTER" and DOTWRITER will continue running. This method is the same as "Stopping DOTWRITER", below.

STOPPING DOTWRITER

Suppose you start printing a file and then realize that you've made a mistake. You want to stop printing. What to do? Hold down BREAK until printing stops and the screen displays:

ENTER COMMAND FROM KEYBOARD.

Release BREAK immediately. (On a Model I or III, you can hold down the SHIFT key instead of BREAK, if you prefer to do so.)

Once the prompt appears, YOU MAY ENTER <u>ANY</u> VALID COMMAND, as we explained under "RUN-TIME OPTIONS". And, one of DOTWRITER's commands is ".EN" ("END"). If you type it, and press "ENTER", DOTWRITER will immediately stop processing text from disk. If a bottom title has been defined, the title will print at the bottom of the page, and then DOTWRITER will go to its termination menu. If there are no bottom titles, DOTWRITER will just stop and show you the termination menu.

CORRECTING MINOR ERRORS

Suppose you turned on the EM command but forgot to turn it off. Just hold the BREAK (any Model) or SHIFT (Models I and III only) key, and when prompted, enter ".EMOFF". The program will continue to print but with the EM turned off. The same is true for all DOTWRITER commands or combinations. ANY LEGAL LINE COMMAND can be issued in this fashion: just be sure to include the dot along with the command(s).

If you stop this way, by accident or on purpose, but want printing to continue without a break, just type ".CM" and press "ENTER". If you haven't moved the paper the printing will continue as if it had never stopped.

END OF TUTORIAL

That's about all there is to the mechanics of preparing text for DOTWRITER and then getting it printed. The next section of this manual contains the COMMAND LIBRARY, and it explains every feature of the DOTWRITER program. It's the part of the book you'll use again and again, when you need to find out how to do something. We hope you've begun to understand the way to prepare a DOTWRITER script, and again, we recommend "practice" and "experimentation." The features of the program make it endlessly interesting, and will let you produce new and different effects, limited only by your own knowledge and creativity.

SECTION TWO
COMMAND LIBRARY

This section describes all commands supported by DOTWRITER 4.0 ("DOT/CMD"). Not all of these commands will work on all printers, because some printers do not have the hardware features needed to implement the commands. The following key will help determine if a particular command is applicable to your printer. These letters appear at the right-hand end of each command's definition:

> No code - Applies to all printers
> C - Applies to C.Itch printers
> D - Applies to DMP printers
> E - Applies to Epson printers
> F - Applies to Epson FX printers
> O - Applies to Okidata Printers

The following codes are also displayed for the commands. These codes indicate the features as noted in the following key.

> % - command will not cause a control break (force a new line).
> G - command applies only to DOTWRITER fonts
> S - command works only with printer fonts

Please read each command carefully so that you are sure you understand its use. Experimentation leads to the best results.

DOTWRITER recognizes two kinds of commands: "dot commands", which begin with a period (a dot) and go on lines of their own; and "escape sequences" (called "emphasis marks" in ALLWRITE), which begin with an exclamation mark "!" and are mixed in with the text. Escape sequences are used to control underlining, boldface, double-width, and the alternate font. They are listed under the ".ES" dot command a bit later in this section, and are identical to the ones used in NEWSCRIPT and ALLWRITE. Please note that DOTWRITER doesn't support all the escape sequences of those other two word processors.

.AD n . . . . . ADJUST LEFT MARGIN (obsolete command -- use .LM)

.AF "name" . . . . . ALTERNATE FONT SELECTION   G

This command selects the letterset "name" as the alternate letterset. The regular letterset is selected with the .BF command. The alternate letters may be inserted among the regular letters with the escape code "!/" to begin printing with the alternate letterset, and the escape code "!?" to return to printing with the regular letterset. The alternate letterset must have a frame size less than or equal to the frame size of the regular letterset (also see ".PR"). For example:

```
.bf ant/pr
.af mes/pr
testing !/ testing !? testing
```

will print like this:

TESTING   testing   **TESTING**

If the alternate letterset is larger than the primary letterset, an error will occur and the Alternate Font will default to the primary letterset. Lettersets that do not fit together in one way can be reversed: see "Note" below.

This command may also be used to specify a letterset for subscripting, or to lower a letterset. For example, if the main letterset is PL and the alternate letterset is selected as .AFPL -2 (that is a dash just before the "2"), the alternate PL will be below the PL line. We've used the same letterset as both the primary and the alternate to do this. All alternate fonts can be selected in this fashion. Experiment to get the subscripting you like best. Note that the number following the AF font name is the number of printer lines that the letterset will be lowered.

You can also use this command to move a small alternate font down so that it lines up with the baseline of the primary font. By specifying ".af mes/pr-2", in the above example, we get the result shown below. If you move an alternate font down too far, the bottom of it will be "clipped" and disappear.

TESTING   testing   **TESTING**

<u>Notes:</u> ".AF" must be the last command on a line. If an ALTERNATE font is too large to fit with a MAIN font, they can still be used together by selecting the larger one with ".BF" and the smaller one with ".AF". Then all you have to do is switch the selection process so that you print most of your text in the alternate font and switch to the main font for the special characters.

.AP "filename" . . . . . APPEND FILE

Append is used to chain text files together. When this command is encountered, the file specified by "filename" will be printed. Control of printing is also passed to this file. Note: This command must be the last command on the last line of a file.

.BF "filename" . . . . . BEGIN FONT (use "filename") G
.BF 1 . . . . . USE PRINTER'S STANDARD CHARACTERS S

This command is used to change from one font to another. The character font specified by "filename" is used. The filename should be one of the supplied fonts or a file created with TGEAP (part of LDS). Your own character sets, or high resolution blocks may be printed using this command. When .BF 1 is issued, the printer's hardware font is used.

<u>Notes:</u> When used, this must be the last command on a line. Also, with the new version, it is possible to use proportional print in very large lettersets or in alternate lettersets (see ".AF" and ".PR"). In short, a letterset which fits into memory will always be proportional if desired. Alternate or large lettersets however, do not fit in memory so the proportional information must be available on the disk. Therefore, homemade lettersets may be proportional if they are the primary letterset and are not too large in size. If you try to use a homemade letterset as an alternate or, if your homemade letterset is very large, it will not be in proportional. If you want to proportionalize your own lettersets, please use our Letterset Design System (LDS), which is available through your dealer or from PROSOFT. Also note, the .BF1 font is not provided for normal processing but rather, is added as a default in error or testing conditions. It cannot be used for proportional printing, only 10 characters per inch.

Examples: .BF OE/PR
.BF MYFILE

The first example would specify that all following text be printed using the Old English Character Font. "OE" is the filename for the Old English Character Set. The second example shows how you could use a font you created with TGEAP.

.BM n . . . . . BOTTOM MARGIN

This command allows you to specify the number of lines you wish to have for your bottom margin. The default value is ".BM 6".

.BR . . . . . BREAK

The .BR command causes a control break. A control break causes text to be printed starting on the next line. Some commands do not cause control breaks. For example:

        This line will be printed as
        .tr 35,35,92
        one continuous line.

    This line will be printed as one continuous line.

Because .tr does not cause a control break. But:

        This line will be
        .br
        broken up because of ".br"

    This line will be
    broken up because of ".br"

Any unrecognized command is treated as a BReak command by DOTWRITER, and no error message will be given.

.BT "title" . . . . . BOTTOM TITLE

This command specifies the title to be printed at the bottom of each page. If this command is not used there will be no bottom title. Otherwise this command works just like the ".TT" command explained later. If you include a dollar sign "$" in a title, it will be replaced by the current page number. Please note that titles are done differently in DOTWRITER than in NEWSCRIPT or ALLWRITE.

                Example: .BT- Page $ -

This example would number the bottom of each page.

Note: This command must be the last command on a line. Also see ".PR".

.CB n,w . . . . . . .COLUMN COMMAND C,D,F

This command allows printers with reverse-feed capabilities to print in multiple columns. (MX-80's cannot do this; FX-80's can; most DMP's and C.ITOH's can, too.) The format of the command is .CB n,w where n = the number of columns and w = optional line length of each column. If you specify .CB 3 for example, the program will automatically separate the line length into three equal columns with 2 spaces between columns. If you specify w, (.CB 3,35) the line length of each column will be equal to w (i.e., w=35 would be 3.5 inches wide). If you specify the w value, be sure that you have enough room on the paper for the columns and the spaces between columns.

Here is how the .CB command works. Suppose that you specify a line length of 75 and a left margin of 5. You will have a page of 7.5 inches wide with a right and left margin of 1/2 inch. The .CB 3 command will then divide the line length of 7.5 inches into three equal columns with 2 spaces between the columns. The first column will print and when the end of the page is reached, the printer will back up to the top of the page (or previously specified location) and will print the next column. This will continue until 3 columns have been printed and then the next page will be started in the same sequence.

If you specify a .VT location, the .CB command will back up to the location of .VT on the FIRST PAGE ONLY. Subsequent pages will be printed with the tops of the columns at the top margin. If no .VT is specified, the first page will also start and return to the top margin.

If you want full-page multiple-columns, you must be sure to keep ".TM" and ".VT" matched. So, if you set ".TM8", you must also set ".VT8" at the same time, or the second column won't start at the right place on the first page. On subsequent pages, DOTWRITER will be able to keep things lined up for you.

Note: Because of paper movement and slight variations in how the print head positions itself, the lines in each column may not be exactly even on some printers. However, even without perfectly lined columns, I'm sure you'll like the look!

Note: On some printers, the paper tends to jam when fed backwards. If this happens, you can "assist" the printer by pulling the paper from behind when it's moving backwards.

.CB-n,w . . . . . . COLUMN COMMAND WITH BORDERS C,D,F

This is another version of the .CB command. It works exactly as the .CB n,w except that the -n signifies that you want vertical lines dividing the columns. Vertical lines will be drawn BETWEEN columns only, not on the left or right margins. You can also issue a .HL command to draw horizontal borders between the vertical lines.

Also see: ".HL" and ".VL".

.CC n . . . . CONDITIONAL COLUMN C,D,F

The .CC n command is exactly like the .CP n command except that it applies to columns. When the computer hits .CC n, it counts the number of lines left on the page. If there are not at least n lines left, it signals the printer to start the next column or move to the next page if the last column has been printed. The value n can be any positive integer.

.CE on,off . . . . . CENTERING LINES OF TEXT (see also CE W)

This command 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 a ".CE ON" command will be centered until a ".CE OFF" command is encountered. Note that ".FO ON" will be temporarily turned to off.

.CE W (formerly CW). . . CENTERING using paper WIDTH . . G

This command is similar to the ".CE ON" command except that the center of the paper is used instead of the center of the line. .CEOFF cancels BOTH CE commands!

.CM "comments" . . . . . COMMENT . . ✻

This command allows you to put private notes within a document. These notes will not be printed by DOTPRINT. The entire line containing .CM is ignored at print time. Note: This command must be the last command on a line.

.CO on,off . . . . . CONCATENATION (see also JU ALL)

The format control must be on (.FO ON) for this command to have any effect. The default is .CO on. This causes the computer to fit as many words on a given line length as possible. When .FO is ON and .CO is OFF then each line will be printed as typed, except spaces will be added between words to fill up the entire line length. Let's look at how this can work for you: First, look at how it works.

        .foon;.coon
        Chapter 1   Page 1

Will be printed as:

        Chapter 1 Page 1

But if we turn the CO off, each space becomes a signal to expand the print so,

        .foon;.cooff

        Chapter 1 Page 1

Will print as:

        Chapter   1   Page   1

As shown, this isn't too useful, but if hard spaces are used with it, the results will be different. A hardspace is a space that MUST be printed as a character, so the words on either side of a hard space will be kept together, not spread out. A normal space can be printed in different sizes and can even be dropped but a hardspace MUST be printed ALWAYS the same size. Most lettersets contain a hardspace in the 127 (ASCII value) location. This character is not accessible from the keyboard in any direct manner but we have created an easy way to use it. We use a normal keyboard character for the hardspace but TRanslate the character to a 127 with the TR command. We'll use the ASCII 35 character for the hardspace in this example. (If you're using NEWSCRIPT or ALLWRITE, you can enter a hard space directly from the keyboard, and don't need to use the "TR".)

        .TR 35,35,92
        .cooff;.juall
        Chapter#1 Page#1

Will now print as:

        Chapter 1   Page 1

Now you can see the benefit of using the CO command: the only division is where the normal space occurs, not where the hardspace is. This is great for Tables of Contents, keeping names unbroken on a line and much more!

.CP n . . . . . CONDITIONAL PAGE EJECT

When encountered, this command automatically starts a new page if the number of lines left on the page is less than "n". This is used to keep things, such as category headings, from being printed relatively alone at the bottom of a page.

.CW x . . . CHANGE COMMAND WORD SYMBOL

In the 3.0 version, "CW" was used as centering by width. That command is now "CEW". "CW x" where "x" is any keyboard character is now used to change the command line character from the DOT (period) to any specified character. This was done to allow the user to use the period as the first character on the line without designating it as a command line. Also, some processor files use other symbols as the command line indicator. For example, ALLWRITE uses the semi-colon ";" as its default control word character. To make your files consistent, you need only place .CW ; as the first command in the file, rather than changing every command line.

.DA on,off . . . . . DARK PRINTING . . C,E,F,G
.DA 0,1,2,3 . . . . . DARK PRINTING . . E,C,F,S

The ".DA on" command is the equivalent of double strike printing, except it is designed for use with the DOTPRINT fonts. Each line is printed twice. You should note that for some fonts such as Microprint and Minicubes the ".DA off" will give better resolution. The following chart is for use with the standard printer fonts.

	Epson	Others
".DA 0"	standard	standard
".DA 1"	emphasized	bold
".DA 2"	overstrike	N. A.
".DA 3"	double emphasized	N. A.

In addition, ".DA on" used with the standard print on some printers makes the print thicker in the vertical direction. Also see ".EM".

.DS ON/OFF . . DOUBLE SPACING

This command will automatically double space your text at print time.

.EM on,off . . . . . EMPHASIZED PRINT . . G

When used with the DOTPRINT fonts this command controls emphasized printing. Emphasized printing is slightly wider than regular print, since each dot is printed twice in one pass.   Using combinations of ".EM" and ".DA" you can control the darkness of the print. Delicate patterns may print "muddy" with ".EM", so check the "Letterset Reference Summary" or run a sample if in doubt. Here are some examples of using ".EM" with and without ".DA":

```
.bf TR/PR
.em off; .da off
testing
.em on
testing
.em off; .da on
testing
.em on; .da on
testing
```

will be printed as:

testing
testing
testing
testing

.EN . . . . . END OF FILE

When the computer encounters an END command 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. With other word processors the .EN command is optional but we recommend always using it. Otherwise the computer may print a string of "garbage" at the end of your nice neat file.

.ES "symbol" . . . . . . ESCAPE CODES and RE-DEFINITIONS

This command changes the symbol for the escape code. (These are also known as "escape sequences" and "emphasis marks".) 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 as:

test <u>test</u> test

The SECOND symbol in the escape code sequence is, however, always the same. Assuming the escape code symbol is "!", then the escape sequences are:

!/ . . . Begin alternate font.
!? . . . End alternate font and resume .bf font.
!& . . . Begin underlining non-blank characters.
!$ . . . Begin underlining blank and non-blank characters.
!% . . . End underlining.
!(. . . Begin double width mode.
!) . . . End double width mode.

These sequences can be intermixed. For example:

```
.bfpl/pr
.afmes/pr
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 ON" 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 on.

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

.FM n . . . . . FOOTING MARGIN

The ".FM" command defines the number of lines to be skipped between the last line of text and the bottom title. This is part of the bottom margin. When the text reaches the bottom margin on any page, the computer will move down "n" spaces before printing the bottom title. The default is ".FM 1".

.HL . . . . . HORIZONTAL LINE . . G

When the computer encounters a ".HL" command it draws a horizontal line across the page between the left and right margins. The Horizontal Line will run the entire line length. The print head pin(s) that draw the line are determined by the ".UP" command. See the UP command for more details. The length of the line is determined by the ".LL" command. For example:

.LL 50
.UP 170
.HL
.UP 56
.HL

Would be printed as:

Note: There is one limitation to the HL command - a FONT, typestyle, or graphic must be loaded with the BF command BEFORE the HL will print. If for example, you want to use HL to print a TOP BORDER line before any text is printed, you must load a letterset first. You need not print anything, but a letterset must be loaded. BF1 (the normal printer font) won't work. It must be a graphic letterset or character.

.HM n . . . . . HEADING MARGIN

The "heading margin" is the space between the Top Title line and the first line of the body of the document. One or two blank lines in this position will give a cleaner, more professional look to your printout. The default value is 1.

.HY "symbol" on,off . . . . . HYPHENATION

This command allows you to define the soft hyphen character and also suppress soft hyphenation. A soft hyphen is printed if it must be used as the last character on a line, and is otherwise discarded as though it never existed. The HY command is defaulted to a nonprintable character but is not off. To start a soft hyphen, command ".HY chr on" is the sequence. "chr" can be any keyboard character. Note that HY CHR OFF will refuse to print the HY character regardless of where it falls on the line. If you need to do a lot of hyphenation, you might want to use Electric Webster's Hyphenation feature. Here is an example of the HY effect.

```
.ll 50
.hy & on
Now is the time for all good men to come to the aid of
their coun&try.  The quick brown fox jum&ped o&ver the la&zy
dog.
```

Will be printed as:

```
Now     is      the
time    for     all
good    men     to
come    to      the
aid     of      their
country.    The
quick       brown
fox jumped o-
ver   the   lazy
dog.
```

.IG on,off,1,2 . . . . . IGNORE

This command is primarily used for preliminary drafts. ".IG on" tells DOTPRINT to skip over all text and dot commands until it encounters an ".IG off" command. When an ".IG off" is reached, processing continues as usual.

".IG 1" tells DOTWRITER to process dot commands, but ignore text.

".IG 2" causes the commands ".BF",".AF" and ".TF" to be ignored, but all other commands will be processed normally, until an ".IG off" command occurs. In other words, it tells DOTWRITER to print all text in the standard printer font. Since the standard printer font is much faster than graphics mode, this is a good way of obtaining quick preliminary drafts.

.IM "filename" . . . . . IMBED FILE COMMAND

When DOTPRINT encounters the ".IM" command, the computer will open the file "filename" and begin printing text from that file. When all the text from "filename" has been printed the computer will resume printing from the original file. Note that ".IM" commands cannot be nested. See the section titled "SAMPLE TEXT FILE AND PRINTOUT" for an example of this command.

.IN n . . . . . INDENT LEFT MARGIN

This command will cause a relative indentation of "n", expressed in tenths of an inch, from the current position of the margin. All indent commands are additive. For example a ".IN 5" and a ".IN 6" given later will cause a total indentation of 11 from the original position. (This is different than NEWSCRIPT or ALLWRITE.) ".IN 0" will always restore the original margin. Also, negative numbers can be used so that IN 5 followed by IN-3 will result in an indent of 2 tenths of an inch. Note that "IN" and "OF" control the same thing, so changing the indentation cancels the offset. If you want to use both of them at the same time, each "IN" must restore the offset, or the offset will be lost.

.IX "phrase" . . . . . INDEX . . X

This command works exactly like the .TC command except that you are creating an Index instead of a Table of Contents. A disk file with the extension "/TSC" will be created during printing and then afterwards you may create an Index from the DOTPRINT menu. The index will be stored with the extension "/IND". You can print this file with DOTPRINT. Exactly the same restrictions apply to the .IX command as for the .TC command. The .IX command does not cause a control break, which means you can insert .IX commands on a separate line in the middle of a paragraph without affecting the printout.

The INDEX and CONTENTS programs are self prompting and the file default name will be the same as the file you are printing. If you are printing "MYFILE/EXT.PASS:2", the default file name will be MYFILE/TSC. Note that each IX entry must be on a separate line and must be preceded by the .IX command. This is a bit different than the indexing features of NEWSCRIPT or ALLWRITE.

Hint: If you want to print a large document in sections, you can combine the table of contents and index files afterwards using the DOS "APPEND" command or your Editor, and then use "TCONINX/CMD" to create a single T/C and a single Index. Just remember to assign starting page numbers to the sections after the first one.

.JU ON,OFF,RIGHT,ALL. . . RIGHT JUSTIFICATION

With ".JU on" all text will be right justified giving a smooth right hand edge (except last line in paragraph). With ".JU off" a ragged right border will be printed. Default for this command is ".JU on". With .JU Right all text will be printed flush right with a jagged left hand margin. The short version of this command is .JU R. When ".FO OFF" is in effect, so is ".JU OFF".

JU ALL is exactly the same as JU ON UNLESS the concatenation (COOFF) has been turned off. If Concatenation is off (COOFF), and JU ALL has been specified, the last line of the paragraph will be justified right. This means that the last line of words, regardless of the number of words in it, will be spread out to fill the entire print line. Breaks or large spaces will come between SPACES though, not within a word so that three words on the last line will be spread out into three columns across the last line.

Also, with the above condition met, a new paragraph can be started on the screen in the following fashion: If a line starts with blanks AND a PP command has not been encountered before the line beginning in blanks, the resulting print out will print the blanks - so that the line will be indented as if it were the start of a paragraph.

.KE . . . . . KEYBOARD ENTRY

This command requests keyboard input at print time. When the computer encounters ".KE" it pauses until a line is typed in from the keyboard. The line is then processed as if it were read from disk. You may want to follow ".KE" with a ".CM" to tell the operator what to do.

.KR on,off . . . KERNING CONTROL (see also KS)

The KERNING command (KRON) allows letters to overlap. In this way, an uppercase T which follows an uppercase L (for example) can actually overlap the base of the L; in lettersets such as TRAJAN (SPQR), "V" and "A" can snuggle closely together. *Kerning is especially valuable when printing italicized lettersets, since the slanted letters can overlap only when ".KR ON".* Here's an example using Modern Uncial:

<div align="center">

With KROFF : LT ToT MULTIPLY
With KRON : LT ToT MULTIPLY

</div>

Note that in the second line, the T overlaps the L and o while in the top line, the T doesn't start until the L or o have been finished! Also, The KRON turns PRON so to go to monospaced print be sure to turn KROFF when you use PROFF.

.KS n . . . . . SET KERNING SPACE

This prevents letters from running into each other when the KR is turned on. 'n' is an integer, and sets the minimum number of *vertical* dots of separation between letters. (".SD" sets the minimum *horizontal* separation.) If ".KR" is "ON" and ".KS" is left at ZERO, certain characters will actually print in each others space. To prevent this, set ".KS" to at least "1". In the example below, with KRON and KS at 0, the following symbols: "C = T - X" overlap:

<div align="center">

GT-X

</div>

But, when KS is set to 2, the result is readable:

<div align="center">

C-T-X

</div>

We set ".KS 1" for most of this book, and ".KS 2" for italics.

.LH n . . . . . LINE HEIGHT . . G

To control spacing between lines ("ledding"), use ".LH n", where "n" is a positive number that specifies the number of dots to space down before beginning the next line. This spacing is <u>in addition to</u> any unused dots at the bottom of the characters just printed: a font that barely uses its last print line will usually have a lot of space between lines. On the other hand, if a font fills its entire vertical area, and the Line Height is small enough, lines can overlap. To restore normal spacing, specify ".LH 12", which is the default. ".LH" gives varied results depending on the font you are using, so experiment to find what you like best. When printing sideways banners ("/PRB" files), you may want to change ".LH" between each letter, since proportional spacing only works horizontally, not for banners. ".LH" affects the result of a ".SK".

.LL n . . . . . LINE LENGTH

This command is used to set the line length. In conjunction with the ".LM" command, it determines your right margin setting. The default is ".LL 70" or 7 inches, from the left margin. On 8 1/2" wide paper, an AD of 5 and a LL of 70 total 7.5 inches leaving 1/2 inches on the left and right margins.

.LM . . . . . . LEFT MARGIN

This command is EXACTLY the same as the AD command. We recommend using "LM" instead of "AD" because it's a better reminder of what it does.

.LS n . . . . . LOGO SPACE

This reserves some space at the top of the first page for your pre-printed logo. It works only on the first page. Also, it must be the very first command in the file if you decide to use this option. You must manually position the first page when using this command. "n" determines the number of lines you wish to reserve for a logo (such as on company stationary). Default for this command is ".LS 0". (In ALLWRITE, the command is ".;LG".)

.MF n . . . . . MAGNIFICATION FACTOR . . G

This command allows you to magnify any character font during printing by a factor of "n". Only positive integers are allowed for "n". If any letter of magnified font would exceed the line length, no magnification occurs. To turn magnification off, use ".MF 1". off.

Note: In earlier versions of DOTWRITER, ".FO" was disabled during magnification. This restriction has been removed in 4.0, so you can use magnification wherever you want it. Also, magnification is very fast in 4.0, so you don't need to hesitate to use it. Of course, the printers will take a while to print big letters, but there won't be any particular delays from DOTWRITER itself.

.MX 80,100 . . . . . PAPER WIDTH SETTING . . E,F

This command allows you to tell the computer what size paper you have in your Epson printer. For 13.6" wide paper you should use ".MX 100" at the beginning of your document. The default is for 8.5" wide paper (.MX 80).

Note: If you have a wide-carriage version of any other brand of printer, use the ".WP" command instead.

.MX-80 . . . . PAPER FEED SETTING . . E

Do not confuse this command with the MX80 command. The purpose of MX-80 is to force the EPSON into 1/72 inch increments. This command has very limited function and should only be used by those who are using MX100 with old Graftrax or those who have an EPSON work alike that doesn't function properly with DOTPRINT. If you think you need the command, try it.

.OF n . . . . . OFFSET COMMAND . . G

This command causes the first line after the ".OF n" command to be printed with normal length, but all following lines to be indented by "n" tenths of an inch. This continues until the next ".OF" command is encountered. To restore normal printing, use the command ".OF 0" or ".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.
.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.
.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.

    Note that "OF" and "IN" control the same thing. See "IN" for details of using them together. (The NEWSCRIPT/ALLWRITE command, ".HI", does not exist in DOTWRITER.)

.PA n . . . . . PAGE EJECT

This command causes a new page to be started. DOTWRITER will print active bottom and/or top titles, and increment the page number by one. You can specify another page number after the command, if necessary.

.PG p1,p2 . . . . . PAGE RANGE

This lets you select a "Page Range" to be printed. It lets you re-print defective pages without having to print anything else. The entire file containing the page(s) must be processed by DOTWRITER, but no printing will occur until page "p1" is reached. Then, pages "p1" through "p2" will be printed, and you'll be asked whether you want to enter another (higher) page range. If so, type in two more numbers without the ".PG". To reprint just one page, give its number as the start and the end.

The printer must be turned on and ready when you use ".PG". You can enter ".PG" from the keyboard as a run-time option by holding down "BREAK" as soon as you've "Entered" the filename, or put ".PG" at the top of the file.

.PI 0,1,2,3 . . . . . PITCH SETTING . . C,D,G

This command is used to set the number of dots per horizontal inch when using the graphics lettersets. Not all printers support pitch changes.

C.Itoh	R/S DMP
.PI 0 is 80 dots/inch	.PI 0 is 60 dots/inch
.PI 1 is 96 dots/inch	.PI 1 is 72 dots/inch
.PI 2 is 136 dots/inch (default)	.PI 2 is 72 dots/inch (DMP 500)
.PI 3 is 160 dots/inch	.PI 3 is 100 dots/inch

The default pitch is usually the maximum dot density for that printer. The only density on the EPSON is 120/inch; the only density on most Microlines is 72/inch. DOTWRITER only supports 60/inch on the DMP 2100. Titles print in the pitch used when ".TF" is encountered, so you can use a different pitch for the body.

.pi0
single !(double!)
.pi1
single !(double!)
.pi2
single !(double!)
.pi3
single !(double!)

prints as follows on a C.ITOH:

single double
single double
single double
single double

.PL n . . . . . PAGE LENGTH

This command allows you to use paper of different length than the standard 11" paper. "n" is the number of standard print lines that will fit on the paper at six lines per inch. For example, ".PL 84" corresponds to 14" legal paper.

.PN on,off . . . . . PAGE NUMBERING

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 a ".PN off" command is encountered. If a ".TT" command is encountered before a ".PN" command then numbering begins on page one. Also, the page number will be printed in the type style that was selected with the TF (title font) command. If no title font has been selected, the page number will print in the BF1, or standard printer font. In this case, special commands such as DArk will not be followed.

.PP n . . . . . NEW PARAGRAPH

This command when used without "n" creates a new paragraph with an indent of 5 tenths of an inch. You can change the indent to "n" tenths by using the command with the value desired.

.PR on,off . . . . . PROPORTIONAL PRINT . . G

Proportional printing is the process of varying the distance from the start of one character to the next, based on the size of each character. This command lets you switch between proportional and monospace printing. When using proportional print, it is a good idea to specify some spacing between letters with the ".SD" command. NOTE: If your operating system uses a lot of high memory, and the letterset you are using is a large one (MB2, for example) and you are not using the most up to date lettersets, then some of the letters may come out monospaced even with ".PR on". The supplied lettersets have been proportionalized and will always be proportionally spaced as long as PR is on. Example:

```
.pr off
.sd3
testing proportional print OFF
.pr on
testing proportional print ON
```

will be printed as:

```
t e s t i n g   p r o p o r t i o n a l   p r i n t   O F F
testing proportional print ON
```

NOTE: Proportional print differs from monospace print in the following manner: Monospace letters take up the same room to print regardless of letter size. The "i" takes as much room as the "M". That means that characters can be closer together or farther apart, depending on the type of letter. Proportional print moves letters together so that narrow letters are the exact same distance from the preceding letter and from the following letter. If the SD is set to 4, every letter will be separated from every other letter by exactly 4 dot spaces.

.PS "symbol" . . . . . DEFINE PAGE NUMBER SYMBOL
When DOTWRITER encounters this "symbol" in a top or bottom title, it will replace it with the current page number. The default is ".PS $". Example:

```
.ps #
.tt Manual Page #
```

Will be printed as: Manual Page 2

on the second page, and the number will increment with each page.

.RD n,n "filename",c "filename" . . . . . .READ COMMAND (Form Letters)

This command is used to insert names or phrases from a mailing list into form letters. There are three ways of using it.

1 ... .RD n where n is a positive integer. This allows you to enter "n" lines from the keyboard when the command is encountered. The lines are printed as if the formatting were off.

2 ... .RD n "filename" When this command is first encountered, the mailing list file "filename" is opened and the first n lines are read from the file and printed 'as is'. Each time afterward when .RD n "filename" is encountered, the next n lines are read and printed 'as is'. If the end of the form letter is reached and there are still lines to be read from the "filename" file, then a page eject is given, and the letter is printed again. This process continues until there are no more lines to be read. NOTE: the form letter cannot contain the .IM command.

3 ... .RD c "filename" where c is a non-numeric character such as # or &. When this command is first encountered, th file "filename" is opened, and the first line in that file that begins with "c" is searched for. When that line is found, the operator is asked to "ENTER SELECTION CODE". You may enter a pre-designed code for selecting certain names from the list or you may just hit <ENTER> to use all the names in the list. Now the computer will find the first line that (1) contains the "c" character and (2) contains the SELECTION CODE you entered. The computer will print the contents of the next line and every line after that until another line beginning with the "c" character. Then the computer will stop printing from the mailing list and resume printing the letter until another .RD c "filename" command is encountered. When the next command is encountered, the computer searches for the next line that meets the above requirements and repeats the above procedure. When the end of the letter is reached the computer will eject the page and print another letter if there are still lines to be read from the file "filename".

NOTE: To speed up printing of form letters we suggest you place an ".IG 2" command at the end of the form letter. This will cause all subsequent letterset commands (.BF,.TF,.AF) to be ignored. Thus, if you use only one regular font and one alternate font, neither will have to be reloaded each time the form letter is printed.

How To Use RD For Form Letters

The RD command is very exciting and it is likely that you will want to jump right in and use it. However, we suggest that you become familiar with the DOT WRITER package, especially creating and printing text files, before you get into the command. We have included several examples to get you rolling, but suggest you try a few simple things first.

Let's look a little closer at the RD commands. If RD n is specified, all text up to the RD will be formatted and printed. When the RD is reached, a prompt will appear on the screen which will ask for a line of text. The line must be entered from the keyboard EXACTLY as you want it in the text. This will continue for each of the n lines that were specified, so in the case if RD 4, you will be prompted for 4 separate line entries. You can use this feature in much the same way as you would a KE command except that a KE must be specified for each line where one RD n command can specify any number of lines. Note that the lines will not be concatenated (joined), so each line will be printed as is - just like the FOOFF command had been issued. If you want to print an address, RD 6 will prompt you for 6 line entries, which can be used for entering name, address, city, state, ZIP, etc. If you specify too many lines, entering .CM will cause the line to be skipped without placing a line space in the text.

Now for the RD n Filename command. When this command is encountered, the file "filename" is opened and the first n lines are read in and printed as if they were entered from the keyboard. You should have prepared a file of addresses or other info before using this command. Once the n lines are read and printed, the rest of the document is finished and the page ejected. The document will now start over but this time, when the RD n filename is encountered, the second group of n lines is printed. This continues until each copy of the form letter is finished. Here is an example of what an address file should look like.

```
John Smith
123 Fourth Street
Buffalo, NY 14221
.cm
.cm
.cm
Bob Brown
Widgets Unlimited
P.O. Box 567
Weston, NH 02468
.cm
.cm
Steve Jones
Vice President
Jones and Sons
```

```
30 Manning Ct.
Eggertsville, NY 14226
.cm
```

Note that each address is 6 lines long - and that the .CM command was used to fill the 6 line requirement. In the above list, the longest address is 5 lines and I could have used RD 5 filename - however you should always be sure that you have enough lines for your longest address and unless you can count them up before entry, you had best specify 6 lines.

Note that each address group is exactly 6 lines long and that unneeded lines are .CM commands so that they will be skipped. With this file, each RD 6 filename command will take one address. If you used an RD n filename where n <> 6, the file would not print correctly. Be sure to create your address file in the format you plan to use.

Suppose that you want to print a variable number of lines from the file. That is the time for RD c filename. The c is any character that will not be part of the address being printed. Create your address file in the same fashion as with the RD n Filename command but set a CODE line as the first line of each address. If we use the & in place of the c, so that the command is RD & filename, then each address must begin with a code line of &. Here is an example:

```
&
John Smith
123 Fourth Street
Buffalo, NY 14221
&
Bob Brown
Widgets Unlimited
P.O. Box 567
Weston, NH 02468
&
Steve Jones
Vice President
Jones and Sons
30 Manning Ct.
Eggertsville, NY 14226
```

Note that each address is a different length but that the addresses have been preceded by the & symbol that I have selected to use as the "c" character in the RD c filename command. In this method, each letter will move to the next & symbol and will print all subsequent lines until the next &. The result is that the entire list will be printed with a variable number of lines in each address.

Note that the record length can vary and the .CM is not used. When the RD &
filename command is reached, the program will start at the beginning of the address
file, look for the &, then it will print every line AFTER the & until it finds another
line beginning with &. At the second &, the RD will end and the rest of the letter will
be printed. On the second copy, the next group of lines will print until the next & is
reached and so on until the file is complete or the copy request has been filled.

That is really quite simple after all, but the average user will run into trouble
if he doesn't learn the basics first!

The "&" line serves two purposes: it separates entries, and it can contain
"selection codes." You can code the entries with letters, numbers, or words; and when
DOTWRITER starts to process the first letter, it will ask you to "ENTER SELECTION
CODE". If you just press "ENTER", it will select all names and addresses. But, if you
type something in, DOTWRITER will select only entries whose code lines contain an exact
match for the selection criterion you typed. Upper and lower-case matter in this
comparison.

This completes the discussion of ".RD" and Form Letters.

.RE 0,1,2 . . . . . REVERSE PRINTING . . G

This command is similar to REVERSE VIDEO only it works on paper. "RE 1" gives blanks between letters. "RE 2" prints black spaces between letters. Default is "RE 0".

This is a full reversal, using RE 2

.SD n . . . . . SPACING BETWEEN LETTERS . . G

This command allows you to adjust the space between letters. Default is ".SD 0". 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 use ".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 exceeded, there is a default to the width or the letter in the current font. If you want more than this, simply use a space character when typing your text.

.SK n . . . . . SKIP LINE(S)

"SK" means "skip" and "n" is the number of lines to be skipped, with "1" as the default. ".SK 5" will skip 5 lines. The size of the first skip depends on the current base font and line height, because DOTWRITER includes clearance at the bottom of each print line before it considers what to do next. The size of the remaining skipped lines depends on the kind of printer you are using (7 or 8 bit). When vertical lines are drawn, spacing is always 7 or 8 vertical dots at a time, even for the first skip.

.SK -n . . . . . BACK SKIP LINE(S) . . F,C,D

This command allows the printer to skip backwards, moving the paper up the page n lines. The line spacing is 6 lines/inch if you are using regular 10 CPI print or 9 lines/inch if you are using a graphics letterset. You can only skip back as far as the top margin. For example ".sk-1000" will skip back to the top margin. See also .VT. This won't work on an MX-80, or any other printer lacking the necessary hardware capability. The FX-80 can do it, but the FX-100 cannot (hardware limitation).

.ST "message" . . . . . STOP, display message

When this command is encountered the computer will stop printing and display your "message" on the CRT display only. It will continue operation when you press the <ENTER> key. This command has many uses. For example:

```
  text text text text
  .st Message Here
  .ap nextfile
```

The computer will print until the ".ST" is hit. Then processing will stop until the <ENTER> key is depressed. In this manner you are given the opportunity to insert the diskette containing the file "nextfile" if necessary.

.SW n . . . . . SPACE WIDTH . . G

This command applies when proportional print is in effect. It sets the minimum width for blanks. The value "n" is an integer between 0 and 9, and determines the minimum width for blanks as a fraction of the maximum character width. The default is ".SW 5". For example:

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

Will be printed:

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

Note that in the 3.0 version a decimal version is used between 0 and 1. In the 4.0 version, the decimal is not used.

.SY e nn,nn,nn . . . . SEND ASCII VALUES DIRECTLY TO PRINTER

This command allows the user to send ASCII codes directly to the printer. 'nn' stands for a number between 0 and 255, and you can send many values at a time this way. The command is intended to let you do things with your printer that aren't directly supported by DOTWRITER. For example, the NEC 8023A can be placed in the unidirectional mode by sending it two characters with the command SY e 27,91. If you sent printable ASCII values this way, they will be printed in the printer's hardware font, and that will throw off the line count.

ADDITIONAL USES OF ".SY e"

".SYe" can tell certain printers to do graphics printing bi-directionally. Most printers will only do graphics from left to right, in order to maintain good regis-

tration of the dots. If you have a C.ITOH 8510, for example, you can include ".SYe27,60" *after* the first ".BF", and the printer will print bi-directionally. However, as soon as DOTWRITER prints a top or bottom title (even a page number), it will revert to single-directional printing. Bi-directional printing is useful for rough drafts, but usually does not give the perfectly straight lines needed for a final copy.

".SYe" can tell certain printers to print in color. The Radio Shack CGP-220 has this capability, and you can find the correct codes in the manual for that printer. The C. ITOH 8510SCP also can print in color. The sequence is: ".SYe27,67,nn", where "nn" is a number from "49" (yellow) through "55" (black). Color printing is effective for an entire DOTWRITER print line, and you cannot get more than one color on the same line, nor print one letterset in multiple colors. We agree it would be nice to be able to print a flag in red, white, and blue; but there just aren't enough color printers (yet) for us to justify the considerable programming effort.

One other comment on color printing: with a "ribbon" printer, such as the C.ITOH, the colors get used up quickly, and as soon as the ribbon begins its second circuit, you are likely to see lighter or blotchy areas. The inks also transfer from one part of the ribbon to another. Trying to print with ".DA ON" sometimes helps, but it also uses up the inks even faster, and often produces false colors (orange comes out red). By contrast, an ink jet printer, such as Radio Shack's CGP-220, produces good colors consistently, because there is no ribbon to wear out. An ink jet printer is also faster on secondary colors (green, orange, purple), because it only has to make one pass instead of two or three.

.TC "phrase" . . . . . TABLE OF CONTENTS . . ∞

Each time this command is encountered, the "phrase" is written to a disk file along with the page number where encountered. The disk file has the same name as the first file printed, with the extension "/TSC". After the text has been printed, the DOTWRITER menu that appears contains a selection to create a table of contents from this file. After selecting "create table of contents", enter the file ID of the "/TSC" file; a new file will be created with the file extension "/TCT". This file may be printed using DOTWRITER.

We suggest that you create an empty "/TSC" file on a diskette that has lots of free space, because the computer will put the file on the first available drive even though it may run out of room later. NOTE: the .TC command does not cause a control break.

There is one restriction on the .TC and .IX command. Entries cannot be recorded in an imbedded file. If a .TC (or .IX) command is encountered in an imbedded file (that is, a file called with the .IM command), it will be ignored.

There is one further restriction when using some versions of Model I TRSDOS. You cannot use the .TC (or .IX) commands after having used the .IM command. This is because the Model I TRSDOS does not process the BASIC command OPEN"E" correctly.

.TF "filename" . . . . . TOP/BOTTOM TITLE FONT

This command determines the letterset in which the top and bottom titles will be printed. The default is the printers standard font. Formatting options such as ".DA", ".CEW", 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. Note: To center the top and bottom titles centering must be turned on before the ".TF command is issued.

.TM n . . . . . TOP MARGIN

This defines the top margin. "n" specifies the number of lines. Default is ".TM 6".

.TR a,b,c . . . . . TRANSLATE COMMAND . . X

This command is used to substitute one set of characters for another set during printing. For example, you can change all upper case characters in the input file to lower case characters when they are printed. The command is given in the format .TR a,b,c where a is the smallest ASCII value to be changed, b is the largest ASCII value to be changed and c is the amount to be added or subtracted to the characters in the range defined by a to b. For example, the command .TR 65,90,32 will change all upper case letters to lower case, and .TR 97,122,-32 will change all lower case to upper case. This can be useful if you are printing with a letterset that contains only upper case characters. Another example: The "hard space" character in NEWSCRIPT is 127, and the PL letterset (as well as most Dot Writer lettersets) contains a "hard space" in position 127. But the hard space character in Electric Pencil is 176. If you have a Pencil file that contains one or more hard spaces, and you want to print it using the PL letterset, you should use the command .TR 176,176,-49. This will translate ASCII characters with a value of 176 to 127.

Here is another example of the use of TR. Suppose you wish to use a hardspace. It can't be printed from the keyboard since it is stored as an ASCII 127. So, use another keyboard character and translate it. The # character is ASCII 35. 127 - 35 = 92. So in order to translate the # to a hardspace character issue the command .TR 35,35,92. Here we have specified a range of 1 character, (35 to 35) and have added 92 to it to make it 127. The result is that the # will now be printed as a hard space.

Note that the .TR command does not cause a control break. Thus, you may translate a character in the middle of a sentence and switch back to normal characters in the same sentence. To turn the translation off, use .TR 0.

.TT "title" . . . . . TOP TITLE

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

The default value is ".TT" followed by 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.

.UP n . . . . . UNDERLINE CONTROL

This command specifies which of the 8 print head pins is used for underlining. The bottom pin is pin 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 values. To start or stop underlining you must use an escape code. If the font being underlined is more than eight pins high the number of pins available for underlining is equal to the number of print lines times 8 minus the height of the font in # of pins. For example, if the font we are about to underline uses two print lines (16 pins) but the height of the font is 9 pins high (font PL) then the bottom seven pins (#1,2,3,4,5,6 and 7) of the second print line are available for underlining. Note that the pin called 0 is not used. This is because in bit image mode, there are only 8 bits so the 9th pin, is not available for programming. Also, the pins will fire in an additive fashion. If you set UP to 129, both the number 8 pin (128) and the number 1 pin (1) will fire. The same UP assignment is used for the HL (horizontal line) control. See the diagram below. The escape codes for underlining are:

       !& starts underlining non-blank characters
       !$ starts underlining blank and non-blank
       !% stops underlining

For example:
```
.BF PL/PR
.up 64
testing !$testing testing!% testing
.sk 1
.up 1
testing !&testing testing!% testing
.up 65
.sk 1
testing !$testing testing!% testing
.up 127

testing !$testing testing!% testing
```
will be printed as:

testing <u>testing testing</u> testing

testing testing testing testing

testing <u>testing testing</u> testing

testing **testing testing** testing

The default value is ".UP 1".

EPS/ ITO
PRINT HEAD PIN LAYOUT 8

PIN NUMBER	.UP VALUE
8	128
7	64
6	32
5	16
4	8
3	4
2	2
1	1
0	NOT USED

DMP/OKI
PRINT HEAD PIN LAYOUT 7

PIN NUMBER	.UP VALUE
8	NOT USED
7	64
6	32
5	16
4	8
3	4
2	2
1	1
0	NOT USED
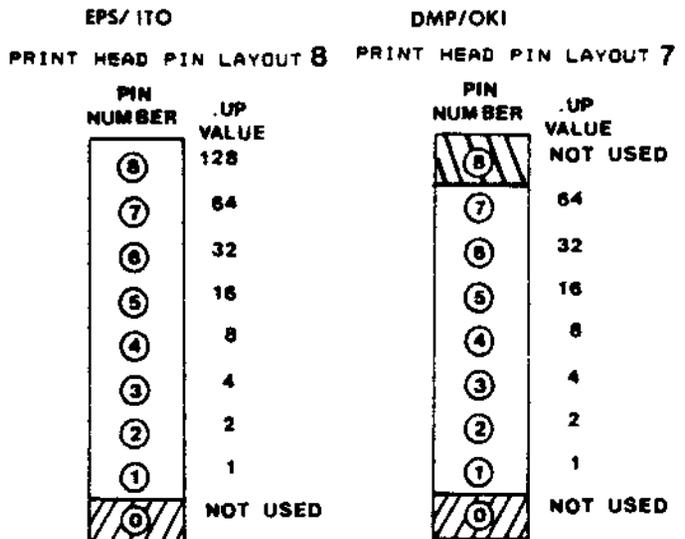
.VL on,off,1,2,3 . . . . . VERTICAL LINE . . G

The commands .VL on or .VL 1 draw a vertical line at the end of the Left hand margin. To ensure that the text does not hit the line, you should indent the text with the .IN command. The command .VL 2 draws a vertical line in both the left and right hand margins. The command .VL 3 draws a line in the right hand margin only. The default is .VL off. You can use .VL along with .HL to enclose your text in a box. For example:

```
.upl;.lh8;.ad30
.ll25;.HL
.VL2
.sk
.in2
.fooff
testing
testing
.sk;.in0
.HL
.foon
.ad3;.ll65;.VLoff;.in0
```
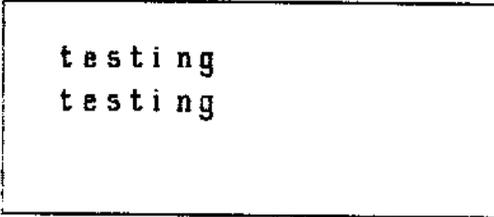
```
+--------------------------+
|                          |
|   testing                |
|   testing                |
|                          |
|                          |
+--------------------------+
```

Note: The VL command will cause double spacing to occur in response to a SK command. This is because a line feed must be sent to print the margin lines. Avoid using SK n in any text that follows the VL command unless you want it spread out. Also, when boxing text, command HL should precede the VL command.

.VT n . . . . . Vertical Tab . . C,F,D

This command is used to advance or return to a specified line on the paper. The line to be "tabbed" to is determined in one of two ways.

The command .vt alone on a line records the current position on the paper. To return to this position use the command .sk-n, where n is a large number such as 1000. The VT command alone must be the last command on a line.

The command .vt n, where n is an integer, sets the "tab position" at n lines from the top of the paper using 6 lines per inch as a measure. The ".SK-1000", after the ".VT, causes the printer to return to the "tab position", even if it has to advance or reverse feed lines to get there.

.WP ON. . . . C.Itoh WIDE PAPER . . C, D, O

Same as the .MX 100 command, but for all other printers.

BLOCK GRAPHICS

(This material is not intended for general use, but if you still have the original "GEAP" program, or desperately need a way of printing screen graphics, DOTWRITER can do it.)

TRS-80 screen block graphics can be used with DOTWRITER, regardless of your printer. The "PL" letterset contains the TRS-80 graphics blocks. To print using these block graphics, you must first draw what you want printed with GEAP (no longer available or supported by PROSOFT), then save it to disk with GEAP's "FG" command. For example: suppose the filename of the saved picture is "pic". Then to print it with DOTWRITER use the following sequence:

```
.bf PL/PR
.fo off;.pr off;.lh 4
.im pic
```

NOTE: You may also want to move the picture toward the center of the paper with the .IN command.

APPENDIX A
THE FIRST PAGE

The top of the initial (first) 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.

POSITIONING THE PAPER

Before starting DOTWRITER, the top edge of the paper should be just above the print head, so that printing could occur on the first physical line of the paper. DOTWRITER will move the paper down from there to allow for the top margin, but neither DOTWRITER nor your printer can "sense" that you've positioned the paper properly. If you start with the paper a couple of lines past (or before) the top edge, your printing will be in the wrong place on every page.

APPENDIX B
USING WORD PROCESSORS AS EDITORS

We have mentioned NEWSCRIPT and ALLWRITE as text editors throughout this manual because they are our products, and have been integrated very well with DOTWRITER. However, most other Word Processors can also be used very nicely with DOTWRITER. In fact, any program that creates an ASCII file can be used.

SCRIPSIT

If you are a Scripsit user, there are some things you will have to know. The first and most important is to always save the document with the "comma A" prefix. That will create the necessary ASCII file. For those of you who are new to Scripsit, that is done in the following format:

S,A filespec/ext.password:drive

You must also forget normal Scripsit formatting procedures. Your Scripsit file should look just like the example file shown earlier in the manual. In addition, it is important to ALWAYS end a file edited with Scripsit with the ".ST" or ".EN" dot commands. This is due to the unusual way Scripsit handles files.

OTHER WORD PROCESSORS

Although we have no direct experience with the other TRS-80 word processors, we've been told by many of our DOTWRITER customers that SUPERSCRIPSIT, ELECTRIC PENCIL, LAZYWRITER, and LESCRIPT can all create "ASCII" files and therefore be used quite well with DOTWRITER. Please refer to the documentation for your word processor if it does not automatically create "ASCII" files. For safety, you may want to end your DOTWRITER scripts with ".EN" to avoid end-of-file problems.

How do you know whether it creates "ASCII" files? Try "LIST"ing a file to the screen. If it looks just the way it did when you were editing it, then it's in "ASCII" format and DOTWRITER can read it, too. If you see special screen graphics, gaps, or reversed upper/lower case, then the file is not in "ASCII" format, and you will need to use a special command in your word processor to convert it.

APPENDIX C
ERROR MESSAGES

When the computer encounters a suspected error it will beep the printer three times (EPSON ONLY) 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 WONT FIT AS ENTERED ..... 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 .... indicates that the ".BF" or ".AF" command specified a font that the computer couldn't find on any of the diskettes in the drives. DOTWRITER prompts you for a correction.

FILE NOT FOUND .... the computer could not find the specified text file. The computer pauses until the operator enters a new name for the text file.

WORD TOO LONG .... occurs only with the .FO ON command in effect. 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 line.

USE POSITIVE NUMBER .... one of the dot commands was followed by a negative number. The computer selects a value of +1.

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 greater than 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 greater than 255. The default is .LL 70.

MUST DEFINE REGULAR FONT BEFORE ALTERNATE FONT ..... self-explanatory.

MAGNIFYING FACTOR TOO BIG .... defaults to no magnification.

ALTERNATE LETTERSET WON'T FIT WITH "filename" ...... the alternate font defaults to the font specified by the BF command. It will be the same as the "filename" font.

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 .... 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 .... defaults to no title.

TOO MANY COLUMNS .... this message occurs if you specify the column widths and have specified too many columns for the available line length. For example, if you had a line length of 6 inches, a column width of 3 inches and you specified 3 columns, you would receive this error message.

INDEX

E

ELECTRIC PENCIL 70
EM 45
emphasis marks 46
emphasized 44
Emphasized 45
EN 45
end of file 45
errors 71
ES 46
escape sequences 20, 46, 62
Escape symbol 46
exclamation mark 46

F

file creation 12
file requirements 11
first page 69
FM 47
FO 47
font selection 39
footing margin 47
form letters 57
formatting 47, 50
full lines 30

G

gap 62
graphics 68

H

hanging indent 53
hard drives 8
hard space 28, 31
heading margin 48
HL 47
HM 48
horizontal line 47
HY 48
hyphenation 48

I

IBM-type printers 9
IG 49
ignore text 49
IM 49
imbed file 49
imbedded files 22
IN 49
indentation 49, 53
indexing 49
IX 49

O

joining lines 43
JU 50
justification 47, 50

K

KE 50
KERNING 14
kerning 51
keyboard entry 50
KR 51
KS 51

L

large characters 52
LAZYWRITER 70
ledding 51
left margin 49, 52, 53
LESCRIPT 70
letterhead 52
Letterset Design 7
letterset disks 6
Letterset Utilities 7
LH 51
line height 51
line length 13, 52
line spacing 44, 51

lines 43, 47, 67
lines between columns 41
lines per page 55
LL 52
LM 52
logo space 52
long files 39
LS 52
LSMU 7

M

magnification 52
mailing list 57
margin 39, 47, 48, 49, 50, 52, 53, 64
messages 71
MF 52
micro-spacing 61
Microline 9
missing files 34
mixed fonts 38
multiple columns 40, 41
MX command 52

N

names and addresses 57
NEC8023A 62
new line 40
new page 44, 54
numbered pages 55

O

OF 53
offset 53
overstrike 44
overview 11

P

PA 54
page 44, 54, 69
page length 55
page numbering 55, 56
page range 34, 54
page symbol 56
paper 69
paper size 55
paper width 52, 67
paragraph 40, 55
period 44
PG 54
PI 54
pitch 54
PL 55
Plug-n-Play 9
PN 55
positioning paper 69
PP 55
PR 55
pre-printed letterhead 52
primary font 39
printer control 62
proportion 14
proportional printing 55
PS 56

R

RD 57
RE 61
reverse image 61
reverse paper movement 67
right margin 47, 52
run-time options 35
running DOTWRITER 33