DOTWRITER 4.0

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

EPS/ITO

INCLUDES THESE FOURTEEN LETTERSETS:

MB2 MEDIUM BOLD 2

MB Medium Bold

TR TypeRite

FF FLAT FACED

GR GREEK ΓΡΕΕΚ

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) 1982, 1983, 1984 by W. K. Mason

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.Ø

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

Copyright (c) 1982, 1983, 1984 by W. K. Mason

PROGRAM LICENSE AGREEMENT

DOTWRITER IS 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 DOTWRITER and/or its 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 DOTWRITER is 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 DOTWRITER diskette 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 DOTWRITER.

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

Date: March, 1984 Printing: 6 5 4 3 2 1

Copyright (c) 1984, W. K. Mason

Published by:

PROSOFT, a division of TTSC
Box 560
North Hollywood, CA 91603
(818) 764-3131

All rights reserved. No part of this book may be reproduced by any means without the express written permission of the publisher. Examples are for personal use only. Every reasonable effort has been made to ensure accuracy throughout this book, but the author and publisher assume no responsibility for any errors or omissions. No liability is assumed for damages resulting from the use of information contained herein.

"DOTWRITER" and "GEAP" are Trademarks of W. K. Mason

"PROSOFT" and "NEWSCRIPT" are Registered Trademarks of The Tesler Software Corporation.

"ALLWRITE!" is a Trademark of The Tesler Software Corporation.

"TRS-80" and "TRSDOS" are Registered Trademarks of Tandy Corporation.

"Electric Webster" is a Trademark of Cornucopia Software, Inc.

TABLE OF CONTENTS

INTRODUCTION TO DOTWRITER 1

- Installation 2
- Components 2

DOTPRINT Overview 4

- Creating A Text File 5
- Starting The File 5

Tricks Of The Trade 12

- Imbedded Files 12
- Appended (Chained) Files 13
- The Standard Print File 14
- Top and Bottom Titles 15

Running DOTPRINT 19

- Run-time options 21
- Stopping DOTPRINT 21

LIBRARY OF DOTWRITER COMMANDS 23

INDEX 55

ACKNOWLEDGEMENTS

DOTWRITER was designed and written by William K. Mason. This book was written by Richard C. McGarvey, and printed entirely with DOTWRITER 4.0 on a C. ITOH 8510A. If your printer has the necessary mechanical capabilities and is supported by DOTWRITER, and you have the character fonts we used here, you can do anything you see in this book.

Cornucopia Software's "Electric Webster" spelling checker was used to check and correct most of the spelling throughout this book, and to generate hyphenation points. When using big fonts, we recommend you use hyphenation also, for improved readability.

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 hundreds 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 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.

This book describes DOTWRITER 4.0, which is an "all machine language" version of DOTWRITER. That means it will run very fast -- much faster than your printer can print, infact. If you've been using an earlier version, you'll be happy to know that 4.0 is almost four times faster (that isn't why its' called "4.0": the previous version was "3.0"), has several new features, accepts the same files and dot commands you've already learned, and is even easier to use.

To get started with DOTWRITER, just follow these five steps:

1. Install it as explained below;
2. Write something with your word processor's editor;
3. Include formatting control commands;
4. Make sure the needed lettersets are "on-line";
5. Print your file with DOTPRINT. DOS command is: DOT

Copyright 1984 by W.K. Mason

INSTALLATION

DOTWRITER for the Models I and III is distributed on a TRS-80 single-density, 35-track "flippy" disk. All popular DOS's on the Models I and III can read this disk: to read side two, just turn the disk over. If you're using a Model III, you will have to "CONVERT" or "REPAIR" both sides of the disk in order to read it. Then, please copy everything from both sides onto disks of your own (backup disks) and work with those backups afterwards. The "BACKUP" command may not work for this, but "COPY" will work just fine. Be sure to keep the original disk where you can find it, because you will need to send it back to us if you ever want updates (corrections) or enhancements.

DOTWRITER for the Model 4 is distributed on a double-density, 40-track data disk. TRSDOS and DOSPLUS can both read this disk. Please make a "BACKUP" of the disk immediately, and then copy the portions you need to your own DOS disk.

These are the components of the DOTWRITER system:

1. DOT/CMD -- the main printing program

   DOTPRINT does the actual text formatting and printing, and is the most important part of DOTWRITER. Because different dot matrix printers work differently, we've had to give temporary names to the programs that support each printer. These names are listed below, and you should pick the one that works with your printer, then COPY and RENAME it to "DOT/CMD" on a disk of your own. We will refer to this program as "DOTPRINT" or "DOT" or "DOT/CMD" throughout the documentation. Note: only one or two of these will be on your disk, and they will be on the FIRST side:

DOTEPS/CMD	Epson MX, RX, and FX
DOTITO/CMD	C. Itoh 8510 and 1550
DOTOKI/CMD	Okidata Microline 84A, 92, 93
DOTDMP/CMD	Radio Shack DMP 200, 400, 500
DOTD21/CMD	Radio Shack DMP 2100

   If you have two of these printers, and the disk you bought only supports one of them, you can order the second version from us for $20.00 (this price is available only to registered owners).

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

   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.

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. We've included over a dozen 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 Microline or DMP printers, the lettersets are set up for use with 7-bit printers. The two kinds of 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.)

5. Optional Font Disks (not included)

   You can buy additional lettersets from us at any time. As we're writing this book (early 1984), there are 15 disks available, and three more in preparation. Each disk contains between four and ten lettersets (the limit is how much we can squeeze onto the 7-bit side of the disk, not how little we think we can get away with). These 15 disks contain over 100 lettersets, and there are a couple of dozen other lettersets almost ready for release. When you decide you want more variety, just look over the samples we've sent you, or check with us for new releases. These disks are all in single-density, 35-track format, and are "flippy" disks: Side 1 has the 8-bit versions, and Side 2 has the 7-bit versions. If you have a Model III or 4, you will have to "CONVERT" these disks to make them readable.

6. Letterset Design System (not included)

   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 not for native Model 4 mode (the screen graphics of the Model 4 are uneven, so you can't design properly with it). 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 shift letters left and right, change their sizes, combine them, make them suitable por proportional printing, or anything else along those lines, then we recommend that you order the Design System. Since since they come with a manual describing their uses, we won't describe them further here.

Copyright 1984 by W.K. Mason

DOTPRINT: An Overview

DOTPRINT 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 DOTPRINT 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 "DOTPRINT" here, we will always be referring to the file you renamed.

You MUST create a TEXT FILE with a Word Processing Editor BEFORE you run DOTPRINT. The text file must be saved in ASCII format; most of the popular word processors can do this. Also, these files must include the text formatting (layout) commands that are recognized by DOTPRINT. 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 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 the user wants to create. You use special commands, which we will explain soon, to describe 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; DOTPRINT 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 by following the instructions included with the EDITOR or WORD PROCESSOR you are using. 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.

Finally, you run DOTPRINT and enter the name of the file that was just created. DOTPRINT loads the file, scans it and begins to print the file, following the commands that you included in the text file.

Copyright 1984 by W.K. Mason

Creating A Text File

DOTPRINT 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 often printed 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. Usual print needs such as letters, can be printed with just the default values. More complex text files may require the use of format command changes and this is easily done. We'll show you a short cut (called STANDARD PRINT FILE) that we use all of the time. But for now. Let's look at the actual creation of a file!

Starting The File

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

That brings us to the next point. DOTPRINT recognizes ONLY DOT commandS. The name DOT command is a coincidence and is not related to the name DOTPRINT. A DOT command is a command used by DOTPRINT and each DOT command is preceeded 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.

Next, let me point out that REGARDLESS of the editor you have chosen to use, DOTPRINT will format the text based on the 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.

Line lengths on the screen do not reflect line lengths on the printed page unless certain commands have been implemented.

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 turned off!
This line was printed with proportional turned on!

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 proportion is all about!

Version 4.0 has another command that can affect the length of the line or word. It is the KERNING command and, though it is explained in the command library, I'll also cover it by example in this section. "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 spaced 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. (It also takes more time to process the text.)

Now that I have covered those few points, let's get in to the actual creation of the text file. First, load in and run your chosen editor. If you use NEWSCRIPT or ALLWRITE, you'll find that the command sequence and operation of the programs are very similar to DOTPRINT's requirements. If you do not have either of those word processors, don't worry. As I said before, any ASCII capable processor or editor will do.

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 just use default values. However, to show you how DOTWRITER works, I'll specify some of the commands. The following command sequence can be done in any order. As you follow the examples in this book, if you don't understand what a particular dot command is doing, please look it up immediately in the "Command Library" section of this book. I'll briefly explain the commands here, but for a full explanation, look up each command as we go!

The first command to learn is very simple: ".CM" ("comment"). It's used to place comments in the text file for future reference. I'll use the CM lines as instructional tools and there is NO NEED FOR YOU TO DUPLICATE THEM in your example file. COMMENT LINES ARE COMPLETELY IGNORED BY DOTPRINT AND WILL NOT ALTER THE TEXT PRINT OUT AT ALL! This narrative won't always show the period along with the dot command, but the examples will, and you must, also.

```
.cm As I mentioned above, this is a comment line and will not
.cm print out. I'm using them so that I can enter explanations
.cm directly in the text file we're creating now. You can
.cm disregard them as you type in this example and in your own
.cm files, use them to make notes to yourself.
.cm First, Let's turn on the proportional spacing feature and
.cm and set up the left, right, top and bottom margins.
.PRON;.AD5;.LL70;.TM6;.BM6
.cm PRON starts proportion. AD5 gives a 1/2 inch left margin
.cm (plus any mechanical margin built into your printer) and
.cm the LL70 gives a line length of 7 inches (in tenths). The
.cm TM and BM commands specify 6 lines at top and bottom of
.cm the page. Also, note that the semi-colon is used to
.cm separate the commands. In the 4.0 version, the ";" is
.cm optional. I use it out of habit. Now we should choose and
.cm load a letterset - we'll use medium bold
.BF MB/PR
.cm the BF (begin font) command could have been placed at the
.cm end of the last command line. It was placed on a separate
.cm line for ease of reading only. THERE IS ONE RESTRICTION
.cm THAT APPLIES HERE - commandS THAT HAVE A FILE NAME OR TEXT
.cm AS PART OF THE command MUST BE THE LAST command ON A LINE!
.cm Now let's select an alternate font -
.AF TR/PR
.cm Using an alternate font is optional. I'm using it for
.cm demonstration purposes only.
.cm We're ready to start the letter now. First the return
.cm address - we'll center it!
```

```
.CEON
John Smith
123 Main Street
West Town, NJ 07142
(291) 987-0000
January 1st, 1984
.CEOFF
.cm All of the text between CEON and CEOFF will be centered.
.cm Each line will be centered separately because the format
.cm is automatically turned off. Now we want to skip a line,
.cm turn the format off and put in the addressee's info
.SK;.FOOFF
Bob Brown
456 Seventh Street
Los Angeles, CA  90000
.cm The SK causes one line to be skipped.  The FOOFF causes the
.cm format to be turned off so that the lines in the address
.cm will not be joined into one long line as would normal text.
.cm Now we'll skip a line, turn the format on, and start the
.cm text of the letter.
.SK;.FOON
Dear Bob;
.PP
.cm The PP command specifies the paragraph indent.  We
.cm mentioned that before!
I was certainly pleased to receive a copy of the !(NEW!)
4.0 version of DOT WRITER.  It is by far the best release
.cm the !( starts expanded print and the !) ends it!
yet.  I am amazed at the speed and the additional commands.
.PP
.cm That ends the first paragraph and begins the new one.  In
.cm the next paragraph I'm going to use the alternate
.cm letterset and underlining.  I'll specify the underlining
.cm parameters now - be sure to look up all of these commands
.cm in the command library as we go so that you'll know what
.cm is going on!
.UP 2
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 !/O!?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.
```

Copyright 1984 by W.K. Mason

```
.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 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.
.PP
Thanks again for a great program and for so many great new
features and lettersets.
.cm Now we'll close with a few tricks.  First we'll turn format
.cm off so that we can treat each line separately.  Then we'll
.cm indent the closing text.
.FOOFF;.IN45
.sk
Sincerely,
.SK4
John Smith
.EN
.cm The final command is ENd and is used to stop DOTPRINT
.cm from printing beyond the end of the file.  It is optional
.cm but it is recommended.
```

(END OF EXAMPLE)

Now that you have typed the letter, and you left out the comment lines, it should look like this:

```
.PRON;.AD5;.LL70;.TM6;.BM6
.BF MB/PR
.AF TR/PR
.CEON
John Smith
123 Main Street
West Town, NJ 07142
(291) 987-0000
January 1st, 1984
.CEOFF
.SK;.FOOFF
Bob Brown
```

Copyright 1984 by W.K. Mason

```
456 Seventh Street
Los Angeles, CA 90000
.SK;.FOON
Dear Bob;
.PP
I was certainly pleased to receive a copy of the !(NEW!)
4.0 version of DOT WRITER.  It is by far the best release
yet.  I am amazed at the speed and the additional commands.
.PP
.UP 2
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 !/O!?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 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.
.PP
Thanks again for a great program and for so many great new
features and lettersets.
.FOOFF;.IN45
.sk
Sincerely,
.SK4
John Smith
.EN
```

This letter will print as shown on the next page:

Copyright 1984 by W.K. Mason

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 certainly pleased to receive a copy of the **NEW** 4.0 version of DOT WRITER. 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

Copyright 1984 by W.K. Mason

Tricks Of The Trade

That is all there is to creating a text file for DOTPRINT! It is really quite easy. However, if you have looked at the command library, you know that I took a lot of short cuts. For example, in the letter we just wrote I DID NOT specify the spaces between letters, the height of the space between lines, or the minimum width of the space between words. I also didn't use numerous lettersets, multiple columns, vertical tabbing, table of contents, index, imbedded or appended files and much more!

Once you have learned how to use the DOTPRINT program and how to create a file that will print properly, the rest is a simple matter of adding new commands to your DOTPRINT vocabulary. In fact, many of the features we have developed for DOTPRINT 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.

No, I don't plan on dropping you here with nothing more than the single example. I'm going to 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. We have an interesting command called IMbed. The IM command allows you to indicate a place in a text file where DOTPRINT should momentarily stop, get another file from disk and print it, and then return to the original file and continue printing.

An example of the IMbedded file would be 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. I use the AP command extensively and advise ALL USERS who write more than one or two page documents to do the same. Here is how I use it.

This manual is quite long. It really isn't much fun to print out 20 pages and find that there is a mistake on page 14. Using the "IG1" command, I can print just the damaged page, but that is more work than I want to go through. Besides, mistakes are not always as simple as just a misspelling. Sometimes a mistake can cause the entire file to go haywire.

Not only that, but editing a file that is 20 pages long is a real problem. Since you can only see small parts on the screen at a time, it becomes difficult to keep track of where you are. There is also the problem of running out of memory space. Why limit your text file to the size of your computer?

With the AP command you don't have to be limited, and you don't have to write in large chunks. AP allows you to flow easily, with no loss of time, page numbering or print continuity, from one file to another. Simply end each file with the command AP ANYFILE/NAM and when the file is finished it will automatically load and print the next file as if it were part of the last. ANYFILE/NAM stands for the filespec of the next file you want to print.

This entire manual was printed using numerous small files. I named them DOT1, DOT2, DOT3 etc. Each file ends with the AP command calling the next in sequence. DOT1 end with the command AP DOT2. When editing, I only have to load and edit a small section, but at print time, the entire manual is printed in one piece. Another hint that will help is that when I print out the manual for the first time, I preceed each AP command with a stop command (.ST). This stops the printing at the end of the file and I can then check the file just printed for errors before continuing. When I have completed the check, I hit ENTER and the printing continues 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 DOTPRINT 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 do break 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

DOTPRINT allows the user to control printing in almost every conceivable way. The user can specify the 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 and much more. This versatility carries with it a small problem - keeping track of what commands you have specified. Also, it can be a real pain if you have to type all of the commands every time you want to create a file. Another problem that arises is that, as you type your file, you may turn on a feature and forget to turn it off. It is really aggravating when you print a file and walk away, only to return to find that you never turned off the centering and now every line of the letter is centered!

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, IN0 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.

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 might 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 I used as standard print for this manual:

```
.DA0;.EMOFF;.MF1;.PI3;.PRon;.RE0;.SD3;.SW5;.PNon;.Ceoff;.Ad5
.LL70;.IN0;.COoff;.JUALL;.UP1;.ES!;.HY£ON
.PS$;.WPOFF;.LS;.TM6;.HM2;.BM6;.FM2
.KRON;.KS1
.TFmodunc2/pr
.TR 35,35,92
.TT # Dot#Writer#4.0 Page#$
.BT # Copyright#1984#by#W.K.#Mason #
.coon .VT6
.TR0
.FOON;.UP32;.ceoff;.PL66;.CB1;.LH7;.DSOFF;.VLOFF;.JU   ON
```

As you can see, I included most of the common commands in this file -- actually, every command I though I might use. I didn't use them all but if I had, I could have easily reset them if needed. I IMbedded this file at the beginning of every text file whether I needed it or not. That way, if I made an error while editing, it would only mess up one file and not the entire manual.

In my own use, I have a file like this for every category of document I produce. One for manuals, business letters, legal work, personal letters, memos, etc. Each file is set up to my personal preferences. This way, instead of starting each document by typing parameters, I simply IMbed the proper STANDARD PRINT FILE. When I need a new type of format, I load in one of my standard files, edit it to the new parameters and save it under a new name. For example, the LEGAL file I use is designed to meet the requirements of the court I deal with most. It is a modification of the BUSINESS/FMT file that allows for a longer page, wider margins and different top and bottom titles. I'm also so lazy, that I have a separate file for the letter close. I have it saved under the name SINCERE. When I want to close a letter, I simply put: .IM SINCERE as the second last command and follow it with .EN - the end command.

Please use this method. You'll find that it saves you a great amount of time and effort and even though most of the commands have default values, you find that the more you use DOTPRINT, the more you'll want to manipulate text. The more special features you use, the more you'll appreciate the STANDARD PRINT FILE!

Manipulating Top And Bottom Titles

There will be times that you need the top title centered, or maybe right justified. Or maybe you'll 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! I'll now show you how to manipulate the titles.

The first thing to remember - and the most important thing - is that WHATEVER COMMANDS ARE 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 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.

That is all you need to know in order to select the functions needed for the top titles. Now I'll show you an example of how this is done. I'll use the standard print file I used for this manual as an example! Note that this time I have numbered the lines and broken them up so that I can easily refer to them in the following example.

```
1)  .DA0;.EMOFF;.MF1;.PI3;.PRon;.RE0;.SD3
2)  .SW5;.PNon;.Ceoff;.Ad5
3)   .LL70;.IN0;.COoff;.JUALL;.UP1;.ES!;.HY&ON
4)  .PS$;.WPOFF;.LS;.TM6;.HM2;.BM6;.FM2
5)  .KRON;.KS1
```

```
6) .TFmodunc2/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;.JU   ON
```

The first thing to note is where the Title Font is selected. Look at line number 6. The command is TF MODUNC2/PR. This selects the title font as modern uncial 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 in 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.

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.

First look at the TR command. It is easy to make any keyboard character into the ASCII 127. Just TRanslate a know 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.

So what? OK, that's a fair question. The HARD SPACE is a character, like a letter but is a blank - it prints nothing. Unlike a space, a blank MUST be printed. If a space falls at the end of a line, it is dropped and the next word is printed in the first space if the next line. If a BLANK falls at the end of a line - it is printed as if it were a letter. Generally, a BLANK is about one half the width of a letterset frame. SPACES can vary in width - that is why we have the SW command. BLANKS are always the same size for a specified letterset. Understand that? To summarize: a space is an optional, variable-sized blank. A BLANK is always printed and is always the same size.

Not to beat a dead horse - but by way of further explanation, if you had a phrase or name in a text, separated by spaces, it could be broken up into parts by the program. The phrase might start on one line and end on the next. If you used blanks instead of spaces, the print out would look the same but it would be ALL ON ONE LINE since the blanks are treated as letters and the phrase or name would look like one long word to the program.

Once you understand that, you can next tackle the second command, CONCATENATION. 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.

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

```
1)  .DA0;.EMOFF;.MF1;.PI3;.PRon;.RE0;.SD3
2)  .SW5;.PNon;.Ceoff;.Ad5
3)   .LL70;.IN0;.COoff;.JUALL;.UP1;.ES!;.HY&ON
4)  .PS$;.WPOFF;.LS;.TM6;.HM2;.BM6;.FM2
5)  .KRON;.KS1
6)  .TFmodunc2/pr
7) .TR 35,35,92
8) .TT # Dot#Writer#4.0 Page#$
9) .BT # Copyright#1984#by#W.K.#Mason #
10).coon .VT6
11).TR0
12).FOON;.UP32;.ceoff;.PL66;.CB1;.LH7;.DSOFF;.VLOFF;.JU   ON
```

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, I have 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. 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 DOTPRINT. 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 (FOON), the justification must be on (.JUALL) and the concatenation must be off (.COOFF). 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!

RUNNING DOTPRINT

I've shown you how to create a file for DOTPRINT. In the next section, you'll be able to look up and learn the MANY commands we have made available for you. But, I haven't yet described what happens when DOTPRINT runs. Actually, it is rather simple as far as the user is concerned.

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.

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.

If you are using single sheets and want to stop after each page so you can change sheets, or if you are using a DMP printer and are printing in 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 that you 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.

Want more than one print of a file? 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.

Finally, you'll be asked for the name of the file you want to print. Just type the name and hit ENTER - the rest is automatic!

When the file is finished, you'll be asked if you want to move to the top of the next page. "No" ends the printing right there. "Yes" moves the page to the top of the next form.

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 NOT PRINT IT, 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!

Copyright 1984 by W.K. Mason

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 DOTPRINT, 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 DOTPRINT 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 and press ENTER. When asked to enter a command, type a line of dot commands, each starting with the period. Then, press "ENTER" and DOTPRINT will continue running. This method is the same as "Stopping DOTPRINT", below.

STOPPING DOTPRINT

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

ENTER COMMAND FROM KEYBOARD.

Release BREAK immediately. Then, YOU MAY ENTER ANY VALID COMMAND, including ENd. If you are printing a bottom title, the title will print and then the program will go to the termination menu. If you are not printing titles, printing will stop and you'll get the menu.

Suppose you turned on the EM command but forgot to turn it off. Just hold the SHIFT key and when prompted, enter EMOFF. The program will continue to print but with the EM turned off. The same is true for all DOTPRINT commands or combinations. ANY LEGAL LINE COMMAND can be issued in this fashion. If you stop printing and want to restart where you left off, just hit ENTER or .CM ENTER. If you haven't moved the paper the printing will continue as if it had never stopped.

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 DOTPRINT program. It's the part of the book you'll use again and again, when you need to find out how to do something.

Copyright 1984 by W.K. Mason

Section Two
Command Library

This section describes all commands supported by DOTPRINT 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.Itoh 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 DOTPRINT 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.

DOTPRINT 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 (see also .LM)

This defines the absolute left-hand margin in tenths of an inch. For example, specifying ".AD 10" would give you a left margin of 1 inch. To set a 0 space margin use ".AD 0" Default value at power up is .AD 5 which equals 5 tenths of an inch. ".AD" is obsolete in DOTWRITER, since ".LM" does the same thing, and we suggest you use it instead of ".AD".

.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 modunc2/pr
.af mes/pr
testing !/ testing !? testing
```

will be printed as:

```
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. If the example above had specified ".af mes/pr-1", it would have printed line this:

```
TESTING testinn TESTING
```

Note: This command must be the last command on a line. Also, if an ALTERNATE font is too large to fit with a main font, they can still be used together. First, select the AF font with the BF command, then select the BF font with the AF command. 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. 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.

Notes: 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 Manipulation Utility 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 that the Old English Character Set is stored under. The second example shows how you could use a font you created while using TGEAP.

Copyright 1984 by W.K. Mason

.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 DOTPRINT, and no error message will be given.

.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".

.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 is one of the most interesting in the program. It allows printers that can reverse-feed the paper to print in multiple columns. 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.

Copyright 1984 by W.K. Mason

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.

.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. 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.

.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. Using combinations of ".EM" and ".DA" you can control the darkness of the print. For example:

```
.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" . . . . . ALTER ESCAPE CODE

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 test 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 . . *

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. Short version of this command is .JU R.

JU ALL is a new command and is used in 4.0 only. 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, regardless of the number of words in it, will be spread out to fill the entire. 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.

.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 Unical:

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

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. If ".KS" is left at zero, there are times when certain characters will actually print in each others space if KRON has been specified. To prevent this rare but possible circumstance, we recommend that KS always be set to at least 1 if KRON has been specified. In the example below, with KRON and KS at 0, the following symbols: "C = T - X" overlap:

C=T-X

But, when KS is set to 1, the result is readable:

C=T-X

Again, we suggest that the KS always be set to 1 or more when KR is on. We set it to "1" for most of this book, and to "2" when we used italics.

.LH n . . . . . LINE HEIGHT . . 6

To specify spacing between lines, use ".LH n", where "n" is a positive integer and specifies the number of dots to space down before beginning the next line. If "n" is small enough, the lines will overlap. To restore spacing to normal specify ".LH 12", which is the default. The LH command gives varied results depending on the style of type printed. Experiment for the results you like best.

.LL n . . . . . LINE LENGTH

This command is used to set the line length. In conjunction with the ".ad" 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"

.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". There is a maximum value for "n", which is determined by the particular font. If the value is exceeded, no magnification will be done.

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).

.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 DOPTRINT. 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.

.PA . . . . . PAGE EJECT

This command causes the page to be ejected and printing to start on the next page.

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

This lets you select a "Page Range" to be printed. It's a great time-saver when you have to reprint some pages in the middle of a long document, but don't want to have to start printing from the first page. The entire document must be processed and formatted by DOTWRITER, but until the first page you've selected, "p1" is encountered, nothing will be sent to the printer. Then, pages "p1" through "p2" will be printed, and then you'll be asked whether you want to enter another page range (higher than the first). If so, type in two more numbers, without the ".pg". If you want to reprint just one page, give its number as the starting and ending page.

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

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

This command is used to set the number of dots per line when using the graphics lettersets. It doesn't work on the Epson.

C.Itoh	R/S DMP
.PI 0 is 640 dots/line	.PI 0 is 480 dots/line
.PI 1 is 768 dots/line	.PI 1 is 576 dots/line
.PI 2 is 1088 dots/line	.PI 2 is 576 dots/line (DMP 500 only)
.PI 3 is 1280 dots/line	.PI 3 is 800 dots/line

The default value is 1088 dots/line. You can print titles with one pitch (the pitch in effect when the .TF command is encountered) and regular text in a different pitch. For example:

```
.pi0
testing
.pi1
testing
.pi2
testing
.pi3
testing
```

would be printed as:

testing
testing
testing
testing

.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. For example: ".PL 60 would correspond to 10" 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
.pr on
testing proportional print
```

will be printed as:

```
testi ng  proportional  pri nt
testing proportional print
```

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

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.

Copyright 1984 by W.K. Mason

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
```

Copyright 1984 by W.K. Mason

```
          .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.

Copyright 1984 by W.K. Mason

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!

.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".

.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)

This is a straight forward command. SK means skip and "n" is any integer. ".SK 5" will skip 5 lines.

.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.

.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 @ 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 @ 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.

.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 DOTPRINT 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 DOTPRINT.

It is suggested 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 . . *

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:

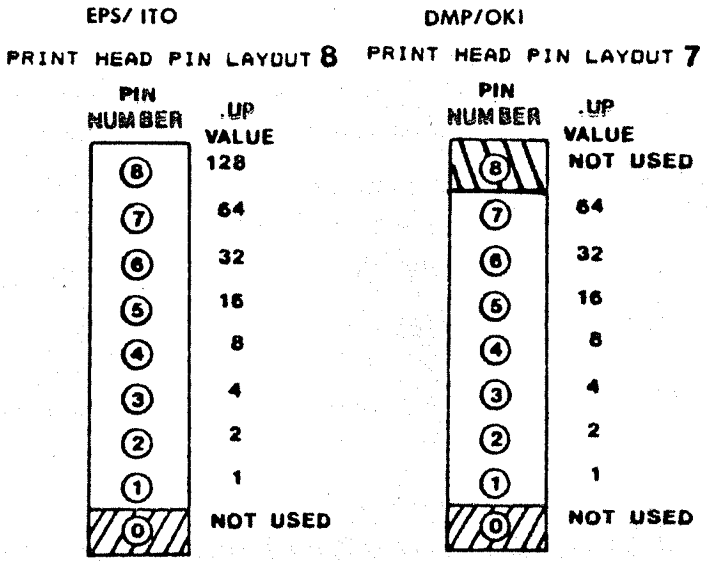
testing testing testing testing

testing testing testing testing

testing testing testing testing

testing testing testing testing

The default value is ".UP 1".

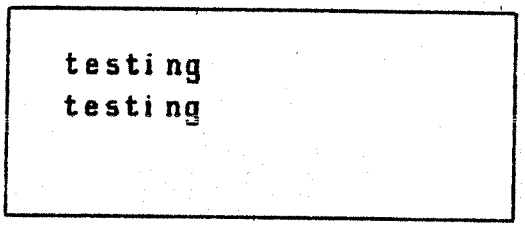


Copyright 1984 by W.K. Mason

.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 insure 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;.ll70;.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

Same as the .MX 100 command, but for the C.Itoh printer.

BLOCK GRAPHICS

Block graphics can be used with DOTPRINT, even with GRAFTRAX-PLUS. 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, 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 DOTPRINT 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.

Copyright 1984 by W.K. Mason

APPENDIX A

THE INITIAL PAGE

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

1 . . . If the first command in the file was ".LS n", then the computer assumes that the paper has been moved by hand "n" spaces from the top. The computer does not print a top title, but starts right in with printing text.

2 . . . If any line feeds have been given (by an .SK command, for example) before the first line of text, then the computer does not print a top title, but starts printing text.

3 . . . If no line feeds have been given before the first line of text, then the computer checks to see if it has received a top title (by the .TT command). If it has, then the top title is printed and printing of text begins at the top margin.

4 . . . If no line feeds have been given and no top title has been given before the first line of text, then the computer spaces down to the top margin and begins printing text. It does not print a top title.

Copyright 1984 by W.K. Mason

APPENDIX B

USING SCRIPSIT AS AN EDITOR

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.

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 procedure. 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.

Super Scripsit requires that ASCII files be generated in a manner different from regular Scripsit. Read the Super Scripsit manual for details: we have no direct experience with that product.

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. The computer defaults to the standard printer font.

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.

ERROR! ERROR CODE="number" ... the computer does not recognize the command, and pauses. It prints out the code determined by the BASIC ERR/2+1 command, and pauses. You can look up the error code in your operating system manual.

ILLEGAL DOT COMMAND .... the computer encountered a line consisting of a single dot with nothing after it. The computer ignores this line.

FILE TOO BIG .... the computer tried to read a record number 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..

Copyright 1984 by W.K. Mason

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.

Copyright 1984 by W.K. Mason

This page intentionally left blank

INDEX

AD 24
AF 24
ALLWRITE 21
alternate font 24, 31
AP 25
appended files 13, 25

banners 36
begin font 25
BF 25
big characters 36
big files 13
black background 43
blank lines 43
blank size 44
blanks and spaces 17
block graphics 49
BM 26
boldface 29
borders 27
bottom margin 26, 31
BR 26
break in text 26

cancelling printing 21
CB 26
CC 27
CE 27
centering 27
chained files 13, 25
changing period 29
character set selection 25
character substitution 16, 46
CM 28
CO 28
columns 26, 27
command word symbol 29
comments 28
concatenation 17, 28
conditional new column 27
conditional new page 29
controlling printer 44
CP 29
CPI 6
CW 29

DA 29
dark 29
default 5
delayed indent 37
Designing lettersets 3
dot commands 4
DOT commands 5
dot spacing 43
DOT/CMD 2
DOTPRINT 2, 4
dots per line 38
double spacing 29
double-width 31
doublestrike 29
drawing lines 32, 48
DS 29

EM 30
emphasis marks 31
emphasized 29
Emphasized 30
EN 30
end of file 30
errors 52
ES 30
escape sequences 31, 44
Escape symbol 30
exclamation mark 30

file creation 5
file requirements 4
first page 50
FM 31
FO 31
font selection 25
footing margin 31
form letters 40
formatting 31, 34
full lines 17

gap 44
graphics 49

hanging indent 37
hard space 16
heading margin 32
HL 32

Copyright 1984 by W.K. Mason

HM 32
horizontal line 32
HY 32
hyphenation 32

IG 33
ignore text 33
IM 33
imbed file 33
imbedded files 12
IN 33
indentation 33, 37
indexing 34
IX 34

joining lines 28
JU 34
justification 31, 34

KE 34
KERNING 6
kerning 35
keyboard entry 34
KR 35
KS 35

large characters 36
left margin 24, 33, 36, 37
letterhead 36
letterset disks 3
Letterset Utilities 3
LH 35
line height 35
line length 5, 35
line spacing 29, 35
lines 28, 32, 48
lines between columns 27
lines per page 38
LL 35
LM 36
logo space 36
long files 25
LS 36
LSMU 3

magnification 36
mailing list 40
margin 24, 26, 31, 32, 33, 34, 35, 36, 37, 45
messages 52
MF 36
micro-spacing 43
missing files 20
mixed fonts 24
multiple columns 26, 27
MX command 36

names and addresses 40
NEC8023A 44
new line 26
new page 29, 37
numbered pages 38

OF 37
offset 37
overstrike 29
overview 4

PA 37
page 29, 37, 50
page length 38
page numbering 38, 40
page range 20, 38
page symbol 40
paper size 38
paper width 36, 48
paragraph 26, 39, 43
period 29
PG 38
PI 38
pitch 38
PL 38
PN 38
PP 39
PR 39
pre-printed letterhead 36
primary font 25
printer control 44
proportion 6
proportional printing 39
PS 40

RD 40
RE 43
reverse image 43
reverse paper movement 48
right margin 31, 35
run-time options 21
running DOTPRINT 19

screen display 5
SCRIPSIT 51
SD 43
second font 24
selected pages 20, 38
short lines 17
SK 43
skip lines 43
skipping text 33
smooth margin 31, 34

Copyright 1984 by W.K. Mason

soft hyphen 32
space character width 44
spaces and blanks 17
spacing 29, 35
spacing between letters 43
special symbols 44, 46
ST 44
standard files 12, 14
stop for operator action 44
stopping DOTPRINT 21
substitution 46
SW 44
SY 44
symbols 46
symbols to printer 44

tab 48
table of contents 45
TC 45
TCONINX/CMD 2
text file 5
TF 45
title 46
title font 45
titles 15
TM 45
top margin 45
top of form 21
top title 46
TR 46
translation 16, 46
TT 46

underline 47
underlining 31
uni-directional 44
UP 47

vertical line 48
vertical tab 48
VL 48
VT 48

white on black 43
width 36
width of paper 48
width of space 44
WP 48

Copyright 1984 by W.K. Mason