WordMaster II

PolyMorphic
Systems

460 Ward Drive   Santa Barbara   California 93111   (805) 967-2351

LIMITED WARRANTY AND LIMIT OF LIABILITY

Interactive Products Corporation (dba PolyMorphic Systems) makes No Warranty, express or implied, concerning the applicability of this program to any specific purpose. It is solely the purchaser's responsibility to determine its suitability for a particular purpose. Interactive Products Corporation accepts no liability for loss or damage resulting from the use of this software beyond refunding the original purchase price.

THIS STATEMENT OF LIMITED LIABILITY IS IN LIEU OF ALL OTHER WARRANTIES OR GUARANTEES, EXPRESS OR IMPLIED, INCLUDING WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

NOTE: This manual is part number 81Ø179, and describes the programs contained in the WordMaster II System Disk. The part numbers for the WordMaster II System Disks are as follows:

SSSD 5" Single User	Part Number	82Ø235
DSDD 5" Single User	Part Number	82Ø236
SSDD 8" Single User	Part Number	82Ø237
DSDD 5" TwinSystem	Part Number	82Ø247
SSDD 8" TwinSystem	Part Number	82Ø246

This manual was written by Jennifer Douglas.

The WordMaster II operating system (WPS) was designed and written by Brian Smith. Special thanks for design suggestions go to Jennifer Douglas, Larry Deran, and Frank Weiss.

The original version 2.1 of the System 88 Formatter was written by Robin C. Soto.

Version 4.Ø of the System 88 Formatter was designed and written by Frank Weiss and revised by Jerry Whitnell.

The System 88 Editor was written by Glenn McComb.

INTRODUCTION

WordMaster II is the PolyMorphic Systems complete word processing system. It is designed to be used in all sizes and kinds of office environments. WordMaster II can be run on any System 88 configuration, from the low-cost single-drive Double Density System 881Ø to the multi-drive TwinSystem 88. This manual and the special form and menu presentation of WordMaster II make it possible for those who are completely unfamiliar with the System 88, or any computer, to learn to do word processing after one or two training sessions.

There are essentially four procedures that the word processing system includes:

writing and changing documents;

typing documents literally;

defining formatting environments and formatting documents;

storage and management of files and disks;

In order to provide complete instructions for System 88 word processing we have divided the manual into the following sections.

Section I: System Start-up Procedures:

This section explains how to power up the system, insert the system disk, and load the contents of the operating system into memory.

Section II: Overview of Operation Procedures:

This section explains how to start up the word processing system. There is also a way to cause the word processing system to be run automatically when the system disk contents are loaded into memory. This automatic start-up of the word processing system is helpful if a dedicated word processing installation is desired. Also included in this section is a simple demonstration of how to create and then change a writing file. Background information is provided about storage of files when using the System 88, and finally the Storage Manager Form, which is involved in most of the word processing options, is introduced. This form is an important aspect of the word processing system. It is used to list the contents of the disk and to select files for any of the word processing functions. It is not used only with the Storage Management Functions option as its name might suggest.

Section III: Storage Management Functions:

There are many functions involving disk initialization; disk and file back up, and renaming, deleting and undeleting files which are associated with word processing. These functions are explained in this section.

Section IV: Writing or Changing a Document:

This section explains the procedures for naming and creating files and sub-directories. Then the procedures for re-opening a file which is located in either a main or a sub-directory are explained. Directions for use of the special writing features like Global Search and Replace; duplication, transferring, or deletion of marked blocks; creating or adding to Key Definition Libraries; and combining or dividing files are included in this section.

Section V: Printer Functions:

This section shows how to use the Printer Functions sub-menu options.

Section VI: Typing a Document Literally:

This section explains how to obtain a literal printed copy of a file. These files are useful when you want to obtain a unformatted copy of your file, embedded commands and all.

Section VII: Formatting a Document:

This is a general introduction to formatter operation. It describes the formatter's start-up state and tells how that state can be modified by defining and loading new environments into memory, and by embedding commands in the text. Procedures for formatting long documents (multi-file) and for only formatting a part of a document are included in this section.

Section VIII: Summary of Formatter Commands:

Once you are familiar with the general formatting procedures explained in Section VII, you can use this command summary and information section as a daily reference guide. It explains the action and placement of each command and tells which other command or commands can be used to cancel the action of a particular command.

NOTE: This manual is a preliminary release. The final release will be a proofed and polished version of this manual with a table of contents, spiral binding, and cover and divider sheets.

PREFACE A

PRINTER INSTALLATION

We recommend that dealers use section 13 of the System 88 User's Manual as a guide for setting up the printer that is to be used for the word processing installation. This is an end-user manual which assumes that the printer has already been defined to the System 88 Printer Driver, that it has been made the "DEFAULT" printer, and has been physically attached to the System 88. If the user does have to perform these procedures himself, he should read the Section 13 of the System 88 User's Manual.

PREFACE B

DIFFERENCES BETWEEN WORDMASTER I AND WORDMASTER II

WORDMASTER OPERATING SYSTEM

The special WordMaster Operating System (WPS) is a new addition to this product. This manual describes its operation. All WordMaster programs and all relevant System 88 functions such as IMAGE, INIT, PACK, COPY, and DELETE are invoked by and then return to the word processing operating system. The operation of word processing programs from Exec is still very much the same as it was with WordMaster I. The differences are outlined in this preface. Anyone who is not entirely familiar with the System 88 and the operation of WordMaster I should not be concerned with this preface, but should read this manual and learn to operate the new word processing system.

Of all of the word processing functions, the one that is the most changed is the formatter. Those changes are listed below. Items numbered 1-3 of the list of changes are pertinent to those who want to use Exec to run the formatter. The rest are changes to and additions to the formatter's capabilities, additions to the editor's features, and changes to the printer driver. If WPS is to be used to run the word processing system, you must read the manual, as it involves too many enhancements to be itemized effectively here.

1. The biggest over-all change to the formatter is its start-up state and what it does right after it is started up. The formatter's start-up state which is determined by internal commands and defaults now insures that attractive documents result from the formatting process even if there are no embedded commands or "environment" files. This start-up state (the start-up modes for all of the formatter's functions) is described in Section 7 of this manual. After the formatter is started up, and before it looks for a file to be formatted, it looks on the system disk for a file called FORMAT.IN. If it finds no FORMAT.IN file it looks at the first file in the command line. The command line can now contain as many file names as will fit on one screen line after the FORMAT command. For example:

   ```
   $FORMAT <2<FILE-A,<2<FILE-B,<2<FILE-C,<2<FILE-D<2<INDEX
   ```

   Commands can be given to alter this start-up state. The formatter consists of a number of text arrangement functions which operate simultaneously. Some of these functions are usually expected to remain in one state for the entire document. Others may change frequently at specific points in the document. The functions which remain in one mode are referred to as long-term functions and the commands which cause these functions to be set in one mode or another are called long-term commands and can be placed in "environment" files. These long-term commands are equivalent to the commands which were referred to as "top-level" commands in the WordMaster I manual. The WordMaster I manual instructed the user to place top-level commands (width, length, margin, line spacing etc,) at the top of each text file. In this release we advise that the long-term commands be placed in environment files. The commands which affect specific parts of the document are embedded in the writing files themselves.

The Summary of Formatter Commands contains all commands understood by the formatter and indicates which commands are long-term. Commands which are used as long-term commands in the FORMAT.IN file can also be used within the text files themselves, but we added the FORMAT.IN feature to minimize the number of commands needed in individual document files and to make them independent of their environment file. This independence makes it possible to change the over-all look of a document without having to edit any of its text files.

2. The term "environment file" is new with this release and it is used to describe collections of long-term formatter commands. The functions which are usually controlled by long-term commands are: margins, line spacing, length, width, error reporting, printer type, indent value, pause (whether or not the printer stops at the end of each page) and characters per inch. If the formatter's start-up modes for these functions are acceptable then no commands need to be placed in environment files. If changes need to be made to any of these functions then a FORMAT.IN file can be created to contain the commands which will affect the needed changes. The formatter automatically looks for this FORMAT.IN file when it is invoked, and it reinitializes its start-up mode accordingly.

   The user needs to determine whether or not all of his documents should have the same environment or whether a different environment is needed for each type of document that is being formatted. The recommended procedure for when and how to create environment files to be used with Exec is as follows:

   > Create a FORMAT.IN file which contains those top-level commands which are needed to alter the start-up mode and which should be in effect for every document that is formatted. Create the FORMAT.IN file by typing the following after the $ Exec prompt.

```
EDIT FORMAT.IN
```

(The FORMAT.IN file must be placed in the main directory of the system disk.)

Create other environment files which are appropriate for particular types of documents. Create these files by typing the following after the $ Exec prompt:

```
EDIT <2<environment-name
```

Environment files only contain commands which are not included in the FORMAT.IN file, are given regular .TX extensions by the system, and can be stored in any directory of any disk in the system.)

Environment files which contain additional modifications to the start-up state (beyond those in the FORMAT.IN file) should be the first file in the FORMAT command line. Below is an example of a FORMAT command line with an environment file as its first file:

```
FORMAT <2<environment-name,<2<File-1,<2<File-2
```

When the formatter is invoked, it will look first for the FORMAT.IN file, re-initialize its start-up state accordingly, and then begin to process the first file in the command line. If that first file contains long-term commands, then it will modify its initial state accordingly and proceed to format the second file in the command line which is actually the first document file.

You can get a good idea of what types of commands should be placed in environment files (FORMAT.IN and otherwise) by reading sections 7 and 8 of this manual.

3. When the formatter is run from Exec, the printer driver parameters for page length, width, and left bias are in effect. There is a control for left bias in the Environment Form, but this form is not in effect when Exec is used to invoke the formatter. Since there is no embedded command for left bias, it is always controlled by the printer driver when the formatter is run from Exec. When Exec is used the printer driver parameters for lines per page and characters per line are also in effect. There are embedded commands for these parameters, however, so they can be over-ridden from the environment or text files.

4. All bugs known to exist in the WordMaster I release have been eliminated.

5. You can now specify multiple line headers and footers. The method for specifying headers and footers is different. See the Summary of Formatter Commands.

6. The header and footers that are defined in the Environment

menu, the FORMAT.IN file or at the top of the file can be turned off and back on again by the no header and start header commands.

7. You can now format a part of a document. See the Format a Document section. If you are using Exec, see the command summary for the commands which allow you to format a part of a document.

8. Underline works differently. See the Summary of Formatter Commands.

9. There are new word fill and word justification commands. Reading the entire section in the command summary on word fill and justification commands is a must.

1Ø. There are new text appearance commands that cause bold type, superscript, subscript and red and black type. See the Text appearance section of the command summary for specifics about these commands.

11. The "word undent" feature is new. It allows paragraph headings to extend out to the left of filled blocks of text. See the command summary for specifics on this command.

12. The center command works differently and is covered under the word justification section of the command summary.

14. There is now a printer type command which takes one of three arguments. We have classed most popular printers into one of three categories. WordMaster II must know what category your printer is in so it can do some of the special case modes. This command is handled by the Environment Form if WPS is being used and can be placed in the FORMAT.IN file if Exec is being used.

14. {pno} is the new command for ## (put page number here.)

15. ~ is the new special symbol for @ (non-expandable blank.)

16. There is a much improved method for reporting errors to the screen and/or printed text.

17. The functions which were formerly controlled by the "Formatting options" are now controlled by the Environment Form, or by the FORMAT.IN file if Exec is being used.

18. A blank line in the input to the formatter causes the formatter to leave a blank line in the output. This minimizes the need for {skp}.

Changes and Enhancements to the Editor:

19. WordMaster II supports the sub-directory capability. This manual contains a complete explanation of the use of the sub-directory structure when writing and changing documents.

20. The System 88 Editor is referred to as a part of the "Write or change a document" function in this manual.

21. There is a new Key Definition Library feature that allows you to save ESC key definitions. See the "Write or Change a Document" section.

22. A completely interactive Global Search and Replace is a new System 88 Editor feature.

23. The Editor now contains warning messages that tell you that there is still more text on your disk that has not been copied into memory. These messages appear when you use the down arrow when there is no more text in memory but there is more text on the disk, and when you type the ESC CTRL-I sequence when you have not yet brought all of the text file into memory.

24. The ESC CTRL-Y sequence now aborts special writing functions rather than the Editor itself. This allows you to terminate Global Search and Replace and other special functions without aborting the writing function itself.

25. ESC CTRL-X is the new Editor abort sequence which causes the system to return to WPS or Exec without recording the text in memory on disk.

26. If Exec is being used to invoke the editor and it is accidentally aborted, the REENTER command will return the system to the editor with its cursor positioned at the same point in the text, and the file still intact in memory.

27. If the system is accidentally re-booted, or if re-booting is necessary to bring the system out of a "hung" state when the TwinSystem is being used, the following sequence of commands can be used from Exec to restore the system to the editor with the file still in-tact and the cursor positioned where it was when the re-booting occurred.

    ```
    $GET Edit
    $REENTER
    ```

    If WPS was used to invoke the writing function (editor) before the re-booting took place and the above sequence is typed, the system will return to EXEC when the ESC CTRL-E sequence is typed. Then WPS will have to be started back up again. If WPS is an INITIAL file, the above sequence can not work as the automatic loading of WPS when the re-booting occurs wipes the text file out of memory.

Printer Driver Changes

28. The printer driver has been changed. This change occurred in the release of Version 4 of the operating system software. At that time the Printer SET and Printer SHOW commands were added and the method for defining printers and establishing one as the default printer were changed slightly. This version of the printer driver is described in the current version of the System 88 User's Manual. We don't discuss these changes in detail in this manual because the user of WPS does not directly encounter the printer driver. The printer driver which was released with Version 4 of the operating system software has been through one minor revision (from Printer 37 to Printer 38.) This revision has introduced an incompatility. Do not invoke your word processing programs from any other system disk as you may have an incompatible version of the printer driver on the other disk. The word processing programs should always be invoked by WPS or by the version of Exec which is on the WordMaster disk when it is shipped. Do not transfer individual programs from one disk to another.

General Changes:

29. The WordMaster Disk operating system allows full use of the storage capacity of any disk format allowed in any configuration of the System 88. It is supported fully by the TwinSystem 88.

3Ø. In general, the integration of all of the word processing related programs into a package with the word processing system at its center has resulted in a product which is easier to learn, easier to use, and far more powerful. Many capabilities and safeguards have been introduced because of the word processing system designed for the WordMaster II release. We encourage all WordMaster II users to read this manual and learn how to use this new system.

PREFACE C

PREFACE TO USERS OF KEYBOARD II

The Keyboard II is no longer sold by PolyMorphic Systems. Therefore this manual assumes that all users have Keyboard III. If you have Keyboard II, you need to know the following.

The four function keys referred to in this manual have equivalents on the Keyboard II.

The ESC H sequence can be substituted for striking Function Key I which invokes the "Help" displays for the word processing system.

The ESC S sequence can be substituted for striking Function Key II which invokes the file select process.

The ESC A sequence can be substituted for striking Function Key III which invokes the process which adds a sub-directory name to the "Current directory specifier" field.

The ESC CTRL-E sequence can be substituted for striking Function Key IV which usually means "leave this display and go on to the next procedure or return to the main menu".

SYSTEM START UP

TURNING ON THE SYSTEM

The first procedure you need to perform is that of powering up your system. This is done **BEFORE** the disks are inserted.

> **WARNING: If you power your system up or down while your disks are in the drives and the drive doors are shut, you may damage the information on your disks. Insert disks after powering up the system and open the drive doors before powering down the system.**

System 8813

Insert the key into the ON-OFF lock on the front panel, press in on it, and turn it to the right until it stops. The red light above the lock is now lit, to indicate that the machine has been turned on. (To turn off the machine, slowly turn the key back to OFF).

If you have an 88/MS add-on unit attached to your 8813, you must power it up as well. Reach around to the back of the system to the ON/OFF switch on the left side (your left as you face the system) of the black power supply box; turn it on.

System 881Ø

The System 881Ø has no lock; instead, it has an ON/OFF switch on the left front of the system. Turn on the system with this switch.

Printer

Make sure your printer is on before you insert your system disk as your printer must be on in-order for the system and the printer to communicate during the load process.

ABOUT THE DISKS

To do word processing on your System 88 you will be using a system disk and, if you have more than one drive, you will most likely want to use one or more document storage disks.

A system disk contains the System 88 Operating System and the System 88 word processing programs. The disk labeled "WordMaster

II" is the system disk to use for word processing on Single-User systems. The disk labeled TwinSystem 88 WordMaster II disk is the disk to use for word processing on TwinSystems.

The Storage Management Functions section of this manual tells how to make back-ups of the system disk and how to create document file storage disks to be used in multi-drive systems. If you have an 881Ø, you will want to create several copies of the WordMaster System Disk, because your system disk must double as a storage disk as well. Continue to follow the directions in this and the next section. Then, when you reach section 3 you will learn how to create back-up and storage disks.

DISK WRITE-PROTECT TAB AND NOTCH

Each disk has a write-protect notch. On a 5" disk it is the rectangular notch. On an 8" disk it is the larger of the three round notches located on the opposite edge from the disk label. It is sometimes desirable to write-protect a disk in order to prevent the altering of information on the disk. Every box of new disks comes with a supply of silver or white write-protect tabs. A 5" disk becomes write-protected when the silver write-protect tab is placed over the write-protect notch. An 8" disk is write-protected when the write-protect tab is removed from the write-protect notch. In order to follow the instructions in the next section of this manual, you can not have the system disk write-protected. Be prepared to write-protect it when we tell you to in the Storage Management Functions section.

DISK INSERTION

When you have turned on the system you can insert your WordMaster or TwinSystem Disk which contains the System 88 Operating System and all of the programs necessary for word processing.

If you have a System 881Ø, insert the system disk in your drive.

If you have a System 8813 insert the system disk in drive 1. Drive 1 is the left-most drive.

If you have a System 8813 with an 88/MS add on unit, insert the system disk in drive 4. Drive 4 if the drive on the left in the 88/MS.

> NOTE: The drives we point out above as the correct drive to hold the system disk are also referred to as "system residence drives" since those are the drives which are the assumed residence of the system disk. The system always looks there for operating system programs. If, however, there is a reason to use a different drive as the system drive, that can be done. If it finds no disk in the system residence drive, the system will look in each drive for the operating system

> disk. This is just an added bit of information; don't worry about it if you don't want to!

The procedure for disk insertion involves taking care not to harm the disk while inserting it correctly into the drive.

Slide the disk out of its paper envelope. Hold the disk lightly, being careful not to touch the exposed sections of the disk surface. If you are inserting your disk into a vertical drive, hold the disk vertically so that the disk label is on the left side of the disk and in the upper corner. The square notch in the edge of the disk cover will be on the bottom edge of the disk. If you are inserting your disk into a horizontal drive, put the disk in with the label up and the round notches facing the rear of the drive.

Open the disk drive door and carefully slide the disk all the way into the slot. Close the door. The drive cannot write or read information to or from the disk if the doors are open!

LOADING YOUR OPERATING SYSTEM PROGRAMS INTO MEMORY

Now that your system is on and the system disk is correctly inserted, you are ready to load the contents of the system disk into memory. To do this, press in and release the "Load" button which is located next to the ON-OFF lock on the front panel of your System 88.

Next you will see the following message on the screen:

```
Exec/83- Top of RAM is FFFF
$
```

In the next section we tell you how to start up the word processing system. There is also a procedure which causes the word processing system to be loaded automatically when the load button is pushed. That method is explained also.

WORDMASTER II: AN OVERVIEW OF OPERATING PROCEDURES

The WordMaster II System Disk makes the System 88 a powerful word processor which can be used by individuals who are completely unfamiliar with computers.

In order to comprehend the demonstrations and instructions presented in this manual, we expect you to be working with WordMaster II. This means that you should follow the start-up procedures in the preceding section. If you have followed those instructions, your WordMaster II System Disk is properly inserted in the system drive and its contents have been loaded into memory. Now you should make sure that you have a new, blank disk nearby. A blank disk and this manual and about 4 hours of your time are all you need to grasp the fundamental concepts and procedures involved in word processing on the System 88.

In this manual, you will learn how to operate the WordMaster functions which enable you to write and change documents, and to format them. The formatting process results in attractive documents which contain all of the special text arrangements you need to establish clarity and effective organization. There are numerous powerful tools which this word processing system makes available to you, that speed up the writing, organizing, proofing and printing of documents. We explain all of these tools in this manual. This first section of the manual is intended to give you an overview of the operating procedures involved in System 88 word processing.

Then, the subsequent sections explain each of the functions that you find on the WordMaster Main Function Menu and its related sub-menus.

Now let's begin.

STARTING WPS

Look at your screen. You should see the Exec $ prompt. Type the following exactly as it appears:

```
     WPS
```

Follow the above entry with a RETURN (strike the RETURN key.)

Once it has been invoked, WPS takes over all communication between the user and the System 88, allowing the user to operate the word processing functions by choosing options, responding to questions, and filling in blanks.

The following menu is displayed:

```
                         WordMaster II
                     WPS Version 1.0  x/xx/80
                        Main Function Menu

     Write or change a document.
     Format a document.
     Partially format a document.
     Type a document literally.
     Load a predefined environment.
     Define or view environment.
     Storage management functions.
     Printer functions.
     Quit this program.
```

The System 88 has many other capabilities besides word processing, but if you want to use your WordMaster System Disk for word processing only, we recommend that a simple, one-step procedure be followed which will cause the word processing system to be invoked automatically every time the WordMaster System Disk is inserted and the Load button is pushed.

Here is that procedure:

Type a Q followed by a RETURN. This is the first letter of the "Quit this program" option which is the last option on the menu that is on your screen. Even though we just started up the word processing system we are going to end its operation momentarily so that you can avoid ever having to start it up manually again.

When the Exec $ prompt appears, type the following after it.

```
     RENAME WPS INITIAL
```

The system will tell you that WPS has been renamed to INITIAL.

Now, push the Load button.

The word processing system is now preparing to take over the communication between you and the system. You will see the same WordMaster II Main Function Menu that you saw when you manually started up WPS. From now on, this menu will appear, automatically, after the Load button has been pushed.

TwinSystem Note: In order to make WPS load automatically when the Load button is pushed, perform the above procedure, unless only one user wants WPS to be loaded automatically. If that is the case, use the above procedure with one of the following modifications,depending on which user wants WPS to be loaded automatically.

RENAME WPS INITIAL1 or,

RENAME WPS INITIAL2

WRITING YOUR FIRST DOCUMENT

You are now going to write your first WordMaster document. This initial exercise is meant to give you a grasp of what its like to work with the menus and forms you will be using. Don't worry if you don't fully comprehend some of the terms or explanations; every aspect of the following demonstration is explained more fully in later pages of this section.

In order to operate WordMaster, you must select options from menus and fill in blanks on screen displays which we call "forms." Below we explain how to operate menus and introduce you to the use of the Storage Manager Form. ,Later in this section, we provide detailed instructions about the purpose and operation of the Storage Manager Form.

SELECT THE WRITE OR CHANGE A DOCUMENT OPTION

A menu is a list of options you may select.

To select options on the WPS menus, type the first letter of the option followed by a RETURN. Striking the RETURN key indicates that you are through typing your selection and the system should process it. Typing the first letter, the first few letters, or the whole function name in upper or lower case is acceptable.

Select the "Write or change a document" option by typing a "W" or a "w" and then strike the RETURN key.

If an "illegal" (incomprehensible to WPS) entry is made, the following message appears:

This is not a valid choice.

If this message appears, merely wait for the "Enter your choice" prompt to return to the screen and type the first letter of the desired option again.

NAMING YOUR DOCUMENT FILE

When you have selected the "Write or change a document" option the following form appears on the screen:

```
         PolyMorphic Systems Word Processor Storage Manager

     There are   9ØØ sectors free
     Current Directory Specifier <1

WP.DX            X






     ESC S   Function Key II to select file to edit.
     ESC H   Function Key I for help.
```

The purpose of this form is to show you the contents of any disk in the system. Right now you are looking at the contents of your system disk. The contents of any disk are revealed by the directories for that disk and the storage manager displays one directory at a time. There can be more than one directory on a disk and there can be more than one disk in some configurations of the System 88. The Storage Manager Form, which is made up of windows and labels shows you which directory is being displayed and it shows the contents of that directory. "Current directory specifier" is a label for the window (windows are called fields) which shows which directory is being specified. Your "Current directory specifier" field should show a <1 or a <4 depending on which drive contains your system disk. We will explain all the details about selecting directories for display later.

On of the other fields (located below the directory specifier) should have "WP.DX" in it. This is the name of a storage area on your disk. This area, called a sub-directory, contains parts of the word processing system. We are now going to show you how to create a storage area called a "file."

Notice the white rectangular box located near the top of the screen. It is the cursor, and it indicates where your input will appear on the form. At present, it is positioned at the end of the current directory specifier field.

Strike the TAB key. The cursor will jump to the next field which is a file-name field containing the name "WP.DX." Strike the TAB key again to move the cursor ahead to a blank file-name field. A file-name field is a type of field which contains the name of a storage area. Notice the X in the above example of the Storage Manager Form. This "X" indicates where our first empty file-name field is. If you have additional file names in the the file-name fields, your first empty field might be in a different spot.

> NOTE: There is a message at the bottom of the screen that says "Function Key I for help." If you strike Function Key I, you will see a special display that tells you how to move the cursor about on the form and other general Storage Manger Form operation instructions. All of these instructions are included in this section, but later on you can use this display as a reminder. It is always accessable when the Storage Manager Form is being displayed.

When your cursor is positioned in the first empty field type the following:

```
SAMPLE-1
```

SELECTING YOUR DOCUMENT FILE

The next step is to tell the system to open a file that is to store document text and to give it the name "SAMPLE-1." Make sure your cursor is still positioned at the end of the name you just entered.

Strike Function Key II. The function keys are the top row of keys in the numeric pad located to the right of the normal keyboard keys.

You then see the following display: (We explain its meaning later in the "Write or change a document" section.

```
Input file: not found
Creating output file:  opened

(Key Definition Library opened)


Hit any key to continue. . .
```

Hit any key to continue as your display requests.

The screen clears and the cursor is positioned in the upper left-hand corner.

TYPING AND STORING YOUR DOCUMENT

Type the following sentence, using the keyboard just like you would a typewriter.

This is the text of my SAMPLE-1 Document.

Now you can tell the system to store your document on disk.

To do this type the following:

ESC CTRL-E
[Strike the ESC key. Hold down the CTRL key while typing the "E" key (either upper or lower case)]

Next you will see the following message:

Exiting...

CHANGING SAMPLE-I

Your document is being written on the disk. Once this process is finished the system returns to the WordMaster Main Function Menu.

In order to see that your new file is actually listed in the directory of the system disk, select the "Write or change a document" option again.

The Storage Manager appears again. This time, "SAMPLE-1.TX" appears in the directory. The ".TX" which has been added to your file name, is an extension, just like the .DX after WP. Extensions indentify the type of a file. .TX stands for text and is automatically appended to all document files. .DX stands for directory and is automatically appended to all sub-directory files. We explain more about extensions, files and sub-directories later.

You can add more text to your file by selecting it for the writing function again. Strike the TAB key to move the cursor from field to field until it is positoned at the end of "SAMPLE-1.TX." Strike Function Key II, and again the display that indicates that your file is being opened appears. Strike any key to continue and you will see the sentence you wrote. Use the right-pointing arrow key to move the cursor to the end of your sentence. Strike the RETURN key. Now type the following sentence:

This is the change I made to my SAMPLE-I document.

Now type the ESC CTRL-E sequence again.

The "Exiting" message appears again while the new text is written on disk.

Again the main function menu is displayed.

You have finished creating a document file, entering text into it, re-opening the same file and changing it. Later, in the "Write or change a document" section, we explain the specifics about typing with the System 88 Keyboard III and you will be creating a longer sample document.

For practice, you can repeat the above instructions and create a new document file, or you can add to your existing file. Next we explain in greater detail how your writing files are stored on disk, how you can organize your files on disk, and how to use the Storage Manager Form to locate and then select a particular file no matter what disk it is on.

USING THE STORAGE MANAGER

Before you begin extensive use of the WordMaster II functions you need to understand more about how files are stored on disk and how the Storage Manager is used to locate them so that they can be selected.

STORAGE OF FILES IN MAIN AND SUB-DIRECTORIES

The System 88 allows storage of text and other data on mini-floppy (5 1/4") and floppy (8") diskettes. A "file" is an area of disk that stores information. One file can contain one whole document, or it can contain a part of a document. You determine how much writing is stored in a file as you did when you created the "SAMPLE-1" file in the previous demonstration.

Files are like folders in a file cabinet. Each is identified by the name the user gives to it, as was the case when you created "SAMPLE-1." Collections of files are grouped together on disks as are folders in file cabinet drawers. The structure that allows this convenient grouping of files is the main disk directory and its branching sub-directories. The Storage Manager displays disk directories so that you can see which disk has what files.

The initialization function, which you will be using later, prepares new, blank disks for use on the System 88. As a result of this function, a directory is placed on the disk. Once the main disk directory exists, you can store files in that directory as you did in the demonstration. Or, you can create a sub-directory which you can use as a sub-grouping of files.

If you plan to store a number of files that all contain letters, you can create a sub-directory called "LETTERS", or any other appropriate name, and store all of your files that contain letters in that sub-directory.

The actual structuring of sub-directories is up to you and your individual storage and organizational needs. However, it is important to note that the Storage Manager form, which displays one directory at a time, contains 21 file-name fields. These fields are like windows which show the names of files contained in a directory. There are 21 of these windows, so there can be 21 files in a directory. However, any one or all of these file-name fields can contain the name of a sub-directory which can also store 21 files or sub-directories. Even though sub-directories are not really files, they are stored just like them, with their names listed in a directory. Therefore, when we speak of the file-name fields in the Storage Manger, we mean the fields which can contain the names of individual files, like your SAMPLE-1 file, and the names of sub-directories which can store groups of files or other sub-directories. Every sub-directory created adds 21 to the total number of files which can be stored on a disk. There is a space limit to the disk in addition to the limit to the number of files in a directory. Usually, however, your files will not be so large that 21 of them will take up an entire disk. This means that use of the sub-directory structure allows you to take advantage of the full storage capability of your disk.

Later in the section called "Writing or changing a document", we show you how and when to create sub-directories. Right now, there is at least one sub-directory (.WP.DX) in your system disk main directory. Soon you will learn how to cause the Storage Manager to view the contents of that sub-directory.

MENU OPTIONS INVOLVING THE STORAGE MANAGER

We already showed that the "Write or change a document" option results in the Storage Manager, but there are other options which can be selected from the main menu that result in this form as well. These options all require that a file be selected to receive some particular action. We list these functions which result in the Storage Manager Form below:

"Write or change a document" When this option is selected, the Storage Manager allows selection of a new or existing file to be opened.

"Format a document" When this option is selected, the Storage Manager displays document file and sub-directory names and allows selection of a file to be formatted.

"Only format a part of a document" When this option is selected, the Storage Manager displays document file and sub-directory names, and allows selection of a file, or files which contain a part of a document to be formatted.

"Type a document literally" when this option is selected, the Storage Manager displays document file and sub-directory names and allows selection of a file to be typed literally.

"Load a pre-defined environment" When this option is selected, the storage manager displays environment files and allows selection of one to be loaded into memory and used in subsequent formatting. We explain the purpose and use of the environment file later on.

"Storage manager functions" When this option is selected, a sub-menu results. The "List a directory (Rename, copy, delete)" option on this menu results in the Storage Manager. It displays files of all "types" and allows selection of a file to be copied, renamed, deleted or undeleted.

In order to learn more about how the Storage Manager operates, select the "Storage management functions" option on the main menu. The following sub-menu results:

```
              PolyMorphic Systems WordMaster II
               WPS Version 1.Ø (release date)
                    Storage Function Menu

     Initialize a diskette.
     Copy an entire diskette.
     Mirror an entire diskette (One drive system).
     Reclaim space on a diskette.
     Single file mirror (One drive systems).
     List a directory (Rename, copy, delete).
     Quit this menu.

Enter your choice.
```

Later in this manual we go through each of the main and sub-menu options, but now we want to explain how to type on the Storage Manager Form and how to use its information to help your word processing efforts. Select the "List a directory (Rename, copy, delete)" function to view the Storage Manager Form. This time the contents of its windows are different and probably somewhat like the following example:

```
             PolyMorphic Systems Storage Manager Form

     There are   9ØØ sectors free
     Current Directory Specifier <1
WP.DX           WPS.GO           CHECKSUM.GO
DUP.GO          MIRROR.GO        FORMAT.GO
TXdef.ED        SAMPLE-I.TX




              Rename, Copy, Delete, or just look.
              Function Key I for help.
              ESC C to copy.
```

FILE TYPES AND THEIR EXTENSIONS

We already looked at the main directory of this disk during the demonstration, but this time there are more files listed. This is because all "types" of files are listed. There are several different types of files; they differ in the way they are created and in their purpose. The three character extension (a period followed by two letters) after a file name indicates its type. The Storage Manager is selective about which files it displays from the specified directory. It always displays sub-directories, because they can contain other files which may be needed for the function being performed. When it is being used as a part of the document writing, typing, or formatting procedures, the Storage Manager shows files with .TX extensions (document files.) When it is being used with the "Load a pre-defined environment file" option it displays files with .FV extensions (environment files.)

Now that the Storage Manager is being shown as a result of selection of the "List a directory (Rename, copy, delete)" sub-menu option, you can see many of the possible extensions. Below we explain what these extensions mean:

> .GO files contain parts of the WordMaster II package and must not be deleted or otherwise altered by the user.
>
> .DX files are really sub-directories which contain files and possibly more sub-directories. The WP.DX sub-directory name appears in your Storage Manager display when the specified directory is the main directory of the system disk. This sub-directory contains the word processing system and must not be changed or deleted. You will soon be adding more sub-directories to your system disk or to a data disk. These sub-directories will contain writing files, maybe more sub-directories and perhaps an environment file which can be used to format the document files which are stored there.
>
> The .ED file is a Key Definition Library. In the "Write or Change a Document" section of this manual we explain how to use this library, how to modify it, and how to create alternative libraries.
>
> The .FV file is an environment file. You can't see this file listed in the main directory, but it is listed in the WP.DX sub-directory. Environment files are created when the "Define or view an environment" option has been selected and the Environment Form is displayed. Environment files establish how a typed literally or formatted document will look. Later in the "Printer Functions" section, we explain how to create and use environment files.

CONTENTS OF THE STORAGE MANAGER

The Storage Manager always shows the last directory viewed with it unless it is being displayed for the first time since start up of the word processing system. Any other existing directory can be viewed through the Storage Manager Form.

As we have explained, the Storage Manager Form consists of field names, fields and special messages. The field names are permanent, but the fields are like windows; their contents change depending on which directory is being displayed in the Storage Manager.

"Sectors free" Field

A sector is a unit of measure of disk space. For example, there are 35Ø sectors on a single sided, single density 5" disk, 14ØØ sectors on a double sided, double density 5" diskette, 2464 sectors on a single sided 8" diskette, and 4928 sectors on a double sided 8" diskette. When a disk is initialized, 4 of those sectors are occupied. Then each time a file is created, more sectors are occupied. The "Sectors free" field always shows how many sectors are left open for storage on the disk. It is good to be aware of this figure, particularly when a function is to be performed that results in a file being added to the disk.

"Current directory specifier Field"

The current directory specifier field shows which directory is currently being displayed. This field can be changed by the user to cause a different directory to be displayed. The current directory specifier field can be made up of one or more components. Each component must be preceded by a left angle bracket. The first and essential component is the drive specifier.

```
     <1
```

In the "Writing a Document" section we explain the other optional components which are sub-directory names.

File and Sub-directory Names and Their Extensions

Beneath the current directory specifier field are three colums of fields. These fields display the names of the files and sub-directories contained by the directory being displayed. There are 21 of these fields, seven in each colum. Only a portion of these fields are in use when the WordMaster Disk is shipped, so the display of the main disk directory may not show 21 file names. We mentioned in our explanation of the System 88 storage system that there are four types of files that the user creates when doing word processing. Any time any of these four file types is created, the new name is placed in a directory. Different main menu options result in different types of files being displayed by the Storage Manager. If you want to see all of the files on the system disk, select the "Storage management functions" option and then select the "List a directory (Copy, rename, delete)" option from the resulting sub-menu.

Special Messages

Each time it is displayed, the Storge Manager Form has special messages at the bottom. These pertain to the option selected from the main function menu, and give reminders as to how to operate the form.

SPECIAL FUNCTIONS OF THE STORAGE MANAGER

There are several functions of the Storage Manager which can always be performed, no matter which option involving file selection has been chosen from the main menu.

Help Function

ESC H

The Storage Manager Form always has a special message which says to type Function Key I for help. The help display that results from typing Function Key I, contains conventions about how to move the cursor about on the Storage Manager Form, how to type on the form, how to change the contents of fields, and what the purpose of the selection of files is for the particular function that is being performed.

File Length Function

In order to find out how many sectors an existing file occupies, position the cursor after the file name and type ESC ?. A message appears at the bottom of the form that tells how many sectors long the file is.

File select function

ESC S

All instances of the Storage Manager Form allow file selection. To select a file, position the cursor after the name of the file to be selected, then strike Function Key II. The file is then selected to receive the action that is indicated. You will learn more about the purpose of file selection as you follow the procedures in the manual.

VIEWING OTHER DIRECTORIES

If you have a multi-drive system, later on you will have other disks in other drives. You can change the disk specifier which is the first component in the current directory specifier field so that it specifies a different disk.

When the cursor is positioned at the end of the current directory specifier field, use the left arrow key to position it over the current disk specifier number. Be sure to leave the left bracket symbol there. Then type the new disk specifier (number of the drive containing a different disk) over it.

```
    <2
```

Strike the RETURN key. If you have another initialized disk in another drive, your present directory display clears and the storage manager shows the main disk directory for the disk you specified. Change the display back to the main directory of your system disk by typing the number of the drive containing the system disk over the current drive specifer and then typing a RETURN. Don't delete the first angle bracket when you are changing the drive specifier. The system will not understand your directory specifer field contents unless its drive specifier is preceded by a bracket. If you do accidentally delete the angle bracket, you won't see a normal directory display. Instead, there will be asterisks in the "Sectors available" field. Reenter the angle bracket and the number of the drive you want to specify and type a RETURN. A normal directory display returns to the screen.

ESC A

Using Function Key III to Cause a Sub-directory Display

We pointed out that your system disk has a storage area called a sub-directory on it. You can see the contents of this sub-directory just as you can the contents of the main disk directory. A sub-directory display occurs when the name of the sub-directory is added to the current directory specifier field.

First of all, note which names in the directory list have .DX extensions. Those are the sub-directories. In order to see what files those sub-directories contain, position the cursor at the end of the desired sub-directory name and strike Function Key III. ESC A The result is that the sub-directory name is added to the current directory specifier and that sub-directory is displayed. Notice that an angle bracket is inserted between the drive specifier and the sub-directory name in the current directory specifier field.

Once a sub-directory is being displayed and you would like to display the main disk directory again, position the cursor after the main disk specifier and before the second right angle bracket; hit RETURN. This causes deletion of the remaining contents of the field and the directory display changes back to that of the main disk directory.

Now that you have a good idea about the general operation procedures for WordMaster, we present each menu option and tell what its purpose is and how to operate its associated forms and sub-menus. We present these options in the logical order of your use, not necessarily in the order in which they occur on the menu. The next two pages contain a summary of the general operating procedures and terms which this section has exposed you to. Use these pages as reference guides as you continue to operate WordMaster II.

SUMMARY OF GENERAL WORDMASTER II OPERATING PROCEDURES

Not all of the WordMaster II operating procedures are summarized here. Those procedures which are unique to a particular function of WordMaster are explained when that function is explained. The remainder of this manual is devoted to describing the detailed operation of each function. The below listed procedures are those which are true no matter which function is selected from the main menu.

SELECTING MENU OPTIONS

A menu is a list of options you may select.

To select options on the WPS menus, type the first letter of the option followed by a RETURN. Striking the RETURN key indicates that you are through typing your selection and the system should process it. Typing the first letter, the first few letters, or the whole function name in upper or lower case is acceptable.

If an "illegal" (incomprehensible to WPS) entry is made, the following message appears;

> This is not a valid choice.

If this message appears, merely wait for the "Enter your choice" prompt to return to the screen and type the first letter of the desired option again.

TYPING ON THE STORAGE MANAGER FORM

There are several conventions for typing on and operating the Storage Manager Form, and when the Storage Manager form is displayed Function Key I invokes a "Help" display which reminds you of these procedures.

They are listed below:

Moving the cursor from field to field

When the Storage Manager Form is being displayed, the cursor is always positioned in, or at the end of, a field. Unless you move the cursor into the midst of a field, it always rests in the first open space of a field. If there are already characters in the field, the cursor will rest in the first blank space after the characters. If there are no characters in the field, the cursor will rest in the first space of the field.

The TAB key moves the cursor ahead to the next field.

Typing ESC left-arrow has the same effect as striking the TAB key.

The BACK SPACE key moves the cursor back one field.

Typing ESC right-arrow has the same effect as striking the BACK SPACE key.

The RETURN key moves the cursor ahead to the next field. If it is typed when the cursor is at the beginning or in the middle of a field with characters, it causes the characters following it to be deleted. This is useful for changing the current directory specifer.

The left-pointing arrow key (without the ESC key typed before it) moves the cursor to the left within a field.

The right-pointing arrow key (typed without the ESC preceding it) moves the cursor to the right within a field.

Typing in a Field

If you want to input characters in an empty field, merely use the appropriate key to position the cursor in the field that you want to type in. The cursor automatically rests in the first empty space. Type in the characters you want. If the field you want to type in already has characters and you want to change them, positon the cursor on the first character that is to be changed and type in the new character. The old character is automatically deleted. Continue this process until you are happy with the contents of the field.

If the field you are changing is the current directory specifier field, type RETURN after changing the contents to cause the specified directory to be displayed.

The Function Keys

Esc H

Function Key I presents a "Help" message which contains all of the conventions for typing on the Storage Manager Form and for using its features. This display also contains special instructions which are pertinent to the function that was selected from the main menu.

Esc S

Function Key II selects a file for the action of the function which is being performed.

Esc A

Function Key III selects a sub-directory to be added to the "Current directory specifier" field and then displayed in the Storage Manager. It saves you from having to type in the sub-directory name yourself.

Esc Cntl E

Function Key IV indicates that you are through using a form and

Esc Cntl G

would like to advance to the next step involved in the selected option. The document formatting options are the only ones which require Function Key IV to be typed before they can be completed. The "Write or change a document" and "Type a Document literally" options are advanced to the next stage following the storage manager when Function Key II Esc S is typed to select a file to receive the action of the selected option. If Function Key IV is typed ESC CNTL G when the storage manager is being displayed as a result of the "Write or change a document", "Type a document literally" or "List a directory (Rename, copy, delete)" options, the system returns to the main function menu after showing the "No file selected" message.

Other Key Sequences

CTRL-Y always interrupts whatever procedure is in progress and returns the system to the main function menu.

ESC ? causes the Storage Manager to place a message near the bottom of the screen which tells how large (in sectors) a file is. In order to find out how large a particular file is, you must position the cursor at the end of its file name field and type the ESC ? sequence.

Viewing Other Main Disk Directories

Positon the cursor over the drive specifer number in the current directory specifier field. Type the new driver number over it, and strike the RETURN key.

Viewing a Sub-directory

If you want to display a sub-directory which is listed in the currently specified directory, just position the cursor behind the file-name field containing the name of the sub-directory you want to display. Strike Function Key III and that sub-directory specifier will be added to the "Current directory specifier" field. The names of files and sub-directories contained in that directory are listed.

ESC A

STORAGE MANAGEMENT FUNCTIONS

The "Storage management functions" option on the main menu invokes a sub-menu that includes options which accomplish the following:

Disk initialization on single and multi-drive systems.

Disk and file back-up on single and multi-drive systems.

Reclaiming disk space that is occupied by deleted sectors.

Listing of directories

Renaming, deleting, and undeleting files.

This section explains each of these procedures. Where the procedures are different for single and multi-drive systems, the two different procedures are described.

DISK INITIALIZATION

One of the first WordMaster functions you need to be familiar with is "disk initialization." In order to be useable on a System 88, a brand new, blank disk must be initialized. The initialization process creates the main disk directory in which files and sub-directories can be stored. Once a new disk is initialized, the "Mirror an entire diskette" and "Copy an entire diskette" functions can be used to make the newly initialized diskette into a back-up of the system disk. It is a good idea to make at least one back-up of the system disk which is kept in a safe place and never used for any other purpose than as a master when new system disks are being created. If a single-drive system is being used, additional disks should be initialized and then the contents of the system disk should be copied onto them. These disks can be used as combination system and storage disks. If a multi-drive system is being used, then additional disks should be initialized and then used as storage disks. First we explain the initialization process on both the single and the multi-drive system. Then we explain how to copy the entire contents of the system disk or a document storage disk onto the initialized disk, thus making it into a back-up.

DISK INITIALIZATION ON MULTI-DRIVE SYSTEMS

Insert a new, blank disk in a drive other than the system drive.

Select the "List a directory (Rename, copy, delete)" option on the Storage Management Functions sub-menu. When the Storage

Manager form appears, change the "Current specifier to show the number of the drive containing the new, blank disk. You should see the "Hard error, preamble bad" message at the bottom of the form, and asterisks in the "Sectors available" field. If you obtain these results, your disk is blank. It is important to verify that a disk is blank before initializing it because initializing a disk with files on it erases those files. Now type <Function Key IV> to return to the Storage Management Functions sub-menu. ESC CNTL E

Select the "Initialize a disk" option.

The following messages are displayed:

Initialize a disk

Which drive?

If you need to abort the initialization process type a CTRL-Y and the system will return to the Storage Management Functions sub-menu. Do not abort the process after you have answered the above prompt.

When you respond to the above question by typing in the number of the drive containing the new disk followed by a RETURN, the following message appears:

(Cleaning disk)

Just before the process is completed, you are requested to type in a name for the disk. This name can be no longer than 8 characters.

You can now begin to use this disk as a document storage disk or you can use the "Copy an entire disk" option to create a back-up for your system disk. Before you finish with this section of the manual, be sure you have initialized at least two disks. One is to be made into a back-up for your system disk, and the other is to be used as a document storage disk. Be sure to label the disks properly and to insert your document storage disk into one of your free drives. You will be using it in the next few sections.

DISK INITIALIZATION ON SINGLE-DRIVE SYSTEMS

Verify that your disk is blank by inserting it in the 8810 drive and pushing the "Load" button. If the disk is blank, the load process will not be completed. You will see an error message at the upper left-hand corner of the screen.

Remove the blank disk, insert the regular system disk, and push the "Load" button. If you have not made your word processing system an initial file which starts up automatically when the system is loaded, then start it up again by typing WPS after the Exec $ prompt.

Select the "Storage management functions" option on the main menu. Then select the "Initialize a disk" option.

The following messages are displayed.

```
                         Initialize a disk

    Which drive?
```

Respond to the above question with a 1 since that is the drive that will contain the disk to be initialized. The following message appears:

```
    Insert a blank disk in drive 1 and hit RETURN to continue...
```

Insert your blank disk and hit RETURN to continue.

The following message appears:

```
    (Cleaning disk)
```

Just before the process is completed, you are requested to type in a name for the disk. This name can be no longer than 8 characters.

When you have typed in the name for the disk, the following message appears:

```
    Insert a System disk and hit RETURN to continue...
```

At this point you should reinsert your system disk and hit RETURN to see the Storage Management Functions sub-menu. Now you are ready to copy the contents of your system disk onto the initialized disk.

Initializing old disks that are to be re-used.

The initialization process can be done on disks which have already been initialized and which have files stored on them. Just be sure that you no longer want the information on the disk because it will all be erased as a result of the initialization function. If you are re-initializing a system disk for your 881Ø, you will see the following message after you have inserted the disk you want to initialize and hit RETURN.

```
    Your system disk is still in the drive.

    Do you want to initialize it?
```

If you really do want to initialize that disk, answer "Y." The above message is an attempt to keep you from initalizing your good system disk.

Label all disks as soon as they are initialized

As soon as a disk has been initialized, label it "blank, initialized" so that you will know that it is ready to be used as a storage disk on a multi-drive system or as a system disk on a single-drive system. Before the initialized disk can be used as a system and storage disk on the single-drive system, it must have the contents of a WordMaster II disk copied onto it.

COPY AN ENTIRE DISKETTE (MULTI-DRIVE SYSTEM)

In order to make a back-up disk for your system disk, you need to copy the contents of the system disk onto a newly initialized disk. In order to make a back-up disk for your document storage disk you need to copy the contents of the document storage disk onto a newly initialized disk. The procedure for copying all of the files from one disk to another at once is invoked by selecting the "Copy an entire diskette." option from the Storage Mangement Functions sub-menu. This procedure can be used if you have two or more drives of the same type as the disk you want to back up. If you only have one drive of the same type as the disk you want to back up, use the "Mirror an entire disk (single-drive system)" option. When the "Copy an entire disk" option is selected, the following message and prompts appear:

```
     Copy entire disk from one drive to another.

     From Which drive?
     To Which drive?
```

In order to copy an entire disk, insert the disk that is to be copied in one of the two available drives that are of the right type for the disk. Then insert the newly initialized disk that is to become the back-up in the other appropriate drive. In response to the above prompts, enter the number of the drive containing the disk to be copied after the first prompt and type a RETURN; then after the second prompt enter the number of the drive containing the disk to be copied onto followed by a RETURN.

The result of this function is that all of the contents of one disk will be copied onto the other disk, making two disks with identical files.

MIRROR AN ENTIRE DISK (SINGLE-DRIVE SYSTEM)

Once a disk has been initialized, you can copy the contents of your system disk onto it so that it can be used as a new work disk on your single-drive system. You should always keep one copy of the WordMaster System Disk that has no files stored on it, so that you can mirror it onto a new, initialized disk and not have to delete text storage files from the new disk before it can be used.

Be careful to note whether or not the disk you are going to copy onto has been ititialized. If it has files on it, make sure you do not want the files because they will be replaced by the contents of the master disk.

When you have your master disk (the disk you are copying from) and the blank, initialized disk (the disk you are copying to) on hand, you are ready to begin. The disk you are copying from is probably already in the system, remove it and write-protect it. To write-protect a 5" disk, place a write-protect tab over the rectangular write-protect notch. To write-protect an 8" disk, remove the write-protect tab from the rounded write-protect notch. If you followed the instructions in the previous section, you have at least one file on your system disk. When you make an exact copy of that system disk as a result of this procedure, you should follow the instructions for deleting a file and reclaiming deleted space which are also included in this section.

The procedure for mirroring a disk on a single-drive system is as follows:

Select the "Mirror an entire diskette" option.

The following display appears:

```
Normally you should wait for the drive light to go off
before removing disks, but in this program you may swap them
after I tell you to.

Insert the copy FROM disk, WRITE PROTECTED

Hit any key to continue....
```

The disk you want to copy from is probably already inserted. However, you are given the opportunity to insert a different "copy From" disk because the disk in the system may have a number of storage files on it and if you have already performed the functions in this section once, you have an extra system disk master which has no storage files on it. If you insert that disk as your "copy From" disk, you will not have to delete the storage files from the "copy To" disk. Remember that the "copy From" disk should be write-protected.

```
Insert copy TO disk.
Hit any key to continue. . . .
```

When you have inserted the "copy To" disk you will see the following message:

```
THIS DISK WILL BE UNUSEABLE UNTIL THE PROCESS IS COMPLETED
```

The above message is warning you not to interrupt the IMAGE process if you plan to be able to use your "copy TO" disk. If the process is interrupted, the "copy To" disk directory will not be properly constructed and the disk will not be useable.

You will be asked to repeat this process of exchanging disks until all of the data on the "copy From" disk has been recorded on the "copy TO" disk. Be very careful not to insert the "copy TO" disk when the "copy FROM" disk is called for and visa versa.

When all of the information has been copied from the "copy From" to the "copy To" disk, the screen will display the following messages and return to the main function menu:

```
OPERATION COMPLETED

Insert a system disk and hit any key to continue.
```

Make sure that you reinsert the system disk that was in the system when you invoked the "Mirror an entire diskette" option.

BACKING UP INDIVIDUAL FILES

There are two available Storage Management Functions which allow you to copy a file from one place to another in the same directory, or from one directory to another directory on the same or different disk. One function [the "Single file mirror (one drive systems)" option] is to be used on single-drive systems, when a file is being copied from one disk to another. The other copy function is a part of the "List a directory (Rename, copy, delete)" option. It is to be used on multi-drive systems when a file is being copied from one disk to another and on single-drive systems when the file is being copied from one spot to another in the same directory or from one directory to another directory on the same disk. These two copy functions are described below.

SINGLE FILE COPY (SINGLE-DRIVE SYSTEMS)

Select the "Single File Copy" option on the Storage Management Functions sub-menu. You will see the following message and prompt:

```
(Duplicate/21)
Normally, you should wait for drive light to go off before
removing disks, but in this program, you may swap them
immediately after I tell you.

Insert master disk. Hit any key to continue...
```

This message is telling you to insert your master disk which is the disk with the file that is to be copied. It is a very good idea to write-protect the master disk before you insert it. To write-protect a 5" disk, cover the rectangular write-protect notch with a write-protect tab. To write-protect an 8" disk, remove the write-protect tab from the rounded write-protect notch. If the system disk already in the drive is the one that has the file you want to make a copy of, you should remove it and write-protect it. Then reinsert it and hit any key and see the following message:

```
Insert slave disk.  Hit any key to continue...
```

The above message tells you to insert the disk which is to receive the copy of the file. The above two messages are repeated until your file has been copied in full. Be careful not to insert the the master disk when the slave disk is called for and vise versa. When the procedure is finished, you will see the following message:

```
Insert a system disk.  Hit any key to continue...
```

Make sure that you insert the same system disk that was in the system when you invoked the file duplication process. If you have write-protected the system disk because it was also the master disk for the duplication procedure, you should reverse the write-protect procedure (remove the write-protect tab from a 5" disk or replace the write-protect tab for an 8" disk) to allow writing on the disk. You need to allow writing on the disk for other word processing procedures since, in a one-drive system, your system disk is also your storage disk that receives document files.

COPYING FILES ON MULTI-DRIVE SYSTEMS

If you have a multi-drive system, you can use the Copy process that is a part of the "List a directory (Rename, copy, delete)" option on the Storage Management Functions sub-menu. This option can also be used on one-drive systems when a file is to be copied from one place on a disk to another place on that same disk.

Here is the Copy procedure:

Select the "List a directory (Rename, copy, delete)" option from the Storage Management Functions sub-menu.

When the Storage Manager Form appears, select the directory for viewing which contains the file that you want to copy.

Position the cursor after the name of the file you want to copy.

Type the ESC key and then the C key. (If, after typing ESC C to invoke the copy function, you decide that you want to abort it, type a CTRL-Y.)

Your cursor is immediately positioned in the first empty file name field of the directory currently being displayed. If all the file name fields are full it is positioned after the first file name in the directory. The following message appears at the bottom of the screen in place of the function key messages.

FILE COPY IN PROGRESS

Move to appropriate directory and enter name of destination file in blank file name space.

If you want to copy the file to a different directory, type CTRL-B to move the cursor to the end of the directory specifier field and change its contents so that they specify the directory in which you want to store your file. When the contents of your destination directory are being displayed, move the cursor to the first empty field and enter the name you want to give to your back-up copy. You do not have to enter the extension as the system assumes that the copy is to have the same extension as the master and adds it for you.

If you want to copy the file to another spot in the same directory, and your cursor is positioned in a blank file name field, you can enter a new name for the back-up copy. You do not need to enter an extension because the system automatically assumes that you want the file to have the same extension as the master and adds that extension for you. Remember that two files cannot have the same name if they are located in the same directory.

If you want to create a new directory in which to store the file, that is possible. Type a CTRL-B to position the cursor at the end of the "Current directory specifier" field, then revise its contents accordingly. If you type in a specifier of a non-existent sub-directory, the new directory display will have no file names listed and will have asterisks in the "Sectors available" field. Move the cursor to the first file name field and enter the name you want to give to your back-up copy. You do not need to enter the extension, the system assumes that you want the copy to have the same extension as the original.

When you have entered the file name for the back-up copy, strike the RETURN key and the copy process is then completed.

RENAME A FILE

Below are the steps to follow to rename a file:

"List a directory, (Rename, copy, delete)" function from the Storage Management Functions sub-menu.

Specify the directory containing the file you want to rename and position the cursor after the file to be renamed.

Use the left arrow key to move the cursor into the present file name. Make the appropriate change and strike the RETURN key. The characters to the right of the RETURN key are thrown away and the characters to the left become the new name. You do not need to specify an extension. The system attaches the same extension that the former file name had unless you specify a different.

When you have completed the above procedure, a message appears at the bottom of the Storage Manager Form which tells you which file has just been renamed and what name it has been given. The screen is updated, causing the new file name with its appropriate extension to be displayed.

DELETE A FILE

To delete a file, select the "List a directory (Rename, copy, delete) option from the Storage Management Functions sub-menu.

Change the contents of the "Current directory specifier field" so that they specify the directory that contains the file you want to delete.

When the directory containing the name of the file you want to delete is being displayed, position the cursor after the name of the file you want to delete.

Strike the CLEAR key. The contents of the field are deleted. Then, strike the RETURN key, or move out of the field with the up or down arrow keys. A message will appear at the bottom of the Storage Manager Form that indicates that the file has been deleted.

UNDELETE A FILE

What if you accidentally delete a file? It's not too late to restore that deleted file to its directory, unless you have reclaimed space on the disk since you deleted the file. The deleted file is not actually removed from the disk until the "Reclaim space" option is used.

To delete a file select the "List a directory (Rename, copy, delete) option from the Storage Management Functions sub-menu.

Change the contents of the "Current directory specifier" field so that they specify the directory that contained the file that you deleted accidentally.

When the directory that used to contain the deleted file is being displayed, type an ESC U. The current directory display clears and a list of the deleted files for that same directory is displayed.

Position the cursor after the name of the file you want to undelete and strike Function Key II.

Esc S

A message will appear at the bottom of the Storage Management Form that indicates that the file is being undeleted. Then the display will clear and the regular display of the directory and its undeleted files returns. This time, the name of the file you just undeleted should appear in the directory.

RECLAIM SPACE ON A DISKETTE

When you have used the "Write or change a document" option several times, the chances are that there are a number of deleted sectors on your disk. We explain why this is true in the next section. Also, if you deleted a file you don't want on your disk, you have some deleted sectors on your disk. When you are sure that you do not want to undelete any of the deleted files on your disk, you can reclaim the space they occupy by selecting the "Reclaim space on a diskette" option from the Storage Management Functions sub-menu. You should never use the reclaim space function unless there is a back-up of the disk on which you plan to reclaim space. This insures that you will not lose the contents of the disk in the event of media failure or system power-down during the reclaim space function.

When you have selected this option, the following message appears:

```
     Reclaim space on a diskette.

     On which drive do you want to reclaim space?
```

Respond to this prompt by entering the number of the drive which contains the disk on which you want to reclaim space.

The result of this function is that the sectors occupied by deleted files are reclaimed for your use, thus the figure in the "Sectors available" field is increased.

SUMMARY

Now that you have read about and practiced the procedures in this section, you should have initialized at least two disks. If you have a single-drive system (an 881Ø), both of those disks should have been made into copies of the WordMaster System Disk. One of the copies should be stored in a safe place as it is a back-up for the system disk and a master to be used when combination system/storage disks are created; the other disk should be kept on hand to be used as a second system/storage disk when the storage space on your present disk is used up, or when your organizational needs require that you begin storing files on a different disk.

If you have a multi-drive system, you should have made one of your initialized disks into a back-up of the WordMaster System Disk and stored it in a safe place. The other initialized disk should be inserted into one of your other drives to be used as a document storage disk.

REMEMBER THAT IT IS VITAL THAT ALL DISKS BE LABELED PROPERLY WITH FELT TIP PEN.

WRITE OR CHANGE A DOCUMENT

In this section, we show you how to use the complete capabilities of the writing function. You will be typing a longer document which you will use again for the "Type a document literally" and "Format a document" sections. Before you can write your document, you must name and open a file for it.

NAMING A WRITING FILE

In our opening demonstration, we showed you how to store a file in the main directory of your system disk. To refresh your memory about how a writing file is opened, select the "Write or change a document" function from the main menu. When the Storage Manager is displayed, make sure that the current directory specifier field is showing the number of the disk on which you want to store your file. If you have an 881Ø, this is drive 1. If you have a multi-drive system, type in the number of the drive containing the storage disk you created while following the directions in the Storage Manager Functions section.

Once the correct main disk directory is displayed, position your cursor in the first open field. Enter the name of your new file just as you did in the opening demonstration. This time, make up your own name and be sure to observe the following limitations:

File names must not have spaces, TABs, RETURNs, +, or *.

File names must not be longer than 17 characters. There are 2Ø blank spaces in a file-name field. Three of them are used by the system for the extension which it automatically appends to the file or sub-directory name.

OPENING A NEW WRITING FILE

After you type the name of your file into the file-name field, strike ~~Function Key 11~~ ESC S to open the file and begin the writing function. The following message appears:

```
Edit 2.Ø (release date)

Input file: not found
Creating output file:  opened

(Key Definition Library opened)


Hit any key to continue. . .
```

The above message says "Input file: not found" because the file is being created for the first time. There is no file on the disk already with that name, so there is no text to be copied into memory. The message also indicates that a new file is being opened on the disk, and in the directory specified by you. If there is a Key Definition Library called TXdef.ED on the system disk it is automatically loaded when the system is invoking the writing function. This library contains phrases which are assigned to individual keys. When a key that has a special phrase assigned to it is typed after the ESC key, the special phrase is entered rather than the character normally signified by the key. We explain this in greater detail later in this section.

Hit any key and the above message leaves the screen. The cursor is positioned in the upper left-hand corner of the otherwise blank screen.

When you type your document it is being placed into memory, not on the disk in the file you opened. The file you opened is ready to receive your text once you have finished with the writing function and have indicated that you are ready to store your text. In a minute you are going to re-open this file, but for now, just type in a sentence or two and then type the following key sequence to tell the system that you are done writing and want your text stored on disk.

```
ESC CTRL-E
```

RE-OPEN A WRITING FILE TO BE CHANGED

The procedure for re-opening a document file is similar to the above procedure for creating a file.

Verify that the current directory specifier field contains the correct directory specifier for the file that is to be re-opened. In this case, you are storing your file in a main disk directory, so the directory specifier is the number of the drive containing the disk which has your file.

Position the cursor at the end of the field containing the name of the file you want to re-open.

CHECK FILE LENGTH AGAINST "SECTORS AVAILABLE" VALUE

There is a special check you should get into the habit of making at this point. Notice the "Sectors available" field at the top of the Storage Management Form. It contains the current number of free sectors on disk. Before you invoke the writing function and ask that an existing file be copied into memory for changing, you should check to see if that file will fit on the disk when you are through changing it. Position the cursor after the name of the file on which you want to check the size. Then type the following:

    ESC ?

At the bottom of the Storage Manager Form, a special message appears which states how many sectors are in the file your cursor is positioned next to. When you know how many sectors are in the file, you can determine whether it will fit on disk when you tell the system you are through changing it and would like it saved on disk. Of course you don't know exactly how large it will be when you are through changing it. It might be the same size, or twice its present size depending on the nature of the changes you are going to make to it. You need to be the judge of whether or not the file can be re-opened and stored on the same disk where it is currently located. The alternative is to use the file copy function to move the file to another disk with plenty of room. Once the file has been moved to the other disk, it can be re-opened and changed.

If you should forget to check the size of the file you are re-opening against the sectors available figure, you are still protected against starting the writing function if there is no room on the specified disk to store the text. Once Function Key II ESC S is typed, and the system begins opening the file, it checks to see that there is enough room on the disk to store the deleted file and the new, changed file. If there is not enough room, there is a special error message which results:

    There is not enough room on your disk for this file.
    Do you wish to open the file anyway?

In most cases you would answer "N" and the write function would be terminated. In some cases, however, you might want to go ahead with the change function because you know that your changes will result in a far shorter file which would fit in the available space. Therefore, you would answer "Y".

WHAT HAPPENS TO RE-OPENED FILES?

When you have determined that the file will fit onto the disk after you have changed it, you can strike Function Key II ESC S (of course the cursor must be positioned next to the name of the file to be re-opened.)

When a file is re-opened the following actions take place:

    It is re-opened

    Its contents are copied into memory.

    It is deleted from its directory.

    A new file is created which will store the changed text from the old sample.

The following message appears when the above re-opening procedures are occurring:

```
EDIT 2.0 (release date)

Input file: opened
(Old input file deleted)

New output file is opened

(Key Definition Library loaded)


Hit any key to continue. . .
```

When a file is re-opened and copied into memory, it is always deleted from the directory, and a new file is created. When the user is finished adding to and changing the text in memory the new output file is used to store the changed text. Although deleted from the directory, the old file is still on the disk. This means that once a file has been re-opened, its text copied into memory, and then deleted from the disk, there are actually 2 files with the same name on the disk. Only one of them appears in the directory, however, because the other one is the older-generation deleted file. If that same file is re-opened again, three files with the same name are stored on disk, two of them are deleted and one is listed in its directory. You may remember from reading the Storage Management Functions section, that there is an "undelete" function which can be used to restore a deleted file to the directory. This is helpful when accidental system power-down occurs while you are typing text into memory. You can restore the next-generation-back file to the directory by using the undelete funtion.

ENDING THE WRITING FUNCTION WITHOUT SAVING THE TEXT

Later, you will be writing a two page document and saving it on disk, but for the present, you needn't save this file. To indicate to the system that you <u>don't</u> want to save a file on disk, type the following key sequence.

```
ESC CTRL-X
```

The following message appears:

```
Edit aborted---FILE NOT UPDATED:

(Type REENTER to resume editing.)
```

The "Type REENTER to resume editing" message is a System 88

Operating System message and does not apply to WordMaster II. All you need to remember is that you should only type the ESC CTRL-X sequence when, for some reason, you do not wish to save the text that is in memory.

When you type ESC CTRL-X, the main menu appears again.

STORING FILES IN A SUB-DIRECTORY

You have learned how to store your files in the main disk directory. We explained in the overview that the sub-directory capability can be used to gather files of a similar type or subject matter together, and that using the sub-directory structure helps you take advantage of the full capacity of your disk.

To store your sample writing file in a sub-directory, type the number of the disk you want to use in the current directory specifier field. This causes the main disk directory of that disk to be displayed. Then position the cursor in the first empty file-name field and enter the name you want to give to your sub-directory, followed by a left angle bracket, which is in turn followed by the name you want to give to your file. The same file name limitations stated above apply to sub-directory names. You might type the following:

```
     LETTERS<Gilcrest
```

When you have entered the name of your sub-directory, the left angle bracket, and the name of your file, strike Function KEY II. ESC 5 The sub-directory is created, the file is opened, the file-opening message appears on the screen, and the hit any key prompt indicates that the writing function can begin.

When you have hit any key, the writing function is invoked and the system is ready to receive your input into memory. Type a few sentences and then type the ESC CTRL-E sequence.

When the system has returned to the main function menu, select the "Write or change a document" option again and note that the Storage Manager is showing you the last directory you selected for viewing. Your new directory file with its .DX extension should be listed in this directory. To see the new file you created, position the cursor after the new sub-directory name and strike Function Key III. ESC 6 The new sub-directory name is added to the current directory specifier field, and the directory display changes accordingly. Your new file should be listed in this sub-directory followed by its .TX extension.

You have just created a first-level sub-directory and stored one file in it. Now you can create a second-level sub-directory and store a file in it. If you are viewing the contents of your new sub-directory, position the cursor to the first empty field and enter the name of a new second-level sub-directory followed by a left angle bracket followed by the name of another new file. For instance, we might enter the following:

```
<PAST-DUE<ARNOLD
```

"PAST-DUE" is a second-level sub-directory for past due letters, and it is listed in the first-level sub-directory called "LETTERS." "ARNOLD" is the name of the new file we are creating, which is stored in our new second-level sub-directory.

When you have entered your new, second-level sub-directory name followed by a left angle bracket and file name, strike Function Key II to cause the file to be opened and the sub-directory to be created. The file-open message appears again with the "Hit any key to continue" prompt. When you have hit any key to begin the writing function, enter a sentence and then type ESC CTRL-E to save your file in its directory.

esc S

When the system returns to the main function menu, select the "Write or change a document" option again. This time, your first-level sub-directory should be displayed because that is the last directory you viewed before you initiated the writing function. This directory should contain the name of your new, second-level sub-directory followed by its .DX extension. To see if your newest file is actually stored in that directory, position the cursor at the end of this new sub-directory name and strike Function Key III to cause the new sub-directory name to be added to the current directory specifier field. The display of the first-level sub-directory clears and the second-level sub-directory display, which contains your newest file, takes its place.

esc A

Now just to make sure you grasp the branching structure of these directories, let's proceed back to the main directory, one directory at a time, viewing each level of directory as we go. Right now the second-level directory is displayed. Notice the name of the file you just created with its .TX extension. Now, to display the first-level directory, position the cursor at the end of the current directory specifier field. Then use the left arrow key to move the cursor into the field and position it after the first sub-directory name and before the third angle bracket. Strike the RETURN key to delete the second-level sub-directory name from the directory specifier and to cause the first level sub-directory to be displayed. Notice the file listed in this directory. Now you can cause the system to display the main disk directory again by positioning the cursor after the disk specifier and before the second angle bracket and striking the RETURN key. The result is that the main disk directory is displayed. The name of the first-level sub-directory you created a little while ago should appear in this directory listing.

Summary: By now it should be clear that the current directory specifier field is instrumental in re-opening or creating a particular file. If the file being created or re-opened is located in the main directory of a disk, then the current directory specifier field need only contain the drive specifier as in the following example:

```
<2
```

If the file being created or re-opened for changing is located in a first-level sub-directory then the directory specifier must contain the drive specifier and the first-level sub-directory specifier as in the following example:

```
<2<LETTERS
```

If the file being created or re-opened for changing is located in a second level sub-directory then the directory specifier must contain the drive specifier, the first-level sub-directory specifier and the second-level sub-directory specifier, as in the following example:

```
<2<LETTERS<PAST-DUE
```

There is a limitation that must be kept in mind when sub-directories are being used. The combination of the directory specifier and the file name is known as a pathname. A pathname must be limited to 31 characters. We have already explained that file names and individual sub-directory names cannot be longer than 17 characters. However, if a pathname is to have several components, i.e. a file name and one or more sub-directories and a drive specifier, the individual components should be kept shorter so that the entire pathname will be within the 31 character limit. For example, here is a legal pathname that you might cause the system to build if you select a file for opening that is located in a third-level sub-directory.

```
<2<LETTERS<PAST-DUE<A-F<DAY.TX
```

The following pathname would not be legal however:

```
<2<LETTERS<PAST-DUE<A-F<FARNSWORTH.TX
```

The names of the three sub-directories in the above two samples are too long to make it practical to name files by last name which is very likely what one would want to do in this instance. Therefore, the individual names of the sub-directories should be shortened.

Below is an example of a main disk directory. Stored in it are two text files and several sub-directories; Note their .DX extensions.

```
        PolyMorphic Systems Word Processor Storage Manager

     There are   900 sectors free
     Current Directory Specifier <2

Letters.DX           Minutes.DX           Coffee-club.TX
Trip-Reports.DX      Baseball-pool.TX     Memos.DX




        esc s Function Key II to select file to edit.
        esc H Function Key I for help.
```

The above example shows the use of the sub-directory structure to organize documents of a particular type. All of the letters branch off of the "Letters" sub-directory, and the trip reports off of the "Trip-Reports" sub-directory and so on. The "Letters" and "Trip-Reports" sub-directories can contain all of the writing files that are letters and all of the writing files that are trip reports, or they can contain sub-directory names if further categorization of letters and trip reports is desired.

Implement this sub-directory capability in a way that is convenient for your organizational needs. Just remember that a sub-directory, like a file name, cannot contain space, TABs, *, or +'s. The entire pathname required to specify a file can not be longer than 31 characters, therefore the number of levels of sub-directories possible depends on the length of the individual sub-directory names and the length of the file name itself.

SELECTING FILES CONTAINED IN SUB-DIRECTORIES

In order to tell the system that you want to change a particular file that is located in a sub-directory, you must use Function Key III to add the appropriate sub-directory or sub-directories to the current driectory specifier until the sub-directory containing the file you want to change is being displayed. Then you can position the cursor beside the name of the file that you want to change and strike ~~Function Key II~~ esc S to select that file.

This same process is used for selection of files to be loaded as environment file, typed literally, formatted or partially formatted. We will refer you back to this section later when it is time to select files to receive the action of these other functions. Right now, it is a good idea for you to practice creating and re-opening files that are stored in sub-directories.

WRITING A DOCUMENT INTO MEMORY

The fact that words and paragraphs are edited into the computer's memory instead of typed right out onto paper, allows the use of special editing functions which eliminate the laborious tasks involved in manual methods of document preparation. Some of these special functions are as follows:

> A portion of text can be marked and deleted from the file. A portion of text can be marked and then moved anywhere else in the file and then deleted or retained in its original position. This process is simple and is accomplished in seconds as a result of the typing of special key sequences.
>
> Every occurrence of a certain misspelling, or other type of error, can be located and corrected in seconds, when the key sequence is used that invokes the Global Search and Replace Function.
>
> Special definitions can be given to keys when they are typed in sequence with the ESC key. These special definitions can be strings of text which occur frequently in a particular document. There is a default Key Definition Library on your disk when it is shipped. The defintions in this library correspond to formatting commands you will be using to cause your "output text" (printed) to be underlined, indented, typed in bold, etc. These, or alternate definitions can be used to speed up the writing process.

We explain all of these writing features later in this section.

For the time being, just use your keyboard like a typewriter and create a sample file with at least two paragraphs.

Before you type in your sample document, you should refer to the following list of differences between typing on the System 88 keyboard and typing on a typewriter. Some of the keys described below result in re-positioning of the cursor in the text. Some of these same keys are also cursor-positioning keys when the Storage Manager Form is being displayed, but the way they function is slightly different. Refer to the summary that follows the WordMaster Overview section to find out how these keys function when the Storage Manager Form is being displayed.

BACK SPACE key: does not result in the cursor jumping back one space as one might think it would. Instead, a special character is inserted into the text file. This character is not understood by the formatter and should not be used while you are writing text. When the Storage Manager Form is being displayed this key is a curser-positioning key.

Blank spaces: typed when the space bar is hit, are actually characters. They can not be seen on the display, but occasionally we refer to the way the formatter responds to the presence of a blank character in a given situation. We are merely referring to the character generated (invisible to you) when you strike the space bar key.

Left-pointing arrow key: The left-pointing arrow key moves the cursor one character to the left. When the Storage Manager Form is displayed, this key is a cursor-positioning key.

Right-pointing arrow key: The right-pointing arrow key moves the cursor ahead one space if there is existing text following the cursor. This key has a different function when the Storage Manager Form is displayed, as it is a cursor positioning key.

Up and down arrow keys: The up and down arrow keys move the cursor to the next line up or the next line down. These arrow keys can only be used where text already exists. The down arrow key can not be used to move the cursor past the last character in the file. When the Storage Manager Form is being displayed, these keys are cursor-positioning keys.

RETURN key: The RETURN key positions the cursor at the beginning of the next line and inserts a character that is invisible to you, but which serves the purpose of marking the end of a line of text. When you are entering text, strike the RETURN key at the end of the screen line. If you don't hit a RETURN, but continue to type past the end of a screen line the cursor will jump to the beginning of the next line. If you only make this mistake once in a while, that's okay, but you should be as consistent as possible about hitting RETURN. At the end of this section, there are some special writing features which work on whole lines of text. For instance, whole lines of text can be deleted with one two-key sequence. One line is defined to be the text between the cursor and the next or previous RETURN. If there is no RETURN character at the end of each line then one line might be several screen lines long. If the RETURN is never used, then the whole file might be one line long. In that case, the function which deletes a line of text would delete all of the text in your file. Not using the RETURN makes typing a document literally and some of the writing features difficult or impossible to use. The formatter does not need to see RETURN characters in the text in order to know when to begin a new line.

When the Storage Manager Form is displayed, this key has special functions which are different than its function while text is being written. They are explained in the WordMaster Overview section.

DELETE key: The DELETE key deletes one character (spaces are characters too) to the left of the cursor.

TAB: The TAB key moves the cursor to the next tab stop. The tab stops are 8 spaces apart. When the Storage Manager Form is displayed, this key is a cursor-positioning key.

SHIFT and CAPS LOCK keys: The SHIFT key works like the SHIFT key on a typewriter, but the CAPS LOCK key works differently. The CAPS LOCK key allows all letter keys to be capitalized without the use of the SHIFT key. The number and symbol keys have two characters each. The upper row of symbols can not be typed unless the CAPS LOCK key is off (up) and the SHIFT key is being used.

When you finish typing approximately two pages, type ESC CTRL-E and exit the editor.

paragraphs

SPECIAL WRITING FEATURES

There are a number of special writing features that you need to learn how to use in order to appreciate the full power of WordMaster II.

Return to the main menu and select the "Write or change a document" option. Select the file you just created to be re-opened for changing by positioning the cursor after it and striking Function Key II. When the file is re-opened and you have "hit any key to continue" the writing function is invoked. You are ready to learn about its special features. There are many of them and you may not fully comprehend their usefullness now. We just want you to try as many of these features as you can and to read about the others. Later, when you want to perform a special function refer back to this section.

THE ESC AND CTRL KEYS

You have seen that your keyboard is much like that of a typewriter. Most of the differences have been stated above. However, there are two keys, ESC and CTRL which when used in special sequences or combinations with other characters cause special writing functions. You have already used the ESC CTRL-E sequence to tell the system that you want to store the text that is in memory on disk. And you have used the ESC CTRL-X sequence to tell the system that you don't want to store what is in memory on disk. Now you need to learn the other sequences that these keys are used with and what their purpose is. Remember that when the ESC key is used in a sequence it is typed once and let up; ther the next key in the sequence is typed. When the CTRL key is used, it is used like the SHIFT key; it is held down while another key is being typed. That is why we insert the hyphen between CTRL and the key typed with it (CTRL-Y.)

INPUTTING AND OUTPUTTING TEXT

When the writing function is invoked, it copies as much text from the input file as will fill half of available memory. It leaves half of memory free so that you can add to the text in memory.

If you have 48K of memory in your system you should keep your text files under 7Ø sectors long. If you do this, your whole text file will fit into memory at one time and you will not have to worry about inputting and outputting text in order to bring the rest of a large file into memory. Read the following directions anyway, as you need to know what to do in the event that an entire file does not fit into memory.

If you have a long text file, all of it may not fit into half of memory. In that case, you will see the following message when you try to move the cursor past the last line of text in memory:

```
WARNING:     You can't move any further,
             but you still have more text
             to read from the disk.  Use
             Control-A to read some of it
             into memory.

         Hit any key to continue.
```

After reading this message, hit any key to continue, and type the following sequence to bring more text from the disk into memory:

```
CTRL-A
```

The above sequence causes enough text to be copied into memory to fill half of the available memory space, unless less text than that exists in the input file. The next CTRL-A sequence will cause half of the remaining memory space to be filled with text, and so on until all of the text is brought in, or until you run out of room in memory for any more text.

You will know when you have run out of space in memory because the CTRL-A sequence will not result in any more text being added, and your cursor will not move, and no characters will appear on the screen when you type. You can make more room in memory in order to bring in more text or to alter or add to the text you have, by using the CTRL-O ("output") sequence. Type CTRL-O. Half of the text in memory is outputted to the currently open output file.

> NOTE: Once text has been output to your output file, it is no longer available for editing; it is gone from memory. You can only reach that text by leaving the writing function and re-opening the file.

Once you have outputted text, you can again copy in text from the input file using the CTRL-A sequence until memory is again full.

You need not bring all text into memory unless you want to change parts of it. Copy in only as much of your file as you need to work with. When you exit the writing function, the rest of the text will automatically be transfered to your new output file even though it was never brought into memory.

MARKING BLOCKS OF TEXT

You can perform the equivalent of the manual cut-and-paste procedure in seconds. When typed after the ESC key, the right and left arrow keys cause arrows to be entered into your text. These arrows can be used to mark blocks of text which can then be moved to a different location in the file, deleted, or just duplicated at various specified places in the file.

To mark a block of text, position the cursor at the beginning of the text to be marked and strike the ESC key, then strike the right arrow key. An arrow will be entered into your text which points at the beginning of the text to be moved. Mark the end of the text to be moved by positioning the cursor to the end of the text, striking the ESC key and then striking the left arrow key. An arrow will be entered into your text which points at the end of the text to be marked.

COPYING A MARKED BLOCK OF TEXT TO ANOTHER PLACE IN THE FILE

You can copy a marked block of text to another spot in the file. Use the arrow keys to position the cursor at the place in the file where you want to move the text. Then type the following sequence:

ESC CTRL-C

To type a CTRL-C, hold down the CTRL key and strike the C key at the same time, in the same manner as you would use the SHIFT key when capitalizing a letter.

When you have typed this sequence your marked block of text will be inserted into the file before the cursor. You can duplicate the marked block in as many places in the file as you want by repeating the above procedure.

Deleting the Markers or the Markers and the Text

Now you can either delete the markers from the marked block and leave the block of text in its original position, or you can delete the entire block of text, markers and all.

The sequence for deleting the markers is as follows:

ESC DELETE (Strike the ESC key and then the DELETE key.)

To delete the entire block of text, markers and all, type the following:

ESC SHIFT-DELETE.

To type a SHIFT-DELETE, hold down the SHIFT key and strike the DELETE key once.

The above sequence is always used when you want to delete a marked block of text.

CURSOR POSITIONING SEQUENCES

There are several special key sequences which help you position your cursor to a desired spot in your file.

CTRL-B: Moves your cursor to the beginning of the file, causing the first 16 lines of your text to be displayed.

CTRL-E: Moves your cursor to the end of the file, causing the last 16 lines of your text to be displayed.

CTRL-N: Positions your cursor at the beginning of a line 15 lines ahead of its present position, displaying the 16 lines which precede the curser.

CTRL-P: Positions your cursor at the beginning of a line 15 lines back from its present position, displaying the 16 lines which follow the curser.

SEARCHING FOR A TEXT STRING

A special key sequence lets you find any word or other combinatioon of characters ("a string"), wherever it occurs in the text currently in memory. Suppose you suddenly realize that you have been misspelling a word, or you want to replace a word with another, everywhere that it occurs. Or you used a certain word or phrase in a particular place, and you want to find that place fast. You can do so by using the Control-F (Find) sequence.

To use Find, position your cursor at the beginning of the file, or at least be sure that it is positioned before the place in the text where the string you want to find is located. Then type CTRL-F. A second cursor appears next to the first one. Now type the string that you want to find. It appears between the two cursors. You can include characters (such as a TAB or a carriage RETURN); they will appear on the screen as Greek letters. (Use the DELETE key to delete mistakes.) When the string is complete, hit the ESC key once.

After you hit the escape key, the second cursor will disappear. If the original cursor does not blink, then the Find function has not found any occurrences of the searchstring; if the cursor does blink, an occurrence has been found and the cursor will be positioned just after the point in the text where the first string is located. You can now change the found string.

To find the next occurrence of the search string, use the Control-C (Continue search) sequence. The Find function remembers the search string and advances in its search to the next occurrence. You can use Control-C as many times as you want: if there are no more occurrences of the search string, the cursor simply will not move from its current position.

To be sure to find all occurcences of the search string, use the Control-B command to get to the beginning of the text in memory. Make sure that your search string is exactly the same as the item you are searching for, misspellings and all. If the item you are looking for contains a space, TAB, RETURN, etc., you must include that within the search string.

COMBINING FILES

The writing function allows you to put two or more files together to form a single file. First write down the full pathnames of the files to be combined, in the order you want them combined. Make sure that the names are exactly correct. If you remember, we explained that you never have to type a complete pathname into the Storage Manager Form because it builds pathnames for you out of the components which you select. However, to accomplish the combining of files procedure you will have to type in a complete pathname. Just to remind you, a pathname is the full specifier that the system needs in order to locate a file. The full specifier consists of the Current directory specifier that is being displayed when the name of the file you want to use is being displayed in the Storage Manager Form. Therefore, in order to write down a pathname, use the storage manager to locate the name of the file you want. Then write down the contents of the "Current directory specifier" field. When you have written down its contents, add an angle bracket followed by the name of the file being specified. For instance, the following pathname tells the system how to locate a file called Benson.

```
<2<LETTERS<New-prod<Benson
```

Everything to the left of the last angle bracket is a directory specifier. The name to the right of the last angle bracket is the name of the file. The system will prompt you for a complete pathname before you can combine files.

Begin the combining procedure by opening the first file in your list, using the "Write or change a document" function. Use CTRL-E to move the cursor to the end of the file. Use CTRL-A if there is more text on disk that has not been copied into memory. When all of the first file has been brought into memory, you can open the second file on your list and bring in the text from it, adding it to the end of your first file. Type the following sequence:

```
ESC CTRL-I
```

If you have not brought all of the text into memory from your first input file you will see the following message:

```
     WARNING:     The current input file was not entirely read
                  into memory! (CTRL-Y to abort)

     New input file name:

     Hit any key to continue.
```

The reason this warning exists is that if you open a new input file before all of the text from the first input file is copied into memory, you will not be able to bring the rest of the text from your first input file into memory anymore. The text that is still on disk is severed from your file.

Once you have brought all of the text into memory, you will see the following message when you attempt to use CTRL-A to bring more text into memory from the input file:

```
                      The current input file is empty.


     Hit any key to continue...
```

When you do see this message, it is safe to open the new input file. Type the ESC CTRL-I sequence. The following display appears:

```
     New input file name:
```

Since you have written down the complete pathname of the second file you want to add, you can type that pathname after this prompt and hit RETURN. The old input file is closed, the new input file opened, and your text file returns to the screen. Your next step is to bring all of the contents of this new input file into memory by typing CTRL-A as many times as necessary. If your cursor refuses to move and you can't bring any more text into memory, it is because memory is full. If this happens, type a CTRL-O in order to ouput the first half of the text in memory to make room for the input file text. This means that you can't make any changes to the text that has been outputted until you re-open the file after the combining process is complete. Remember that you have opened one output file and are outputting the contents of at least two input files to that output file. This is the purpose of the combine function. Repeat the above steps for each file you want to add to the end of your first file. When you have combined the files you want in one file, type ESC CTRL-E to write the remainder of the contents of memory into the output file, and to close the ouput file and return to the main menu. Now you can re-open the one, big file you created, and move through it verifying that your several files were combined in the right order.

DIVIDING FILES

To divide one file into several files, re-open the file you want to divide. When that file is opened and you are looking at its text, mark the beginning of the text with a right arrow (type the ESC key and then the right arrow key.) Then move the cursor to the point in the text which marks the dividing point you want. Mark the end of the first part of the file with a left arrow. You can now create two files, one which will contain the marked block of text and one which will contain the rest of the file from the left arrow to the end. To deposit the first part of the file (the marked block) in a separate file, type the following:

```
ESC-CTRL-D
```

The marked text is then written onto disk in the output file you opened.

Usually, at this point, you should use the ESC SHIFT-DELETE sequence to delete the marked block of text that you just outputted from the text remaining in memory. If you don't delete it, it will be in the next file you create.

The next step is to create a separate output file for the rest of your text, or for the next section of text that you want to be a separate file. Type the following to open a new output file:

```
ESC CTRL-O
```

When the new ouput file is opened you can either type ESC CTRL-E to cause the remainder of the input file (including text in memory and text still on disk that has not yet been added into memory) to be placed in a new separate file, or you can mark a portion of it and use the ESC CTRL-D sequence to ouput that marked portion to the new output file. If you use the ESC CTRL-D sequence, you still have text left in memory and maybe in the input file on disk. Repeat the above procedure to create a new output file for that text. The purpose of this function is to make two or more ouput files for one input file, thus dividing a file into more than one smaller files.

DELETING A WHOLE WORD

You already know that striking the delete key deletes one character to the left of the cursor. If you want to delete a whole word, type the following:

```
CTRL-W
```

To type the above sequence hold down the CTRL key and strike the

"W" key. The word to the left of the cursor is deleted.

DELETING A WHOLE LINE OF TEXT

To delete a whole line of text (text to the left of your cursor all the way to the next RETURN character) position the cursor after the line you want to delete and type the following:

CTRL-X or **CLEAR** on the numeric pad

To type the above sequence hold down the CTRL key and strike the "X" key. This causes an entire line (from the cursor to a RETURN character) to be deleted. Remember that lines of text are defined by where you entered RETURNS. If you forget to enter a RETURN at the end of a line, the cursor automatically jumps to the next line. This means that when you use CTRL-X, you may cause more than one screen line of text to be deleted if there is no RETURN at the end of the previous screen line. You can not see RETURN characters but they are there and they tell the writing functions where lines of text begin and end.

RESTORING DELETED CHARACTERS, WORDS, OR LINES

What if you accidentally delete a whole line or word or part of a word and you wish you hadn't? You can bring back the accidentally deleted text one character at a time. The secret to this feature is not to move your cursor away after you accidentally delete some text. Leave it where it is after the accidental deletion and type the following, holding down both keys until the characters, words, or lines that you want to restore are restored.

CTRL-U

CHANGING THE CASE OF LETTERS

If you have typed several letters, a whole word, or even a whole line of text in the wrong case, there is a sequence which will reverse the case of all of the letters between the cursor and the next RETURN character. Make sure that the characters to be reversed are on a line by themselves by positioning the cursor after the last letter in the line to be reversed and striking a RETURN after it. Then position the cursor before the first character to be reversed and type the following:

CTRL-V

CHANGING OR CREATING KEY DEFINITION LIBRARIES

Type the following:

ESC b

This results in the following formatter command being inserted in your text:

{bold}

Delete this command for now, because we just want you to see evidence of the the Key Definiton Library capability.

The reason that {bold} is entered into your text when you type ESC b is that we have defined the lower case b key to be {bold} when it is preceded by an ESC. Likewise, we have defined a number of other keys to be formatter commands when they are preceded by an ESC. You can see what other definitions we have placed in the Key Definition Library by typing the following:

ESC ?

A display of all the currently available key definitions appears. You can add to or change this list. We show you how below. The fact that the formatter commands are assigned to keys as definitions is a feature which will be quite useful to you as you begin using the format feature of WordMaster II. We explain later how this feature operates. Once you understand what effect the various commands have on the text, you will be able to insert those commands taking advantage of the special key definitions to avoid having to type them in. You can insert these commands in your text as you are writing your document. They are used to cause the formatter to begin a new page at a particular point, to underline a specific portion of text, or to center a title, to indent a paragraph and many other special text arrangement and appearance functions of which the formatter is capable.

You will recall that when you select a file to be changed or created for the first time, the file-opening message includes the phrase "Key Definition Library loaded." The Key Definition Library is the file on disk that contains the definitions we have given to keys. It is automatically loaded at the time that a file is opened because it is given a name that the system understands to be the default Key Defintion Library. That name is TXdef.ED. If you select the "List a directory (rename, copy, delete) option on the Storage Management Functions sub-menu, and list the contents of the WordMaster System Disk main directory, you will find that the file name TXdef.ED is in one of the file name fields. This default library contains formatter commands and their associated keys (key definitions.)

You may want to use this key definition capability to assign different definitions to keys that are temporary, or you may want to change our default definitions. This is up to you. Let's say that you want to create a document file that contains the same phrase numerous times. Instead of having to type that phrase each time, you can type it once and assign it to a definable key as a definition of what that key does when it is typed after ESC.

In this manual we repeat the phrase "Current directory specifier" dozens of times. We can assign this phrase to the ESC C key sequence, and thereby eliminate the need to type it again.

Here's how to define a key.

> Type ESC=k (where k is any key you want to define.)
>
> A second cursor appears on the sceen right next to the regular one.
>
> Enter the phrase you want to assign to a particular key.
>
> Next, strike the ESC key: the second cursor and the phrase you typed will disappear.
>
> Now position your cursor at the point in the text where you want the special phrase to appear.
>
> Type ESC k (where k is the key you defined)
>
> The phrase you assigned should appear in your text.

Following the above procedure, you can define as many keys as you want while the writing function is invoked. There are some keys that the system knows as control keys. In general they are the keys used for cursor positioning, like TAB, RETURN, arrow keys, etc. You can not define these control keys. If you attempt to do so, you will see a message to that effect, and you should pick a different key.

You can decide to save your definitions in the already existing default Key Definition Library. You would choose this option only if the definitions you assign are phrases or words that you will need to insert in most of the writing files you do. When you save new definitions in the default TXdef.ED file they are added to the existing definitions unless you use a key that is already used in the default file, in which case your definition replaces the older definition for that key.

The procedure for saving definitions in the default TXdef.ED file is as follows:

> After you have made the key assignments according to the above procedure, type ESC CTRL-W. You will then see the following prompt:
>
> > **New Key Definition Library Name:**
> >
> > Enter TXdef and a RETURN after the above prompt, and then hit any key to return to your text. All definitions made by you prior to your using the ESC CTRL-W sequence are saved. The new Key Definition File

which you call TXdef contains all of the key definitions currently in memory (those are shown when you type ESC ?.) These include the definitions that were automatically loaded in from the default library and the new definitions you made prior to typing ESC CTRL-W. If you decide to define more keys after this procedure is completed, you have to repeat it in order to tell the system to update the current default file to include your newest definitions.

Besides saving definitions in the TXdef library, you can save them in alternative libraries. The procedure for doing this is the same as the one above except that you supply your own name when prompted for the Key Definition Library name. Your new library will contain all of the TXdef definitions plus your new definitions, unless your new definitions use keys already in use by the default library, in which case they replace the older definitions.

Once you have created an alternative Key Definition Library, it will be written on disk and will be available the next time you are writing a document. As we have already explained, the default TXdef library is automatically loaded into memory when a writing file selected to be written or changed. Once the writing function has been invoked, you can ask that your alternative library be loaded into memory for your use. To do this type the following:

ESC CTRL-L

The above sequence results in the following prompt:

Key Definition Library name:

When you have typed the name of your alternative library, followed by a RETURN, the system loads your alternative library into memory and you can use its defined keys.

If you just want your new key definitions to be temporary, don't use the ESC CTRL-W sequence. Just define the keys as shown above and use them for the duration of the writing procedure in progress.

GLOBAL SEARCH AND REPLACE

Global Search and Replace is a writing feature that is similar to Find (CTRL-F) but more powerful. This feature is used to find every occurrence of a particular text string and then replace it with a correct string. If you have misspelled a word several times in your document and you would like to replace it with its proper spelling, this is the feature to use. Type the following while the writing function is invoked.

```
     ESC CTRL-G
```

The following display appears:

```
     Enter the search string:
     Enter the replacement string:
     Validate each change?  (Y or N):
```

This display is a series of self-documenting prompts. Enter the name of the string (incorrect word or phrase) you want to find, type RETURN, and enter the name of the replacement string followed by a RETURN. You can request that the system stop at each occurrence of the search string so that you can validate that the occurrence should be changed. This is probably a good idea unless you are very familiar with this procedure. If you type Y you will see the following display, and if you type N your changes will be made instantly.

```
When the question mark appears, type one of the following:


               CTRL-Y:    quits global search/replace
               ESC:       skips this change and continues
               RETURN:    makes remaining changes
               (any key)  does this change and continues

     Type any key to continue...
```

When you type any key to continue, the first search string is found and the question mark is positioned just to the right of it, waiting for you to type one of the key strokes or key sequences listed above. The first option allows you to abort the Global Search and Replace process if you realize that you did not request the right search string. The next option allows you to skip the change for the occurrence shown and to proceed to the next occurrence. The third option ends the each occurrence validation process and causes all other occurrences of the search string to be replaced. If you select this option, the changes will be made and the cursor will be positioned after the last change made in the file. If there is still more text in the file that has not been brought into memory, you can type a CTRL-A to bring that text in to memory and then repeat the entire Global Search and Replace process on that text if you believe there might be any more occurrences of the search string you want to replace. The fourth option allows you to okay the change for the occurrence shown and to proceed to the next occurrence.

The best way to learn how to use Global Search and Replace is to practice it on sample files. Try all the options and think of possible applications of this feature.

SUMMARY

Below is a reference list of the key sequences that invoke special writing features:

Arrow keys: The arrow keys are used to move the cursor.

Up-pointing arrow: Move the cursor to point directly above present cursor position; if there are no characters directly above the cursor, it moves to the right of the nearest character above.

Down-pointing arrow: Move the cursor to the point directly below present cursor position; if there are no characters directly below the cursor, it moves to the right of the nearest character above.

Right-pointing arrow: Move cursor one space to the right; if the cursor is at the right end of the line, it moves to the beginning of the next line down.

Left-pointing arrow: Move cursor one space to the left; if the cursor is at the left end of the line, it moves to the right end of the next line up.

ESC up-arrow and ESC down-arrow: Move the cursor to the BEGINNING of the line above by typing ESC up-arrow. Move the cursor to the BEGINNING of the line below by typing ESC down-arrow.

CTRL-A: Add to text in memory from input file.

CTRL-B: Display beginning of text in memory.

CTRL-C: Continue search for search string. (used with CTRL-F (Find) function)

CTRL-F: Find a string. Type Control-F, then the string. Then hit ESC (escape key) once.

CTRL-N: Display next fifteen lines from present position.

CTRL-O: Output text in memory to output file.

CTRL-P: Display fifteen lines before present position.

CTRL-U: Undelete deleted characters.

CTRL-V: Reverse upper and lower case.

CTRL-W: Delete one word.

CTRL-X: Delete one line.

ESC right-arrow, ESC left-arrow: To mark a block of text for moving, copying, or deletion, insert a right-pointing arrow at the beginning of the text to be marked (ESC right-arrow.) Insert a left-pointing arrow at the end of the text to be marked (ESC left-arrow.)

ESC CTRL-C: Copy a marked block to the present cursor location.

ESC CTRL-D: Copy a marked block to the output file. This is useful when dividing files. Be sure to create a new output file for the remaining text.

ESC SHIFT-DELETE: Delete a marked block.

ESC-DELETE: Delete block markers only.

ESC CTRL-L: Load an alternative Key Definition Library.

ESC CTRL-W: Create a new Key Definition Library which will contain all of the definitons currently in memory. Those include the defintions loaded automatically from the TXdef file and those created after the writing function is invoked and before the ESC CTRL-W sequence is typed. If more keys are defined after this sequence is typed, the sequence must by typed again to update the new library. It new definition file be given the default Key Definiton Library name if permanent alterations are to be made to the default library, or it can be given a different name if the default library is to remain unchanged.

ESC ?: Causes the key definitions currently in memory to be displayed. If a new defintion file is created it will contain all of the definitions listed in this display.

ESC CTRL-G: Invokes the Global Search and Replace function.

ESC CTRL-E: Causes the writing function to terminate while the text in memory is saved on disk. If all of the text in the input file did not fit into memory and was never added in, the rest is added to the end of the text in memory when it is written on disk.

ESC CTRL-W: Aborts the writing function. No text is saved on disk. If a file is re-opened, its contents placed in memory and then ESC CTRL-W is used, that file will no longer be listed in its directory; it must be undeleted. The Storage Management Functions section explains the "undelete" process.

CTRL-Y: Aborts any writing feature in progress, leaving the writing function invoked.

TYPE A DOCUMENT LITERALLY

Now that you know how to use the "Write or change a document" function, create a file that has about two pages worth of text in it. Leave blank lines between the paragraphs. Make some lines one or two words long and others a full screen line. We want you to make your lines as irregular as possible because you will be typing this document literally and formatting it and we want to emphasize the difference between the outputs of the two functions. Remember to enter RETURNs at the end of each line.

TYPE YOUR DOCUMENT LITERALLY

Select the "Type a document literally" option from the main menu.

Change the value in the current directory specifier field, if necessary, to cause it to specify the directory that contains your sample document file.

When the directory that contains your sample file is being displayed, position the cursor after the name of your sample file and strike Function Key II. esc s

Your document is printed out literally.

Now look at the paper copy of your document. Notice that the lines are uneven, just as they were in you input.

TYPE FUNCTION PARAMETERS

The type function does some minimal text arrangements in accordance with the parameters shown when you select the "View page parameters" option from the Printer Functions sub-menu. Your printer was instructed to leave a top and bottom margin in accordance with these parameters. The first character in each line is placed on the first space to the right of the left bias area. The total number of spaces and characters between the left bias and the right perforation or edge should equal the "Characters per line figure." The total number of printed lines and blank lines on the first page should equal the "Lines per page" figure.

The reason you had to type a RETURN at the end of your input lines when you created this sample file, is that when the type literally function is used your printer can only print as many characters on a line as your width limit ("Characters per line") allows. If you don't enter RETURNs in your input, your input lines may be longer than than the width limit allows. The printer will either truncate (throw away) the portion of the line that exceeds the width limit, print the excess characters on the platen of the printer itself, or continue the overflow on the next line. It is best to avoid this by remembering to insert RETURNs.

The purpose for typing a document literally is that it provides a guick method of obtaining a paper copy of your document. And, if you have entered formatting commands, (see the Format a Document" section) the type function does not process them, rather it prints them as they appear in the input. Some people like to obtain paper copies of files before they format them so they can see the cause of problem areas in the formatted paper copy.

We are having you print your sample file as a part of a two-part demonstration. In the next section you will be formatting the same file. You will see the difference in the text arrangements made by the formatting function.

PRINTER FUNCTIONS

Selecting the "Printer functions" option on the main function menu results in the following sub-menu.

```
Printer functions

Define a new printer name.
View page parameters.
Connect a different printer name.
Move paper to top of page.
Quit this menu.
```

Any time you are using a function which involves the printer (i.e. "Type a document literally." or "Format a document." or "Only format a part of a document.") The System 88 Printer Driver is involved. This is the part of the operating system which communicates to the printer. You can communicate to the printer driver by selecting the "Printer functions" option on the main menu.

DEFINE A NEW PRINTER NAME

This option is intended for use by dealers. It is the function which allows a particular type of printer to be defined to the System 88 Printer Driver so that it can prepare to communicate with it. When you receive your system disk, your dealer has already defined your printer by using this option. When this option is selected, the printer definition function of the System 88 Operating System is invoked. This function is explained in full in Section 13 of the System 88 User's Manual. If you ever change printers, you will have to make sure that your new printer is properly defined to your Printer Driver. This operation requires that you have some specifications for the printer. These specifications can be obtained from the manual that comes with the printer. It is probably best if you ask your dealer to help you in the event that you need to perform this operation.

DEFAULT Printer

One of the features of the "Define a printer" option is the ability to select one of the known printers (one of the printers that has been defined to the system) to be the DEFAULT printer. Once this has been done, the system prepares to run the DEFAULT printer each time the operating system programs are loaded into memory (each time the "Load" button is pressed.) If you are only using one type of printer, your dealer has already made this your DEFAULT printer. Therefore, you should not need to concern yourself with connecting your printer each time you need to use it. (See "Connect a different printer name" below.)

VIEW PAGE PARAMETERS

Select this option to see the parameters that are in effect when a document is typed literally. Parameters are figures which tell the printer the size of the paper. The printer must know the paper size in order to know when to begin a new line or page.

Below is a sample page parameter display:

```
     Lines per page (form size): 66
     Characters per line (page width): 85
     Lines for TOP margin: 3
     Lines for BOTTOM margin: 6
     Offset for left EDGE: 1

     Strike any key to continue. . .
```

WHAT IS A PRINTING ENVIRONMENT?

Select the "Define or view an environment" function. Selecting this option results in the display of the Environment Form which has a number of formatter and page parameter controls. Among these controls are the above parameters. Notice that each of the listed controls and parameters has a value or argument. These values are taken from the "default environment values file" which was on your disk when it was shipped. This file is called WP<Word.FV and its contents are automatically loaded into memory when the word processing system is started up. They affect any printing function unless you change them on the Environment Form before you perform a printing function, or unless you load an alternative environment values file into memory.

Before you type a final document on a typewriter, you make many decisions. You decide what the margins should be, whether or not the document should be single, double or triple spaced, whether or not to number the pages, where to put the page numbers, whether or not to indent, how many spaces each indent should be and etc. These decisions are equivalent to the kinds of decisions you make when you adjust the Environment Form values to meet your printing needs. The term "Environment" means the conditions under which the document is printed, the over-all appearance of the document that is determined by values that you expect to remain constant throughout the document.

With this word processing system, you can define a number of different environments which are suitable for all kinds of printing that you do. Then, when you need to print a document you can load the appropriate environment.

There are two kinds of printing you can do. You can have a document typed literally and you can format it. The next two sections explain these two processes. At this time, you should realize that the literal typing function uses only the parameters listed when you view the page parameters from the Printer Fuctions menu. These are just a few of the controls in the environment form. Even though the other controls are in effect whenever a printing function occurs, the special formatting environment controls are ignored when the document is typed literally. Below we show you how to change the values in the Environment Form, and how to store your changes in the default environment file or in an alternative environment file. Later, when you learn to format, you will be referred to this section to follow these instuctions for creating alternative environment values files.

CHANGING THE VALUES SHOWN IN THE ENVIRONMENT FORM

To change the values in the Environment Form, use the arrow positioning keys in the same way you do when moving about on the Storage Manager Form. Position the cursor at the end of the field you want to change. You can then use the right arrow key to position the cursor over the value in the field and then type the new value, causing the old one to be deleted. Or, you can delete the old value with the CLEAR key and then replace it with the new one. When you have finished making changes to the parameter values in the Environment Form, strike Function Key IV. ESC CNTL-E

Any time you change any values or arguments in the Environment Form, you are asked the following question after striking Function Key IV: ESC CNTL-E

> **Do you want to save these changes as a pre-defined environment?**
> **(Y or N)?**

If you just want to use your changes for one printing and you don't want to save them in any file, answer "N" to this question.

If you answer "Y" to the above question, you will be asked the following:

> **New values filename?**

Then you will be asked to provide the drive number of the disk on which you want to store the file.

At this point, you can either enter the name of the default environment file, thus changing its contents, or you can enter a new name which will result in a new alternative values file being created. The three alternatives that you have at this point are explained in full below:

1) You can save the changes in the default file called WP<Word which is stored on the system disk. If you do this you should answer "Y" and then provide the following filename:

   WP<Word

   Then you will be asked to name the drive containing the disk that is to store this file. In response to this prompt enter the number of the drive containing your system disk.

   If you store the changed environment in the default values file, its orignninal values will be altered by you. From that point on, the values file with your alterations will be loaded automatically when the word processing system is started up. You can change the parameters in effect at any time by changing your default environment file again, just as you did the first time. Or, you can change the environment in one of the remaining two ways.

2) The second alternative is to make the desired changes to the Environment Form, but to opt to save the values in a different file which is given any name you feel appropriate. You can create as many different values files as you want to, depending on your needs. If the above option is taken, the default file must be stored on the system disk in the WP sub-directory and must be called Word. But, the alternative files can be stored on any disk, in any directory and can have any appropriate legal filename. For example:

   ENVIR<MANUALS

   Remember that if you want your values file stored in a sub-directory, you must include it in response to the prompt. There must be a bracket between the sub-directory name and the file name. When the system asks you which drive you want to store the file on you can provide the number of any drive that has a disk in it.

   If you do store the changed environment in an alternative file, the word processing system will still prepare to use the parameters in the default values file every time it is started up. To cause the word processing system to use the values from your alternative values file, select the "Load a pre-defined environment" option from the main function menu. This results in the Storage Manager Form that shows you the names of values files in any directory you specify. You can follow the usual procedures for selecting a file (specify the directory containing the file, position the cursor after the name of the file and strike Function Key II.) The result of loading an alternative environment file is that your special values file is loaded into memory. From that point, any printing done is affected by this special values file, unless a different file is loaded into memory, the

   Esc S

system is re-booted and WPS is started up again, or the Environment Form is altered by you before a printing.

3) If you just want to use your changed Environment for one printing and you do not care to save the changed environment in a file, you can answer "N" to the prompt. The result is that your changed environment will affect all printing operations until the Environment Form is accessed and changed, the system is re-booted, or a different values file is loaded into memory.

We have explained in full the procedure for creating alternative values files. However, at this point, you only know about the page parameter values. In the "Format a document" section we explain the purpose of the other environment controls listed in the Environment Form. Soon you will be creating environments for all of your main types of documents. Then when any document is to be formatted, you can load the appropriate environment and select the file to be formatted. This procedure of creating and saving environments for printing saves time and insures uniformity of your documentation.

CONNECT A DIFFERENT PRINTER NAME

Normally, your printer is connected automatically when the system is loaded. This is because your dealer made it the DEFAULT printer. This option is only used if the printer is not already automatically connected when the system is loaded. You may need to use this option if your dealer has defined two different printers to your System 88 operating system software. If two printers are defined, only one can be the DEFAULT printer that the system prepares to run automatically when it is loaded. As you switch back and forth between two types of printers, you may need to use this function to connect the printer currently being used. After the following prompt:

```
Printer name to connect?
```

type in the name your dealer used when he defined your printer to the system software.

If you ever attempt to perform a printing function and see a message that indicates that you cannot print because no printer is connected, you know you must select this option and type in the name of the printer you are using.

Your dealer will be able to tell you what name to use for your printer. Also, you can select the first option on the Printer Functions sub-menu to see a list of "known printers." These are all of the types of printers (as defined by us and your dealer) which your system knows how to run. One of these known printer names is the one the dealer gave for your printer.

MOVE PAPER TO TOP OF PAGE

Selecting this option immediately results in your printer executing a "form feed" which causes the paper to advance upwards until the top of the next page is positioned under the printer head.

QUIT THIS MENU

Select this option when you are finished with the Printer Functions sub-menu and want to return to the main funtion menu.

FORMAT A DOCUMENT

Now that you have created a two page document and have typed it literally, you can format it and compare the results of these two printing functions.

FORMAT YOUR SAMPLE DOCUMENT

Select the "Format a document" option from the main function menu.

Select your sample writing file to be formatted, by positioning the cursor next to its file name, striking Function Key II and then striking Function Key IV.

ESC Cntl E ESC S

The Storage Manager Form leaves the screen and is replaced by the pathname of the file to be formatted. If this is the file you meant to select, type any key to begin the formatting process. Your sample file should be printed out in formatted form on your printer.

When the process is finished, type any key to return the system to the main function menu.

Compare the results of the typed literally and formatted versions of your sample file. You will note that the output lines of your formatted sample are even and that they are flush with both the left and the right margins. There are a number of other less obvious arrangements that the formatter is making. In this section we explain all of the functions that the formatter is capable of and then tell you how you can request that these functions perform exactly the type of text arrangements that you want them to.

HOW THE FORMATTER WORKS

The formatter performs a number of text arrangement functions simultaneously. The result of these functions is an attractive document that can have any or all of the following:

Titles that appear at the top of every even, every odd, or all pages

Titles that appear at the bottom of every even, every odd, or all pages

Page numbering wherever you want it on the page

Line spacing and characters per inch settings of your choice

Several types of indentation

Extra-dark text, and underlined words or phrases

Centered titles

Superscript or Subscript letters and many more features

In order for the above features to be applied to a particular document, the formatter needs instructions from you. Each instruction you give tells one of the formatter's funtions to operate in a certain mode. For instance, there is a function that does underlining. The formatter does no underlining unless you provide a command which tells that funtion to start underlining. Then it continues to underline until you tell it to stop. There is a command to tell the formatter to underline and one to tell it to stop. In fact, there is another command that tells the formatter to underline words only, not spaces or punctuation. These three commands match the three modes that the underline function has.

Each of the formatter's text arrangement functions has at least two possible modes. There is a command for each of these modes. There are essentially two kinds of formatter functions, long-term and short-term. Both of these kinds of functions can be controlled by commands, but the long-term functions are usually controlled by the values entered in the Environment Form. If you select the "Define or view an environment" option, you will see the types of long-term functions the formatter has. All functions, long-term and short-term are set in a multi-purpose mode by the formatter's internal commands. Then, from the Environment Form and from the text files, you can alter this multi-purpose state.

In the Printer Functions section we showed you how to enter values into the Environment Form and then to save them in environement values files which you can then use every time you format a document that requires that kind of environment. These values control the formatter's long-term functions. These same functions can be controlled from the text file as well, if it is necessary to over-ride the environment value for a particular long-term function.

INTERNAL COMMANDS

You have probably noticed that your sample document was formatted attractively without your giving any instructions beyond telling the system to format it. This is because the formatter's start-up state is determined by a group of internal commands. There is an internal command for every function that the formatter can do. These internal commands set each of the functions in a general-purpose mode. For instance, one of the formatter's functions is capable of two different types of underlining. Certainly the formatter would not be very useful if it always underlined unless you told it not to. Therefore, we gave it a command which set this function in an "off" mode. At any point in the document you can embed a command which will place this function in one of its underlining modes.

The formatter is always capable of doing something reasonable even if given no other commands. If you want to change these general purpose settings to settings that are appropriate for your purposes you can issue commands in two ways: by altering the values in the environment values file, and by embedding commands in the text.

ENVIRONMENT VALUES FILE

When the formatter starts up, it receives commands from the "environment values file" which is in effect at the time. We explained in the Printer Functions section that there are certain controls which are in effect every time printing occurs. These controls tell the printer how big the paper is, how large the margins should be, and where to position the printer head so that it begins printing on the right side of the left perforation. There are many more controls that are in effect when a document is formatted than were in effect during the literal typing process. This is because the formatter is capable of many more text arrangement functions. All of the controls that affect the environment are listed in the Environment Form.

PolyMorphic Systems Environnment Form

Lines per page	66	Characters per line	85
Top margin	3	Bottom margin	6
Offset for left edge	1	Left margin	10
Right margin	10	Line spacing	1
Standard indent	5	Printer type	stan
Characters per inch	10	First page number	1
Print errors	N	Pause after each page:	N

Headers and Footers on following pages.

This display lists all of the environment controls. The figures and letters which are positioned to the right of each control are the values which we have spoken of in the Printer Functions section.

When we ship the WordMaster II disk, the values for these long-term functions which match the formatter's internal commands have been pre-set. This means that the default values file is not actually changing the formatter's start-up state. Soon you will change the values in this form, and thereby change the formatter's start-up state.

EMBEDDED COMMANDS

The third possible source of commands for the formatter is the document file itself. In the document file, you can request that some specific portion of text be underlined, indented, printed in extra-dark type, etc. You do this by embedding commands in the text file. All of the formatter's functions which are expected to change from one mode to another during the formatting process are set in a multi-pupose mode by the internal commands and can be set in special-purpose modes by you when you enter an embedded command in the text file.

HOW THE FORMATTER WORKS: A SUMMARY

When you formatted your sample document file, it had no formatter commands in it. Therefore, the formatter printed your document according to its internal commands which establish its multi-purpose start-up state.

How can you alter the formatter's start-up state? You can change the values in the Environment Form and then opt to save these changed values in the default values file or in an alternative values file, or just to have those values remain in memory, temporarily affecting subsequent formatting procedures. When a document is printed with your changed environment in effect, the long-term formatter functions will be set according to your preference rather than the multi-purpose start-up state.

In addition to altering the environment values in effect, you can embed commands in the document files which cause specific portions of the file to have a different appearance than they would if formatted according to the start-up state.

Now let's review how these three sources of commands are handled by the formatter. When you invoke the formatter by selecting a file to be formatted, and then type any key to continue to indicate that you are ready for that file to be formatted to the printer, the formatter is started up. It initializes itself in accordance with its internal commands which set each of its functions in what we consider to be a useful multi-purpose mode. Then the environment values currently in effect are translated into commands and given to the formatter. If any of these commands are different than the formatter's internal commands for long-term funtions, the formatter reinitializes its start-up state accordingly. Finally, the formatter begins processing the document file you asked it to format. If it finds any commands in that text file, it adjusts the mode of the functions affected by those commands accordingly. There is a command equivalent for every mode of every long-term function as well as a command for every mode of specified or short-term functions. Any of these commands can be embedded in the text. Even though a long-term function is expected to remain constant throughout a document, there are occasions when it is necessary to over-ride the environment file value that is in effect for a particular function with an equivalent command that has a different value and which is embedded in the text file.

Before we provide instructions for altering the formatter's start-up state to meet your needs exactly, we list and explain each of the formatter's functions and tell what mode each function is in when the formatter is started up. Even though the default values file automatically alters the long-term start-up functions, your sample should still reflect the start-up (internal commands) for these functions because our default file values match the formatter's internal commands. Our list of fuctions is divided into two parts; the long-term functions that you will be controlling with the values files you create using the Environment Form, and the short-term functions that you will be controlling using with embedded commands.

START-UP MODES FOR LONG-TERM FUNCTIONS

Offset for left edge: The offset for left edge function is the only long-term control that is not determined by an internal command. In fact, there is no formatter command for this function. This is because this one value is given directly to the printer from the Environment Form. It controls the printer's starting point on a line. Our default value for this function is 1, but your dealer may have changed this value so that it is appropriate for your needs.

The same offset for left edge which affected the typed-literally sample affects the formatted sample. The offset value is just large enough to cause the printer to begin printing on the first printable space to the right of the perforation. This will allow the margins to be equal and the text to be centered attractively on the paper.

Characters per line: If your default values file figure for this function is still 85, as it was when we shipped the WordMaster II disk. There should be 65 characters per output line (spaces count as characters too.) The spaces for left and right margins are subtracted from the Characters per line value and the resulting figure is the maximum number of characters on a line.

Below we explain the Justify Expand mode which is also in effect when the formatter is started up. This mode results in the insertion of extra blanks to make each line exactly 65 characters wide. If the Justify Expand mode were not in effect the number of characters in a line would vary slightly but would never be more than 65.

Lines per page: The total number of blank and printed lines on the page should equal 66. When the formatter has counted 66 blank and printed lines it begins a new page.

Margins: The internal commands for margins establish a top margin of 3, a bottom margin of 6, a right margin of 1Ø, and a left margin of 1Ø. These are apparent in the formatted output.

Characters per inch: The internal command for the characters per inch function is {cpi 1Ø}. This means that there are 1Ø characters per horizontal inch.

Word Fill: Notice that the words have been placed in even lines, regardless of where the gaps and uneven lines appear in the text file. This is because the Word Fill mode is in effect. The "characters per line" value tells the formatter when to begin a new line. If that value is 85, for example, then the formatter puts as many full words on a line as it can until the total of the number of characters it has printed, plus the margin values equals 85.

Notice that we said that the Formatter only places full words on a line. If it can't put another full word on the line without exceeding the characters per line limit, it begins a new line.

The WPS Environment Form contains a control for the formatting mode function. The internal command for this is "fill" as is the default values file setting that is shown in the Environment Form. Fill is actually a composite command which means word fill and justify expand. Justify expand is one of several modes that the justification function can be set in. For convenience, we developed composite commands which represent popular combinations of word fill and justification. The command summary explains all justifiction and word fill modes and shows examples of their effects on text.

Justify Expand: The right and left edges of text are straight along the right and left margins. This is because of the internal Justify Expand command. The action of the Justify Expand command causes both edges to be straight along the margins. We have explained that the Word Fill mode causes the formatter to begin a new line when it can't fit another full word on the present line being filled. In order to make all of the lines exactly the same length the Justify Expand mode causes the formatter to insert extra blanks in the output text where there are already blanks in the input text. This is why there are occasionally more than one blank between words and more than two blanks between sentences. There is a command which will cause the formatter to leave the right edge uneven as is sometimes desirable for letters. All of the alternatives to the default Word Fill and Justification modes are explained in the System 88 Command Summary.

The WPS Environment Form lists a control for the formatting mode function. The default mode for this is "fill."

Blank lines between paragraphs: Since Word Fill causes the formatter to place words one after the other on a line, regardless of their placement in the input text, it fills line after line in the same fashion. However, there are blank lines in the formatted sample which correspond to the blank lines in the input text. This is because the formatter assumes that a line left completely blank in the input should be left completely blank in the formatted output. Therefore, when two RETURN characters are found one after the other in the input, the formatter interrupts its word fill function to leave a blank line and continues to fill on the next line. This function is built-in and not controlled by commands. There is no way to disable it except by not leaving any blank lines in the input text. If there is a formatting command alone on a line, that line is not blank.

Single Spacing The formatter's start-up mode for line spacing is single spacing. We have matched this with our environment value for this function.

Two space after sentence-ending punctuation:
The formatter automatically places at least two spaces after sentence-ending punctuation marks. This is a built-in function, and is temporarily alterable when an embedded command is inserted for that purpose. If there are more than two spaces after a sentence-ending punctuation mark, the extra space is a result of the insertion of extra blanks done by the justify expand function.

START-UP MODES FOR SHORT-TERM FUNCTIONS

Printer Type: For purposes of the above demonstration, the internal command for printer type mode works just fine. However, to do bold, superscript, subscript, and red/black printout, the formatter needs to know what type of printer is being used. We have divided all of the printers that the formatter can run into categories. These are: standard, spin, and diablo. Included in the standard category are TI 81Ø, decwriter, Printronix, and Paper Tiger. The spin category consists of the non-Diablo compatible models of the NEC Spinwriters. Included in the diablo category are Diablos, Qumes, and Diablo compatible printers, and the Spinwriter 5515. There is a command for each printer type. The start up mode for the printer type function is "stan." This means that if your printer falls into one of the other categories, the appropriate printer type command should be used (see command summary) before bold, superscript, subscript, or red/black modes are used.

Text Appearance Functions: The initial commands for the formatter are to have all text printed non-bold, not superscripted or subscripted, and black. All of the text appearance functions are set in their "Off" modes.

Page Numbering: In the formatted sample document, there is no page number anywhere on the page. This is because the formatter's page numbering function begins in the off mode. However, the formatter keeps track of the pages as it formats them and is always prepared to print the current page number. The command summary explains what command to use to cause the formatter to place numbers on the pages. Unless directed otherwise, the formatter assumes that the first page printed is page 1. The WPS Environment Form lists a function called First page number. The internal default number for this function is 1. The default number for this function in the environment value file was 1 when the WordMaster disk was shipped. This mode can be changed in the Environment form, or with an embedded command.

Pause after each page: Internal commands turn this off, so that no pause occurs after each page. This mode can be changed in the Environment Form or with an embedded command.

Error Print: Since no commands have been used to obtain this sample, there can be no errors to report. Errors result from illegal use of commands. The Error Print start-up mode causes errors to be displayed on the screen only.

Stop on error: Since there are no formatting commands in the text, there can be no errors. If the formatter does find errors, there is a function which can be turned on which causes it to stop when it finds an error. This function is set in the off mode by the internal command.

Now that you understand the formatter's start-up state, you can make it do exactly what you want it to by altering the values in the default values file, and embedding commands in your text.

DEFINING OR VIEWING AN ENVIRONMENT FOR FORMATTING

When you are ready to format a document, you need to decide what long-term modes are appropriate. Select the "Define or view an environemt" option from the main function menu. The Environment Form is displayed as a result of selecting this option.

You can change the values in the Environment Form and save them in an alternate values file which is appropriate for a particular type of a document, or you can store the changed values in the default values file, thus altering our defaults permanently. If you don't want to save the changed environemt values, you don't have to. Once you make changes in the Environment Form, those changes remain in memory and affect all printing functions until they are changed again, a different values file is loaded using the "Load a pre-defined environment option" or the system is turned off. All procedures related to using the Environment Form are explained in the Printer Functions section. The only additional information you need about the effects of any individual long-term function mode are explained in the command summary.

Once you are happy with the formatting environment and have returned to the main menu, you can select the "Format a document" option, select the file to be formatted, strike Function Key IV to indicate that you are finished selecting files, and then strike any key to begin the formatting process.

ESC Cntl-15

EMBEDDING FORMATTER COMMANDS

If the file you formatted had no embedded commands in it, it may not look exactly the way you want it to. Perhaps you want to underline, center, or bold specific portions of it. Or, maybe you want your paragraphs indented, and would like to tell the formatter to begin a new page at a particular point. There are many special changes you can make to the formatter's multi-purpose start-up state besides the alterations you make to the long-term functions. These changes which affect specific portions of the text are accomplished when you embed commands in your text file.

The Formatter Command Summary tells what command to use for each type of text arrangement that the formatter can make. It also explains where to place the command in order to cause it to have the exact effect you want. We suggest that you look over the command summary, and pick out some of the modes that you think you will be using most. Make a note of the proper syntax for the command that accomplishes the mode you want and then select the "Write or change a document" function. Select the text file to be changed and enter the curly bracketed commands in the appropriate places. Format your file again. Now your document should look exactly the way you want it to.

Soon you will be familiar enough with the formatter commands that you will be entering them as a part of the writing process. Remember that the Text Definition Library, which is loaded into memory every time the writing function is invoked, contains the key assignments appropriate for formatting. When the writing function is invoked, you can type the following to see what these key assignments are.

```
ESC ?
```

Now to enter your first commands.

While the writing function is invoked, decide on a portion of text that you want to underline and position the cursor directly in front of it.

Type the following to enter an Underline command.

```
ESC u (lower case u)
```

{ul} is entered into your text. Position the cursor directly after the text to be underlined and type the following:

ESC U (upper case U)

{nul} is entered into your text. Type ESC CTRL-E to save your text file on the disk, select the "Format a document" function, strike Function Key II to select the file you just changed for formatting, strike Function Key IV and then any key to begin the formatting process. The portion of text that was enclosed by the {ul} and {nul} commands in your text file is now underlined in the formatted output.

You will be embedding all of your commands in a similar manner. Not all of the commands have been given key assignments; you can type them in yourself or attempt to assign them to unused keys.

Before you begin using the command summary there are a few general principles you need to grasp about the commands. We explain these below.

SYNTAX OF FORMATTER COMMANDS

Formatter commands are enclosed in curly brackets (for example {bold}). The command summary shows the correct syntax for each command. You must enter the correct syntax or the formatter will not understand your command and will report an error.

You do not need to enter curly bracketed commands in the Environment Form. It prompts you for values and arguments which are turned into curly bracketed commands and given to the formatter when it starts up.

WHY EMBED COMMANDS IN THE TEXT?

Consider what the formatter does without any embedded commands. First, it is initialized in its start-up state. Then it processes the commands given to it from the environment file in memory at the time of the formatting operation. Then, it begins to format your text file in accordance with its internal commands as they are modified by the environment file. If word fill is in effect, the formatter is putting as many words on a line as it can until it reaches the width limit, and as many lines on a page as it can until it reaches the "Lines per page" limit. All text appearance modes are in their multi-purpose ("off") settings.

You will be embedding commands to alter this mode of operation.

WHEN DOES AN EMBEDDED COMMAND AFFECT THE TEXT?

Some commands take effect immediatly. If you place them in the middle of the input line, they affect the very next character that follows them. The effect of some commands is delayed until the formatter begins a new line. Other commands take effect immediately and cause a new line to be started before the text following the command is processed.

There is a "Placement and Effect" paragraph for each function listed in the command summary. This paragraph tells when the commands that set the different modes for that function affect the text.

SOME COMMANDS REQUIRE VALUES

You already know that there are internal commands which set the formatter functions in their start-up modes. There are also default values that are set internally. These default values are for commands which require values to clarify their functions. When you embed a command that requires a value, you have the choice of supplying the value you want or allowing the default value to be applied. The value follows the name of the command inside the curly brackets. There is a space between the name and the value, i.e. {skp 12} The Skip command is an example of a command that requires a value. I t causes the formatter to skip whatever

number of lines is specified in the command. If no value is supplied, the internal default value for Skip is applied. The internal default value for the Skip command is 1.

HOW THE COMMANDS ARE CANCELLED

Formatter commands which establish some formatting mode other than that established by the internal commands are either one-time-only commands or they require cancellation. One-time-only commands are entered in the text at the appropriate spot. When the text is formatted, they have a one-time-only effect on the document. All other commands must be cancelled by a command which sets the function in a different mode or by a command which sets the function in the off mode. In the command summary we explain how each command is cancelled.

COMMAND EQUIVALENTS FOR LONG-TERM FUNCTIONS

There are command equivalents for the values that are entered into the Environment Form. We have already explained that the formatter turns the Environment Form values into curly bracketed commands. You may need to know the equivalent in the event that you want to over-ride a long-term mode in the document file itself. For example, you may want to change a header or a footer that is defined in the Environment Form because you want a page or a series of pages to have a different header. In a case like this, you need to embed the command equivalent for the long-term mode in the text. We tell what these equivalents are in the command summary.

FORMATTING LONG DOCUMENTS

When you want to format a long document that has more than one text file, there are some special procedures you should know about.

First, you can select as many files as you want for formatting. The Storage Manager Form tells you to type Functon Key II ESC S to select a file for formatting and Function Key IV ESC CNTL E when you are through selecting files. Select as many files as you want, in the order that you want them to be formatted. That is how we printed this document. We selected all of the files in the document in the order that we wanted them printed.

If you are creating text files that you know are going to be a part of a long document, there are some pointers we can give. Restrict an individual file to an individual portion of the document. For example, we have a file for each section of this document.

Place a Begin Page command at the end of each file unless you want the next file to be added to the last file on the same page.

Use the "Only format a part of a document" option when you have printed the whole document and you find you have some errors on one of the higher numbered pages. This prevents you from having to re-run the entire document.

If you want to place headers or footers in your document, read the Header and Footer section of the command summary. You can also place a Reset Page Number command at the beginning of each file of the long document if you want each section to begin with a page 1.

SUMMARY OF FORMATTING TERMS

Functions: The formatter operation is a composite of several functions which all operate simultaneously.

Modes: Each of the formatter's functions has at least two modes. The internal commands cause each of the functions to be set in a particular mode when the formatter is invoked. The user can change the mode of a function with the appropriate command.

Commands: There is a command for every mode.

Curly brackets: Commands are enclosed in curly brackets. For example: {bold}

Values: Some commands require values to clarify their function. For instance, an indent command requires a value which tells the formatter how many spaces the indent should be.

Word, Line, and Page Breaks: A break is the beginning of a new word, line or page. Even though there are no commands embedded in the text the formatter's normal operation includes word, line and page breaks. Breaks can also be generated by commands when it is desired that the formatter begin a new line or page in a particular spot rather than the next time a line or page break occurs as a result of its normal operation. The sole action of some of the commands is to cause line or page breaks at a specific point. There are other commands which must cause a line or page break in order to accomplish some other kind of text arrangement; i.e. the indented first line of a paragraph, or the centering of text. In the next section we explain when the formatter does breaks without commands and how commands which cause breaks affect the output.

Printer Instruction Commands: Some functions control the aspects of the printing process, i.e. whether or not the printer will pause at the end of each page so that the user can change paper, which printer type is being used, whether or not the error messages are printed in the text as well as on the screen, and whether or not the formatting process terminates when a user error is discovered. These functions are explained and listed in the Formatter Command Summary.

Text Appearance Commands: Some commands change the way the printer would normally print the characters, i.e. red or black type, bold or normal darkness of type, and super or subscript type. We call these text appearance commands. They are listed and explained in the Formatter Command Summary.

Default Value: All commands which require values , have an internally set default value which is used for that command if it is given in the text without any value at all.

Action: Each command has a specific effect on the text because it causes one of the formatter's functions to be set in a different mode. The effect that the command has is a result of its action. In the Formatter Command Summary we explain what the action of each command is.

Long-Term Functions: Some functions usually remain constant throughout the document. In order to have a uniform effect on the document, they must be processed by the formatter before it begins to process any of the document text (margin and line spacing commands are examples of long-term commands.)

Environment: The environment for a document is established by the settings for the long-term functions. The internal commands establish settings for these functions, but these internal settings are immediately revised by the commands given to the formatter by the environment values file. The long-term functions which determine the environment are all listed in the Environment Form.

Environment files: You can create as many environment values files as you want, following the procedures in the Printer Functions section. You'll probably want to create an environment values file for every major document type that you work with.

Then when you want to format a document, select the "Load a predefined environment" option, view the environment values files that are listed in the Storage Manager Form. Select the appropriate one for the job you are about to do. Then, if you think you may want to make some changes to this special-purpose environment file, you can select the "Define or view an environment" option and make the changes using the Environment Form. You can save your changes in the same file, thus altering that file, or you can use your changed file temporarily (while it is in memory) to format your document.

Input line: One line of the text file is an input line.

Output line: One line of the formatted text is an output line. The formatter's Word Fill mode causes the formatter to place as many words on a line as it can until it reaches the width limit, so when Word Fill is in effect, the ouput lines do not have exactly the same words on them as the input lines. No Word Fill causes the same words that are in an input line to be placed on an output line. When the Word Fill or No Word Fill modes have determined what words are to be placed on a line, the justification mode determines how they will be placed on the line. This is explained in detail in this section.

FORMATTER COMMAND SUMMARY

In the following list, the commands are divided into groups according to function. The groups are as follows:

Printer Instruction Commands
Page Numbering Commands
Special Formatter Symbols
Singular Word Fill and Justification Commands
Word Fill and Justification Composite Commands
Margin Commands
Header and Footer Commands
Indent Commands
Line or Page Break Commands
Word Appearance Commands

PRINTER INSTRUCTION COMMANDS

The printer instruction functions affect the printing process. All but two of them are long-term functions and are determined by the Environment Form. One function that is not determined by the Environment Form is invoked by you when you select the "Only format a part of a document" option. The other function that is not long-term is the Page number function. It is either embedded in the text or in the header or footer. Since it is often embedded in the header or footer it has long-term effect on the document in the same way that headers and footers do.

Even though these functions almost always remain constant throughout a document, we provide you with command equivalents just in case you need to change the function part-way through the printing of a formatted document. These command equivalents can be placed in the document file itself.

The functions described in this section are as follows:

Print errors
Pause after each page
Stop on errors
Characters per inch
Printer type
Only format a part of a document.

Print errors

{errpr}

ACTION: The Print errors function can be either on or off. When it is on, it causes errors to be reported on the printed document as well as on the screen.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command causes errors to be reported to the screen only.

BACKGROUND: The System 88 Formatter provides error messages for essentially two kinds of user mistakes. The user might provide a command that the formatter does not understand, in which case a message like the following results:

```
     [file=<2<filename.TX, page=1, line=2][balk]
     *I don't recognize this command.*
```

If you provide an indent command with a value that is larger than the maximum characters per line limit, then clearly the formatter can not implement your command. This is the second kind of error. It results from your asking the formatter to do something that is impossible. It will do something that it considers to be the most reasonable under the circumstances and then send an appropriate error message like the following example:

```
     [file=<3<Gilcrest-letter<.TX, page 3,line 3Ø][ind 8Ø]
     *Your indent value is greater than your width.*
```

When you see one of these two error messages, you should realize that the formatter could not be doing what you want since it recognizes that you have made an error and is performing in what it considers to be an acceptable alternative mode. Use the error isolation message to find the spot in the input where you placed the unknown or impossible command.

Note that the first line of the error message tells the user the name of the file containing the command, the page number of the printed text where the command is located, the line number of the text file where the error occurred and the incorrect user command is placed in brackets at the end of the error message line. The second line of the error message contains an English explanation of the user's error.

The Print errors function is helpful for the first runs of documents which may contain many user errors. Since the screen can only hold a limited number of error messages, the top message is eventually scrolled off the screen as messages are added to the bottom. If there are more errors than will fit on the screen at one time, the user might miss the earlier messages unless he is watching the screen during the formatting process. The Print errors function avoids this inconvenience as the errors are printed right in the text at the location where the incorrect user instruction (embedded command) occurs.

PLACEMENT AND EFFECT: Error Print is a long-term function. There is a control for it on the Environment Form. Answer "Y" to the "Error print" option and the formatter will send errors to the printer as well as to your screen. It affects the entire document.

COMMAND EQUIVALENT: The command equivalent is {errpr}.

Stop on errors

{errst}

ACTION: Stop on errors is a long-term function. It has two possible modes, on or off. When it is on, Stop on errors causes the printing function to halt when the formatter finds an error. You have two choices when the formatter finds an error and halts. Check over your output and determine if you want the formatting run to continue or not. If you want to continue formatting anyway, strike any key. If you want to abort the process and fix the error, type a CTRL-Y.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command for Stop on errors sets it in the off mode as does our default values file.

PLACEMENT AND EFFECT: Stop on errors is a long-term function. There is a control for it in the Environment Form. Enter a "Y" after the Error stop option in the Enviornment Form and the printer will stop when it finds a user error. User errors that are caught by the formatter are explained under the above command description for Print errors.

COMMAND EQUIVALENT: The command equivalent for Stop on errors is {errst}. If it is placed in a text file, it takes effect immediately, causing the formatting process to halt when a user error is discovered by the formatter.

Pause after each page

{pause}

ACTION: This long-term function causes the formatter to pause at the bottom of each page right before it would normally advance to the next page. This function is useful when single sheet paper is being used for the printing. When the formatter has paused, the paper can be removed and a new one inserted. Once the paper has been changed, the user can type any key to cause the formatting process to continue.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command for this function sets it in the off mode as does our default values file.

PLACEMENT AND EFFECT: Pause after each page is a long-term function and there is a control for it in the Environment Form. Answer "Y" to the "Pause after each page" option and the formatter will pause at the end of each page until you type any key to continue.

COMMAND EQUIVALENT: The command equivalent for Pause is {pause}. If it is placed in the text, it takes effect after the formatter has done one regular page advancement, causing the printer to halt at the end of the page and wait for the user to type any key to indicate that the printer should advance to the next page.

CANCELLATION: It is possible to turn off Pause during the formatting process by entering an embedded command in your text. The embedded command is as follows:

```
     {pause Ø}
```

The above command takes effect right after the formatter does its next pause. (After it pauses one more time it will stop pausing.)

Line Spacing

{lsp n}

ACTION: The Line Spacing command is used to change the formatter's internal command of single spacing. The alternatives are double and triple line spacing.

INTERNAL COMMAND OR DEFAULT: The internal command for this function sets it in the single spacing mode as does our default values file.

PLACEMENT AND EFFECT: The {lsp} command is a long-term command. There is a control for it in the Environment Form. Type in the value 1, 2 or 3 after the "Line spacing" control.

COMMAND EQUIVALENT: The command equivalent for Line spacing is {lsp n}. The numbers 2 or 3 must be entered for n. If no value is provided, the internal default value of 1 is applied.

If this command equivalent is placed in the input file it causes an immediate line break and the text following the command appears at the beginning of the first line after the appropriate spacing is accomplished.

CANCELLATION: Any of the three modes cancels the other two.

Characters Per Inch

{cpi n}

ACTION: The Characters Per Inch command establishes how many characters the printer will place in one horizontal inch.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command for this function sets it in the 1Ø characters per inch mode as does our default values file.

PLACEMENT AND EFFECT: Characters per inch is a long-term function. There is a control for it in the Environment Form. Enter a 1Ø or a 12 after the "Characters per inch" control and your documents will be printed accordingly.

COMMAND EQUIVALENT: The command equivalent for this function is {cpi n} where n is 1Ø or 12. If no value is given, the default value of 1Ø is applied.

When the command equivalent is placed in the input text it takes effect immediately, causing a line break. The text immediately following the command is affected by it.

CANCELLATION: Either of the two modes cancels the other.

"Only format a part of a document"

{pgstart n} and {pagestop n}

ACTION: When the "Only format a part of a document" option is selected from the main function menu, the following prompts are displayed:

From which page?

To which page?

Type the number of the first page to be printed after the first prompt and the number of the second page to be printed after the second prompt.

BACKGROUND: Normally, when you invoke the formatter you ask it to format an entire document. Sometimes, however, it is convenient to be able to tell the formatter that you only want one page or some fraction of the total number of pages printed. The "Only format a part of a document" funcion is used for this purpose. Let's say that your document is three text files long, and the two pages you want to format are in the third file. You will still want the formatter to progress through the first two files so that it processes all of your commands and arrives at the page you want printed with the correct commands in effect. Select the "Only format a part of a document" function and enter the approriate page numbers after the prompts. Then select all of the files from the Storage Manager that contain the document from its beginning though the portion that is to be printed. For instance, if you want to print pages 75 through 85 and there are 3 document files that contain the text of your document from its first page through its 85th page, you would select all three files from the Storage Manager Form.

Clearly this function saves paper and ribbon and time when all but a few pages of a document are the way you want them and you only need to re-print a fraction of it.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command for this function is to instruct the printer to print every page of the document as does our default values file.

PLACEMENT AND EFFECT: Each time a partial printing is desired, the "Only format a part of a document" option must be selected.

COMMAND EQUIVALENT: The command equivalents for this function are {pgstart} and {pgstop}. If they are placed at the top of a file, they will have the same effect as does the partial print option. It makes no sense to use these commands unless WPS is not being used and the formatter is being run from Exec. These are considered long-term commands and therefore should be placed in the FORMAT.IN or alternative environment file, or at the top of the first document file to be formatted.

Printer type

{diab}, {stan} and {spin}

ACTION: Printer type is a long-term function that tells the formatter what type of printer is being used. It is important the formatter have this information when special text appearance modes are being used.

All popular printers have been categorized by us for the purpose of this function. Every popular printer is either a "diab", a "stan", or a "spin".

INTERNAL COMMAND OR DEFAULT: The internal command sets this function in the "stan" mode as does our default values file.

PLACEMENT AND EFFECT: This is a long-term function and there is a control for it in the Environment Form. Enter either "diab" "stan" or "spin" after the Printer type function control. Your dealer should tell you which type your printer is.

COMMAND EQUIVALENT: Since the entire document is printed with one printer, there is never any reason to enter the command equivalent for this function in the document file. However, there is a command equivalent which can be used if Exec is being use to run the formatter. The command equivalent is {diab}, {stan}, or {spin}.

PAGE NUMBERING COMMANDS

The next two pages describe page numbering commands which determine where and how the formatter numbers pages.

First page number

{rpgn n}

ACTION: The First page number function allows you to tell the formatter that you want it to consider the next page it formats to be some page number other than what it would automatically assume. This is a long-term funtion and has a control in the Environment Form. If you supply a value of 6 after the control in the Environment Form, the first page of your document will be given the number 6. Even though the page is numbered 6 its text is the first page of text in your file. The formatter does not actually place page numbers on your pages unless you request it to. We explain how to do this in this summary when we explain the header and footer functions. Whether or not you request that page numbers be placed on every page, the formatter always keeps track of what page is being printed just in case you want to place a page number on some page in the middle of the document, or in the event that you ask it to print only some portion of the document. If either of these events occur, the formatter checks to see if you have used the First page number function to specify a number other than 1.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command for this function causes the first page of the document to be numbered 1, as does our default values file.

PLACEMENT AND EFFECT: First page number is a long-term function and so it has a control in the Environment Form. It can change the formatter's normal mode for page numbering by making the first page of the document some page other than 1.

COMMAND EQUIVALENT: The command equivalent for First page number is {rpgn n} where n is the value of the first page number printed after the appearance of the {rpgn} command in the text file. Although First page number is most frequently used as a long-term command, it is sometimes necessary to change the page numbering in the middle of a document (at the beginning of each section for instance.) If the {rpgn n} command is placed in a text file, the formatter will begin numbering its pages accordingly as soon as it begins a new page.

Page Number

{pno}

ACTION: The Page Number command tells the formatter to insert the current page number in place of the command.

BACKGROUND: The most efficient means of telling the formatter to place page numbers on every page of your document is to define a header or footer command which contains the {pno} command (see the explanation of Headers and Footers later in this summary.)

INTERNAL COMMAND OR DEFAULT: The formatter's internal command for this function sets it in the off mode. The formatter does not put page numbers on the pages unless instructed to do so by the embedded command.

PLACEMENT AND EFFECT: The {pno} command is always used as an embedded command. It takes effect immediately causing the printer to print the current page number in place of the command itself. It has a one-time-only effect on the text, unless it is placed in a header or footer definition, in which case it functions every time a header or footer is printed on a page. See the description of the header and footer commands. These are long-term functions and since they frequently include the {pno} command, it too is often considered a long-term function.

CANCELLATION: This command has a one-time-only effect on the text unless it is placed inside a header or footer. The only time it makes sense to cancel it is when it is used in a header or footer. Headers and footers can be cancelled by the No Header or No Footer command.

SPECIAL FORMATTER SYMBOLS

The next three pages describe three special symbols which the formatter understands.

Non-Expandable Blank

~

ACTION: The non-expandable blank symbol tells the formatter that it is to leave one and only one blank space in place of the symbol. Two non-expandable blank symbols in a row tell the formatter to leave two, and only two, blank spaces in place of the special symbols.

BACKGROUND: This symbol is most often used when the Justity Expand or Fill modes are in effect. Both of these modes cause the formatter to insert extra blanks where there are blanks in the input text in order to bring the left and right ends of the output lines even with the left and right margins. There are some spaces which should not be expanded and the non-expandable blank symbol prevents the formatter from doing so.

PLACEMENT AND EFFECT: The non-expandable blank special symbol is placed where it is desired that there be only one blank space in the formatted output.

EXAMPLE: The following input technique is used to cause there to be only one space between the words in the above paragraph heading.

```
     PLACEMENT~AND~EFFECT:~
```

Curly Brackets

{}

ACTION: The left curly bracket tells the formatter that the characters following it are to be one or more commands. If the characters following the left curly bracket do not form one or more commands that the formatter recognizes, then the formatter will send an error message to the screen and to the formatted text if the Error Print function is being used. The right curly bracket signals the end of a command or command string. There can be more than one command within one set of curly brackets if each is separated from the other by a comma.

EXAMPLE: {bold} or {ind,bold,wund}

Back Slash

\

ACTION: The Back Slash symbol tells the formatter that it should print literally the special formatter symbol that follows it. You may be wondering how we can even print curly brackets in our formatted text; we precede each one with a \. We printed the \ by entering two of them in a row in our input text.

PLACEMENT AND EFFECT: The Back Slash is entered right before the special symbol that is to be printed literally.

EXAMPLE: In order to print a curly bracketed command in our manual we enter the following: \{dind}

SINGULAR WORD FILL AND JUSTIFICATION COMMANDS

Word fill and justification commands control the way the words are placed on lines. Word fill functions determine whether the input line (one line of the text file) will become an output line (one line of the printed, Formatted copy) as is true with the No word fill Mode, or whether the Formatter places as many words as it can on a line until it reaches the width limit, ignoring the RETURNs in the text file, as is true with the Word Fill mode. Justification controls where the words that are to be placed on one line will go. Lines of printed text can be justified to the right, left, center, or expanded to both the right and left at the same time.

There are composite (justification/word fill) commands and there are singular commands which affect either word fill or justification. The composite commands represent the most frequently used, practical combinations of word fill and justification commands. Most of the time, you will be using the composite commands. We provide the singular command capability, just in case there is some special arrangement you desire that we do not make possible with our composite commands. The Formatter's internal command for word fill and justification is {fill} which is a composite command. This is also the agrument provided by us in the default values file. Any of the other composite commands will over-ride this default completely. A singular justification or word fill command will only over-ride one part of the fill command's double function. This is an important point to remember when you are using the singular commands. In order to see what a singular command does, embed the {nfil} command in your sample text file. This command turns off both parts of the start-up formatting mode. Now provide any singular command and compare the two outputs.

Before you read the detailed descriptions of each command that sets a word fill or justification mode, you should check over the Composite Word Fill and Justification Commands Demonstration. Notice the effect of each composite command. You will rarely need to use the singlular commands, and the demonstration of the effects of the composite commands will help you decide which command to use for your different needs. Once you have a good idea of the effects of each of these commands, you can go back and read the individual command descriptions.

Word Fill

{wf}

ACTION: Word Fill causes the Formatter to place as many words on a line as it can until it reaches the maximum character width specified by the "Characters per line" control. This means that very irregular lines in a text file are corrected in the formatted document. It also means that the Formatter, when filling, ignores the way words (in the text file) are distributed on lines. If you have extra spaces and TABS in a line (in your text file) the formatter will throw them out and place words one after the other in a regular fashion, making the left and right margins even. The Formatter can not break words; if placing one more word on the line being filled would cause that line to be longer than the width in effect, the Formatter must begin a new line before it prints the next word. This means that the number of characters on a line will vary somewhat. When filling, the Formatter subtracts the right and left margins, and any indents in effect, from the specified width to arrive at the number of characters to place on a line. Spaces (blank characters) are counted as characters too in this filling process. If the specified width is 85 and the right and left margins are 1Ø, the Formatter will place 65 characters on a line.

Word Fill is handy for producing pages of even looking lines. There are other modes that are appropriate for charts and other text arrangements that cannot be obtained when Word Fill mode is in effect.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command sets this function in the Word fill mode, as does our default values file. Actually the internal command is a composite command which turns on Word Fill as well as justify expand. This is also true of our default values file instruction for this function.

PLACEMENT AND EFFECT: Once the standard formatting mode has been determined from the Environment Form, you may want to change the formatting mode at specific points in the document to allow different word fill and justification modes to take effect. This you do by inserting embedded commands. The command for Word Fill is {wf}, but since Word Fill is actually turned on by the default values file composite word fill and justification command {fill}, you will probably want to use a composite command to change , the formatting mode. You may have some unusual need that is not met by any of the composite commands, in which case you should embed singlular word fill and justification commands.

The placement of a word fill or justification command at the beginning, middle, or end of a line causes an immediate line break and the immediate implementation of the new mode. The text immediately following the command appears at the beginning of the next printed line and is affected by the command.

CANCELLATION: {wf} can be cancelled by {nfil},{nwf}, {ce} or {exp}. If you place any of these commands in the Environment Form after the Standard formatting control function, remove the curly brackets. The Environment Form will add brackets to your entry when it passes it to the formatter.

See the Singular Word Fill and Justification Commands Demonstration.

No Word Fill

{nwf}

ACTION: No Word Fill causes the Formatter to make the input line equal to the output line, what ever appears in the text file on one line will appear on one line of the formatted copy. This means that if there are spaces before the first characters on an input line, or if there are extra spaces between words on a line, those spaces will be placed in the same positions on the output line.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command sets this function in the Word Fill mode, as does our default values file. Actually, the internal command is a composite command which turns on word fill as well as justify expand. This is also true of our default values file instruction for this function.

PLACEMENT AND EFFECT: Once the Standard formatting mode has been determined by the Environment Form, you can cause a specific portion of text to be formatted in a different mode by placing a formatting mode command in the text file. Since {nwf} only cancels a part of the standard formatting mode composite command {fill}, you probably will want to use one of the composite commands that includes {nwf} They are {nfil}, {exp}, and {ce}.

Position the command directly in front of the text to be affected. The text immediately following the command will appear at the beginning of the next printed line and will be affected by the command.

CANCELLATION: {nwf} can be cancelled by {fill}, {wf}, or {fjl}.

See the Singular Word Fill and Justification Commands Demonstration.

Justify Left

{jl}

ACTION: Justifies (positions) to the left, whatever text is to be on one line of output.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command sets this function in the Justify Expand mode, as does our default values file. Actually, the internal command is a composite command which turns on word fill as well as justify expand. This is also true of our default values file instruction for this function.

PLACEMENT AND EFFECT: Once the Standard formatting mode has been determined by the Environment Form, you can change it with an embedded command. The {jr} mode turns off the justify expand portion of the {fill} mode. If given after the {nfil} mode has been set, the {jr} command causes the no filled lines to be justified to the right.

Position the command directly in front of the text to be affected. The text immediately following the command will appear at the beginning of the next printed line and will be affected by the command.

CANCELLATION: {jl} can be canceled by {ce}, {jr}, {je}, {jc}, {exp} or {fill}.

See the Singular Word Fill and Justification Commands Demonstration.

Justify Right

{jr}

ACTION: Positions text, that is to be on one output line, to the right. so that the last character in the line is against the right margin.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command sets this function in the Justify Expand mode, as does our default values file. Actually, the internal command is a composite command which turns on Word Fill as well as Justity Expand. This is also true of our default values file instruction for this function.

PLACEMENT AND EFFECT: Position the command directly in front of the text to be affected. The text immediately following the command will appear at the beginning of the next printed line and will be affected by the command. This command is most often embedded in the text file.

If the Fill mode is in effect, the {jr} command will cancel the Justify Expand portion of this composite command. Text will be filled and will be justified to the right only. If the No Fill mode is in effect, the {jr} command will cancel the Justify Left portion of the this composite command. Text will not be filled but it will be justified to the right only.

CANCELLATION: {jr} can be canceled by {jl}, {fjl}, {jc}, {je}, {nfil}, {ce}, {exp}, or {fill}

See the Singular Word Fill and Justification Commands Demonstration.

Justify Center

{jc}

ACTION: Positions in the center of the line any text that is to be output to that line.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command sets this function in the Justify Expand mode, as does our default values file. Actually the internal command is a composite command which turns on Word Fill as well as Justify Expand. This is also true of our default values file instruction for this function.

PLACEMENT AND EFFECT: Position the command directly in front of the text to be affected. The text immediately following the command will appear at the beginning of the next printed line and will be affected by the command.

If this command is given while the Fill mode is in effect, it cancels the Justify Expand mode and text is justified to the center and filled. If it is given when the No Fill mode is in effect, it cancels the Justify Left portion of No Fill and the text is justified to the center and not filled.

CANCELLATION: This command can be cancelled by {jl}, {jr}, {exp}, {je}, {fill}, {nfil}, and {fjl}

See the Singular Word Fill and Justification Commands Demonstration.

Justify Expand

{je}

ACTION: Positions the line of input text on the printed output line so that the first and last characters of the line are against the left and right margins. This is accomplished by means of the insertion of extra blank characters where there are already blank characters between words.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command sets this function in the Justify Expand mode already. Actually, the internal command is a composite command which turns on Word Fill as well as Justify Expand. This is also true of our default values file instruction for this function.

PLACEMENT AND EFFECT: Position the command directly in front of the text to be affected. The text immediately following the command will appear at the beginning of the next printed line and will be affected by the command.

CANCELLATION: {je} can be canceled by {jl}, {jr}, {jc}, {fjl}, {exp}, {ce}, or {nfil}.

See the Singular Word Fill and Justification Commands Demonstration.

SINGULAR WORD FILL AND JUSTIFICATION COMMANDS

DEMONSTRATION

In order to show how singular word fill and justification commands function, we provide the following examples of possible combinations of them. Please note that, in each example, we use both a justification command and a word fill command. Since both are present, both parts of the Formatter's internal default instruction for word fill and justification are replaced by singular commands. The Formatter's internal instruction is equivalent to the composite command {fill} which set the Word Fill and Justify Expand modes. If only one singular command is embedded, half of the Formatter's default instruction would still be in effect.

The first sample paragraph below is a copy of the paragraph as it appears in the text file. Compare it with the other samples so you can see how the different combinations of singular commands affect the text.

Sample Paragraph as it Appears in the Text File

```
There are only a few concepts that you need to
grasp
before      you can     begin to effectively use
the System 88
Formatter.  First, you need to understand the formatter's
start-up state.
Secondly, you need to understand
how you can use the Environment Form
to establish a formatting environment.
Finally, you need to know how you can use embedded commands
to affect specific portions of the document or to
over-ride the long-term controls.
```

Word Fill and Justify Left

{wf,jl}

```
There are only a few concepts that you need to grasp before you
can begin to effectively use the System 88 Formatter.  First, you
need to understand the formatter's start-up state.  Secondly, you
need to understand how you can use the Environment Form to
establish a formatting environment.  Finally, you need to know
how you can use embedded commands to affect specific portions of
the document or to over-ride the long-term controls.
```

No Word Fill and Justify Left

{nwf,jl}

```
There are only a few concepts that you need to
grasp
before       you can     begin to effectively use
the System 88
Formatter.  First, you need to understand the formatter's
start-up state.
Secondly, you need to understand
how you can use the Environment Form
to establish a formatting environment.
Finally, you need to know how you can use embedded commands
to affect specific portions of the document or to
over-ride the long-term controls.
```

Word Fill and Justify Right

{wf,jr}

```
  There are only a few concepts that you need to grasp before you
can begin to effectively use the System 88 Formatter.  First, you
need to understand the formatter's start-up state.  Secondly, you
       need to understand how you can use the Environment Form to
   establish a formatting environment.  Finally, you need to know
 how you can use embedded commands to affect specific portions of
           the document or to over-ride the long-term controls.
```

No Word Fill and Justify Right

{nwf,jr}

```
                  There are only a few concepts that you need to
                                                            grasp
                before       you can     begin to effectively use
                                                    the System 88
        Formatter.  First, you need to understand the formatter's
                                                start-up state.
                                 Secondly, you need to understand
                             how you can use the Environment Form
                           to establish a formatting environment.
      Finally, you need to know how you can use embedded commands
               to affect specific portions of the document or to
                                over-ride the long-term controls.
```

Word Fill and Justify Center

{wf,jc}

```
 There are only a few concepts that you need to grasp before you
can begin to effectively use the System 88 Formatter.  First, you
need to understand the formatter's start-up state.  Secondly, you
   need to understand how you can use the Environment Form to
 establish a formatting environment.  Finally, you need to know
how you can use embedded commands to affect specific portions of
     the document or to over-ride the long-term controls.
```

No Word Fill and Justify Center

{nwf,jc}

```
         There are only a few concepts that you need to
                              grasp
        before       you can    begin to effectively use
                          the System 88
    Formatter.  First, you need to understand the formatter's
                        start-up state.
                Secondly, you need to understand
              how you can use the Environment Form
             to establish a formatting environment.
   Finally, you need to know how you can use embedded commands
       to affect specific portions of the document or to
                over-ride the long-term controls.
```

Word Fill and Justify Expand

{wf,je}

```
There  are  only a few concepts that you need to grasp before you
can begin to effectively use the System 88 Formatter.  First, you
need to understand the formatter's start-up state.  Secondly, you
need to understand how  you  can  use  the  Environment  Form  to
establish  a  formatting  environment.  Finally, you need to know
how you can use embedded commands to affect specific portions  of
the document or to over-ride the long-term controls.
```

No Word Fill and Justify Expand

{nwf,je}

```
There   are   only   a   few   concepts   that   you   need  to
grasp
before              you  can          begin to   effectively  use
the                           System                           88
Formatter.    First,  you  need  to  understand  the  formatter's
start-up                 state.
Secondly,         you         need         to          understand
how      you      can      use      the      Environment     Form
to       establish        a        formatting        environment.
Finally,  you  need  to  know  how  you can use embedded commands
to  affect  specific  portions   of   the   document   or   to
over-ride            the            long-term           controls.
```

COMPOSITE WORD FILL AND JUSTIFICATION COMMANDS

So that you don't always have to enter a separate command for word fill and one for justification, we have merged selected pairs of these singular commands and created composite commands. One composite command will cause the formatter to do the equivalent of one justification and one word fill command.

Most of the time you should be able to use a composite command to obtain the word fill and justification you desire. One composite command cancels out any other composite command.

The default environment values file instruction for word fill and justification is the composite command known as {fill}. You can stop or alter this standard fill function by entering a different composite command, or by entering a singular word fill command, a singular justification command, or both. Explained on the following pages are the composite commands. Each composite command stands for a specific pair of singular commands.

Fill

{fill}

ACTION: The Fill command causes the Formatter to do the Word Fill and Justify Expand modes. Whole words are placed on the ouput line until the number of the characters on the line plus the right and left margin values plus the indent values, if there are any, equal the prescribed page width. When adding a new word to a line will cause the line to be longer than the established width, the new word is placed at the beginning of the next printable line. When no more words can be placed on a line without violating the width, the line is justified to the right as a result of the insertion of extra blank spaces where there are already user inserted blanks.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command sets this function in the Fill mode as does our default values file.

PLACEMENT AND EFFECT: Position the command directly in front of the text to be affected. The text immediately following the command will appear at the beginning of the next printed line and will be affected by the command.

CANCELLATION: {fill} can be canceled by {nfil}, {exp}, or {ce}. It can also be canceled by {nwf} when used with any of the following: {jc}, {jr}, {jl}.

See the Composite Word Fill and Justification Commands Demonstration.

No Fill

{nfil}

ACTION: The No Fill command causes the Formatter to do the equivalent of No Fill and Justify Left. Words appear on the output line exactly as they appear on the input line. If there are spaces or tabs before the first characters on the input line they will still appear on the output line with the first space or tab being justified to the left margin.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command sets this function in Fill mode, as does our default values file instruction.

PLACEMENT AND EFFECT: If you want to change the standard formatting mode to No Fill, enter **nfil** after the long-term control for formatting mode. If you want to ask the formatter to print a specific portion of text in the No Fill mode, but want the Fill mode to remain the standard, position the command directly in front of the text to be affected. The text immediately following the command will appear at the beginning of the next printed line and will be affected by the command.

CANCELLATION: {nfil} can be canceled by {fill}, {exp}, {fjl} or {ce}. It can also be canceled by {wf} when used in combination with any of the following: {jr}, {jc}, and {je}.

See the Composite Word Fill and Justification Commands Demonstration.

Center

{ce}

ACTION: The Center command causes the formatter to do the equivalent of the No Word Fill and Justify Center commands. The words on one line of input will appear on one line of output. The words are centered on the output line.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command sets this function in Fill mode, as does our default values file instruction.

PLACEMENT AND EFFECT: If you want to change the standard formatting mode to Center, enter ce after that function control on the environment form. If you want to specify that some portion of text be affected by the command, position the command directly in front of the text to be affected. The text immediately following the command will appear at the beginning of the next printed line and will be affected by the command.

CANCELLATION: {ce} can be cancelled by {fill}, {nfil}, {fjl} and {exp}. It can also be cancelled by {wf} when used with any of the singular justification modes except {jc}.

See the Composite Word Fill and Justification Commands Demonstration.

Expand

{exp}

ACTION: The Expand command causes the Formatter to do the equivalent of the Justify Expand and No Word Fill commands. The words on one line of input will appear on one line of output. Blank characters will be inserted where there are input blank characters for the purpose of expanding the length of the output line to be flush with both the left and right margins.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command sets this function in Fill mode, as does our default values file instruction.

PLACEMENT AND EFFECT: If you want to change the standard formatting mode to Expand, enter exp after that function control on the Environment Form. If you want to specify that some portion of text be affected by the command, position the command directly in front of the text to be affected. The text immediately following the command will appear at the beginning of the next printed line and will be affected by the command.

CANCELLATION: {exp} can be canceled by {fill}, {fjl}, and {nfil}. It can also be canceled by {wf} when used in combination with any of the following singular justification commands: {jr}, {jc}, {je}

See the Composite Word Fill and Justification Commands Demonstration.

Fill Justify Left

{fjl}

ACTION: Causes the Formatter to do the equivalent of Fill and Justify Left. This is a useful alternative to the internal default instruction which is Fill. Fill Justify Left does not cause the insertion of blanks in order to justify the right edge of the line with the right margin. This makes it useful for letters and other documents where it is desirable for the lines to be regular but not absolutely straight along the right edge.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command sets this function in Fill mode, as does our default values file instruction.

PLACEMENT AND EFFECT: If you want to change the standard formatting mode to Fill Justify Left, enter fjl after that function control. If you want to specify that some portion of text be affected by the command, position the command directly in front of the text to be affected. The text immediately following the command will appear at the beginning of the next printed line and will be affected by the command.

CANCELLATION: This command can be cancelled by {fill}, {exp}, {ce}, {nfil} and by {nwf} when used in combination with {je}, {jc}, {jr}, or {je}.

See the Composite Word Fill and Justification Commands Demonstration.

COMPOSITE WORD FILL AND JUSTIFICATION COMMANDS DEMONSTRATION

Below are five samples of formatted output. Each is preceded by one of the Word Fill and Justification Composite commands.

The first sample paragraph below is a copy of the sample paragraph as it appears in the text file. Compare it with the other samples to see how the different composite commands affect the text.

Sample Paragraph as it Appears in the Text File

```
There are only a few concepts that you need to
grasp
before      you can     begin to effectively use
the System 88
Formatter.  First, you need to understand the formatter's
start-up state.
Secondly, you need to understand
how you can use the Environment Form
to establish a formatting environment.
Finally, you need to know how you can use embedded commands
to affect specific portions of the document, or to
over-ride the long-term controls.
```

Fill

{fill}

There are really only a few concepts that you need to grasp before you can begin to effectively use the System 88 Formatter. First, you need to understand the formatter's start-up state. Secondly, you need to understand how you can use the Environment Form to establish a formatting environment. Finally, you need to know how you can use embedded commands to affect specific portions of the document, or to over-ride the long-term controls.

No Fill

{nfil}

There are really only a few concepts that you need to
grasp
before   you can   begin to effectively use
the System 88
Formatter. First, you need to understand the formatter's
start-up state.
Secondly, you need to understand
how you can use the Environment Form
to
establish a formatting environment.
Finally, you need to know how you can use embedded commands
to affect specific portions of the document, or to
over-ride the long-term controls.

Center

{ce}

```
 There are really only a few concepts that you need to
                          grasp
    before       you can     begin to effectively use
                      the System 88
Formatter.  First, you need to understand the formatter's
                    start-up state.
            Secondly, you need to understand
          how you can use the Environment Form
         to establish a formatting environment.
                      Finally, you
                    need to know how
              you can use embedded commands
   to affect specific portions of the document, or to
            over-ride the long-term controls.
```

Expand

{exp}

There are really only a few concepts that you need to grasp before you can begin to effectively use the System 88 Formatter. First, you need to understand the formatter's start-up state. Secondly, you need to understand how you can use the Environment Form to establish a formatting environment. Finally, you need to know how you can use embedded commands to affect specific portions of the document, or to over-ride the long-term controls.

Fill Justify Left

{fjl}

There are really only a few concepts that you need to grasp before you can begin to effectively use the System 88 Formatter. First, you need to understand the formatter's start-up state. Secondly, you need to understand how you can use the Environment Form to establish a formatting environment. Finally, you need to know how you can use embedded commands to affect specific portions of the document, or to over-ride the long-term controls.

MARGIN FUNCTIONS

This section of the summary desribes the four margin functions. These are long-term functions and are listed in the Environment Form.

Right Margin, Left Margin, Top Margin, Bottom Margin

{rm n}, {lm n}, {tm n}, {bm n}

ACTION: These functions establish the margins for the formatted document. They are long-term functions which are listed in the Environment Form. However, it may be necessary to change one or more of the margins for some portion of the document printing. The commands can be embedded in the text if this is the case.

INTERNAL COMMAND OR DEFAULT: The formatter's internal commands for these functions are:

Right margin 1Ø
Left margin 1Ø
Top margin 3
Bottom margin 6

The original default environment file matches these internal margin settings. The values specified by these internal commands are defaults. The formatter will always apply the margins that are specified in the Environment Form. If you enter a margin command in your text file with a different value than it has in the Environment Form, your embedded command value will over-ride the Environment Form value. If you enter a margin command in your document file and you do not supply a value with it, the internal default value will be applied.

PLACEMENT AND EFFECT: When embedded in the text, the margin commands take effect the next time the formatter begins a new line. It is useful to embed margins in the text when it is necessary to increase or decrease the printable width of the lines without changing the "Characters per line" value in the Environment Form.

CANCELLATION: The margin commands are cancelled by other margin commands of the same type with a different value. When a margin command is supplied with no value, the Environment Form value for that function is applied.

HEADERS AND FOOTERS

This section of the summary describes the Header and Footer functions. These are long-term functions that are accommodated by the Environment Form. Frequently it is appropriate to embed header or footer related commands in the text file so that the Environment Form headers or footers can be temporarily or permanently cancelled. This is explained in full in the following pages.

Odd Header, Even Header, Header
Odd Footer, Even Footer, Footer

{ohe}, {ehe}, {he}, {ofo}, {efo}, {fo}

ACTION: This function specifies a segment of text (including the current page number if desired) to appear at the top and the bottom of every page. This is a long-term function because it affects the entire document. However, if desired, a header or footer can be changed, temporarily cancelled and started back up, or permanently cancelled by commands which are embedded in the text.

ENTERING HEADER AND FOOTER COMMANDS IN THE ENVIRONMENT FORM: The second and third pages of the environment form contain fields which are used to define headers and footers. There is a field which is to be used for the odd-page header definition, one that is to be used for the even-page header definition, and one that is to be used if a header is to be placed on every page. There are also three fields available for footer definitions, one for even-page footers, one for odd-page footers and one for every-page footers. You should be aware that you cannot enter a header definition in the field for every-page headers, and one in the field for odd page headers. They will both be printed on the odd pages, one over the other.

If you want a certain title page, number, or phrase to appear at the top or the bottom (or both) of every even, odd or all pages of the document, fill in the text you want to place at the top or bottom of the page; be sure to select the right field in which to enter this information.

If you have supplied a header or footer in the Environment Form, it will appear on every appropriate page in the manual, unless some embedded command prevents its printing.

When you are typing in the header or footer, there are several things to remember. You probably don't want your header or footer filled. If you don't provide a special word fill/justification command you'll get a filled header. The most common justificication commands used for headers or footers are {ce} or {exp}. Place these commands in the header or footer field before you begin entering the header or footer text. There are examples of each of these modes and their use in headers in the following pages.

The header or footer command can have blank lines and text, one or more lines of text, or just blank lines. This multiple-line header and footer capability is useful for printing headers and footers with more than one line, and for setting the header or footer text apart from the regular text with an appropriate number of blank lines. The headers in this manual have three lines, one line of text and two blank lines to separate the header from the regular text of the manual.

Usually, it is a good idea to skip at least one line (in this manual we skipped two) between between the headers and the regular text and between the footers and the reqular text. The formatter does not skip any lines for you. You have to instruct it in how many lines are to separate the header and the text and the footer and the text, by including a {skp n} command in your footer and header definitions. This command should come right after the header text in a header definition, and right before the footer text in a footer definition.

Each footer and header field is 3 lines long. This means you have room for a justification command, a skip command, and 3 lines of header or footer text.

Here are some header definitions used for this manual as they appear when entered in the Environment Form header fields.

```
{jl}{pno}  WordMaster II {skp 2}


{jr} Formatter Command Summary  {pno}
```

The following is a sample footer with the center justification command and the {pno} command which causes the current page number to be placed at the bottom center of every page. Note that we insert a skip command before the footer text to separate it from the regular document text.

```
     {jc}{skp 2} {pno}
```

CANCELLATION: Header and footer commands are cancelled by other header and footer commands with different definitions. They are temporarily cancelled by the No Header and No Footer commands. The syntax for these header on/off commands are as follows:
No Header and No Footer commands: {nohe}, {nehe} {nhe}, {nofo}, {nefo} and {nfo}.
Start Header and Start Footer commands: {sohe}, {sehe} {she}, {sfo}, {sefo}, and {sofo}.

PLACEMENT AND EFFECT OF HEADER AND FOOTER COMMANDS IN THE TEXT FILE: Headers and footers are normally controlled by the Environment Form. However, if it is desirable to replace the header or footer specified in the form with a different header or footer that is to appear at some point in the text, the appropriate header or footer command can be embedded in the text. When the header and footer commands are embedded in the text, they will take effect after the formatter does its next page break.

If you force a page break with a Begin odd page or Begin page command and then place any kind of Header or Footer command right after the page break command, the Header or Footer or No Header or No footer, or Start Header or Start Footer command will take effect on that same page.

The method for supplying Header and Footer commands in the text is different than defining them with the Environment Form. You must enter the appropriate Header or Footer command before the definition and you must follow the definition with an {end} command which indicates that the definition text is complete. All of the header and footer commands are listed at the top of this section. Each specifies even, odd or every and header or footer.

You can include skip, and justification commands in the header and footer commands that you place in the text file in the same way that you provide them in your Environment Form header and footer definitions.

The header and footer on/off (No Header and Start Header) commands always work on the last defined header that is of the type specified by the No Header or Start Header command, whether or not that last header or footer was defined in the Enviornment Form or in the text file.

CANCELLATION: The cancellation of headers and footers defined in the text is the same as described above for headers and footers that are defined in the Environment Form.

SUMMARY OF FACTS ABOUT DEFINING AND USING HEADERS: You can specify odd, even or every page headers. They can contain page numbers and can be justified to the center, to the right, or to the left of the line. Note the headers in this manual which are justified to the left on even pages and to the right on odd pages. Headers and footers can have multiple lines of text, a combination of blank lines and printed lines, or all printed lines. The blank lines are important for visibility and for distinguishing the header or footer from the regular text.

As you can see, headers and footers can be defined in either the Environment Form or the text file. If you are going to use the same header or footer throughout a document you can easily select the "Define or view an Environment Form" option and enter the header or footer. Even if you want some of the pages in the document not to have headers or footers, you can still define them in the Environment Form and then use the No Header or Footer and Start Header or Footer commands to control the turning on and off of the long-term header or footer at specific parts of the document.

If you want to have more than one header or footer of a specific (odd, even or every page) type in the document, you should enter the first one in the Environment Form and then enter its replacements at appropriate places in the text.

When embedding a header or footer definition in the text file, you must begin it with the appropriate header or footer command which specifies whether the header or footer is to be printed on even, odd or all pages. The last part of a header or definition that is embedded in the document file is the {end} command.

HEADER AND FOOTER EXAMPLES

There are many possibilities for header and footer commands. The best way to find them out is to try some, but to give you an idea this page has a few samples.

The following is a definition of an odd page header that is to be placed in a text file. Notice that it begins with the appropriate header command and ends with the {end} command.

```
{ohe}{jr}System 88 Formatter Command Summary  56


{end}
```

The above sample is the proper construction for an embedded header command. Notice the {ohe} and the {end} commands. These are required when the header is being defined right in the text file because the formatter must be able to tell the difference between your regular text and your header or footer text.

The {pno} (Page Number) command is used frequently in headers and footers. In the formatted text it is replaced with the current page number.

The {jr} command is used to turn off the {je} portion of the Fill mode which is automatically in effect for headers.

Below is an example of a footer with multiple lines of text.

```
{fo}{ce}


                         WordMaster II
                   PolyMorphic Systems{end}
```

If a multiple line header (with multiple lines of text is being defined, the formatting mode must be set in No Fill (i.e.{nwf,jr}, {ce}, or {nwf,jl}, {exp}) so that the lines will break in the desired spot. Rarely would you want the formatter to fill your header. Usually there are only a few words on each line and they are to be centered, or right justified.

The following input results in the header that is at the top of the even pages of this section:

```
{ehe}{jl}{pno} WordMaster II


{end}
```

Note the placement of the {end} command. The two blank lines (a {skp n} command is equally sufficient) make a satisfactory separation between the header and the regular text of the manual.

The following input results in an expanded header:

```
{he}{exp}PolyMorphic Systems TwinSystem Page~{pno}


{end}
```

The non-expandable blank special symbols (~) are used to prevent the formatter from expanding the header in the wrong places, as the {exp} command causes the formatter to insert extra spaces wherever there are already spaces in order to spread the three parts of the header across the full length of the line.

Below is the formatted output of the above header as it would look if it were at the top of a page.

```
PolyMorphic Systems           TwinSystem           Page 57
```

INDENTATION COMMANDS

There are a number of indentation functions that allow you to present your text more clearly and attractively. Text can be indented from the left margin or from both the left and the right margins at once. There are special commands that cause temporary indents (one line only) that are used at the beginning of a new paragraph, or to set off an example.

Indent, Double Indent

{ind n}, {dind n}

ACTION: These two functions cause indentation. The Indent commmand causes the text to be indented n number of spaces from the left margin. The Double Indent command causes the text to be indented n number of spaces from both margins. The Standard Indent control in the Environment Form establishes the standard value for indents. If you want your indents to be 5, set this at 5. Then, when you are supplying the Indent and Double Indent commands, you do not need to specify a value with them unless you want to specify a value that is different from the standard. When an Indent or Double Indent command is issued with a value that is different from the standard indent value, that new value is in effect until the next Indent or Double Indent command is issued. Each time one of these commands is issued, it cancels out the command of the same type issued before it. The values are not added. So, if you supply an Indent command with a value of 8 ({ind 8}) and then an Indent command with a value of 10, the indentation will change to 10 spaces from the margin, not 18.

INTERNAL COMMAND OR DEFAULT: The internal commands for the Indent and Double Indent functions place them in their off modes. The Standard Indent function sets the value of n for these commands when they are supplied without a value. The internal default value of the Standard Indent function is 5. This value is matched by that placed in our default values file.

PLACEMENT AND EFFECT: Since these commands are meant to affect specific portions of text. They are always placed in the text file. They take effect the next time the formatter begins a new line. In order to know exactly when an indent command will take effect, (which text is the first to be indented) you should place it at the beginning of a line, and be sure that there is a line break command after the last line of text, this will cause the text immediately following the indent command to be affected.

There may be times when the first word that is indented is not the important factor. Perhaps you want to indent in order to hollow out a space for a diagram. In this case, you would not want to have a line break or blank line before the command. Just place the command anywhere on the line and the indentation will occur the next time the formatter begins a new line.

CANCELLATION: The No Indent command ({nind}) cancels both of these commands. It takes effect the next time the formatter starts a new line just as the Indent and Double Indent commands do.

Temporary Indent, Paragraph Indent

{tind}, {par}

ACTION: The Temporary Indent command causes an immediate and temporary indent to occur n spaces from the left margin. The {tind n} command requires a value, and when it is given without the value, the value of the Standard Indent control in the Environment Form is applied.

The Paragraph command causes the formatter to leave one blank line and and then to perform an immediate and temporary indent n spaces from the left margin or the present indent if one is in effect. The {par} command takes no value; the number of spaces it indents is dependent on the Standard Indent control in the Environment Form.

If there is already an Indent or Double Indent command in effect when the {tind n} or the {par} commands are issued, they will cause the formatter to indent n spaces from the present indent.

INTERNAL COMMAND OR DEFAULT: The formatter's internal commands for these two functions set them in their off modes. The number of spaces they indent is dependent on the internal default value for the Standard Indent function which is 5. Our default values file also gives a value of 5 to the Standard Indent function.

PLACEMENT AND EFFECT: Place these commands directly before the text that is to be affected by them. They cause a line break, and in the case of the {par} command two line breaks in a row are executed to leave a blank line.

CANCELLATION: Both of these commands have a one-time-only effect on the text. They do not need to be cancelled.

Word Undent

{wund n}

ACTION: When there is an Indent or a Double Indent command in effect, the Word Undent command is used to temporarily cancel the indent so that one word, or a series of words, can extend out to the left of the indented text n spaces. The first word after the command is extended, and the next word is placed at the current indent position. If no value is supplied for n the Standard Indent value determined in the Environment Form is applied. This function is useful for numbering or labeling filled blocks of text.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command for this function sets it in the off mode. The internal default of 5 for the Standard Indent function is applied when no value is given for n. Our default values file has the same default for the Standard Indent function.

PLACEMENT AND EFFECT: This command always affects specific portions of text, so it is embedded in the text file. It takes effect immediately, causing the word after it to extend to the left of the current indent. A word is any string of characters that is not interrupted by a space, TAB, RETURN, or other word or line break. If more than one word is to extend, non-expandable blank symbols must be inserted between the words in order to make them look like one word to the formatter. The second word after the Word Undent command is placed at the current indent.

The following input shows you how to provide a {wund} command.

{ind}{wund 3} 1. This is the first filled block of text in a series of filled blocks which we want to number. If you flip through this manual you will find several places where we have used this function.

{wund 3} 2. This is the second block of filled text in the series. The indent value is not provided with the command because we want the Standard Indent value to be applied. Note that we repeat the {wund} command each time. We use a value of 3 for the Word Undent command because we only want 3 characters to extend to the left of the indent.

The following is the result (output) of the above example:

1. This is the first filled block of text in a series of filled blocks which we want to number. If you flip through this manual you will find several places where we have used this

function.

2. This is the second block of filled text in the series. Note that we repeat the {wund} command each time. We use a value of 3 for the Word Undent command because we only want 3 characters to extend to the left of the indent.

Below is an example of the use of Word Undent to extend titles of filled blocks to the left.

The input:

```
{ind 10}{wund}
WordMaster II: This is PolyMorphic Systems word processing
system that is presented in a menu and form format.  This
application runs on all configurations of the System 88
including the low-cost Double Density 8810 and the
TwinSystem 88.

{wund}
DataMaster I: This package
allows the creation of data files through
simple, interactive procedures.  Once constructed,
these files can be altered to accommodate future
changes in the information or operational needs of
the System 88 user.

{wund}
TwinSystem 88: This system supports two independent users
on one microcomputer for a fraction of the cost
of a second microcomputer.  Each of the two users has
a separate video display board and 48K byte memory
board.  Both users share
the central processor, disk drives, and printer.  All these
shared resources are managed by the TwinSystem 88 for the two
users.{nind}
```

The output:

```
     WordMaster II: This is PolyMorphic Systems  word  processing
          system  that  is  presented  in a menu and form format.
          This application runs  on  all  configurations  of  the
          System  88  including  the low-cost Double Density 8810
          and the TwinSystem 88.

     DataMaster I: This package allows the creation of data files
          through    simple,    interactive    procedures.    Once
          constructed, these files can be altered to  accommodate
          future  changes in the information or operational needs
          of the System 88 user.
```

TwinSystem 88: This system supports two independent users on one microcomputer for a fraction of the cost of a second microcomputer. Each of the two users has a separate video display board and 48K byte memory board. Both users share the central processor, disk drives, and printer. All these shared resources are managed by the TwinSystem 88 for the two users.

CANCELLATION: Word Undent is a one-time-only command so it does not require cancellation.

Standard Indent

{sind}

ACTION: This function establishes a default value for all indentation commands. When the indentation commands are given without values, the value for the Standard Indent function is applied.

INTERNAL COMMAND OR DEFAULT: The internal default value for this function is 5. This is matched by our default values file.

PLACEMENT AND EFFECT: This is a long-term function because it establishes the default value for all indentation commands. If a different value is provided with a specific indentation command when it is embedded in the text, that value over-rides the Standard Indent value. There is a control for Standard Indent listed in the Environment Form.

CANCELLATION: This function is always in effect, but it can be temporarily over-ridden when a different value is entered with any embedded indentation command. It can be permanently over-ridden when the command equivalent of Standard Indent {sind n} is embedded in the text file and given a different value.

LINE OR PAGE BREAK COMMANDS

We have already explained that the formatter begins new lines when it reaches the "Characters per line" limit if it is in Fill mode, and when it sees a RETURN character when it is in the No Fill Mode. We have also explained that the formatter begins a new page when the total number of blank lines and printed lines equals the "Lines per page" value. You can interrupt this normal operation when it is important for you to begin a new line or page at a particular point in the text. This section tells which commands you can use to do this.

Line Break, Skip, Blank Line

{br}, {skp n}, RETURN RETURN

ACTION: These three functions all interrupt the normal fill or no fill process so that a new line can be started at a specified point, n blank line(s) can be left between lines of text or commands, and one blank line can be left between text lines when a blank line is created in the input text.

Inserting the Break command ({br}) in the text file causes the formatter to begin a new line immediately.

Inserting the {skp n} command in the text causes the formatter to begin a new line immediately, after skipping n line(s).

A blank line in the input file (two RETURNs in a row) equals a blank line in the output file.

INTERNAL COMMAND OR DEFAULT: The start-up state of the formatter causes it to begin new lines when it reaches the width limit specified in the Environment Form. The Break and Skip functions are set in their off modes. The formatter's start-up mode causes it to begin a new line on the next printable line as is dictated by the line spacing mode. The internal default value for the Skip command is 1. This means that if the Skip command is embedded with no specified value for n, the default of 1 is applied.

PLACEMENT AND EFFECT: The Break and Skip commands take effect immediately. Break causes a line break and skip causes two line breaks in a row n times. The text right after these command appears at the beginning of the new line.

CANCELLATION: The Break and Skip commands are one-time-only commands so there is no need to cancel them. The function that leaves a blank line in the output, if there is a blank line in the input, can not be cancelled. It is always in effect.

Begin Page, Begin Odd Page

{bpg}, {bop}

ACTION: The Begin New Page and Begin Odd Page functions cause the formatter to begin a new page at a specified point. The Begin Odd Page command causes the formatter to place the text following the {bop} command at the beginning of the next odd numbered page. The Begin Page command causes the formatter to begin a new page on the next page whether it is odd or even. If the formatter has to skip a page to begin on an odd page when the {bop} command is used, it places the header or footer (if any are defined) on the skipped page. This allows for continuity in the printing of a long document. We used {bop} commands at the end of every section of this manual to ensure that the first page of each section would be an odd numbered page.

INTERNAL COMMAND OR DEFAULT: The formatter's start-up state causes it to begin new pages when the total number of blank and printed lines on the page equal the "Lines per page" figure.

PLACEMENT AND EFFECT: The Begin Page and Begin Odd Page commands take effect immediately. The text directly following them appears after the header (if any is in effect) on the next printed page.

CANCELLATION: These are one-time-only commands. They need no cancellation.

Needs

{ne n}

ACTION: This function allows you to specify that you do not want the formatter to start formatting n lines of text unless there is enough room on the present page to fit all n lines. If there is not enough room for n lines, the formatter begins a new page and then formats the n lines.

The Needs command can be used in conjunction with the Skip command. It allows you to save n blank lines so that when the output copy has been created, a diagram can be inserted in the blank space.

INTERNAL COMMAND OR DEFAULT: The formatter's start-up mode causes it to begin a new page when the total number of blank and printed lines is equal to the "Lines per page" figure. The Needs function is in the off mode at start up. The internal default value for needs is 1. If no value is supplied with the Needs command the default is applied.

PLACEMENT AND EFFECT : This command should always be placed right before the n lines that it is to affect. If there is enough room on the page for n lines, those lines are printed on that page. If there is not enough room they are printed on the next page.

For example, the following input shows the right way to provide this command:

```
{ne 6}{nfil}
Justification Modes

1.  Justify Center
2.  Justify Left
3.  Justify Right
4.  Justify Expand
```

CANCELLATION: This is a one-time-only command that requires no cancellation.

WORD APPEARANCE COMMANDS

There are several commands which can be embedded in the text that cause specific words or sections of text to be printed differently than the formatter's default instructions call for. These commands cause underlining, bold face type, red or black type, super or subscript type, only one space to be printed after a period, or all capital letters. They are listed and explained in the following pages.

UNDERLINE and WORD UNDERLINE

{ul} and {wul}

ACTION: You can instruct the formatter to underline with the {ul} command which causes all spaces, punctuation and characters to be underlined. Or, you can instruct the formatter to underline with the {wul} command which causes only characters, not spaces and punctuation to be underlined.

INTERNAL COMMAND OR DEFAULT: The internal command for this function sets it in the Off mode.

PLACEMENT AND EFFECT: The {ul} and {wul} commands take effect immediately. Both the {ul} and the {wul} commands should be placed at the beginning of the text that is to be underlined. The {nul} command is then placed at the end of the text to be underlined. These commands do not cause the formatter to begin a new word or line. If one of these commands is placed in the middle of an input word, underlining will begin in the middle of the output word without any break occurring in the word or the line.

CANCELLATION: Both the {ul} and the {wul} commands are cancelled by the {nul} command.

See the following page for a demonstration of the uses of these commands.

UNDERLINE AND WORD UNDERLINE USES

In the following input, a portion of text, (spaces and punctuation included) is enclosed by the {ul} and {nul} commands.

In this sentence, {ul}this text (characters, punctuation
and spaces) is underlined. We can
continue underlining for as many {nul}lines as we want.

Below is the formatted output resulting from the above input example:

In this sentence, this text (characters, punctuation and spaces) is underlined. We can continue underlining for as many lines as we want.

In the following input a specific portion of text is enclosed by {wul} and {nul} commands.

In this sentence, {wul}this text (punctuation
and spaces excluded) is underlined. We can continue
underlining for as many {nul}lines as we want.

Below is the formatted ouput resulting from the above input example:

In this sentence, this text (punctuation
and spaces excluded) is underlined. We can continue
underlining for as many lines as we want.

Underline and Word Underline Uses Continued

Creating a line in No Word Fill mode

Sometimes you may want to create a line that is just a line, that does not underline any text. In No Word Fill mode all you have to do is enter an {ul} command followed by the number of spaces you need to make your line as long as you want. Follow the last space with a {nul} command.

Here is the input that accomplishes this:

```
{nfil,ul}                         {nul}
```

Here is the formatted output that results from the above input:

```
__________________________
```

Creating a line in Word Fill mode

Sometimes you may want to create a line that is just a line, that does not underline any text. In Word Fill mode, all you have to do is enter a {ul} command followed by enough non-expandable blank special symbols to make the line as long as you want. Follow the last non-expandable blank symbol with the {nul} command.

Below is the input that accomplishes this:

```
{fill,ul}~~~~~~~~~~~~~~~~~~~~~~~~~~~{nul}
```

Here is the formatted output that results from the above input:

```
___________________________
```

BOLD

{bold}

ACTION: The bold command causes your formatted text to be extra dark.

INTERNAL COMMAND OR DEFAULT: The formatter's internal command for this function sets it in the off mode.

PLACEMENT AND EFFECT: Bold takes effect immediately and stays in effect until the {nbold} command is given. It does not cause a word or a line break. If embedded in the middle of an input word, the output word will have bold letters starting at that mid-point. The formatter introduces no breaks in the word or the line in order to accomplish thıs action.

CANCELLATION: {bold} is cancelled by {nbold}.

Example: The title of this page is in bold type.

CAP

{cap}

ACTION: The Cap function causes all letters following it to be printed in upper case.

INTERNAL COMMAND OR DEFAULT: The internal command for this function sets it in the Off mode.

PLACEMENT AND EFFECT: This command takes effect immediately. The formatter does not introduce a word or a line break in order to perform this action. A Cap command can be embedded in the middle of an input word causing the output word to have all capital letters from that point on without the introduction of a word or line break.

CANCELLATION: {cap} is cancelled by {ncap}.

Example: The following line of input contains the {cap} and {ncap} commands:

```
The let{cap}ters immediately following the Cap command
are capitalized in the formatted output.  All letters
are capitalized {ncap}until the No Cap command is used.
```

The following is the formatted output that results from the above input text.

```
The letTERS IMMEDIATELY FOLLOWING THE CAP COMMAND ARE CAPITALIZED
IN  THE  FORMATTED OUTPUT.  ALL LETTERS ARE CAPITALIZED until the
No Cap command is used.
```

POINT

{point}

ACTION: The Point command is inserted instead of decimal points and periods that are used in abbreviations. Point is useful because the formatter's automatic default instruction is to place two spaces after sentence ending punctuation. The Point command prevents this. It is replaced by a period, and is followed by one, and only one, blank space. When the Point command is used when the Justify Expand mode is in effect, it prevents extra spaces from being inserted in the single space following the period that is inserted in its place.

INTERNAL COMMAND OR DEFAULT: The formatter's internal default that is relevant to this function is that two spaces are always placed after periods. Then, when Justify Expand is in effect, the formatter inserts more spaces where there are already spaces in order to accomplish right and left justification.

PLACEMENT AND EFFECT: The Point command is inserted in place of the period that is to be followed by only one space.

CANCELLATION: {point} is a one-time-only command requiring no cancellation.

Example: The following input shows the use of the point command:

```
J{point} R{point} Gilcrest
```

The following is the formatted output that results from the above input.

```
J. R. Gilcrest
```

RED and BLACK

{red} and {black}

ACTION: The {red} and {black} commands change the color of the print if a red/black ribbon is being used on the printer.

INTERNAL COMMAND OR DEFAULT: The internal command for this function is {black}.

PLACEMENT AND EFFECT: These commands take effect immediately. Place them directly before the text they are to affect. This formatter does not need to introduce a word or a line break in order to perform this action.

CANCELLATION: {red} is cancelled by {black} and {black} is cancelled by {red}.