THE WRITER'S TOOL

The most natural and complete word processor available for ATARI Home Computers!

Precision Software Tools

A TUTORIAL AND REFERENCE MANUAL

FOR

THE WRITER'S TOOL

VERSION 2.2

A Professional Word Processing Program

Designed For Use on Atari Home Computers

TRADEMARK ACKNOWLEDGEMENTS

MX-80, Graftrax-Plus, and FX-80 are trademarks of EPSON Corporation. PROWRITER is a trademark of C. ITOH Electronics Incorporated. NEC 8023-PC is a trademark of Nippon Electric Corporation. Microline 82A and Microline 92 are trademarks of OKIDATA Incorporated. ComRiter CR-II is a trademark of COMREX International, Incorporated. Gemini-10X is a trademark of Star Micronics, Incorporated.

The following are trademarks of Atari Corporation: ATARI, ATARI 800 Home Computer, ATARI 1200XL Home Computer, ATARI 800XL Home Computer, ATARI 810 Disk Drive, ATARI 1050 Disk Drive, ATARI 825 80-Column Printer, ATARI 1025 80-Column Printer, ATARI 1027 Letter-Quality Printer, and ATARI 850 Interface Module.

THE WRITER'S TOOL

TUTORIAL AND REFERENCE MANUAL

TABLE OF CONTENTS

(Page number letters are ordered by section name, not alphabetically.)

INTRODUCTION

TUTORIAL

REFERENCE GUIDE

INTRODUCTION

1.0 OVERVIEW

THE WRITER'S TOOL is a cartridge and disk-based machine-language program for fast, convenient, and integrated word processing on ATARI computers. With The Writer's Tool you can enter and edit text, save and retrieve text files on disk, and print text with a wide variety of formats. [The author used The Writer's Tool to write, edit, and print this manual.]

The Writer's Tool provides exceptional support for special printer capabilities, including multiple fonts, proportional spacing, graphic characters, graphic images, and many others too numerous to mention.

The Writer's Tool also includes a convenient spelling checker which operates in complete harmony with word processing functions.

The Writer's Tool is convenient as well as powerful. The simple command structure, the use of prompts and error checking, and a handy reference card minimize the need for memorization or frequent references to the user's manual. When you do need help, the extensive reference guide, and the example-filled tutorial encourage you to use all of the considerable powers of The Writer's Tool.

2.0 HARDWARE REQUIREMENTS

COMPUTER: The Writer's Tool can be used with any ATARI computer with at least 48K of memory. This includes the 1200XL and 800XL, as well as the ATARI 800.

DISK DRIVE: The Writer's Tool is provided in single-density disk format and can be used with the ATARI 810 or 1050 disk drives, or with a number of ATARI-compatible single-density and double-density drives.

PRINTER: Almost any printer can be used, provided it connects directly to the ATARI serial bus, or it has a Centronics-compatible interface which allows it to be connected to the parallel port on an ATARI 850 Interface Module, or equivalent. Special print features of most popular printer are fully supported.

3.0 HOW TO USE THIS MANUAL

Hands-on experience is the most efficient way to learn how to use The Writer's Tool. Begin by working through the first five sections of the **TUTORIAL.** This provides many work-along examples, and step by step instructions in the use of The Writer's Tool. Although the tutorial is mainly written for the novice user, it will also be useful to the advanced user.

Once you gain some experience, the **REFERENCE GUIDE** will become your primary source of information. The Reference Guide provides summaries of all commands and functions of The Writer's Tool, as well as more specific information about the printers supported by The Writer's Tool. This is primarily written for the experienced user.

As you explore the more powerful functions, you will probably use the reference guide for brief explanations and reminders, and the tutorial for detailed examples when you need them.

Don't be frightened by the amount of information presented in this manual. You don't need to know it all to make effective use of The Writer's Tool. Many of the functions are handled through prompts and menus. Once you get started, you may not even need the manual. When you do need to return to the manual, don't forget to use the **INDEX** at the back.

A **REFERENCE CARD** can be found in the inside pocket of the manual cover. This briefly summarizes most The Writer's Tool commands, and is useful as a reminder once you have learned how to use the program.

4.0 SUMMARY OF FEATURES

The following list highlights the features of The Writer's Tool. These are the essential ingredients which make it a word processor. Users who have no previous word-processing experience should not expect to understand immediately all of the features listed. This list is provided mainly to whet your appetite for learning how to use the program.

INTEGRATED FUNCTIONS

- Editing commands are available during text entry.
- Editing and printing functions are integrated into one program, allowing rapid transfers between them.
- Special support programs, such as the merged printing and spelling checker programs, are accessible from the word processing environment.

CONVENIENT TEXT ENTRY AND EDITING

- Two text entry options are provided (TYPE-OVER or INSERT), each with different flashing cursors.
- Dynamic word wrap keeps text readable at all times. Word wrap can be turned off for viewing and editing program files.
- Cursor commands allow movement to beginning or end of text, to top or bottom of screen, to next or previous screen, to beginning or end of line, to next word, to next or previous line or character.
- Insertions or deletions can be done by character, word, line, or block.
- Undelete commands can restore the last 5 deleted lines or words.
- A Join command deletes all spaces to next character.
- Text can be erased before or after cursor.
- Block commands include Mark, Copy, and Delete, or delete block markers. Marked blocks are highlighted in inverse video.
- The Search and Replace function allows search only, continue search, or search with replace, skip, or quit options. Wild card search characters are allowed.

- Special print format modifiers can be inserted with a single keystroke.

- Previously entered text can be converted from upper case to lower case, or vice versa.

- CAPS-LOCK, INVERSE VIDEO, and CURSOR EXCHANGE key states displayed when active. CURSOR EXCHANGE feature allows one-handed movement of cursor.

DISK FILE MANAGEMENT

- Disk directories are available without return to DOS. Directories can be displayed or printed in double column format. Two disk drives supported (can be configured for four drives).

- Disk commands include Initialize, Load, Save, and Delete disk files. Warning messages appear when text buffer may be in jeopardy. Save automatically uses name of last file specified (unless another name is chosen).

- The Writer's Tool text files are compatible with standard DOS files, allowing viewing and editing of a wide spectrum of disk files regardless of record length.

- Double density disk drives are supported. Drive density can be controlled from within the program.

FLEXIBLE PRINT FORMATTING

- Pica, Elite, Compressed, Proportional, and Correspondence Quality fonts are supported.

- Single imbedded characters are used to start/stop superscripts, subscripts, underlining, boldface, red print, italics, double-strike, and double-width printing.

- Greek/Math character sets of NEC 8023 and C.ITOH PROWRITER printers can be printed.

- Word wrap on printout can be turned off; useful for listing BASIC and other program files.

- Imbedded printer control codes and even graphics can be used without losing word-wrap and justification capabilities.

- Right justification, split justification, centering, and column alignment with imbedded tabs; supported for both proportional and standard character sets.

- Format parameters can be set externally or by means of imbedded format commands. Multiple format changes can be made in one format line.

- Print format options include font, page length, footer start, left margin, line length, indent, up to 8 tab positions, line spacing (four options), justification (four options), word wrap on/off, header and footer blocks with imbedded page numbers, text centering, vertical grouping, page eject, space between double columns, and reset to default values.

- Headers and footers can be of any length, with centered, left, or right-justified page numbers combined with textual information. Header and footer formats can be different from main text and from each other. Split justified lines can be alternated automatically on even pages.

- Soft hyphens can be inserted to produce automatic hyphenation. Hard spaces are also supported.

- Default formats allow printing most text without using format commands. Defaults can be customized by user.

- Tables and other text blocks can be automatically kept together on a single page.

- Print control includes continuous or wait-at-page-end options, abort or pause after any line, start at any point in text, and linked printing of a sequence of files.

WORD COUNTER

- Word count, character count, and memory usage displays are available.

SCREEN PREVIEW

- Page breaks and format commands can be verified by screen preview prior to printout. Printing can be deferred to any page number.

- A direct transfer from Preview to Edit functions allows rapid corrections of problems found during a preview.

LINKED PRINTING

- An unlimited number of files can be printed as one document. Disk files can be linked internally or externally and/or merged with text in the memory buffer or with keyboard input.

- Single or multi-line printer graphic images can be mixed with text during linked printing.

MERGED PRINTING

- Template documents can be selectively merged with data base items or keyboard entries during printout.

- Data base files can be created with the help of prompting derived from labels in a template document.

CUSTOMIZATION

- Cursor flash rate and luminance, screen colors and luminance, sound output, default printer, default format parameters can easily be modified.

- Printer and format default files can be loaded when desired or set to load automatically when the program starts.

- Printer data files (defining printer command codes) can be created and edited, allowing new printers to be supported, as well as the customization of existing printer files.

SPELLING CHECKER

- The spelling checker is accessible from the word-processing environment, and returns to that environment when checking and correction is completed.

- A master dictionary of 20,000 words is provided; any number of user dictionaries can also be created.

- During proofreading, a total word count and distinct word count is displayed.

- Errors can be corrected in context from within the spelling checker, or marked in inverse video and corrected from the word processing environment.

TUTORIAL

This tutorial provides a step-by-step tour through the main functions of The Writer's Tool. By detailed examples and illustrations we hope to encourage your full use of the considerable powers this program provides. You will often be given numbered steps to perform, as well as explanations and screen images of what should happen after each step. If you read through each step before performing it, you will be more aware of what's happening and less likely to make mistakes. But don't be afraid of making mistakes; it's a natural part of the learning process.

The tutorial begins by showing you how to load the program and get it running. This is followed by lessons on simple text entry and editing, on using the disk drive to save and retrieve text, and, finally, on using a printer to get your text on paper.

The essentials are covered in the first five sections. More detailed information about producing special print formats and using special printer capabilities is presented in Sections 6-9. Among the External Menu functions, you should not miss the **spelling checker** and **customization** programs described in Sections 11 and 12.

TUTORIAL CONTENTS IN BRIEF

1.0 LOADING THE WRITER'S TOOL

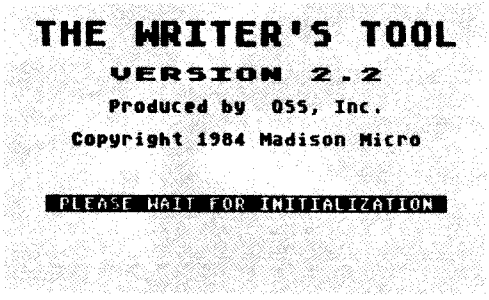
(1) Turn off your ATARI computer and insert The Writer's Tool cartridge.

(2) Turn on disk drive #1 and wait for the busy light to go out.

(3) Insert The Writer's Tool disk in drive #1.

(4) Turn on your computer.

The Writer's Tool program will now start loading. In about 10 seconds it will display the following **Sign-on screen:**



After another 20 seconds the program will complete the load process. You should then see a mostly blank screen with a flashing cursor in the upper left hand corner. This is the **EDIT screen.** If you do not see this screen, remove The Writer's Tool disk and carefully repeat the load procedure described above.

Once the EDIT screen appears, The Writer's Tool is ready for text entry and editing. [The message at the bottom of the screen should be ignored until later.]

2.0 ENTERING TEXT

Entering text is much like typing on a typewriter, except that you should

enter **RETURN** at the end of a **paragraph,**

but **not** at the end of every line. As you enter characters on the keyboard, they will appear on the screen at the position of the cursor (the flashing rectangle). If you type an incorrect character,

press **DELETE** or **BACK S** to move the **cursor back** one character,

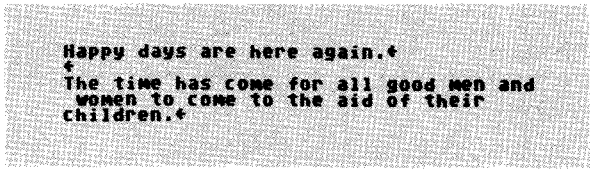
then type the correct character. Before pressing a letter key,

hold down **SHIFT** to produce an **upper case** character.

Now enter the following text:

> Happy days are here again.**RETURN**
> **RETURN**
> The time has come for all good men and women to come to the aid of their children.**RETURN**

Each **RETURN** will produce a **left arrow** on the screen, marking the end of a paragraph or a blank line. When you are done typing, the display should look like this:



3.0 EDITING TEXT

Editing what you've typed is accomplished by moving the cursor to the place that needs changing and then deleting, inserting, or retyping the characters needed. Exercises in this section will make use of the text entered in Section 2.0.

3.1 MOVING THE CURSOR

(1) Press and hold down the **CTRL** key located on the left edge of the keyboard.

(2) While you are holding down the **CTRL** key, press one of the four **arrow** keys to move the cursor in the direction of the arrow. If you keep the arrow key depressed longer than 3/4 second, it will begin to repeat.

(3) Move the cursor to the beginning of the text just entered, then try each of the following methods of cursor movement:

CTRL	+	**W**	to move to the **next word**
CTRL	+	**A**	to move to the **beginning of a line**
CTRL	+	**Z**	to move to the **end of a line**
CTRL	+	**B**	to move to the **beginning of the text**
CTRL	+	**E**	to move to the **end of the text**
TAB			to move in **5-character steps**

Enter **CTRL** + **RETURN** to move the cursor to the beginning of the following screen line. This has the same effect as **CTRL-A** and **CTRL-↓** in sequence.

3.2 INSERTING AND DELETING CHARACTERS

One way to insert a character is to press

CTRL + **INSERT** (**>**) to **insert a space** at the cursor,

then type the character in the space provided. You can use

CTRL + **DELETE** (**BACK S**) to **delete a character** at the cursor.

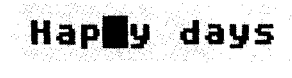
EXAMPLE

This example uses text entered in Section 2.0.

(1) Use **CTRL-[arrow]** to move the cursor to the 2nd 'p' in 'happy':

(2) Press **CTRL-DELETE** once. This will delete the 'p' and move the following characters one space to the left, leaving the cursor at the 'y':

(3) Press **CTRL-INSERT** once. This inserts a space at the cursor:

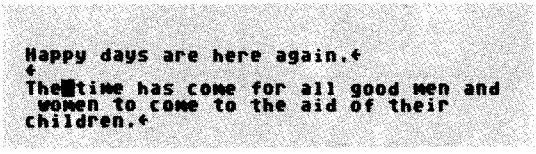


(4) Type a 'p' to restore the deleted character:

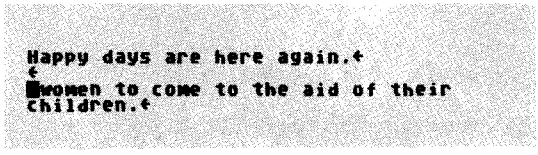
3.3 DELETING AND RESTORING WORDS AND LINES

The Writer's Tool can delete by words or lines and remember the last five deletions.

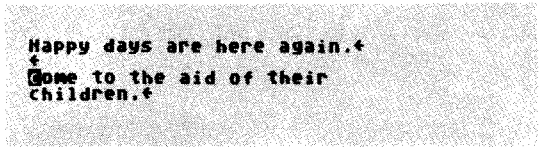
(1) Move the cursor to the first line of the second paragraph entered in Section 2.0 (the cursor can be anywhere on that line):



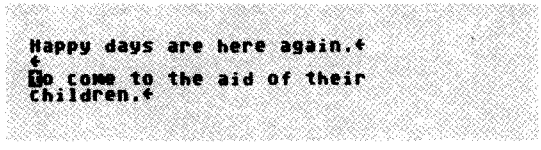
(2) Enter **SHIFT** + DELETE BACK S to **delete the screen line:**



(3) Enter **CTRL-D** twice to delete two words:



(4) Enter **CTRL-U** to **Undelete** the most recent deletion:



(5) Enter **CTRL-N** to restore the Next most recent deletion. Enter **CTRL-N** twice more to restore the remaining two deletions.

3.4 USING INSERT AND JOIN

INSERT AND JOIN is an editing technique which first inserts a large space for text entry, then joins the trailing text to the inserted text by deleting the unused space.

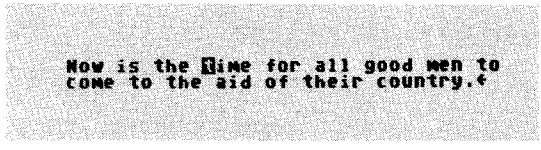
Before starting this demonstration, use **CTRL-B** to move the cursor to the beginning of the text and **SHIFT-DELETE** to clear the screen.

Now type the following text:

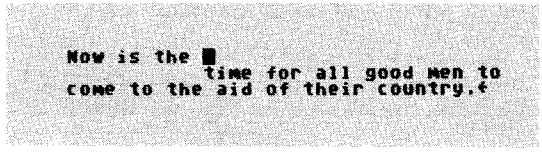
```
Now is the time for all good men to
come to the aid of their
country.RETURN
```

Suppose you want to insert the word 'best' before the word 'time' in the sentence you just typed. Here is one technique for doing it:

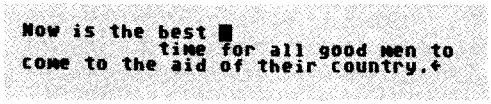
(1) Move the cursor to the beginning of the word 'time'. The display should now look like this:



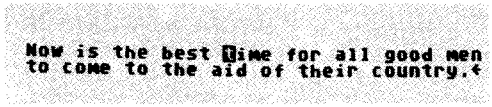
(2) Enter **SHIFT** + **INSERT** (**>**) to **make space:**



(3) Type the word 'best ' (don't forget the space at the end of the word). This still leaves a large gap between 'best' and 'time' as shown below:



(4) Enter **CTRL** + **J** to re-**Join** the text:



SUMMARY OF INSERT AND JOIN

1. Move the cursor to the point where the insertion is needed.

2. Open up space by entering **SHIFT-INSERT.**

3. Type the insert.

4. **Remove unused space after the cursor** by entering **CTRL-J.**

INSERTING LARGE AMOUNTS OF TEXT

If you need to insert several lines of text you can open up as many blank lines as needed by pressing **SHIFT-INSERT** once for each line. For very large inserts press and hold both **SHIFT** and **CTRL** keys, then press **INSERT.** This will open up all the space available, and the text after the cursor will disappear from view temporarily. When you finish the large insert, press **CTRL-J** to remove the unused space after the cursor. The text that disappeared will then reappear at the cursor position.

3.5 INSERT AND TYPEOVER MODES OF TEXT ENTRY

For all of the previous tutorial examples EDIT has been set to the TYPEOVER mode of text entry. In this mode a character entered from the keyboard writes over the character at which the cursor is positioned.

The Writer's Tool also provides an **INSERT mode** of text entry. The way this mode works is illustrated in the following example.

(1) Clear the screen and enter this text:

```
Now is the time for all good men to
come to the aid of their country.
```

(2) Press **CTRL** + **I** to turn on the **INSERT mode.**

[Note that the cursor now changes to a flashing vertical bar. The vertical bar is on the left edge of the character at the cursor position.]

(4) Move the cursor to the first character of the word 'time'. The vertical bar should be flashing at the left edge of the 't'.

(5) Type the word 'best '. Note that all of the **text beyond the cursor is pushed aside as new characters are entered.**

(6) Now delete the word by pressing the **DELETE** key repeatedly. Note that the text to the right of the cursor is pulled along with the cursor as it moves to the left during the deletion process. This is a characteristic of the INSERT mode only.

WORD REPLACEMENT IN INSERT MODE

(1) Position the cursor at the first letter of the word 'good'.

(2) Type the word 'evil'.

(3) Use **CTRL-DELETE** until 'good' is deleted. As you can see, **CTRL-DELETE** works the same in the INSERT mode of text entry as it does in the TYPE-OVER mode.

(4) Press **CTRL** + **T** to restore the **TYPEOVER mode.**

3.6 USING BLOCK COMMANDS

Here the term **text block** means any sequence of ordinary characters between two special characters called **block markers.** A block can be as short as one character or longer than 18,000 characters. It can be a word, paragraph, or anything. Once a block is marked, it can be copied, deleted, or moved.

The block commands allow you to rearrange the order of paragraphs, to delete specific parts of a document, or copy one part of a document into another part.

Before these functions can be demonstrated, some text is needed to work with. Clear the screen, put the cursor at the beginning of the text and enter the following:

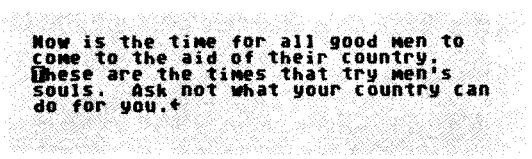
```
Now is the time for all good men to
come to the aid of their country.
These are the times that try men's
souls.  Ask not what your country can
do for you.RETURN
```

The following demonstration shows how to rearrange the order of the first two sentences.

MARKING A BLOCK

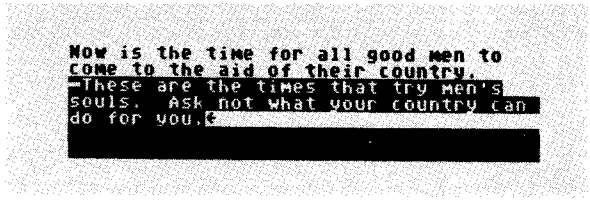
(1) Move the cursor to the first letter of the word 'These':

BEFORE
1st MARK:



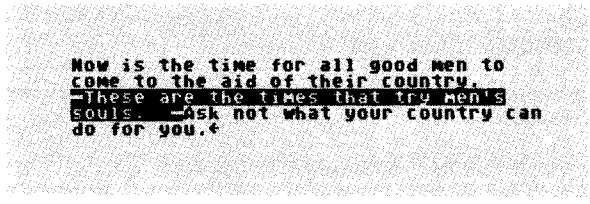
(2) Enter **CTRL-M.** This inserts a special marker into the text and causes all following text on the screen to be displayed in inverse video. The cursor is left at the marker character (the white rectangle with a horizontal bar through the middle). To get a better view of what happened use **CTRL-DELETE** to get rid of the marker, then enter **CTRL-M** again. Now the first marker is in position:

AFTER
1st MARK:



(3) Move the cursor to the 'A' in 'Ask', then enter **CTRL-M** to insert the second block marker. Now only the marked sentence will be highlighted in inverse video:

AFTER
2nd MARK:

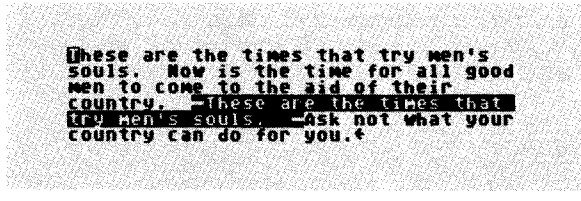


COPYING THE BLOCK

(4) Now that the desired block is marked, move the cursor to the beginning of the text using **CTRL-B.**

(5) Enter **CTRL-C.** This inserts a copy of the marked block at the beginning of the text and moves the rest of the text forward:

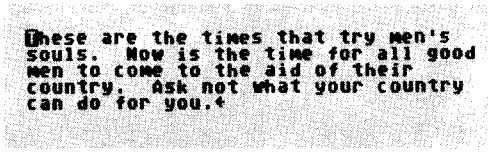
AFTER
CTRL-C:



DELETING THE BLOCK

(6) Enter **CTRL-X.** This deletes the marked block and the marker characters as well:

AFTER CTRL-X:



This completes the move which reversed the order of the first two sentences.

DELETING MARKERS ONLY

Enter **CTRL-CLEAR** to erase all block markers. The cursor position doesn't matter.

SUMMARY OF BLOCK COMMANDS

Important facts to remember from the block command examples are as follows:

1. **CTRL-M** inserts a block marker.
2. **CTRL-C** inserts a copy of the marked block at the cursor.
3. **CTRL-X** deletes a marked block and its markers.
4. **CTRL-CLEAR** erases block markers only.

The most important restrictions on using block commands are these: (1) you can't copy or delete a block unless it has been marked, (2) you can't copy a block into itself, (3) you can't copy a block past the end of text already entered. Error messages which result from attempting illegal block commands are fully described in the REFERENCE GUIDE.

3.7 CHANGING CASE

The case or appearance of previously entered text can be changed without retyping. Move the cursor to the beginning of what you want changed, then

enter **CTRL** + **L** to convert to **lower case**

enter **CTRL** + **K** to convert to **UPPER CASE**

enter **CTRL** +  to flip between normal and inverse video characters (see page T-14).

These commands only affect text which has already been entered.

3.8 PAGING COMMANDS

Paging commands can move the cursor (and scroll the text) in 20-line steps either forward or backward. If the cursor is not at the top or bottom of the screen, the first action of a paging command is to move the cursor to the top of the screen (page reverse) or to the bottom of the screen (page forward). These commands allow you to move the cursor rapidly through a large amount of text.

Enter **CTRL** + **F** to page **forward.**

Enter **CTRL** + **R** to page in **reverse.**

Here the term page is used to mean 20 screen lines of text.

4.0 CHANGING KEYBOARD AND DISPLAY CHARACTERISTICS

4.1 SETTING UPPER AND LOWER CASE

The Writer's Tool begins in the lower case mode. In this mode you must hold down the **SHIFT** key to produce an upper case letter. If you want to enter mainly upper case letters,

press **SHIFT** + **CAPS LOWR** to set the **upper case** mode.

This will display **"CAPS LOCK"** on the status line and produce upper case letters from unshifted key-ins. [NOTE: The 800XL CAPS/LOWR key only has the CAPS label.]

press **CAPS LOWR** to return to **lower case** mode.

4.2 INVERSE VIDEO CHARACTERS

Normal characters are white on a blue background. Inverse characters are blue on a white background. To enter inverse video characters, you must first set the keyboard to **INVERSE** mode:

press  or  to **turn on** the **INVERSE** mode.

The first key appears on the old ATARI computers, while the second appears on the XL series computers. The key function is the same. Just the labels are different.

When the inverse mode is turned on, you should hear a beep and see **"INVERSE"** displayed on the status line. Keyboard entries will then produce inverse video characters until you return the keyboard to normal:

press  or  to **turn off** the **INVERSE** mode.

The use of inverse characters will be explained later. Now just remember how to turn off the inverse mode when it is accidentally activated.

4.3 ONE-FINGERED CURSOR MOVEMENT

In some cases, using a two-key combination to move the cursor is a nuisance. When you need one-fingered cursor control,

press **CTRL** + **CAPS LOWR** to **activate** CURSOR EXCHANGE.

This will produce the message **"CURSOR EXCH"** on the status line. At this point you will be able to move the cursor just by pressing the arrow keys. The cursor controls have been exchanged with the unshifted characters **+**, **-**, *****, and **=**. To enter these characters you will thus need to hold down the **CTRL** key, while pressing **+**, **-**, *****, or **=**. When you want to return to the normal mode of cursor control,

press **CTRL** + **CAPS LOWR** to **deactivate** CURSOR EXCHANGE.

4.4 CONTROLLING THE STATUS LINE DISPLAY

If you find the status line display of the keyboard settings annoying, then

press **SHIFT** + **CLEAR** to **erase the status line.**

If you want to know what the keyboard settings are, then

press **CTRL** + **?** to **restore the status line.**

4.5 CHANGING THE TYPEOVER CURSOR

In the typeover mode of text entry, the cursor position is indicated by flashing between the character at the cursor and its inverse video representation. If this doesn't appeal to you,

press **SELECT** to **switch cursors.**

The first time you press **SELECT,** the cursor will change to a **flashing underline.** Pressing **SELECT** again will restore the original **inverse video** cursor.

4.6 WORD WRAP CONTROL

You may have already noticed that The Writer's Tool keeps words from being split between the end of one line and the beginning of the next. This function is called **word wrap.** Its purpose is to make the displayed text as readable as possible. However, there are times when you may want to turn off the word wrap function: (1) when you want to know exactly how much space there is between the last word of one line and the first word of the next, or (2) you are editing program files for which words are not the natural text unit.

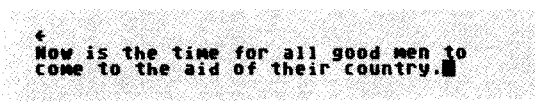
EXAMPLE

(1) Press **RETURN,** then enter the following text:

```
Now is the time for all good men to
come to the aid of their country.
```

(2) Press **START** to **turn off word wrap:**

(3) Press **START** to **turn on word wrap:**

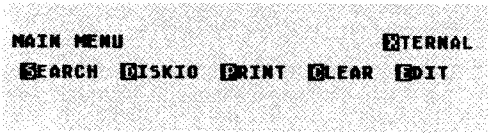


5.0 USING MAIN MENU FUNCTIONS

So far, the tutorial has focussed on using the EDIT system of The Writer's Tool. Through the **MAIN MENU** you can activate other necessary word processing functions, the most important of which are printing, saving, and retrieving text.

5.1 THE MAIN MENU

Press **OPTION** or enter **CTRL-O** to display the **MAIN MENU**:



This display will occupy the bottom four lines of the screen (also used as the "command window"). When the main menu is displayed, you cannot type characters into the text buffer (the memory space reserved for text).

The first letter of each function name on the menu is highlighted in inverse video. Pressing the corresponding letter key will activate the function. To see how this works,

press **E** to return to **EDIT.**

The other functions on the menu are summarized below:

SEARCH manages search and replace activities.

DISKIO (stands for DISK Input/Output system) manages the transfer of text between disk storage and computer memory.

PRINT handles formatting and printing of text.

CLEAR handles deletion of large blocks of text from memory.

XTERNAL handles external routines including the spelling checker and customization programs.

The following subsections show how to access and use these functions.

5.2 USING SEARCH AND REPLACE

The search and replace function provides a fast and convenient way to find particular words or phrases within a large body of text, and to make repetitive replacements of one word or phrase with another, no matter where they are needed and no matter how many are needed.

The Writer's Tool supports three variants of this useful function: (1) search for first occurrence and return to EDIT, (2) continue search in EDIT, and (3) search and replace with verify, skip, or quit options. Each of the three variants is illustrated by examples.

Before working through the examples clear the screen and enter the following sentence:

```
Now is the time for all good men to come
to the aid of their country.RETURN
```

SEARCH AND RETURN

(1) Enter **CTRL + B** to move the cursor to the beginning of the text (The Writer's Tool only searches from the cursor forward).

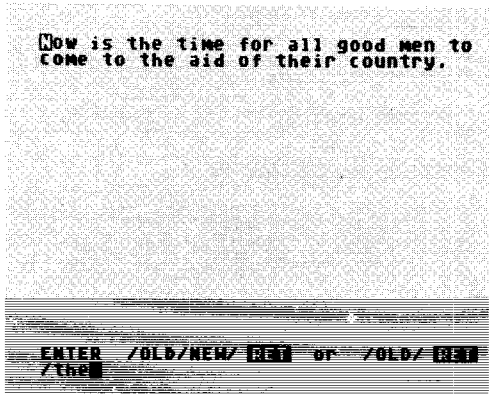
(2) Press **OPTION** to display the main menu at the bottom of the screen. (When this 4-line window is displayed, normal keyboard text editing is temporarily suspended.)

(3) Press **S** to activate the **search** function:

[In this prompt, "**old**" refers to the object of the search and "**new**" refers to a replacement string. The slash character "**/**" is a delimeter and can be any character which does not appear in **old** or **new** strings.]

(4) Answer the prompt by entering "**/the**", but **don't** press RETURN yet.

The screen should now look like this:



(5) While you are watching the screen, press **RETURN** to activate the search. A beep of medium pitch will be sounded when the first occurrence of "**the**" is found, and the cursor will appear at the first letter. The command window will then disappear and control will return to EDIT.

CONTINUE SEARCH

Once a search string is defined, as was done in the previous steps, it then becomes possible to continue searching without leaving EDIT:

(6) Enter **CTRL-S.** This moves the cursor to the next occurrence of "**the**", which was the last search string defined. Note that the line containing the found string is always moved to the top of the screen.

(7) Enter **CTRL-S** again. This time the search routine finds the 'the' in "**their**". [This could have been avoided by entering "**/ the /**" instead of "**/the**" in step 4.]

(8) Enter **CTRL-S** once more. This time you should hear a high-pitched beep, and see "NOT FOUND" temporarily displayed on the status line.

SEARCH AND REPLACE

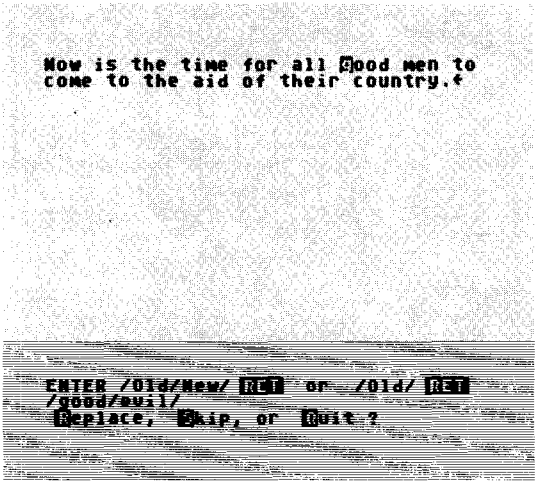
The following example makes use of the same text entered at the beginning of this section.

(1) Move the cursor to the beginning of the text. (Use **CTRL-B**)

(2) Press **OPTION** or **CTRL-O** to display the main menu.

(3) Press **S** to activate the SEARCH function.

(4) Answer the prompt by entering **"/good/evil/",RETURN.** The cursor will move to the first letter of "good", a medium pitched beep will sound, and the prompt **"CHOOSE Replace, Skip, or Quit"** will appear:



(5) Press **R** to make the replacement. Immediately afterwards a slightly higher pitched beep will indicate that there are no more occurrences of the word "good" in the remaining text. Then you will see the message "NOT FOUND", and control will return to EDIT.

SKIP AND QUIT

Using the same text as the previous examples, suppose you want to replace "men" with "women".

(1) Move the cursor to the beginning of the text.

(2) Press **OPTION** for the main menu, then press **S** to activate SEARCH.

(3) Answer the prompt by entering "/me/wome/",**RETURN.** [This is not the best way to replace "men" with "women", and is only used to show the use of SKIP and QUIT responses.]

(4) The first occurrence of "me" is in the last two characters of "time". Since this is not where the replacement is wanted, answer the "Replace, Skip, or Quit" prompt by pressing the **S** key. This will leave "time" unchanged and the cursor will advance to the "m" in "men".

(5) Answer the prompt by pressing the **R** key. This will cause "men" to be changed to "women", and the cursor will then move to the "m" in "come".

(6) This time answer the prompt by pressing the **Q** key. Control will then return to EDIT, and the search and replace function will be aborted.

(7) Press **CTRL-B** to bring all the text back into view so you can see the effects of the replacement.

SEARCH AND DESTROY

Here the search and replace function is used to delete the word "time" by replacing it with nothing.

(1) Make sure the cursor is at the beginning (Enter **CTRL-B**).

(2) Press **OPTION**, then **S** to activate SEARCH.

(3) Enter "/time //",**RETURN** (the double slash indicates "replace with nothing", in other words, delete).

(4) Press the **"R"** key to effect the replacement. Since there are no more occurrences of "time", control will again return to EDIT.

5.3 USING A DISK DRIVE

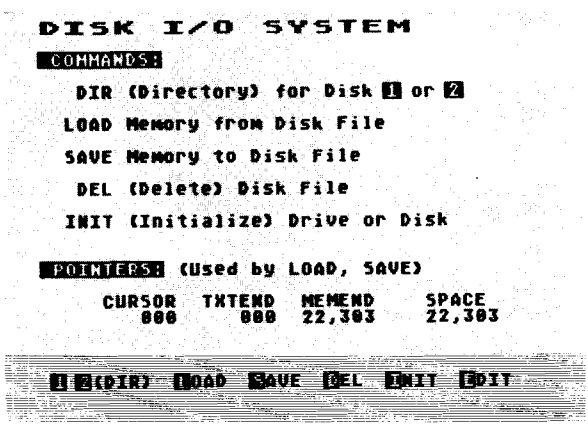
This section demonstrates the functions of the Disk Input/Output System (DISKIO).

[Before proceeding any further, use **CTRL-B** to move the cursor to the beginning of the text buffer, then press and hold **SHIFT-DELETE** until all text is deleted.]

GETTING TO THE DISK I/O SYSTEM

(1) Press **OPTION** to display the main menu.

(2) Press **D** to activate the disk system. You should now see the following screen display:



DISK I/O SYSTEM commands are briefly described in the upper part of the screen and presented in a new menu at the bottom of the screen. (This menu works the same as the main menu.)

SUMMARY OF DISKIO COMMANDS

Read the following command descriptions but don't try any just yet.

1 2 (DIR) This displays the names of the files on a disk (the disk directory) and the disk space used by the files. A directory can be obtained for drive #1 or drive #2 by pressing **1** or **2.** If you have more than two disk drives, see Appendix 2.

LOAD This reads characters from a disk file and copies them into the text buffer (the part of your computer memory reserved for text). The copied text will start wherever the cursor was positioned. This command does not change anything on the disk.

SAVE This copies characters from the text buffer to a disk file. The save process usually starts at the cursor position and ends at the last character in the text buffer. This command does not alter the text in the buffer.

DEL (DELETE) This is used to erase a disk file, and thus makes more space available for storing new information.

INIT (INITIALIZE) This is used to prepare a blank disk for storing data or to change between single and double density disk formats. It can also be used to completely erase an old disk.

EDIT Pressing **E** returns program control to EDIT. (You can return directly to the main menu by pressing **OPTION.**)

DESCRIPTION OF POINTERS

CURSOR This is where the cursor is positioned relative to the beginning of the text buffer. LOAD and SAVE commands usually begin at the cursor position.

TXTEND This is where the last character was entered in the text buffer. A LOAD command will set TXTEND to the position of the last character loaded.

MEMEND This is the size the text buffer (the total number of characters which can fit in the buffer).

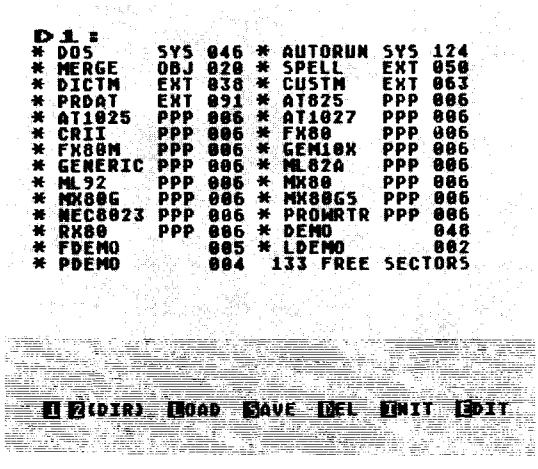
SPACE This is how much room is left for entering more text (the difference between MEMEND and TXTEND). When the text buffer is empty, SPACE and MEMEND will be the same.

DISPLAYING A DIRECTORY

If you can't remember the name of a text file you saved previously, or you can't remember on which disk you saved it, or you want to save a file and you don't know which disk has enough room to hold it, then the first thing you will need is a directory of the disk. To see how this works, do the following:

(1) Make sure that the Disk I/O System screen is displayed. If it isn't use **OPTION**, then **D** to activate DISKIO.

(2) Make sure that The Writer's Tool disk is in drive #1, then press **1** to display the directory for drive #1:



This directory listing has a header and two major columns, each with three sub-columns of information. The header (which reads **D1:** in this example) tells which disk drive the directory came from. The first two sub-columns list the first and last parts of the filenames: the file "DOS.SYS" has "DOS" listed in the first sub-column and "SYS" in the second sub-column. The third sub-column lists the number of disk sectors used by each file. The last item in the directory display (FREE SECTORS) shows how much room is left on the disk. The asterisks indicate that files on the master disk have been "locked" to protect against their being erased or modified.

PRINTING A DIRECTORY

To produce a printed copy of the disk directory, follow the same procedure used to display a directory, except press the **SELECT** switch before pressing **1** or **2,** and hold down the **SELECT** switch until the directory begins printing.

LOADING A DISK FILE

This example continues the previous one, and assumes that The Writer's Tool disk is in drive #1, and the directory is displayed.

(1) Press **L** to activate the load process. This will produce the prompt "LOAD WHICH FILE?".

(2) Enter **'DEMO',RETURN** (**D1:DEMO** is also acceptable; if you later want to load a file from drive #2, you must use **"D2:"** at the beginning of the filename).

(3) If you made a mistake in step (2), then answer the prompt "ARE YOU SURE (Y/N)" by entering **'N',RETURN,** and go back to step (1). Otherwise, enter **'Y',RETURN.** This will start the load process (you should now hear the disk drive clicking and whirring). When the load is complete, a low pitched beep will sound and program control will return to EDIT. This means you have a successful load.

FILE-NAMING RULES

A disk file name must begin with an upper case letter and use only upper case letters or numbers for the remaining characters. The name may have two parts: a first name having as many as eight characters, and a last name up to three characters long. The first and last names must be separated by a period. The period is not displayed in the directory listing.

A filename may also need a prefix. For example **D2:LETTER** has the prefix **D2:** to specify that the file **LETTER** is on disk drive #2. Filenames specified without a prefix are assumed to refer to drive #1.

INITIALIZING A DISK

Before a new blank disk can be used for data storage, a special magnetic pattern must be written on the disk surface. This process is called "initializing" (or "formatting") the disk. The following example shows how to do it.

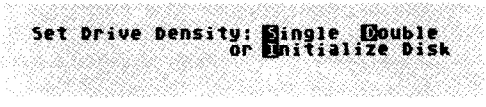
(1) Remove The Writer's Tool disk from drive #1.

(2) Insert a blank disk in drive #1.

(3) Enter the DISK system as follows: use **OPTION** to display the main menu, and then press **D** to activate DISKIO.

(4) Now that the DISK I/O SYSTEM screen is displayed, press **I** to begin the initialization. You will then see the prompt "Which Disk Drive (1 or 2)?".

(5) Enter **'1',RETURN** or just **RETURN.** This will display the prompt:



(6) Press **'I'** to initialize the disk in the selected drive. (Do not attempt to change disk densities before reading Appendix 3.) Because initializing a disk will completely erase it, you will be asked "ARE YOU SURE (Y/N)?".

(7) Answer with **'Y',RETURN.** The message "INITIALIZING DISK" will then appear. After about 40 seconds, when the initialization process is completed, the directory of the initialized disk will be displayed. This will show only the number of sectors available for storage.

(8) Press **E** to return to EDIT.

[NOTE: initializing a used disk will erase all previously stored information.]

SAVING TEXT ON A DISK

This example requires a formatted (initialized) disk and The Writer's Tool disk. If you don't have a formatted disk, go back to the section on initializing a blank disk.

To save text in a disk file, you first need some text in the memory buffer. If you've worked through the previous two examples, you should have a copy of the **DEMO** file already in the buffer. If not, you should either go back and work through those sections or enter something of your own choosing. Once you have some text in memory, proceed as follows:

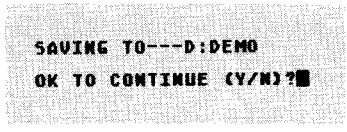
(1) Move the cursor to the beginning of the text buffer. This specifies that you want to start the SAVE at the beginning of the text.

(2) Press **OPTION** to display the main menu, then press **D** to display the main DISK I/O SYSTEM screen.

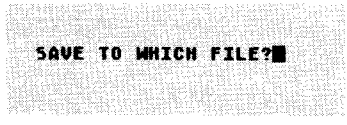
(3) Make sure that a blank formatted disk is inserted in drive #1.

(4) Press **S** to activate the save function. This will produce one of the following prompts:

IF DEMO WAS THE MOST RECENT FILE LOADED OR SAVED:



IF NO FILE WAS PREVIOUSLY LOADED OR SAVED:



If you got the second prompt, go to step (5). The first prompt gives you a chance to save the current text under the last filename you used. This is especially useful if you are editing an old file and want the new version saved under the old name. However, to continue with this example, answer the "OK to CONTINUE..." prompt with **'N',RETURN,** and proceed with step (5) when the next prompt appears.

(5) Answer the prompt with 'JUNK',**RETURN** (as usual, do not enter the quotes or the comma). This will produce the prompt "ARE YOU SURE (Y/N)?"

(6) Answer the last prompt with **'Y',RETURN.** The save will then begin. While the save is taking place, the message "SAVING TO D:JUNK" will be displayed. When the save is complete, the directory will be displayed for the disk which now contains the saved file. This verifies that the save was completed successfully.

(7) Press **E** to return to EDIT.

SAVING ONLY PART OF THE TEXT

If you start a save with the cursor not at the beginning of the text buffer, then a warning will be presented. If you ignore the warning then only text from the cursor on will be saved. If you heed the warning and choose the "SAVE ALL TEXT" option, then the cursor will be moved to the start of the text before the save process begins.

DELETING A DISK FILE

This example assumes that you have a disk containing the file "JUNK", and that this disk is inserted in disk drive #1. This will be the case if you have worked through the previous section.

(1) Enter the DISK I/O SYSTEM.

(2) Press **I** to display the directory for the disk in drive #1. The directory listing should show that JUNK is one of the disk files (probably the only one).

(3) Press **D** to start the DELETE function. This will display the prompt "DELETE WHICH FILE?".

(4) Answer the prompt with 'JUNK',**RETURN.** This will produce the prompt "ARE YOU SURE (Y/N)?"

(5) Answer the last prompt with **'Y',RETURN.** While the delete is taking place the message "DELETING D:JUNK" will be displayed. When the delete is completed, the directory for the disk in drive #1 will be displayed. If it shows that "JUNK" is no longer present, then the delete was successful.

(6) Press **E** to return to EDIT.

5.4 USING A PRINTER

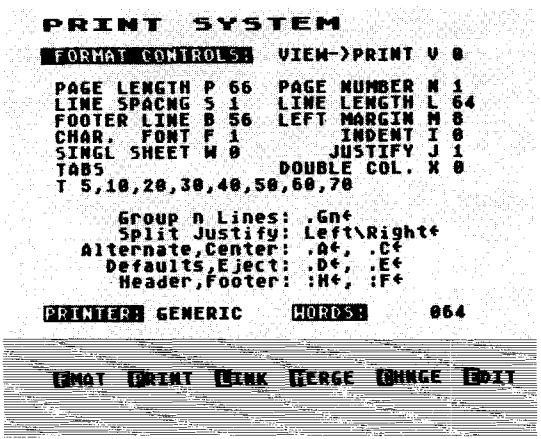
The contents of the text buffer can be printed using the PRINT SYSTEM.

To provide a standard sample of text for the examples of this section, move the cursor to the beginning of the text, then use the Disk I/O System to LOAD the "PDEMO" file from The Writer's Tool disk. [If you get a warning that the text buffer will be written over, just enter **'Y',RETURN** to continue.].

ACCESSING THE PRINT SYSTEM

(1) Once the "PDEMO" file is loaded in the text buffer, press **OPTION** or **CTRL-O** to display the main menu.

(2) Press **P** to display the **Print System** screen:



The print format parameters and their default values are listed just below the heading "FORMAT CONTROLS". Each item in this list has a descriptive name, a letter code, and a default value. The first item, for example, has the descriptive name "PAGE LENGTH", the letter code "P" and the default value of "66". This means that the printed page will have a length equal to 66 single-spaced print lines (at six lines/inch, this gives the standard 11-inch page). The meaning and use of the other format parameters will be described later.

OVERVIEW OF PRINT MENU FUNCTIONS

At the bottom of the screen is a new menu through which you can access the functions of the print system. Read the following brief descriptions, but **don't try any functions yet,** and don't worry about understanding them.

FMAT (FORMAT) This is used to change the value of the format parameters listed near the top of the screen.

PRINT (NORMAL PRINTING) This reads the text buffer, formats the text into print lines, and sends them to the printer.

LINK (LINKED PRINTING) This enhanced version of **PRINT** allows continuous printing of a sequence of disk files.

MERGE (MERGED PRINTING) This loads a utility program from The Writer's Tool disk which supports the printing of template documents merged with information from data base files.

CHNGE (CHANGE PARAMETERS) This allows you to load special data files to change the current printer or to change the default print format and screen display characteristics. The name of the printer currently installed will appear just above the menu. The display now shows that the GENERIC printer is installed, providing a restricted print capability compatible with almost any printer.

EDIT Pressing **E** returns program control to EDIT. (You can return directly to the main menu by pressing **OPTION** instead of **E.**)

WORDCOUNTER

The number displayed after **"WORDS :"** is the number of words between the cursor and the end of the text currently in the text buffer.

PRINTING IN DEFAULT FORMAT

Before going any further you should verify that the ATARI 850 Interface Module is turned on, that your printer is turned on, and that your printer is selected (if it has a select button). Of course, you should also have the printer connected to the interface module and the interface module connected to the computer. [It's a good idea to turn your printer off and then on, to make sure it starts out in its default mode.]

Once this is done, press **P** to start printing the text buffer. During printout you will see the following message in the command window:

While this message is displayed, you can temporarily halt the printout by pressing **P**, then resume printing by pressing **P** a second time. Pressing **Q** aborts the printout and returns to the print system. Pressing **E** aborts the printout and returns to EDIT, with the cursor positioned at the first character of the last line printed.

When the print process is completed, the Print System Menu will be redisplayed and you should have a printout that looks like this:

```
    Venus, the morning or evening star, is the third brightest
object in the sky. Only the sun and moon are brighter.
    Venus has a thick atmosphere of carbon dioxide, with a
surface pressure of 90 atmospheres, and a surface temperature of
800 degrees Farenheit.
    Venus is surrounded by a cloud of concentrated sulfuric
acid droplets which has a thickness of more than 10 miles.
```

This was printed with the default format. The next example shows how to change the format.

NOTE: If your text was all printed on one line, then your printer has not been properly configured for use with ATARI computers. Consult your printer manual to find out how to make your printer automatically perform a line feed after every carriage return.

CHANGING FORMATS FROM THE PRINT SYSTEM

The following example uses text loaded from disk at the beginning of this section and assumes that the Print System screen is displayed.

(1) Press **F** to activate the format changing routine. This will produce the following prompt:

A **format line** starts with a period, ends with a carriage return, and has letters and numbers in between. The letters must be either upper or lower case letters standing for one of the format parameters, and the number (no longer than three digits) following each letter defines a new value for the corresponding format parameter. Spaces in the format line are ignored. Continue with the following steps to see how to use a format line.

(2) In response to the prompt, enter **'.M20L50',RETURN** (skip the quotation marks and comma, but don't forget the leading period).

(3) Now you should see the print screen redisplayed with LEFT MARGIN (M) set to 20 and LINE LENGTH (L) set to 50. To see the impact of these changes on the appearance of the print, press **P.**

Shown below is one paragraph of the default printout, then the modified version just printed:

DEFAULT:

Venus, the morning or evening star, is the third brightest object in the sky. Only the sun and moon are brighter.

MODIFIED VERSION:

Venus, the morning or evening star, is the third brightest object in the sky. Only the sun and moon are brighter.

Note that the print format parameters **M** and **L** have now returned to their original default values. More permanent format changes are possible using format lines imbedded within the text (explained in Section 6).

TURNING OFF JUSTIFICATION

In the last print example the right margin was straight, as well as the left margin. This is called right justification. The Writer's Tool does this automatically by inserting a few extra spaces between the words. If you would rather have equal spaces between words and don't mind a ragged right margin, then set the **J** parameter to **0** (enter **'.J0',RETURN**).

WHERE TO FIND OUT MORE ABOUT FORMAT

Now that you have some idea of what a format line is and how it works, you will be able to understand the more detailed discussion of format specifications presented in the REFERENCE GUIDE. Examples showing how to use most of the format functions are presented in Tutorial Section 6 (USING FORMAT CONTROLS).

WHERE TO FIND OUT ABOUT OTHER PRINT SYSTEM FUNCTIONS

The other print system functions are covered in Tutorial Sections 7-9.

5.5 CLEARING PART OF THE TEXT BUFFER

This section assumes that you have the PDEMO file loaded into the text buffer. If not, go to Section 5.3 and load the PDEMO file before continuing.

The CLEAR command allows you to delete text from the memory buffer. This method of deletion is mainly used to remove large amounts of text quickly. The following example shows how to remove everything but the second paragraph of the PDEMO file.

(1) Go to EDIT and move the cursor to the beginning of the second paragraph.

(2) Press **OPTION** to display the main menu.

(3) Press **C** to activate the CLEAR function. This will produce the prompt "CLEAR AFTER OR BEFORE CURSOR (A/B)?".

(4) Enter **'B',RETURN.** This will result in the second prompt "ARE YOU SURE (Y/N)?":

(5) Enter **'Y',RETURN.** This will erase all text before the cursor, move the cursor and remaining text to the top of the text buffer, and finally return the program control back to EDIT.

(6) Now move the cursor to the beginning of the new second paragraph.

(7) Activate the clear function again: use **OPTION** to display the main menu, then press **C.**

(8) This time answer the prompts with **'A',RETURN,** then **'Y',RETURN.** This will delete all text after (and at) the cursor position, leaving only one paragraph in the text buffer. This was originally the second paragraph of the PDEMO file. Thus, the objective stated at the beginning of this example is accomplished.

6.0 USING FORMAT CONTROLS

The number of different ways the same words can be printed is almost unlimited. The character shapes and sizes, the length of printed lines, the space between lines, the number of lines per page, the numbering of pages, the style of headers and footers, and the justification and indentation of paragraphs are just a few of the numerous factors which can change the appearance of the printout. A specific choice of all these variables is what is meant here by the term **format.**

The Writer's Tool automatically makes a choice of the format variables which produces reasonably good-looking printout on most printers. This section shows how to change the format to suit your own special purposes.

You have already seen how to change print format by entering format lines from within the Print System. A more powerful technique uses format lines imbedded within the text itself. Imbedded format lines, together with imbedded header and footer blocks, and a few special format control characters, provide great flexibility in controlling format.

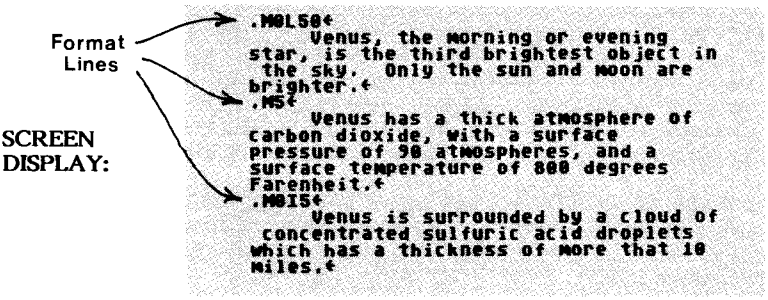
6.1 IMBEDDED FORMAT LINES

An imbedded format line is a format line placed within the text. To function there it must be a paragraph of its own. This means it must be preceded by a carriage return symbol (unless it's the first line in the buffer) and terminated by a carriage return symbol.

EXAMPLE

(1) Move the cursor to the beginning of the text buffer, CLEAR all text, then load the file PDEMO from The Writer's Tool disk. Do this even if you already loaded PDEMO earlier.

(2) Enter three format lines in the text as shown in the following screen display (use **SHIFT-INSERT** to make room for the additions, then **start with a period** and press **RETURN** at the end of each format line):



(3) Move the cursor to the beginning of the text, then go to the Print System. Note on the Print System screen that **M**, **L**, and **I** have not been changed by the imbedded format lines. The print routine does not yet know about the imbedded commands.

(4) Press **P** to produce the following printout:

PRINT EXAMPLE:

Venus, the morning or evening star, is the third brightest object in the sky. Only the sun and moon are brighter.

Venus has a thick atmosphere of carbon dioxide, with a surface pressure of 90 atmospheres, and a surface temperature of 800 degrees Farenheit.

Venus is surrounded by a cloud of concentrated sulfuric acid droplets which has a thickness of more than 10 miles.

Here the line length is set to 50 (**L50**). The first and last paragraphs have the same margin (**M0**), but the last paragraph is indented (**I5**). Note that the indents do not affect the right hand margin. The second paragraph has a different left margin (**M5**), which also affects the right margin.

EXPLANATION

Imagine that the printing program reads the text buffer much the way you would, a word at a time. As it reads the words, it copies them into a special place in memory called the line buffer. When it collects as many words as can be fit in the line buffer, it sends the line to the printer and then continues reading more words to make the next line.

How long the line buffer is and how it's printed are determined by numbers stored in other places in the computer memory. These numbers are the format parameters. At the beginning and end of any print operation these memory locations contain the default format values. But these values can be changed by the imbedded format commands.

While the program is reading through the text, it keeps an eye out for paragraphs that start with a period. These are interpreted as format commands. When it encounters a format line, it decodes the line and stores any new format parameters in memory (erasing the old values). It then reads on, looking for more words to print. The remaining text is printed using the revised format parameters, until another format line is encountered. Thus, **imbedded commands have priority over formats entered from the print system.** Even though you can change the parameter values using the **FMAT** command, these changes will be superceded during printout by whatever imbedded format statements appear in the text.

IMPORTANT FACTS ABOUT IMBEDDED FORMAT LINES

1. An imbedded format line must be a paragraph that starts with a period.

2. Imbedded formats only affect the text which follows them.

3. Imbedded formats take precedence over format commands entered from the print system.

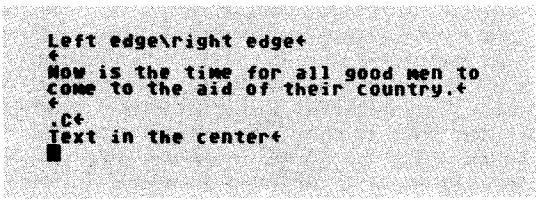
4. Printing halts when a bad format line is encountered. Control will then return to EDIT with the cursor positioned at the beginning of the bad format line.

6.2 CENTERING AND SPLITTING

Centering titles and headings between left and right margins requires a special format line (**'.C',RETURN**) just before each line you want to center. Splitting a line requires a special character (**backslash**) to be inserted at the point where you want the line divided. The part of the line left of this point will be printed flush against the left margin, and the rest of the line will be printed flush against the right margin. Both centered and split lines must be shorter than the line length specified by the **L** parameter.

To see how to use these functions, do the following:

(1) Move the cursor to the beginning of the text and enter what's needed to produce the following screen display:



(2) Move the cursor to the beginning of the text and print the text buffer. The printout should look like this:

```
Left edge                                                right edge

Now is the time for all good men to come to  the  aid  of  their
country.

                          Text in the center
```

The **backslash** symbol is used to split a short text line so that the left side of the line is left justified, and the right side is right justified.

The centering command is a format line with only one parameter (**C**) and no numbers. It will center only the text line immediately following the centering command.

6.3 USING TABS

You can use tabs to align columns of text, and indent or outdent the first line of a paragraph. Using tabs requires inserting special characters in the body of the text. To see how this works, begin with the following example.

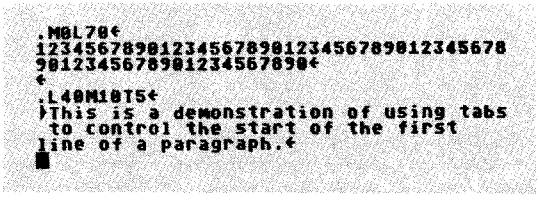
USING THE **ESC** KEY

(1) Go to EDIT, delete all text, and put the cursor at the beginning.

(2) Press the **ESC** key once. This has no visible effect. Now press the **TAB** key. This should produce a solid right-pointing triangle on the screen. Normally, pressing **TAB** would just move the cursor. By pressing **ESC** first, the command function of the keyboard keys is bypassed, and the key-ins are interpreted as characters instead of commands. Now clear the text and proceed to the next example.

CONTROLLING PARAGRAPH INDENTATION WITH TABS

(1) With the cursor at the beginning, enter text to produce the following screen display:



(2) Move the cursor to the beginning of the text and print the buffer. This will produce an outdent on the first paragraph line:

OUTDENT:

```
1234567890123456789012345678901234567890123456789012345678901234567890
    This is a demonstration of using tabs to
        control the start of the first line of a
        paragraph.
```

[The sixty numbers just above the paragraph show the column position of each character.]

(3) Now go to the second format line and change **T5** to **T10,** making sure that nothing else is changed. Move the cursor to the beginning of the text and print the buffer again. This will make the first line flush with the rest of the paragraph:

FLUSH:

```
1234567890123456789012345678901234567890123456789012345678901234567890
This is a demonstration of using tabs to
control the start of the first line of a
paragraph.
```

(4) Go back to the second format line again and change **T10** to **T15.** Move the cursor to the beginning and print this version to obtain a paragraph with the first line indented by 5 spaces from the left margin:

INDENT:

```
1234567890123456789012345678901234567890123456789012345678901234567890
          This is a demonstration of using
tabs to control the start of the first
line of a paragraph.
```

IMPORTANT FACTS ABOUT TABS

1. The column position of the first tab is controlled by an imbedded format command using the **T** and a 1 to 3 digit number following it. Additional tab columns can be set by appending a comma and number for each additional tab column up to a total of eight tabs (**.T10,20** will set the first two tabs).

2. To get the text to align with a tab position requires inserting a tab character in the body of the text (using **ESC, TAB**).

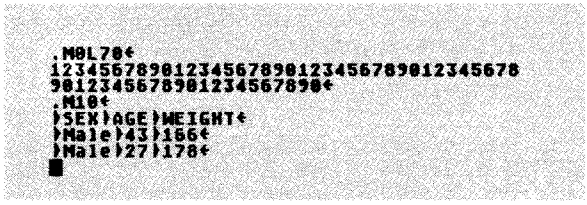
3. Text following the tab character will begin printing at the next tab column to the right of the current print column.

4. Pressing the **TAB** key without a preceding **ESC** only moves the EDIT cursor and has no format control function.

The use of multiple tab settings to align multiple columns of text is shown by the following example.

USING TABS TO MAKE TABLES

(1) Move the cursor to the beginning of the text, clear any old text, then enter new text to produce the following screen display:



(2) Move the cursor to the beginning of the text and print the text buffer. The printed output should look like this:

```
1234567890123456789012345678901234567890123456789012345678901234567890
    SEX  AGE       WEIGHT
    Male 43        166
    Male 27        178
```

This example used the default tabs at columns 5, 10, and 20. **Note that characters are printed in columns 6, 11, and 21.** This offset makes margin and tab settings of the same value produce print in the same column. The next example shows how to change the tab columns.

(3) Move the cursor to the beginning of the text. Then insert the format line **'.T15,30,45', RETURN .**

(4) Move the cursor back to the beginning of the text and print this version:

```
1234567890123456789012345678901234567890123456789012345678901234567890
               SEX             AGE             WEIGHT
               Male            43              166
               Male            27              178
```

By changing the tab column settings you can thus change the format of the table without retyping the entries in the table, an obvious convenience.

6.4 HEADERS AND FOOTERS

A header is a block of text printed at the top of each page. A footer is a block of text printed at the bottom of each page. Both header and footer blocks are defined by imbedding them in the main text and surrounding them with special marker lines. Headers and footers can each have their own formats and can be any length, provided their total length doesn't exceed the page length.

VERTICAL PAGE FORMAT EXAMPLE

To illustrate how the vertical format of a printed page is determined, suppose that a 3-line header and a 2-line footer are defined. Further suppose that the page length is set to 12 (using **P12** in a format line) and that the first footer line is set to line 9 (using **B9** in a format line). Then the printed page will have the following vertical organization:

Page Number	Page Line	Print Line	Contents
1	1	1	Header line 1
1	2	2	Header line 2
1	3	3	Header line 3
1	4	4	Text line 1
1	5	5	Text line 2
1	6	6	Text line 3
1	7	7	Text line 4
1	8	8	Text line 5
1	9	9	Footer line 1
1	10	10	Footer line 2
1	11	11	Blank line
1	12	12	Blank line
2	1	13	Header line 1
2	2	14	Header line 2
2	3	15	Header line 3
2	4	16	Text line 6
2	5	17	Text line 7
2	6	18	Text line 8
2	7	19	Text line 9
2	8	20	Text line 10
2	9	21	Footer line 1
2	10	22	Footer line 2
2	11	23	Blank line
2	12	24	Blank line

This example shows a total of 24 printed lines. Since the page length is set to 12, this corresponds to two complete pages. The combination of **B9** and **p12** format commands reserves four footer lines starting at line 9 of each page. Since there are only two lines in the example footer, this produces two blank lines between the end of the footer and the beginning of the next page. If no footer were defined, then all four of the reserved lines would be blank.

To produce blank lines between the header text and the main text, blank lines should be inserted in the header block. To produce blank lines between the end of the main text and the beginning of the footer text, blank lines should be inserted at the start of the footer block.

As far as The Writer's Tool is concerned, the top of a page is wherever the print head happens to be when printing is started. You can adjust the header and footer blocks to account for whatever starting position is convenient.

Now that you know how headers and footers are positioned on the printed page, you should be ready to learn how to use them.

IMBEDDING A HEADER

(1) Make sure that The Writer's Tool is in EDIT, then clear the text buffer.

(2) Starting at the beginning of the text buffer, type in what's needed to produce the following screen display:

```
:H¢
.D¢
This is line 1 of the header block¢
Left title\Right date 5-12-83¢
¢
:¢
.P10L50I10¢
This is the main text, and will be
printed after the header is printed.
To show that headers repeat at the top
 of each page, without using a lot of
paper, the page length was set to the
ridiculous value of 10 lines, and the
line length to 50 characters.  The
header block is always printed in the
default format because of the '.D'
command at the beginning of the header
 block.¢
```

The **':H ←'** identifies the start of a header, and the **': ←'** defines the end of the header. The header format (line length, font, etc.) will be printed the same as the main text unless a separate format command is placed within the header block (as in this example). **A format defined within a header block only affects the header itself** and has no affect on the format of the main text.

(3) Move the cursor to the beginning of the text and print the text buffer. Your printout should look like this:

```
This is line 1 of the header block
Left title                                   Right date 5-12-83

        This is the main text, and will be
        printed after the header is printed. To
        show that headers repeat at the top of
        each page, without using a lot of paper,
        the page length was set to the
        ridiculous value of 10 lines, and the

This is line 1 of the header block
Left title                                   Right date 5-12-83

        line length to 50 characters. The
        header block is always printed in the
        default format because of the '.D'
        command at the beginning of the header
        block.
```

The blank header line was used to produce a one-line margin between the header text and the main text (which begins on the 4th line of each page). The last line of the main text is printed on the 9th line, with the tenth line reserved for a possible one-line footer.

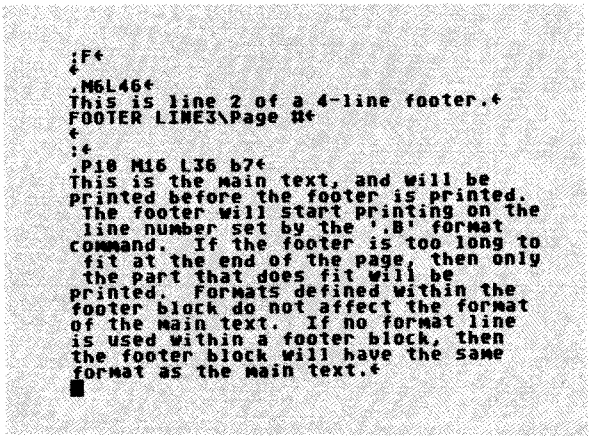
SINGLE KEY-STROKE HEADER INSERTION

To quickly insert a simplified header block, move the cursor to the point of insertion, then enter **SHIFT-CTRL-H.** You can then edit this block as needed.

IMBEDDING A FOOTER

Although a header always starts at the first line of each page, **the starting line of a footer is set by the 'B' format parameter.** In the default format, **B** is set to line 56 on a page of 66 lines. If no footer block is defined, this provides a 10-line gap between pages. If a footer is defined then the margin between the end of one page and the start of another page depends on the contents of the footer. The next example shows how to use a footer.

(1) Clear the text buffer, move the cursor to the beginning and enter text to produce the following screen display:



The start of the footer block is indicated on the display screen by **':F ←'** and the end is marked with **':←'**. The first footer line is blank to provide a one-line gap between the end of the text and the first printed footer line. **[Note that this footer format is different from that of the main text.]** The footer will start printing on line 9 of each page, and the main text on line 1. The **#** symbol (on the third line of the footer) **will be replaced by the page number** on printout.

(2) Move the cursor to the beginning of the text buffer and print the buffer. Your printout should look like this:

```
        This is the main text which will be
        printed before the footer is
        printed. The footer will start
        printing on the line number set by
        the '.B' format command. If the
        footer is too long to fit at the end

This is line 2 of a 4-line footer
FOOTER LINE 3                          Page 1

        of the page, then only the part that
        does fit will be printed. Formats
        defined within the footer block do
        not affect the format of the main
        text. If no format line is used
        within a footer block, then the

This is line 2 of a 4-line footer
FOOTER LINE 3                          Page 2

        footer block will have the same
        format as the main text.
```

NOTE: The footer was not printed at the bottom of the last page because the text ended before the footer line was reached. To insure that a footer is printed on the bottom of the last page, a page eject command (**'.E',RETURN**) must be inserted at the end of the text. The use of page ejects is described in Tutorial Section 6.6.

MAKING THE FIRST PAGE DIFFERENT

To make the best use of headers and footers (and format commands as well), it is important for you to understand the sequence of events described below.

Recall that The Writer's Tool print formatting program reads through the text just far enough to compose one print line, then prints the line and continues reading. As it encounters special format commands, it interprets each command as it finds it, makes the necessary alterations in its table of format parameters, then continues reading again. When it encounters a header or footer block, it remembers two things: where the header or footer text is stored in the text buffer, and the fact that there is a header or footer.

Before printing the first line of a page, the print program checks its memory to see if a header block has already been encountered. If it has, then the program prints the header before printing part of the main text. If not, then the main text is printed at the top of the page. If the program encounters a second header block, it forgets about the first header and remembers only the more recent one.

To defer the start of header printing to the second page it is only necessary to make sure that there is some printable text just before the header is defined. [Blank lines preceding the imbedded header will also work.] You should try this by modifying the text left over from the previous example.

SUMMARY OF HEADER AND FOOTER USAGE

1. A header is printed at the top of every page following its definition. If you want a header on the first page, make sure the header block appears first in the text file.

2. A footer is printed starting at the line set by the **'.Bnn'** format command. The footer is printed on each page following its definition.

3. A header is defined by an imbedded block of text starting with the marker line **':H',RETURN** and ending with the marker line **':',RETURN.**

4. A footer is defined by an imbedded block of text starting with **':F',RETURN** and ending with **':',RETURN.**

5. If more than one header or footer is imbedded in the text, only the last one affects the pages following it.

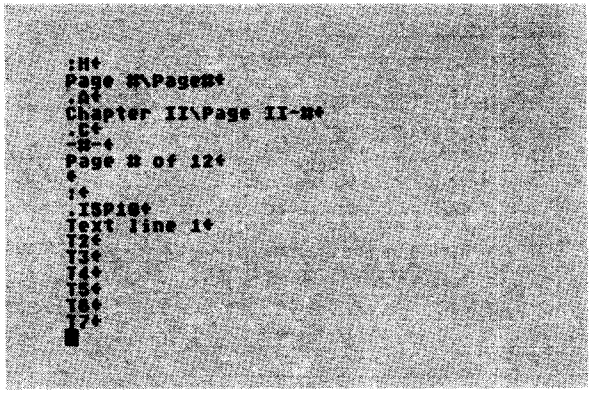
6. Formats defined within a header or footer block have no effect on the main text. If no formats are set within a header or footer, they will have the same format as the main text.

7. A simple header block can be inserted in the text by holding down both **SHIFT** and **CTRL** keys, then pressing **H** .

6.5 PAGE NUMBERING

Page numbers can appear in headers or footers, and may be left justified, centered, right justified, or placed anywhere within the header or footer text. The following example demonstrates most of these options.

(1) Go to EDIT, clear the text buffer, and enter text to produce the following screen display:



[The short page length of 10 lines is used to save paper and still let you see the page number advance from one page to the next.]

(2) Move the cursor to the beginning of the text and print it. Wherever the # symbol appears within the header (or footer), the printout will contain the current page number. Your printout should look like the sample shown on the next page.

[**Note that the second header line in this sample has left and right sides reversed on page 2.** This will happen on all even pages because of the alternate command (**'.A',RETURN**) preceding the split-justified line. This is useful if you are printing (or Xeroxing) on both sides of the paper and want page numbers to appear near the outer edge of each page.]

```
Page 1                                          Page 1
Chapter II                                      Page II-1
                              -1-
Page 1 of 12

     Text line 1
     T2
     T3
     T4

Page 2                                          Page 2
Page II-2                                       Chapter II
                              -2-
Page 2 of 12

     T5
     T6
     T7
```

(3) Go back to EDIT and at the beginning of the text buffer insert the format line **'.N101',RETURN** (this will start page numbers at 101 instead of 1). Move the cursor back to the beginning and print this version.

(4) Return to EDIT and insert five **RETURN**s at the beginning of the text buffer. This will prevent printing of the header on the first page (five blank lines will replace it).

(5) Move the cursor to the beginning of the text buffer and print again. Your printout should look the same as before except that no header will appear on the first page.

IMPORTANT FACTS ABOUT PAGE NUMBERS

1. Page numbers are produced on printout wherever a **#** symbol appears within a header or footer block.

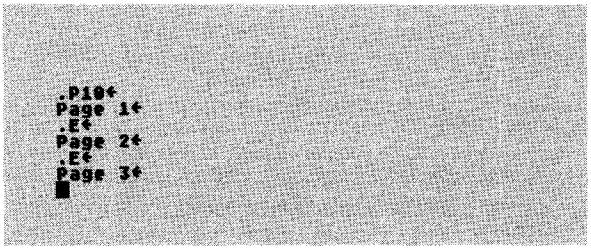
2. The starting page number can be set using the **'.Nnn'** command in a format line.

3. The alternate command can be used with a split-justified line to put page numbers on the outer edge of double sided pages.

6.6 PAGE EJECTS

The page eject command causes the printer to advance to the top of the next page. Its use is demonstrated in the following example.

(1) Go to EDIT, clear the text buffer, and enter text to produce the following screen display:



(2) Move the cursor to the beginning of the text and print the text buffer. Your printout should look like this:

Page 1

Page 2

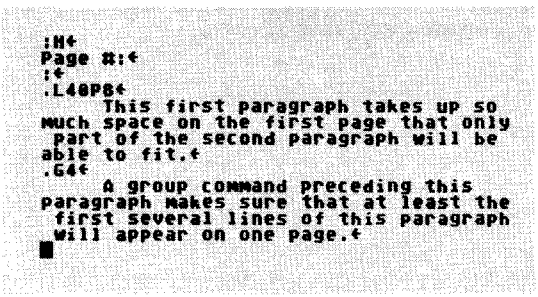
Page 3

6.7 USING THE GROUP COMMAND

The end of one page and the beginning of another (the page break) can often occur at awkward points in the text. For example, you may find a table heading at the bottom of page 3, and the rest of the table at the top of page 4. Or you may find a major section heading at the bottom of page 5, and the first paragraph of the section at the top of page 6. Even if your present version of a document prints without any awkward page breaks, they might appear after revising the document. Although these awkward page breaks can be avoided by inserting page ejects at appropriate points, you can handle this problem automatically using the group command (also known as a **conditional page eject**).

The group command is an imbedded format line with one parameter, and has the form **'.Gnn',RETURN.** This causes the printer to advance to the next page unless there are nn lines remaining on the current page. Thus the nn print lines following this command will not be split between two pages. The following example demonstrates the use of the group command.

(1) Go to EDIT, clear the text buffer, and enter text to produce the following screen display:



To save paper the page length is set to eight lines (.p8). There will be only seven lines per page available for main text (the eighth line is a blank footer line). By the time the first five lines are printed there will only be two lines available on the first page. The group command (set to four lines with **.G4**) will thus force the printer to advance to the next page.

(2) Move the cursor to the beginning of the text and print the buffer. Your printout should look like this:

```
Page 1:
     This first paragraph takes up so
much space on the first page that only
part of the second paragraph will be
able to fit.



Page 2:
     A group command preceding this
paragraph makes sure that at least the
first several print lines of this
paragraph will appear on one page.
```

(3) Go back to EDIT and change the group command to **'.G2'** instead of **'.G4'**.

(4) Move the cursor to the beginning of the text and print the text. This time there is sufficient room for the grouped text to appear on the first page:

```
Page 1:
     This first paragraph takes up so
much space on the first page that only
part of the second paragraph will be
able to fit.
      A group command preceding this
paragraph makes sure that at least the

Page 2:
first several print lines of this
paragraph will appear on one page.
```

6.8 USING SOFT HYPHENS

The readability of justified text is often marred by the large gaps between words. This problem is worst with short print lines and long words. The text can be made more readable by turning off justification. But this can produce very ragged right-hand margins. The best solution to both of these problems is to hyphenate long words so they can be split between two lines.

Soft hyphens are marker characters indicating where a word can be split if needed. If a split is not needed, The Writer's Tool will ignore soft hyphens.

EXAMPLE

(1) Load the PDEMO file at the beginning of the text buffer, then delete all but the second paragraph.

(2) Insert a format line which sets the line length to 50, then print the paragraph (and note the large gaps between words):

```
    Venus has a thick atmosphere of carbon
dioxide, with a surface pressure of 90
atmospheres, and a surface temperature of 800
degrees Farenheit.
```

(3) Use **SHIFT-|** to insert soft hyphen markers (|) in the words **diox|ide, atmo|spheres,** and **Faren|heit.** Now print the paragraph again (and note the improved readability):

```
    Venus has a thick atmosphere of carbon diox-
ide, with a surface pressure of 90 atmospheres,
and a surface temperature of 800 degrees Faren-
heit.
```

(4) Now comes the best part. Insert the word **very** in front of the word **thick,** then print the paragraph once more:

```
    Venus has a very thick atmosphere of carbon
dioxide, with a surface pressure of 90 atmo-
spheres, and a surface temperature of 800 degrees
Farenheit.
```

Note how the soft hyphens get turned into printed hyphens only when needed to improve text readability. You can also use soft hyphens after hard hyphens without having double hyphens printed.

6.9 LINE SPACING

The Writer's Tool allows four different settings of the vertical space between print lines. Within a format line preceding the text you want spaced, use

S1 for single-spaced lines (6 lines/inch)

S2 for double-spaced lines (3 lines/inch)

and, for some printers (see Reference Section 6), you can also use

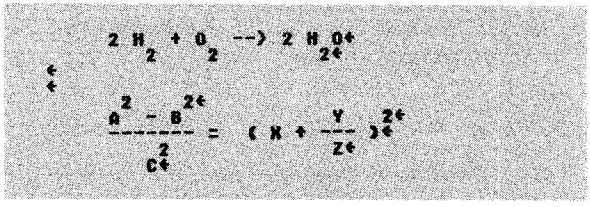
S3 for one-and-a-half spaced lines (4 lines/inch)

S4 for half-spaced lines (12 lines/inch)

[Other spacings are also possible, though not as convenient; you can imbed direct printer controls to set any line spacing your printer is capable of (see Tutorial Section 11).]

These spacings can be mixed in any way you want, without changing the length of the page, which is always measured in units of single spaced lines.

The half-spaced mode is especially useful for laying out equations and diagrams. The following screen display shows one example (**don't try it until you find out more about your specific printer capabilities**):



When printed in half-space mode (**S4**), the result is this:

$$2\,H_2 + O_2 \longrightarrow 2\,H_2O$$

$$\frac{A^2 - B^2}{C^2} = \left(X + \frac{Y}{Z} \right)^2$$

6.10 JUSTIFICATION OPTIONS

The Writer's Tool supports four different options for handling word breaks and alignment of the right-hand margin. These are all controlled by setting the **J** format parameter within a format line. To illustrate the effects of these options, the second paragraph of the PDEMO file is shown printed below for each option (all with a line length of 50).

J0 Justify off, ragged margins, word wrap on

```
    Venus has a thick atmosphere of carbon
dioxide, with a surface pressure of 90
atmospheres, and a surface temperature of 800
degrees Farenheit.
```

J1 Justify by full spaces, straight margins, word wrap on

```
    Venus has a thick atmosphere of carbon
dioxide, with a surface pressure of 90
atmospheres, and a surface temperature of 800
degrees Farenheit.
```

J2 Justify off, straight margins, word wrap off

```
    Venus has a thick atmosphere of carbon dioxid
e, with a surface pressure of 90 atmospheres, and
a surface temperature of 800 degrees Farenheit.
```

J3 Justify by microspacing, straight margins, word wrap on

```
    Venus has a thick atmosphere of carbon
dioxide, with a surface pressure of 90
atmospheres, and a surface temperature of 800
degrees Farenheit.
```

NOTE: This last option is only possible with some printers (the COMREX CR-II, and OKIDATA Microline-92 are two examples).

6.11 USING PRINT PREVIEW

The print preview function is controlled by the **'V'** format parameter, and is usually set from the print system rather than being placed in an imbedded format line.

If **V** is set to a non-zero value (say n), then all printout for page numbers less than n will be sent to the screen display for preview; pages with numbers equal to n or greater will be printed on the printer. Thus **V** is the View-to-print transition page number.

EXAMPLE

(1) Clear the text buffer, then load the DEMO file from The Writer's Tool disk.

(2) Move the cursor to the beginning of the text, then press **SHIFT-CTRL-H** to insert the simple header block. Edit the header block to remove the backslash character.

(3) With the cursor at the beginning, go to the print system, press **F**, then enter **'.V20',RETURN.** This sets the stage for printing everything to the screen, since the DEMO file doesn't have page numbers as high as 20.

(4) Press **P**, and watch what happens. The first 38 columns of each line of text will be shown on your display. Look for page numbers in the header blocks.

(5) This time, after you press **P** to get the preview started, press it again to pause the preview. Do this a few times, ending up with a pause.

(6) When you have the preview paused, take note of the first word on the last line printed to the screen. Now press **E.** This will return control to EDIT with the cursor positioned at the beginning of the first word of the last line previewed. This makes it easy to edit immediately any problems you see while previewing.

(7) Move the cursor back to the beginning, then go to the print system and change **V** to a value of 2. Press **P** to preview page 1 and print starting on page 2.

(8) Before leaving this section, reset **V** to zero, turning off the preview function.

HOW TO PRINT A SELECTED PAGE

Set **V** to the number of the page you want printed, start the print process, and press **Q** when the page has been printed. If your printer stores characters before printing, you may have to push the select button on your printer at the end of the page.

NOTES ABOUT PRINT PREVIEW

The preview print to the screen does not show exactly what your printed output will look like. Its main functions are (1) to show where page breaks will occur and (2) to allow starting the print at any page. These functions alone will probably save you a significant amount of time and paper. You can also see which justified lines have large numbers of spaces inserted between words (these probably would benefit from hyphenating the first word on the following line).

The preview will not show fractional line spaces, changes in the number of characters per inch, and most important, you can't see more than the first 38 or so characters of each line.

In some cases you will see special characters in the preview that do not show up on printout. These are printer control characters, and are best ignored.

6.12 DOUBLE-COLUMN PRINTING

Double-column printing is activated by setting the **'X'** format parameter to a non-zero value and deactivated by resetting it to zero. The value of **X** is the number of spaces between the left and right columns. The horizontal width of each column is the same and is set by the **'L'** format parameter (line length).

In double-column mode, printing proceeds as follows: (1) the header is printed; (2) the first column of text is printed; (3) the paper is reversed to the beginning of the second column; (4) the second column is printed; and (5) the footer is printed. This sequence is designed for one header and footer spanning both columns, and does not allow double headers or footers. While the second column is printing, tabs are positioned as if the left edge of the page were shifted by the same amount as the column shift.

The method of backing up the page depends on printer capabilities. Some printers (the C.ITOH PROWRITER and ATARI 825, for example) support reverse paper feeds. These can print double columns automatically. Others, like the EPSON MX-80, can only execute forward line feeds. In the latter case printing stops at the end of the first column, and a message is displayed requesting manual reversal of the paper. Once the paper is re-positioned, printing can be continued by pressing **START.** The manual mode can also be used with printers that don't require it, and may allow somewhat more accurate alignment of the two columns.

SETTING UP THE FORMAT

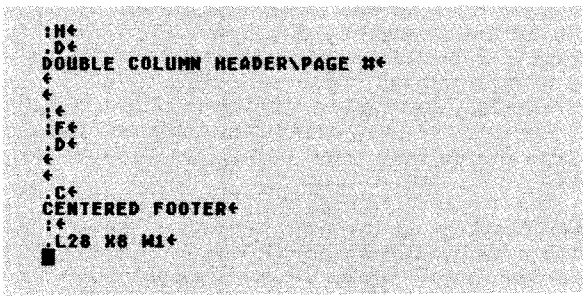
To obtain good looking double-column printout requires a carefully chosen set of format parameters. Line length (L), left margin (M), and the space between columns (X) are the three format parameters which are particularly important. Since headers and footers are not double printed, you will probably want to use longer line lengths in the header and footer blocks, so they can span both columns.

For convenience, suppose you want to use the default format for both the header and the footer. The default format has a line length of 64 and a left margin of 8. To obtain the same left and right margins in double column mode, and also allow for an 8-space margin between the two columns, the line length must be 28 characters (28+8+28=64). These parameters are used in the following example.

EXAMPLE OF DOUBLE-COLUMN PRINTING

(1) Go to EDIT and clear the text buffer.

(2) To set up the header, footer and formats, enter text to produce the following screen display at the beginning of the buffer:



The last line is a format line setting each column width to 28 characters and the spacing between columns to 8 spaces. The **'W1'** command forces manual page reversal, even if your printer does support reverse line feeds.

(3) Make sure the cursor is still positioned just past the end of the text just entered.

(4) Now go to the Disk I/O System and load the file DEMO from The Writer's Tool disk (this will save you the trouble of typing several pages of text).

(5) When control automatically returns to EDIT, move the cursor to the beginning of the text buffer and go to the Print System.

(6) Before proceeding any further, you should try to adjust the top of the paper in reference to some mark or fixture on the printer so that you will later be able to roll back the paper to the same position. [The alignment of the first and second columns depends on how well you can return the paper to its initial position.]

(7) Press **P** to execute the PRINT function. After printing the header and the first column you will see the following message displayed in the command window:

```
MOVE PRINT HEAD TO TOP OF PAGE
then press START to print 2nd COL
```

(8) Now move the paper back to where it was just before step 7, then press **START**. This will print the second column, then the footer of page 1. At this point printing will again stop (because of the W1 format setting), and you will see the message:

```
PAGE DONE...PRESS START TO CONTINUE
```

At this point your first page should look like this:

```
DOUBLE-COLUMN HEADER                                    PAGE 1

Man has tried again and          that extremely hot air must
again to harness the wind        meet cold air before a
for his own pleasure and         tornado can be produced, and
profit. He has built             it is probable that the
windmills which drive            so-called jet streams also
millstones or complicated        play an important part in
electric generators, and he      their creation.
has constructed ships and
          .                                .
          .                                .
          .                                .
Humphries believes that at       stirred in a pot, rising
least twenty-six different       higher up the sides the
fulfilled before a tornado       whirling motion pressed the
can develop. It is certain       air outwards leaving a

                    CENTERED FOOTER
```

(9) If you want to continue with the second page, position the paper as described in step 6, press **START**, and continue following the prompts until printing is completed. If you have seen enough already, just press **OPTION** to abort the print process.

SUMMARY OF DOUBLE-COLUMN PRINTING

1. The format command **'Xnn'** activates double-column printing with nn spaces between first and second columns. The command **'X0'** turns off double-column printing.

2. In double-column mode the line length format parameter (L) sets the width of each column (not the total width of both columns).

3. Manual page reverse is required for printers that do not support reverse paper feeds.

4. Headers and footers are not printed in double-column format. Their formats should be set to cover both columns.

7.0 USING SPECIAL PRINTER CAPABILITIES

The Writer's Tool provides a convenient way to use special printer capabilities such as underlining, superscripts and subscripts, boldface, compressed print, proportional spacing, and more. You can easily control these print features, provided that special information about your printer is made available to the program. The needed information about command languages and personalities of most popular printers is stored on The Writer's Tool disk in special printer data files listed below:

PRINTER FILE	PRINTER NAME	SIMILAR OR EQUIVALENT PRINTER
AT825.PPP	Atari 825	Centronics 737, 739
AT1025.PPP	Atari 1025	
AT1027.PPP	Atari 1027	
CRII.PPP	COMREX CR-II	Brother HR-15
FX80.PPP	Epson FX-80	
GEM10X.PPP	Gemini 10X	
GENERIC.PPP	Simple driver for use with any printer	
ML82A.PPP	Okidata Microline 82A	
ML92.PPP	Okidata Microline 92	
MX80.PPP	Epson MX-80	Epson MX-100
MX80G.PPP	Epson MX-80 with Graftrax-Plus	
MX80GS.PPP	(same as above)	
NEC8023.PPP	NEC 8023A-PC	
PROWRTR.PPP	C.ITOH PROWRITER 8510A-P	
RX80.PPP	Epson RX-80	

If your printer is not listed in this table you probably should stick with the GENERIC print mode (in which case you can skip the rest of this section, although you should read Tutorial Section 11).

The following subsections show how to install your printer data file and how to use your printer's capabilities.

7.1 INSTALLING A PRINTER DATA FILE

The data file for your printer must be read by The Writer's Tool before you can conveniently use any special capabilities of your printer:

(1) Go to the Print System and press **C** to activate the **CHNGE** function. You will then see the prompt: "INSERT DISK WITH CUSTOM FILE IN D1, **START** WHEN READY or **OPTION** to QUIT".

(2) In this case, insert The Writer's Tool disk and press **START.** This will first display a directory, then produce the prompt "LOAD WHICH FILE?".

(3) The printer data files are those ending with **'PPP'.** If you had an MX-80 with Graftrax-Plus, you would enter **'MX80G.PPP',RETURN.** After entering the appropriate file name, the print system screen will be redisplayed, with the name of your printer now appearing after the word **PRINTER.** After the installation, The Writer's Tool will be able to support the special features of your printer.

CHANGING THE DEFAULT PRINTER

If you would like The Writer's Tool to automatically load your printer data file when the program itself is first loaded, then follow this procedure:

(1) Use the Disk I/O System to LOAD your printer data file into the text buffer. **Be sure not to modify this file in any way.**

(2) Remove the write-protect tab from The Writer's Tool disk, then re-insert The Writer's Tool disk into Drive #1.

(3) Go to the disk system again and SAVE the contents of the text buffer using a **new filename.** The new name should be the same as the old one except that the extender should be **PDF,** instead of **PPP.**

(4) Re-install the write-protect tab on The Writer's Tool disk. Now, when you first boot load The Writer's Tool, your printer name will appear on the print system screen. Try it.

7.2 USING DIFFERENT FONTS

Strictly speaking, a font is a set of characters having specific shapes and sizes. Here, the shapes of the characters are not as important as the number of characters which can fit in a given length (the character pitch).

The Writer's Tool can recognize five different primary fonts, and seven font modifiers, some of which may also affect character pitch. The detailed characteristics of each font and font modifier vary from printer to printer. For convenience, this general discussion will use two of the most popular printers as examples: the Epson MX-80 (with Graftrax Plus), and the C.ITOH Prowriter. **The meaning of the fonts and font modifiers for other printers are presented in Reference Section 6.**

PRIMARY FONTS

The five primary fonts which can be used with a C.ITOH PROWRITER are illustrated below:

FONT NUMBER	FONT NAME	PITCH	PROWRITER PRINT SAMPLE
1	PICA	10 Char./Inch	**Pica**
2	ELITE	12 Char./Inch	**Elite**
3	COMPRESSED	17 Char./Inch	**Compressed**
4	PROPORTIONAL	22.86 Spaces/Inch	**Proportional (f4)**
5	SPACED PROP.	20 Spaces/Inch	**Proportional (f5)**

The pitches of the two proportional fonts are defined in spaces per inch because the number of proportional characters per inch depends on which characters are printed: an **M** takes up more space than an **i.** The spaced proportional font uses the same characters as font 4, but a small additional space is inserted between characters to make them more readable.

The Epson MX-80 allows only PICA and COMPRESSED fonts. The fonts available with other printers are described in Reference Section 6.

SETTING FONTS IN FORMAT LINES

The five primary fonts are selected with the **'Fn'** command placed within a format line (**'n'** is the font number listed on the previous page). For example, you can use **'.F3',RETURN** to force text after the format line to be printed in the compressed font.

ADJUSTING THE OTHER FORMAT PARAMETERS

Because different fonts take up different amounts of space, when you change a font you may also need to change the left margin, line length, and possibly tabs and indents.

If you want to calculate your own format parameters, it is important to note that most printers start printing about 1/4 inch from the left edge of the page. You should also realize that the left margin parameter used by The Writer's Tool sets the number of spaces from the printer starting point, and not from the edge of the page. Thus, if you want a one-inch left margin with a 10 CPI font, don't use an 'M10'. This would produce a margin of 1.25 inches: 1/4 inch printer offset + 1 inch from the 10 spaces produced with the 'M10' command. Instead, you should use 'M8', giving a margin of 0.8 inches + 1/4 inch (from the offset), for a total of 1.05 inches.

If you would rather not calculate your own format parameters, you can use those listed in the following table. Margins are calculated assuming that the printer starts 1/4 inch from the left edge of an 8.5-inch wide page. In this table, **RETURN** at the end of each format line is denoted by **<RET>**.

FONT	PITCH	Format lines for 1-inch margins	Format lines for 1.5-inch margins
Pica	(10 CPI)	.F1 M8 L64 <RET>	.F1 M12 L56 <RET>
Elite	(12 CPI)	.F2 M9 L78 <RET>	.F2 M15 L66 <RET>
Compressed	(17 CPI)	.F3 M13 L110 <RET>	.F3 M21 L94 <RET>
Proportional	(22.86 sp/in)	.F4 M17 L149 <RET>	.F4 M28 L127 <RET>
Spaced Prop.	(20 sp/in)	.F5 M15 L130 <RET>	.F5 M25 L110 <RET>

NOTE: Recall that these formats are not available on all printers (see Reference Section 6).

FONT DEMONSTRATION

(1) Go to EDIT and put the cursor at the beginning of the text buffer.

(2) Insert The Writer's Tool disk in drive 1, and load the demonstration file **FDEMO.**

(3) Enter the PRINT system and make sure that your printer is the one installed (if it isn't, then go back to Tutorial Section 7.1 and follow the installation procedure).

(4) Make sure your printer and Interface Module are connected and turned on. Then press **P** to print the demonstration file. If you used a PROWRITER, or NEC8023, your printout will look like this:

(.F1M20L40) Four score and seven years ago our fathers brought forth on this continent a new nation.

(.F2M24L47) Four score and seven years ago our fathers brought forth on this continent a new nation.

(.F3M24L67) Four score and seven years ago our fathers brought forth on this continent a new nation.

(.F4M46L91) Four score and seven years ago our fathers brought forth on this continent a new nation.

(.F5M40L80) Four score and seven years ago our fathers brought forth on this continent a new nation.

The format settings for each paragraph are indicated in parentheses. In each case the left margin is about 2.25 inches and the line length about 4 inches.

NOTE: If you used an MX-80 printer, your printout of Elite and Proportional fonts will not work properly. Since MX-80's cannot print these fonts, the default font will be printed instead. To avoid ugly results like this, don't try to use fonts that your printer can't print (see Reference Section 6.).

7.3 USING FONT MODIFIERS

Besides setting different fonts within format lines, the appearance of the print can also be changed within a line of printed text by means of FONT MODIFIERS. These are special characters which act like start/stop switches marking the beginning and ending points of a particular print modification.

MODIFICATIONS SUPPORTED BY THE WRITER'S TOOL

The following table summarizes the seven font modifications supported by The Writer's Tool, including how to insert the modifier characters, and examples of what they look like on the screen and how they affect the printout.

MODIFICATION	Keyin to Insert Modifier Char.	DISPLAY SAMPLE	MX80G PRINT SAMPLE
None		Normal	Normal
Emphasized	SHFT-CTRL-E	Emphasized	Emphasized
Double-Strike	SHFT-CTRL-D	Double Strike	Double-Strike
Italics	SHFT-CTRL-I	Italics	*Italics*
Double-Width	SHFT-CTRL-W	Wide	W i d e
Underlining	SHFT-CTRL-U	_Underlined_	Underlined
Superscript	SHFT-CTRL-↑	A↑2↑ + B↑2↑	$A^2 + B^2$
Subscript	SHFT-CTRL-↓	A↓i↓ + B↓ij↓	$A_i + B_{ij}$

In each of these examples, a special character was inserted just before and just after the modified word in order to produce the indicated modification. The 'DISPLAY' column shows how the special characters look on your monitor. The 'PRINT SAMPLE' column shows what effect they have when the text is printed on an MX-80 with Graftrax-Plus. To achieve the desired effect you must also have a printer which can support these modifications. [The GENERIC print mode, doesn't support any of the modifications.]

HOW TO INSERT MODIFIER CONTROLS

The Writer's Tool provides special conveniences for inserting the font modifier characters in your text. The appropriate keystrokes to insert each modifier are shown on the previous page and discussed below.

Modifiers can be inserted by pressing and holding down both **CTRL** and **SHIFT** keys, then pressing a letter or arrow key. The letter and arrow keys are easy to remember: **E** for Emphasized, **D** for Double-strike, **I** for Italics, **W** for double-WIDE, **U** for Underline, and the up and down arrow keys for superscripts and subscripts. The appearance of the special modifier character suggests their functions. For example, the inverse video exclamation mark clearly suggests Emphasized print, and the inverse video slash suggests the slant of Italic characters. The up and down arrow symbols are even more obvious indicators of their superscript and subscript functions.

The font modifier characters can also be produced using a sequence of keystrokes. The inconvenient way to produce an inverse video exclamation mark is to press and release the **INVERSE** key, then enter **SHIFT-!**, then turn off the inverse function by pressing the **INVERSE** key again. To enter an up-arrow you can use the following sequence: **ESC**, then **CTRL-[↑]**. [These alternate methods will be needed if you want to find or modify these characters using the SEARCH system, since this system will not respond to the **SHIFT-CTRL-** key-ins.]

IMPORTANT FACTS ABOUT FONT MODIFIERS

1. Font modifiers are special characters used to start and stop a modification in the appearance of printed text.

2. Font modifiers should always be used in pairs.

3. Not all modifiers can be used with all printers.

4. The underline start/stop symbol (entered with **SHFT-CTRL-U**) is different from the underline character (entered with **SHFT-[_]**). Thus, if you want a fixed number of blank spaces underlined, just enter the appropriate number of underline characters.

8.0 LINKPRINTING

Linkprinting allows you to print an unlimited number of disk files as one continuous document. You can also print a mixture of text in memory with text on disk, just by inserting specially formatted linknames which identify the diskfile to be printed. The procedure for linkprinting is illustrated by the following example:

(1) Make sure that The Writer's Tool disk is in drive #1. This disk contains a demonstration file (LDEMO) which is needed for this example.

(2) Go to EDIT and enter the following text at the beginning of the buffer:

```
<D1:LDEMORETURN
THAT WAS PRINTED IN DEFAULT FORMAT.RETURN
.I20RETURN
NOW AN INDENTED VERSION IS PRINTED.RETURN
<D1:LDEMORETURN
```

The one-line paragraphs displayed as '**<D1:LDEMO←**' define links to the disk file D1:LDEMO. The first character (<) alerts the printing program that a linkname is present.

(3) Move the cursor to the beginning of the text buffer, then enter the Print System (but don't try printing yet).

(4) Once the Print System screen is displayed, press **L** to start the LINKPRINT function. This will display the warning message:

```
WARNING:Text now in memory may be
modified by linked printing
OK TO CONTINUE (Y/N)?
```

In some cases you may want to save the text buffer on disk before printing. In this example, it's OK to destroy the text buffer.

(5) Answer the prompt with **'Y',RETURN** and the linkprinting will begin. During the printing you will see the message "Printing Link File D1:LDEMO" displayed in the command window.

When linked printing is completed, your printout should look like this:

```
    This is the text stored in the LDEMO file used to
demonstrate linked printing. Since this file does not contain
any imbedded format commands its print format is determined by
whatever format commands preceded it.
THAT WAS PRINTED IN DEFAULT FORMAT
                    NOW AN INDENTED VERSION IS PRINTED
                        This is the text stored in the LDEMO
                    file used to demonstrate linked printing.
                    Since this file does not contain any
                    imbedded format commands its print format is
                    determined by whatever format commands
                    preceded it.
```

Here is what happened. The printing program read and printed text from the buffer until it found the linkname. At this point, it deleted the part of the buffer which was already printed, then loaded D1:LDEMO into the text buffer, in front of the text which follows the first linkname. Once loaded, the linkfile becomes part of the text buffer and printing continued until another linkname was encountered. Again the printed part of the buffer was deleted and the new link file was loaded, in the same manner as before. Printing stops when the last character is printed. [The most recently defined format, and the header and footer lines are not erased by the linking process.]

(6) Return to EDIT and note that the original text entered in step 2 is no longer in the buffer.

SOME APPLICATIONS

Suppose you want to print a copy of your recently revised history paper which you have split into five chapters each stored in a single diskfile and all on one disk. This could be done by entering the following into the text buffer:

```
<D1:CHAP1RETURN
<D1:CHAP2RETURN
<D1:CHAP3RETURN
<D1:CHAP4RETURN
<D1:CHAP5RETURN
```

Executing the LINKPRINT function would then print all five chapters. If you first saved the text buffer under the name 'D1:HIST.PAP', next time you wanted to print a complete copy of the paper you would only need to enter

<D1:HIST.PAP**RETURN**

and then execute the LINKPRINT function. This is an example of a nested link: the first link file contains links to other files.

Suppose you send out a Christmas letter each year to a dozen different relatives. Most of each letter is the same general family news and some of each letter is personalized to each individual.

An easy way to handle the Christmas letter problem is to write all the common news and store it in a disk file (say D1:COMNEWS), then link this into each individual letter in the following manner:

Dear Uncle John,**RETURN**
RETURN
<D1:COMNEWS**RETURN**
RETURN
By the way, we will be able to accept
your invitation to spend the week following
Christmas with you and Aunt Helen at Vail.**RETURN**
RETURN
Sincerely,**RETURN**

These are just a few of many possible applications of the LINKPRINT function.

IMPORTANT FACTS ABOUT LINKPRINTING

1. A linkname is specified in a one-line paragraph starting with '<' and followed by the name of a diskfile. The diskfile name must have the form Dn:NAME, where n refers to the drive number.

2. In addition to the imbedded linkname, linked printing requires that the LINKPRINT function be selected from the Print System.

3. The LINKPRINT function modifies the text buffer during the printing process.

9.0 USING THE MERGE SYSTEM

The MERGE system allows you to print a merged document which combines a document template with information either entered from the keyboard or automatically read from a diskfile. The template might be a form letter which needs specific names and addresses to be filled in. The merge system can read a diskfile with names and addresses, fill them into the template, and print the result automatically. If you have only a few letters to send, then merged printing is not a great advantage. If you have a large number of letters to send, merged printing can be invaluable.

The MERGE system also allows you to create a data base of information which is designed to match the requirements of a particular template. Creating a data base and producing merged printout both rely on the template document. The next section shows how to create a template.

9.1 PREPARING A TEMPLATE

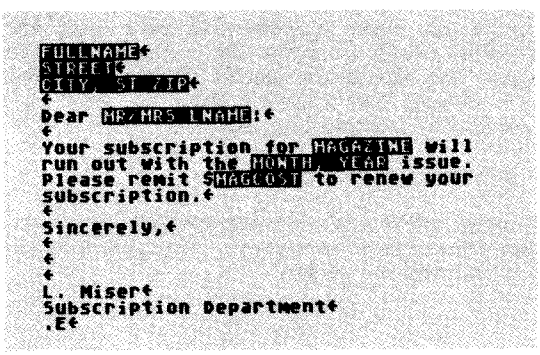
A template is different from any other document in only one respect: it has entries which define the location and names of variables. Wherever you want a variable to appear, just type the name of the variable in inverse video. These inverse video names will be used by the merge system to prompt you when you create a data base, and will be displayed next to items read from the data base during the merged printing process. The names themselves do not become part of the data base. Work through the following steps to create a template which will be used in the following sections.

(1) Go to EDIT and clear the text buffer.

(2) In the following, **[INV]** stands for the INVERSE key (this is used to start and stop inverse video text entry). To create a template, type the following text into the buffer:

```
[INV]FULLNAME[INV]RETURN
[INV]STREET[INV]RETURN
[INV]CITY, ST ZIP[INV]RETURN
RETURN
Dear [INV]MR/MRS LNAME[INV]:RETURN
RETURN
Your subscription for [INV]MAGAZINE[INV] will
run out with the [INV]MONTH, YEAR[INV] issue.
Please remit $[INV]MAGCOST[INV] to renew your
subscription.RETURN
RETURN
Sincerely,RETURN
RETURN
RETURN
RETURN
L. MiserRETURN
Subscription DepartmentRETURN
.ERETURN
```

After entering this text, you should see the following screen display:



(3) You have now completed a template. Move the cursor to the beginning of the template, insert an initialized blank disk in drive #1, then save the template using the filename **SUBLET.**

9.2 CALLING UP THE MERGE SYSTEM

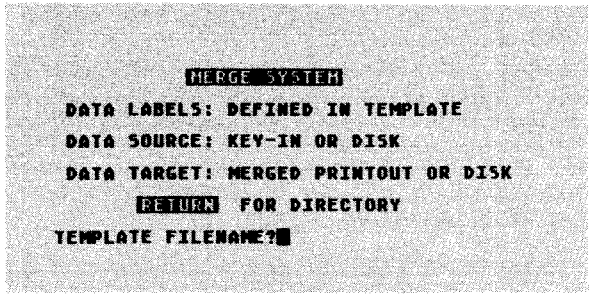
(1) Go to the Print System, then press **M** to activate the MERGE system. This will produce the prompt

```
WARNING: Text in Memory will be Erased.
OK TO CONTINUE (Y/N)?
```

(2) Answer the prompt with **'Y',RETURN** to continue with the loading of the Merge System. This will produce a second prompt:

```
INSERT Master Disk in Drive 1
START WHEN READY or OPTION to QUIT.
```

(3) After inserting The Writer's Tool disk, press **START.** This will load the merge system, after which you will see the Merge System Sign-on screen:



DATA LABELS are the inverse-video names placed in the template document. The SOURCE of the data input can be either the keyboard or a diskfile containing data items in the same order as the labels. The TARGET (or destination) of the input data can be either (1) a printed document in which the data items will replace the labels, or (2) a diskfile which will contain just data items matching the data requirements of the template. Now go to the next section.

9.3 PREPARING A DATA BASE

DATA LABELS in the TEMPLATE document are used to prompt for data input in the creation of a data base. The following example assumes that you have already created the template document named SUBLET, and that you have the Merge System Sign-on screen displayed on your monitor.

(1) Before Merge functions can be activated, you must define which template document should be used. Insert the disk containing the file SUBLET, then press **RETURN,** followed by **1** to obtain a directory for disk drive #1. After the directory you will again see the prompt "TEMPLATE FILENAME?"

(2) After verifying that you have the right disk, answer this prompt by entering **'SUBLET',RETURN.** After SUBLET is loaded, you will see another prompt:

DATA SOURCE? KEYBOARD DISKFILE

(3) Answer this prompt by pressing **K.** The next prompt will be

DATA DESTINATION? PRINT APPEND CREATE

(4) In this prompt "PRINT" means that the data input will be merged with the template and printed; "CREATE" means that the input data items will be stored in a new diskfile; and "APPEND" means that the input data will be added to the end of an already existing diskfile. For this example press **C** to create a new data base file. The next prompt will ask you to specify the name of the new diskfile: "DESTINATION FILENAME?"

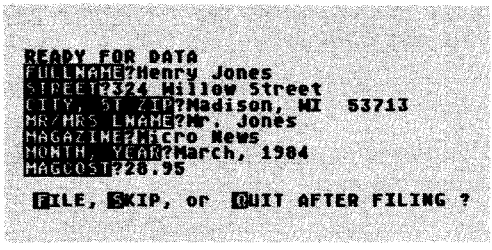
(5) Enter **'SUBDAT',RETURN** as the name of the new data base file. Finally, you will be asked "VERIFY EACH DATA GROUP (Y/N)?"

(6) Here "DATA GROUP" means one complete set of data items matching all the labels in the template. "VERIFY" means to accept or reject each data group just after completing it. On keyboard input, this gives you a chance to catch mistakes before they get written to your data base file. On diskfile input this allows you to select which data groups will be printed or written to another file. For this example, answer by pressing **Y.**

(7) This completes setting up the data base creation function. Now begins the data entry. Below the line saying "READY FOR DATA", you will see the first data label presented as a prompt for data input. Answer this prompt and the remaining prompts in this first group as indicated below:

LABEL PROMPT	YOUR ANSWER
FULLNAME	Henry Jones**RETURN**
STREET	324 Willow Street**RETURN**
CITY, ST ZIP	Madison, WI 53713**RETURN**
MR/MRS LNAME	Mr. Jones**RETURN**
MAGAZINE	Micro News**RETURN**
MONTH, YEAR	March, 1984**RETURN**
MAGCOST	28.95**RETURN**

(8) After the last entry is made you should see the following screen display:



(9) The prompt at the bottom of the screen asks for verification. In this case press **F** to accept the data group and have it filed in the data base. Next you will be prompted for entry of another group of data items. This time make up your own entries, but when you get to the end of this second group, press **Q** to file the second group and quit. This completes the data base, and presents a new prompt:

NEW TEMPLATE NEW DATA or QUIT ?

Press **Q** to return to the Print System. Your data base is now safely stored in the diskfile SUBDAT.

9.4 PRINTING A MERGED DOCUMENT

This section assumes that you have already created a template called SUBLET and a data base called SUBDAT, and that you are in the Print System.

(1) Press **M**, then follow the prompts needed to load the Merge System. When the Merge Sign-on screen is displayed, go to step (2).

(2) Remove The Writer's Tool disk from drive #1 and insert your disk containing the SUBLET and SUBDAT files.

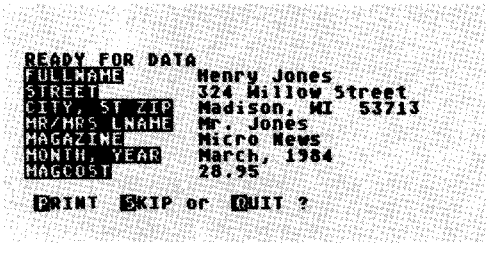
(3) Answer the "TEMPLATE FILENAME?" prompt by entering **'SUBLET', RETURN.**

(4) Answer the "DATA SOURCE?" prompt by pressing **D.** This means that data will be read from a diskfile.

(5) When prompted for the filename of the data source, enter **'SUBDAT', RETURN.** This is the diskfile containing the data items to be merged with the template. If you had forgotten the filename, pressing just **RETURN** would have given you a chance to see a disk directory to jog your memory.

(6) Answer the "DATA DESTINATION?" prompt by pressing **P.** This is needed to produce a merged printout.

(7) Answer the "VERIFY?" prompt by pressing **Y.** You should then see the following screen display:



(8) Make sure that your printer is turned on, selected, and connected. When your printer is ready, press **P** to produce merged printout of the first data group:

```
Henry Jones
324 Willow Street
Madison, WI  53713

Dear Mr. Jones:

Your subscription for Micro News will run out with the
March, 1984 issue.  Please remit $28.95 to renew your
subscription.

Sincerely,


L. Miser
Subscription Department
```

(9) After this printout is completed, the second data group will be read from the diskfile and displayed for your verification. You can print or skip this second group as you wish. When all of the data has been read from the data base file, you will see the message "REACHED END OF DATA FILE", and then the prompt for "NEW TEMPLATE, NEW DATA, or QUIT?". Press **Q** to return to the Print System.

9.5 MERGED PRINTING WITH KEYBOARD INPUT

Producing merged printout directly from data entered from the keyboard requires only a slight change in the procedures followed in the previous section: you should press **K** when presented with the "DATA SOURCE?" prompt. Of course you will still need to specify a TEMPLATE FILENAME, and press **P** when presented with the "DATA DESTINATION?" prompt.

9.6 INSERTING NON-PRINTING COMMENTS

Any one-line paragraph which begins with an exclamation point will be treated as a comment and will not be printed by The Writer's Tool. If you make such a one-line comment a data label (by entering the text after the ! in inverse video), then whatever data you enter to replace that label will also not be printed. However, the data will be written to a diskfile when you create a data base. This provides a means by which you can place markers at the beginning of each data group. These markers will help you find your way around the data base file if you decide to edit the file using The Writer's Tool.

If you made the first data item a comment, then entered * when prompted for that data, each group of items in your data base file would begin with a *. You could then use the search function of The Writer's Tool to quickly scan from one data group to the next. This would greatly simplify the editing of the data base.

9.7 EDITING A DATA BASE FILE

Once you create a data base file with the Merge System, you can edit the file with the EDIT system of the Writer's Tool. Just load the file into the text buffer, and proceed as usual. When you load a data base file, you will see that there is one line for each data item, but no marker to tell you when one data group ends and another begins. Usually this is obvious from the data items themselves. For editing convenience you may want to insert group markers using the procedure described in the previous section.

IMPORTANT NOTE

When editing a data base file, **do not add or delete lines, unless you add or delete an entire group.** If you change the number of lines within a group, the data will no longer be in phase with the template document and the data items following the error will be inserted in the wrong place.

9.8 REPEATED DATA ITEMS.

All previous examples have used a template that requires a fixed number of data items. This also results in a data base with a fixed number of data items per group. The Writer's Tool can also support data bases with variable length groups.

An example of a variable length data group is a typical invoice. Suppose you run a small mail order business and you want to keep track of each customer's order. You might want your data base to contain the following data items:

```
customer's name
customer's address

description of ordered item #1
unit price of ordered item #1
quantity of ordered item #1
price of total item #1 order

description of ordered item #2
unit price of ordered item #2
quantity of ordered item #2
price of total item #2 order

total price of all ordered items
```

This kind of data base presents a problem because the number of different kinds of items isn't the same for each customer. Dealing with this problem requires the following template structure:

```
NAME LABEL
ADDRESS LABEL

--BEGIN REPEAT MARKER--

DESCRIPTION LABEL
UNIT PRICE LABEL
QUANTITY LABEL
SUBTOTAL LABEL

--END OF REPEAT MARKER--

TOTAL COST LABEL
```

The Merge System requires an Inverse-Video < to begin the repeat block and an Inverse-video > to end the repeat. When you prepare a data base using a template structure like this, you will see "----BEGIN REPEAT BLOCK--" at the beginning, and at the end you will see "R=REPEAT or C=CONTINUE". Entering **r** (or **R**) will cause the block to be repeated. Entering **c** (or **C**) will cause the merge function to move to the next item after the repeated block. The R's and C's will become part of the data base, so you can easily edit the data base to add or delete repeated items.

9.9 ADDING TO A DATA BASE

To add new data groups to the end of a previously created data base, choose the APPEND option when asked for the data destination. Then enter the filename of the old data base. The append operation will not destroy data already entered into the data base.

If you keep adding to a data base, you eventually will create a data base that is too long to be edited by the EDIT system of The Writer's Tool. If you ever need to edit a large data base you will first have to break it into smaller files. This can be done as follows: use the large data base as a source, create a new file as the destination, then use verify to select which groups are copied from the source to the destination.

If you want to add a new data item to a data base, you will have to use EDIT to insert an extra item in each data group, then add a corresponding data label to the template.

10.0 EXTERNAL MENU FUNCTIONS

The Writer's Tool Master Disk contains auxiliary programs which do not normally reside in memory with the word processor. These programs can be accessed from the word processor environment, but must be loaded from the Master Disk each time they are used. They are loaded and executed by making selections in an **External Function Menu** which is accessible from the Main Menu. When one of these programs completes its task, it transfers control back to the word processing environment, without having to reboot The Writer's Tool. This provides convenient and rapid access to significantly expanded capabilities.

ACCESSING THE EXTERNAL MENU

First press **OPTION** to display the main menu, then press **X** to display the External Menu:

The function names on this menu have the following meanings:

Spell This program can spot spelling and typographical errors, and mark the errors, as well as aid you in making corrections. It always consults a master dictionary, but can also check words against any number of user dictionaries.

Dictm This dictionary management program is useful in building and correcting a large user dictionary. Small user dictionaries can be created with **SPELL**.

Custm This program helps you create custom default files which control default format settings and screen display characteristics.

Prdat This program can be used to modify or create your own specialized printer data files.

Main This refers to the main menu. Press **M** to leave the External Menu and return to the Main Menu.

Once this menu is displayed, and the Master Disk is in drive #1, the desired function can be activated by typing the first letter in the function name. This procedure is demonstrated in the following tutorial sections.

11.0 THE WRITER'S TOOL SPELLING CHECKER

The Writer's Tool Spelling Checker can help you produce error free text by finding spelling and typographical errors in your documents. It will remove much of the tedium of proofreading, but it does have two important limitations: it will not spot incorrect word usage; and, because of limited dictionary size, some correct words may be flagged as possible errors. This second limitation can be largely avoided by creating an additional dictionary which contains the proper names, technical words, or less commonly used words which occur in your writing.

Here is a brief description of the checking and correction process. The first step is to read a text file and compile a list of distinct words appearing in the file. The second step is to compare these words with one or more dictionary files, and compile a new list containing only words which were not found in the dictionaries (this is the list of possible errors). The final step is to re-read the original file, point out possibly misspelled words, then write a new file which has these words either marked or corrected.

11.1 USING THE SPELLING CHECKER

To illustrate the use of the spelling checker you should first create a file to be checked:

(1) Load DEMO from The Writer's Tool Master Disk.

(2) Using Search and Replace, change the word "believes" to "beleives" and then save the the text under the file named JUNK.

Now that you have a file to proofread, proceed with the following steps.

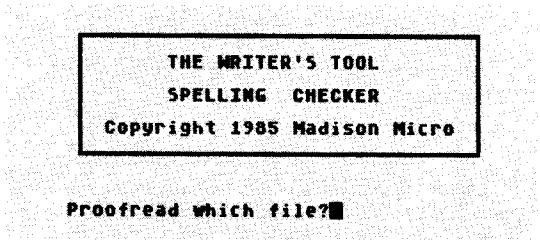
(1) After calling up the External Menu, insert the Writer's Tool Master Disk, then press **S.** If the text buffer is empty, the program **SPELL.EXT** will immediately load and execute.

If the text buffer is not empty when the function is selected, then a warning will be issued:

```
WARNING:Text in Memory Will be Erased
OK to CONTINUE (Y/N) ?
```

If you want to save what's currently in the text buffer, then answer **N.** This will return control to the MAIN MENU from which you can access the disk I/O system and perform the save.

If you answered with with **Y**, or you got no warning at all, then you should see the following screen display:



(2) Insert in drive #1, the disk containing the file to be proofread. (If you have more than one drive, you could use drive #2, as long as you include 'D2:' as part of any filename specification.)

(3) At this point you should enter the name of the file to be proofed. (For this example the filename is JUNK.)

Anytime you aren't ready to answer a filename request, just press **RETURN.** This will produce the prompt

Pressing **Q** at this point would allow you to return to the word processing environment. Pressing **1** or **2** would produce a directory, followed by another request for the filename.

(4) Once the file to be proofed is specified, **SPELL** will begin reading words from the file, making an alphabetical list of all the distinct words, and displaying a running count of the total number of words read and the number of distinct words. (A typical 2000 word file contains about 700 distinct words.) When **SPELL** is done reading the file, the display will change to the following:

```
   TOTAL WORDS:  993
DISTINCT WORDS:  494

Done Reading from D:JUNK

Save Alphabetic Word List (Y/N)?
```

(5) In this case, and most others, you should answer the prompt with **N.** (This save option is not needed for proofreading. It is only provided as a convenience for creating alphabetical word lists.) Answering this prompt will produce one more question:

```
Save Compressed List (Y/N)?
```

(6) At this point you could save the word list in the compressed form used in dictionary files. This also should be answered with **N.** (This save option is only provided for creation of a user dictionary, as described later.)

Answering **N** to both of these prompts will activate the next step in the proofreading process, which begins with a prompt to insert the Master Dictionary:

```
Insert Master Dictionary in Drive 1
START when Ready or OPTION to Quit
```

[The master Dictionary is provided either on a separate disk or on the reverse side of The Writer's Tool Master Disk.]

(7) After inserting the Master Dictionary in Drive #1, press **START** to initiate the comparison process. This will produce a new screen display and the dictionary will be scanned and compared with the list of words already in memory. Each word not found in the dictionary will be displayed on the screen as it is identified. (These will be in alphabetical order.) The time to do a complete dictionary scan is slightly more than two minutes. When the scan is done, the screen display will look something like this:

```
These Words Not Found in Dictionary:

APOCALYPTIC BELEIVES CATASTROPHIC
DYERSBURG EYEWITNESS HUMPHREYS
RD REELING TH WACO WHIRLWINDS

UNMATCHED WORDS:  11

CHECK Another Dictionary (Y/N)?
```

When you first begin to use the proofreader, you will discover that the master dictionary doesn't include all the words in your vocabulary. By saving these missing words, and then adding them to your own specialized dictionary, you can significantly increase the utility of the program. Once you have created one or more of your own customized dictionaries, you can answer the prompt "CHECK another Dictionary (Y/N)?" with **Y**, and then enter the name of your custom dictionary.

(8) When you are done with dictionary comparisons or if you are just using the Master Dictionary (as in this example), answer the prompt with **N**. This produces the prompt "SAVE ERROR LIST (Y/N)?". This gives you a chance to save an uncompressed version of the error list which you can later edit with the word processor, and/or combine with another user dictionary. Once you have saved the error list, or after answering **N**, the correction/mark procedure will begin with the following screen display.

```
Insert Disk Containing D:JUNK

THEN CHOOSE...
Mark Errors Automatically
Correct Errors (or Mark Selectively)
Quit (Skip Correction)
```

(9) To correct or mark the file which was proofread, it is necessary to re-read the original file. The disk containing this file should be inserted into the drive specified in the filename appearing at the top of the screen. (If only D: appears as the drive specifier, then use drive 1.)

The next step is to specify whether to Mark the file, Correct the file, or Quit. The meanings of these options are as follows:

MARK This creates a file in which each unmatched word is replaced by its inverse video version. This option should be selected if you want to make corrections later using the word processor. When prompted for a destination file, you may enter the same filename as the original.

CORRECT This option creates a corrected or selectively marked file which must have a different name than the original file. In this case the text will be displayed on screen, with the unmarked word highlighted. For each "error" you will be asked to choose between marking, correcting, ignoring, or omitting the word in question. If you choose **correct**, then you will be prompted to enter the correct spelling. If you choose **ignore**, then this particular occurrence of the word will be copied to the correction file without any change. If you choose **omit**, then the word will be deleted from the error list and all further occurrences of the word will not be flagged.

QUIT This allows you to choose between returning to word processing or restarting the spelling checker.

To get a feel for the way the Mark and Correct options work you should work through the tutorial twice, first ending with the **Correct** option, and second, ending with the **Mark** option. After using the Mark option be sure to return to the word processing environment and read the marked file into the text buffer and try making corrections with the search and replace function.

11.2 CREATING A SMALL USER DICTIONARY

Creating your own specialized dictionary will greatly add to the usefulness of the spelling checker. You may want to create one dictionary with proper names, another dictionary with technical words used only in biology reports, and a third dictionary containing words you commonly use but just don't happen to be in the Master Dictionary. Of course all these different words could be put in one dictionary. The main advantage of such user dictionaries is their efficiency. A generalized master dictionary large enough to handle everyone's special vocabularies would occupy several disks and take a half hour or more to read. But a modest sized master dictionary coupled with a customized user dictionary can easily fit on one single-density disk and take only a few minutes to check. The following general steps are required to create a small user dictionary (1,000 words or less). Once you have created a dictionary, you can continue to add words using the dictionary management program described later.

1. To create a user dictionary you must begin with a list of words, generally words not present in the master dictionary. The most common way to collect such words is to proofread your documents and then save the unmatched words in individual files. You could also proofread address files, and save the unmatched words (these would mainly be proper names and names of streets and cities).

2. Using The Writer's Tool word processor, load error files or other word lists one after the other into the text buffer (total words should not exceed 1000).

3. Edit this combined word list as needed to make sure that it doesn't contain misspelled words. It doesn't matter what the word order is, whether words are in upper or lower case, or whether words appear more than once. Words can be separated by one or more spaces or carriage returns. When you've completed your editing save the final result in a diskfile.

4. Use the spelling checker to "proofread" this file. Save the alphabetic word list under the same filename so you'll have an organized list of which words are in your dictionary. Finally save the compressed version using the extender **'.DIC'** to remind you that the file is in dictionary format. Once the compressed list is created, it can be used from then on as a dictionary.

Once you start a small dictionary using the above procedure, you can add words or remove words using the dictionary management program described in the next section.

11.3 MANAGING A LARGE USER DICTIONARY

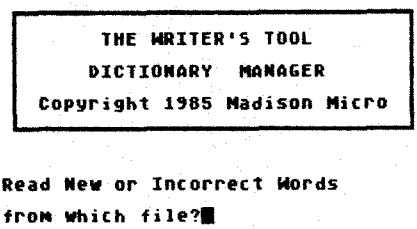
The dictionary management program **DICTM.EXT** allows you to modify an existing dictionary by adding or subtracting words. Although you can't add or subtract more than about 1,000 words at a time, there is no limit on the total number of words added (except for disk space). Words to be added or subtracted must be placed in a diskfile before starting the management program. The step-by-step procedures are as follows.

1. Create a small user dictionary using the "SAVE COMPRESSED WORDLIST" function of **SPELL.EXT.**

2. Create a disk file containing words which are to be removed from or added to the dictionary. These words do not have to be in any particular order. Upper case or lower case is also allowed. The words do need to separated from each other by spaces or some other common word separator.

3. Go to the External Menu (by pressing **X** from the main menu).

4. Select the dictionary management function by pressing **D.** This program will sign-on with the following display:



5. Answer the prompt by specifying which file contains the words which you want to add or remove from the dictionary. [Whenever you are asked for a filename, you can press **RETURN** to get access to a disk directory listing.] After the filename is entered, words from this file will be read into memory, sorted, and counted. Total and distinct word counts will be displayed while the file is being read. When this process is completed you will be prompted to enter the name of an existing dictionary file.

6. You should insert the disk which contains the dictionary to be modified, then enter the appropriate filename (say **MYOWN.DIC**). As soon as the name of the dictionary is specified, the old version will be renamed by changing the extender to **.OOO** (e.g. **MYOWN.OOO**).

7. The next prompt will ask whether the distinct words on the modification list are to be added to the dictionary or removed from the dictionary. If addition is selected, only new words will actually be added; words which are already in the dictionary will be listed on the screen, along with the total count of new words added. If word subtraction is selected, words which are removed from the dictionary will be displayed and counted. After either addition or removal, the total number in the revised dictionary will be displayed.

Since the new dictionary and the original dictionary must both reside on the same disk, the largest dictionary which can be created this way is limited to about half of the total space available on the disk. A second limitation is that the list of words to be added or removed must fit into the available memory space (a thousand words is probably the most you should allow on a single update list).

In case you are still wondering why words might be removed from a dictionary, the answer is to eliminate misspelled words which were inadvertently added.

12.0 USING THE CUSTOMIZER PROGRAM

The Writer's Tool Master Disk actually contains two programs which might be called customizers: **PRDAT.EXT** and **CUSTM.EXT.** The first program allows you to modify or create customized printer data files, and is described in the reference guide. The second program is the main topic of this section. It allows you to personalize The Writer's Tool to suit your own purposes and tastes.

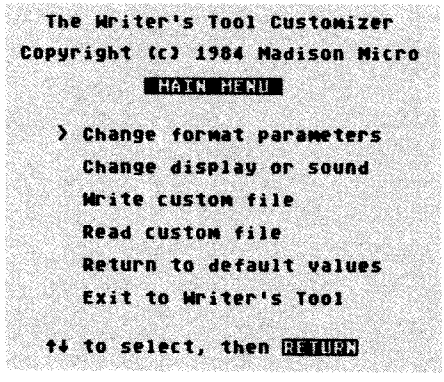
The customizer program **CUSTM.EXT** presents lists of format and screen display parameters and their original default values. You can select those you want to change, then alter them as desired. When you've finished making changes, your changes can be written to a disk file for later use by The Writer's Tool.

LOADING THE CUSTOMIZER PROGRAM

The customizer program is accessible from the External Menu.

(1) Press **OPTION** to display the Main Menu, then press **X** to display the External Menu.

(2) Insert The Writer's Tool disk in drive #1, then press **C** to load the customizer program (listed as **Custm** on the menu). When the load is completed you should see the following screend display:



OPERATING THE CUSTOMIZER PROGRAM

The main menu of the customizer contains four options:

CHANGE FORMAT PARAMETERS This is for setting new default values of page length, line spacing, beginning footer line, font, single sheet option, line length, left margin, justify, and all tabs.

CHANGE DISPLAY or SOUND This is used to change the cursor flash period and brightness, the left screen margin (allows full 40 column display), the color and luminance (brightness) of the background, and the luminance of the characters. If you have an old ATARI 800 you can also choose whether beeps are sounded by the console or TV monitor. (With an XL computer, all sound is piped through the monitor).

WRITE CUSTOM FILE This is for writing your selections into a disk file for later use by The Writer's Tool.

READ CUSTOM FILE This allows you to read a previously created custom file which you can then modify using the commands listed above.

RETURN TO WORD PROCESSING This quits the customizer and returns to the main menu of The Writer's Tool.

To create a new custom file, you should first select one of the change options and press **START**. This will display a new screen containing a list of the parameters which can be modified, their original settings, and the current custom values that you may have set. You can select which parameter to modify by using the up and down arrow keys. To increase the parameter value press the right arrow key; to decrease its value press the left arrow key.

When you've completed all your changes, press **OPTION** to return to the main menu, then **SELECT** and **START** the write option. You will then be prompted to insert in drive #1 the disk on which you want the custom file written.

Once you have the appropriate destination disk inserted, press **START** to continue to the next step. This will produce a request for an identifier (up to 8 characters long). [Since this identifier is part of a filename, it must contain only upper case letters and numbers, and start with a letter.] This identifier is not used by The Writer's Tool; it is only there to help you later distinguish one custom file from another. The CUSTOM program will add the extender ".FFF" which allows The Writer's Tool to identify it as a format file.

HOW TO LOAD THE CUSTOM FORMAT FILE

There are two ways in which The Writer's Tool can load a custom file:

MANUAL LOAD: This method is best suited for occasional use. If you have a custom format file designed only for a certain kind of document, and you keep these documents on a separate disk, then you should write the custom file on this disk, and use the following procedure whenever you are working with this disk.

(1) Go to the Print System, then press **'C'** to activate the **CHNGE** function. This will produce the prompt: "INSERT DISK WITH CUSTOM FILE IN D1, **START** WHEN READY or **OPTION** to QUIT".

(2) At this point you should insert your disk containing the custom (.FFF) file which you want to load, then press **START.** This will produce a directory of the disk, followed by the request "LOAD WHICH FILE?"

(3) Enter the name of your custom file (including the .FFF extender) then press **RETURN** to install the new default format parameters. Control will return to the Print System screen with the new default parameters displayed.

AUTOMATIC LOAD: This method is best for loading parameters which you almost always want used.

(1) Use The Writer's Tool Disk I/O System to load the '.FFF' file into the text buffer. Be careful not to modify the file in any way.

(2) Remove the write-protect tab from The Writer's Tool disk, then put it back in drive 1.

(3) Go back to the Disk I/O System and use the **SAVE** function to write the text buffer into a file with the extender **'.FDF'** [the first part of the name doesn't matter].

(4) Replace the write-protect tab on The Writer's Tool disk. Now, whenever you load The Writer's Tool, your custom default file will automatically be loaded also.

WHY USE THE CUSTOMIZING FEATURE?

There are both practical and esthetic reasons for using the customizer feature. A few examples are listed below:

If you always use the same printer, and almost always use the same font and margin settings which are different from the original defaults, you can save yourself the time and trouble of entering format commands each time you start up the program.

If you are preparing a complex document with an unusual format, and many small insertions of a different format, you can avoid having to enter the unusual primary format every time you finish a text block with a modified format. Just create a custom default file that sets defaults to the unusual format. Then every time you need to reset it you can just use the '.D' command.

If you find the original cursor flash rate to be irritatingly fast, or you dislike hearing a beep every time a search target is located, why not slow down the cursor flash and send the sound to the monitor (where you can control the volume)?

13. PHRASE BUFFERS AND DELETE BUFFERS

PHRASE BUFFERS

A relatively new feature of The Writer's Tool allows you to store text strings (phrases) at the beginning of your document, then recall them as needed, while you are entering or editing text. This can be a great convenience if you are using a very long, difficult-to-remember word, and using it often. This feature can also simplify the process of changing back and forth between several different format settings within the same document. You can store each format in a phrase buffer and recall them when needed, as many times as needed.

The phrase buffers are actually comment lines. These are non-printing text strings defined by a leading exclamation point (following a RETURN) and a terminating carriage return. To make a comment line serve as a phrase buffer, the following syntax is required:

!num=phraseRETURN

where **num** denotes one of the single-digit numbers 0,1...9, and **phrase** denotes the text that will be recalled from the buffer and inserted in the document. Up to 10 phrases can be defined, but all must appear at the beginning of the document.

The command to insert a pre-defined phrase is just **OPTION** followed by the number of the phrase to be inserted.

EXAMPLES

To see how conveniently phrase buffers can be used, try entering the following examples at the beginning of the text buffer:

```
!1=Optimized Systems Software RETURN
!3=Krzysztof M. Serkowski RETURN
!2=staphylococcus bacteria RETURN
!4=.M12 F2 L76 T12,16,24,36 B52 RETURN
```

Now move the cursor somewhere beyond the phrase definition area, and try entering **OPTION-1** through **OPTION-4** to insert the pre-defined phrases. Note that the cursor moves to the end of the inserted phrase just as if you had typed it in the old fashioned way. Entering **OPTION-5**, or trying to insert any other undefined phrase just produces a buzzer sound. As you can see by this example, the order of the phrase definitions doesn't matter. What does matter is where the buffers are located: they must all be in one contiguous block at the beginning of your document.

Each phrase can be as long as 255 characters (including the leading exclamation point and the terminating RETURN). Since the phrases are stored in comment lines they will not print, or affect the format of the document. Thus the phrase definitions can remain with the document at all times.

In some cases you will probably want to include a carriage return within the phrase itself. This can be done by entering **CTRL-TAB** wherever you want a **RETURN** to appear when the phrase is recalled. The Writer's Tool will do the translation when you insert the phrase.

DELETE BUFFERS

As described earlier, words deleted with **CTRL-D,** and lines deleted with **SHFT-DELETE** or **SHFT-CTRL-DELETE,** are stored in a delete buffer large enough to save the last five deletions. As more deletions are made, the oldest ones are removed while new ones are added. The most recent of these deletions can be recalled and inserted into the text by entering **CTRL-U.** However this command does not affect the delete buffer; entering **CTRL-U** a second time produces the same effect as the first (provided nothing was deleted in the meantime). To get at other items in the delete buffer, you must use **CTRL-N.** This command shifts all the items in the buffer by one position each time it is called, moving the 2nd item to the top, the 3rd item to the 2nd position, etc. (the first item ends up in fifth position). This command also copies the new top item into the text. After **CTRL-N,** using **CTRL-U** would insert the same item as did the last **CTRL-N** command.

Thus to recall the last 5 items deleted you should enter **CTRL-U** once, followed by four successive **CTRL-N** commands.

REFERENCE GUIDE

This reference guide describes all of The Writer's Tool commands and subsystems: what they are, how to use them, and how they function. The reference guide is complete, but brief, and not introductory. It does not present the step-by-step examples which are found in the tutorial sections of the manual, and it assumes some familiarity with the terminology and concepts presented in the tutorial. If you are already familiar with word processing software, you may be able to use The Writer's Tool effectively just by consulting the reference guide. Most of you will find the tutorial to be a useful preliminary.

REFERENCE GUIDE CONTENTS IN BRIEF

1.0 CONTENTS OF THE WRITER'S TOOL DISK

The Writer's Tool disk files and their functions are as follows:

DOS.SYS This is the file management portion of the DOS XL disk operating system (a product of OSS, Inc.), and is capable of supporting both single-density and full double-density disk drives.

AUTORUN.SYS This file contains the RAM-resident part of the word-processing program. It automatically loads after DOS.SYS, and, in turn, loads printer and/or format default files (if present), then starts The Writer's Tool.

MERGE.OBJ This is a disk-based subroutine of The Writer's Tool which assists in the creation of data bases and merged printing.

SPELL.EXT This spelling checker program is one of four programs accessible from the External Menu of The Writer's Tool.

DICTM.EXT This External Menu program performs dictionary management functions.

CUSTM.EXT This External Menu program is used to create custom format files which later can be installed manually or automatically as the default format.

PRDAT.EXE This External Menu program can be used to create and edit printer data files.

Pname.PPP Each of these printer data files contains specific information needed to make use of special printer capabilities.

DEMO This file and the PDEMO, LDEMO, and FDEMO files contain text used in the TUTORIAL to demonstrate The Writer's Tool functions.

MASTER.DIC This is a compressed dictionary file containing 20,000 words. It is provided either on the reverse side of the Master Disk or, in some cases, on a separate disk.

2.0 STARTUP PROCEDURES

Part of The Writer's Tool is cartridge-resident and the remainder boot loads from disk. With your computer off and disk drive on (and ready), insert The Writer's Tool disk and cartridge and turn on your computer.

After turn-on, DOS.SYS will be loaded first (this takes about 10 seconds). Next AUTORUN.SYS will be loaded and executed. A sign-on message will be displayed while the word processing program is being loaded (this load takes about 20 seconds).

When the load is finished, a brief initialization sequence will be performed; then control will be transferred to The Writer's Tool EDIT system. As part of the initialization, custom printer (PDF extender) or format (FDF extender) files will be installed automatically if they are present on The Writer's Tool disk.

3.0 THE EDIT SYSTEM

This is the initial operating mode and the central function of The Writer's Tool. EDIT provides for both text entry and screen-oriented text editing without mode changes. Text stored in your computer memory can be edited by moving the cursor to where a change is desired, then inserting, deleting, or typing new text over the old.

3.1 TEXT ENTRY MODES

There are two modes of text entry: type-over and insert. Each will remain active until the other is selected.

CTRL-T Activates the TYPEOVER Mode. This is the default mode of text entry. The cursor position is indicated by flashing between a character and its inverse video representation, or by a flashing underline (you can switch between these two cursor options by pressing **SELECT**). The character at the cursor is replaced by the new character entered. When used in conjunction with **SHIFT-INSERT** or **SHIFT-CTRL-INSERT** (to insert space) and **CTRL-J** (to remove unused space), this is the fastest mode for inserting text near the beginning of a large text file.

CTRL-I Activates the INSERT Mode. This mode uses a flashing vertical bar for the cursor. The character at the cursor and those to the right of the cursor are pushed forward as new characters are entered. This mode of entry becomes too slow for fast typing when there are more than several thousand characters after the cursor.

3.2 KEYBOARD CONTROL

What happens when a key is struck depends on the state of the keyboard. The keyboard is normally in lower case mode, but can be changed to other modes and returned using the following controls.

SHIFT-CAPS Locks the keyboard into upper case mode, and displays the message "CAPS LOCK" on the status line (bottom screen line).

CAPS/LOWR Pressing this key returns the keyboard to lower case, and removes the "CAPS LOCK" message from the status line.

INVERSE Pressing the ATARI logo key (on an 800) or the key with a triangle (on an 800XL) turns the INVERSE mode on and off. When it's on, inverse video characters are produced. This control is independent of the upper case/lower case control. When the **INVERSE** key is hit a warning beep will sound. While this mode is active, the message "INVERSE" will be displayed on the status line.

CTRL-CAPS Exchanges cursor and unshifted functions on the arrow keys, and displays "CURSOR EXCH" on the status line. This allows cursor movement without pressing the **CTRL** key, but when this function is active you must use the **CTRL** key to enter '+', '*', '-' , or '=' characters. Pressing **CTRL-CAPS** a second time restores normal functioning of the arrow keys.

The display of keyboard status can be erased by entering **SHFT-CLEAR** and restored by entering **CTRL-?**. **SHFT-CLEAR** also turns off the INVERSE mode.

3.3 CURSOR MOVEMENT COMMANDS

CTRL-[→] Moves the cursor one character to the right. If the cursor starts at the end of screen line, it will move to the beginning of the next screen line.

CTRL-[←] Moves the cursor one character to the left. If the cursor starts at the left margin, it will move to the last character of the screen line above the initial line. If the cursor is at the beginning of the text buffer, it won't move at all.

CTRL-[↑] Moves the cursor up one screen line. If the cursor starts at a horizontal position to the right of the end of the destination line, then it will move to the end of that line. If the cursor is at the beginning of the text buffer, it won't move at all.

CTRL-[↓] Moves the cursor down one screen line. If the cursor starts at a horizontal position to the right of the end of the destination line, then it will move to the end of that line.

CTRL-B Moves cursor to the beginning of the text buffer and puts the first line of text at the top of the screen.

CTRL-E Moves the cursor to the end of the text.

CTRL-A Moves the cursor to the beginning of the line.

CTRL-Z Moves the cursor to the end of the line.

CTRL-RETURN Moves the cursor to the end of the next screen line. This has the same effect as CTRL-[↓], followed by CTRL-A.

CTRL-W Moves the cursor to the next word. This command will skip the cursor past return symbols, comment lines, format command lines, and header and footer marker lines.

CTRL-F Page Forward. If the cursor is at the bottom screen line, the text will move up twenty screen lines. If the cursor starts at some other position, this command will first move the cursor to the bottom screen line. If you attempt to page past the end of the entered text, the message "CURSOR PAST END OF TEXT" will appear and paging will not occur.

CTRL-R Page Reverse. If the cursor is at the top of the screen, the text will move down 20 screen lines. If the cursor is not at the top, then this command will first move the cursor to the top screen line.

TAB Moves the cursor to the next screen column which is a multiple of five. This is designed for fast cursor movement. To use imbedded tabs for controlling print format, see the section on special symbols.

CTRL-S Continue search. Moves cursor to the next occurrence of the search string defined in the last use of the SEARCH subsystem. If no search string has been defined, this command will have no effect.

ALTERNATE CURSOR MOVEMENT

Another important method for cursor movement is by means of the SEARCH function. This is especially useful for finding a specific location in a long text file.

The print preview can also be used to move the cursor. Pressing **P** during the preview will pause the preview. Pressing **E** after a pause will return control to EDIT with the cursor positioned on the first character of the last line printed to the screen.

3.4 DELETE COMMANDS

CTRL-DELETE Deletes the character on which the cursor is positioned. The deleted character is replaced by the character to the right of the cursor.

DELETE(BACK S) This deletes the character to the left of the cursor and moves the cursor one character to the left. In the type-over mode of text entry the deleted character is replaced by a space. In the insert mode the deleted character is replaced by the character immediately to the right of the deleted character.

CTRL-D This deletes the word following the cursor and saves the deleted word in a 5-item buffer.

SHIFT-DELETE Deletes the line on which the cursor resides and saves the line in a 5-item buffer. This command also produces a beep to warn against accidental repeats. The most recently deleted line or word can be "undeleted" by entering **CTRL-U.** The next most recent line or word can be restored by entering **CTRL-N.**

SHFT-CTRL-DELETE Performs the same function as **SHIFT-DELETE,** but does not give a warning beep and is thus faster.

CTRL-J Join command. This deletes spaces between the cursor and the next non-space character. This is primarily used to rejoin text that is split open by the **SHIFT-INSERT** or **SHFT-CTRL-INSERT** commands, after adding new text in the space opened.

CTRL-X Deletes a previously marked block and the block markers (see BLOCK COMMANDS, Page R-10).

CTRL-CLEAR Deletes all block markers present in the text buffer (see BLOCK COMMANDS, Page R-10).

Other ways to delete text are provided by the CLEAR and SEARCH functions (see Pages R-18, R-29 for explanation).

3.5 INSERT COMMANDS

CTRL-INSERT Inserts a space between the character at the cursor and the character to the left of the cursor.

SHIFT-INSERT Inserts a blank line on the screen. In the type-over mode a screen-line's worth of blank characters (usually 38) is inserted between the character at the cursor and the character to the left of the cursor. The cursor is left positioned at the first of the inserted blanks. This is to be used to open up space for inserting new text. Extra space can then be removed using **CTRL-J**. In the insert mode this command just inserts a return.

SHFT-CTRL-INSERT Inserts all available space at the cursor position. Text after the cursor is moved as far forward as possible (all the way to the end of the memory buffer) to make room for large text inserts. This command is only effective in the type-over mode of text entry. After new text is entered, the unused space should be removed by placing the cursor at the beginning of the blank region and pressing **CTRL-J** to rejoin the text that was moved to the end of the buffer. **NOTE**: when text after the cursor is pushed to the end of the buffer, attempts to insert more space (using SHIFT-INSERT for example) will result in "OUT OF MEMORY" messages.

CTRL-U The "UNDELETE" command inserts the last deleted word or line at the cursor position.

CTRL-N Should be used after **CTRL-U** to insert the Next most recently deleted item (word or line). Up to 5 deleted words or lines can be recalled.

SHFT-CTRL-H This inserts a header block and header markers in one keystroke.

OPTION-n Inserts previously defined phrase #n at the cursor, then moves the cursor to the end of the insertion. Up to 10 phrases can be defined at the beginning of a document using comment lines in the form **!n=phrase RETURN,** where n=0,...9. Any CTRL-TAB character (displayed as an inverse-video left arrow) within **phrase** will be translated to RETURN when the insertion takes place.

Another way to insert text is to use the insert mode of text entry (explained on Page R-4).

3.6 BLOCK COMMANDS

Blocks of text can be marked, then copied or deleted. To move a block of text, first mark it, then copy it to the new location, and finally delete the original.

CTRL-M Marks a block. This inserts a block marker symbol between the character at the cursor and the character to the left. When the first marker is inserted, all text beyond the marker will be displayed in inverse video. When the second mark is inserted, only text between the two marks will be highlighted in inverse video, provided at least one mark is on the screen. A block marker cannot be inserted past the current end of text (this point can be found using **CTRL-E**). If you try to do so, the message "CURSOR PAST END OF TEXT" will be displayed on the status line.

CTRL-C Copies a marked block. This command inserts a copy of the marked block between the cursor and the character to the left. The original block and its markers are not affected by this command. When using this command you may encounter one of four error messages, each accompanied by a buzzer. If the cursor is past the end of text, the message "CURSOR PAST END OF TEXT" will appear on the status line, and no copy will be inserted. If you try to insert a copy within the marked block itself, the message "CURSOR ERROR" will appear. If the text buffer is nearly full, there may not be enough room to insert another copy of the block. In this case "OUT OF MEMORY" will be displayed. If there are fewer than two block markers, the block is not defined. Trying to copy an undefined block will produce the message "MARKER ERROR" on the status line. If there are more than two block markers, the first two will be used to define the marked block and no error message will appear.

CTRL-X Deletes a marked block (and block markers). If less than two markers are present, the "MARKER ERROR" message will be displayed and no deletion will occur. If more than two markers are present, the first two will define the block to be deleted. For this command cursor position doesn't matter.

CTRL-CLEAR Deletes only block markers. This command will delete all block markers in the text buffer, regardless of cursor position.

3.7 CASE CONVERSION COMMANDS

These commands do not affect the state of the keyboard. They only act on text already entered.

CTRL-K Converts to upper case (K sounds like C in CAPS). If the cursor is positioned at a lower case letter, the character will be converted to the upper case letter and the cursor will step forward to the next character. If the initial character is not a lower case letter, the cursor will merely advance to the next character.

CTRL-L Converts to lower case. If the cursor is on an upper case letter, then the character will be replaced with the corresponding lower case letter and the cursor will step forward to the next character. Otherwise, the cursor will just step forward.

CTRL- /|\ or **CTRL-** ◪ flips previously entered characters between normal and inverse video representations. (This command does not affect carriage returns.)

3.8 DISPLAY COMMANDS

The EDIT display screen has 23 lines reserved for showing part of the text buffer, and one line (at the bottom of the screen) reserved for displaying status and error messages. The appearance of the screen display of text, cursor, and status information can be modified using the following commands (active only within the EDIT system):

START Pressing this console switch toggles the word wrap function on and off. Turning off the word wrap is most useful for editing programs and other text files for which "words" are not the natural text unit. It is also useful for showing the exact number of spaces after the last word on a screen line. Another benefit to turning off the word-wrap function is that more text can be displayed on each screen line.

SELECT Pressing this console switch toggles the type-over cursor between the inverse video cursor (the default mode) and the flashing underline cursor. This command does not affect the cursor used for the insert mode of text entry.

SHIFT-CLEAR This erases whatever is displayed on the status line and turns off the INVERSE control if active.

CTRL-? Restores the display of keyboard status at the bottom of the screen.

In the default mode the screen lines can have no more than 38 characters. This can be increased to 40 characters using the customizer program described in Reference Section 8. With the customizer program you will also be able to change the color and luminance of the screen background and the luminance of the characters displayed.

If there is no keyboard entry for nine minutes, the display will enter the 'ATRACT' mode which produces periodic color and brightness changes designed to prevent burning in patterns on your TV monitor. Normal display colors will reappear when you use the keyboard again.

3.9 SPECIAL KEYS AND EDIT MODE FUNCTIONS

RETURN Marks the end of a paragraph and is displayed as a left arrow. It also marks the end of both a print line and a screen line. When this key is pressed the cursor moves down to the beginning of the next line. The cursor cannot be positioned to the right of the **RETURN** symbol.

SHIFT Hold this key down while pressing other keys to produce upper case letters or to activate secondary commands (shown at top of key legends).

CTRL Hold this key down while pressing other keys to produce graphic characters, or activate command functions.

TAB Causes cursor to move to the next screen tab column. (Tab columns are set at 5-space intervals and cannot be changed using tab SET or CLEAR functions.) To use tabs to control print format requires inserting the **TAB** character, accomplished by pressing **ESC,TAB.**

ESC The escape key is used to insert characters in the text which would otherwise be interpreted as commands. For example, to insert the **TAB** symbol in the text, enter **ESC,TAB.** Without the initial **ESC**, the **TAB** would cause the cursor to move to the next tab column. To insert the escape character in the text, enter **ESC,ESC.** As these examples indicate, pressing **ESC** allows the next character to bypass the command interpreter.

BREAK Has no effect in EDIT or any of the subsystems.

OPTION-n Inserts phrase defined in one of ten (n=0,...,9) comment lines of the form **n=phrase, RETURN** placed at the beginning of a document.

Special keys used to control the keyboard are described in Reference Section 3.2

3.10 SPECIAL CHARACTERS

The Writer's Tool reserves special characters for controlling printer output. You should avoid using these characters unless you want to invoke the print formatting functions associated with them.

MARKER CHARACTERS

These Characters have special significance when they appear at the beginning of the text or immediately following a RETURN character.

[.] Marks the beginning of a format line. (Format lines are explained on Page R-34.)

[:] Starts a marker line which delineates header and footer blocks (see Reference Section 5).

[!] Marks the beginning of a comment line (not printed).

[<] Marks a line containing a link name. The following characters are interpreted as the name of a LINK file. If LINKPRINT is active, the named file will be loaded and printing will resume at the beginning of the newly loaded file.

[@] The format line **'@',RETURN** will cause a print pause and the prompt "HIT ANY KEY TO CONTINUE". To avoid an inadvertant pause, don't start a paragraph with **'@'**.

MISCELLANEOUS IMBEDDED CHARACTERS

ESC,TAB Imbeds a tab character in the text (displayed as a right pointing triangle). On printout this causes the text following it to start one space beyond the next tab column set in the most recent print format line.

SHFT-CTRL-P Produces the same character as **INV-ESC-CTRL-I** (it looks like a fat, filled-in **'P'**). This symbol is used to delimit a printer control code or graphics sequence. During printout, characters between pairs of these symbols will be transmitted to the printer, but will not be counted as printable characters in calculating line length or justification. The delimiting characters will not be sent to the printer. This can be used (in conjunction with the null character) to insert printer graphics without disturbing print formatting.

CTRL-[,] Produces the null character (displayed as a heart). This is not printed by the printer but is treated as a character by formatting routines. This is most useful for adjusting for the space used by inserted printer graphics.

INV-ESC-CTRL-R This produces the same symbol inserted by the block mark command (a white rectangle with a horizontal bar through the middle). When searching for block markers you should search for the **INV-CTRL-R** character (you don't need to press escape when defining the search string).

[|] This vertical bar is used as the soft hyphen symbol. It marks the point where a word can be hyphenated at the end of a print line.

INVERSE SPACE This character denotes a hard space. The space will appear on printout, but is treated as a normal character in determining word breaks, and justification.

[\] The backslash character marks the breakpoint for split justification.

FONT MODIFIER CHARACTERS

To enter the font modifier characters, **hold down** both **SHIFT** and **CTRL** keys, then

press **E** to insert  (**emphasize** start/stop character)

press **D** to insert  (**double**-strike start/stop character)

press **I** to insert  (**italics** start/stop character)

press **W** to insert  (double-**wide** start/stop character)

press **U** to insert (**underline** start/stop character)

press [↑] to insert  (**superscript** start/stop character)

press [↓] to insert  (**subscript** start/stop character)

Font modifiers should be used in pairs. Text between a pair of markers will be printed with the modification defined above.

NOTE: Because of varying printer capabilities, the print functions of the modifiers may differ from those listed above. (See Section 6.0.)

3.11 EDIT USE OF CONSOLE SWITCHES

The console switches are on the right side of the ATARI 800 and 800XL keyboards and at the top of the ATARI 1200XL keyboard. Their functions in the EDIT system are as follows:

OPTION Pressing this switch will display the MAIN MENU which then provides access to other system functions.

OPTION n Pressing a number key (0, .., 9) after pressing **OPTION** will insert phrase #n into the text and return to EDIT with the cursor placed just after the phrase. The phrases must be stored in a contiguous block of comment lines at the beginning of the text in memory.

SELECT Toggles between the two possible type-over cursors (underline and inverse-video).

START Toggles between two possible modes of text display: word wrap on or off. This does not affect the word wrap function on printout.

SYSTEM RESET Whenever this switch is pressed, the program halts whatever it's doing and returns control to EDIT. This does not affect text stored in memory or any of the system parameters.

OPTION+RESET If **OPTION** is pressed before **RESET**, and held down during the RESET process, control will be transferred to the disk operating system. The effect of this transfer will depend on which DOS was loaded during the autoboot process. The version of DOSXL provided with The Writer's Tool will just return control back to EDIT. **In all other cases, text in memory will be lost** when control is transferred to DOS. To re-run The Writer's Tool from DOS, just use the binary load option to run the AUTORUN.SYS file on The Writer's Tool master disk.

4.0 USING MAIN MENU FUNCTIONS

4.1 THE MAIN MENU

The MAIN MENU can be accessed from EDIT by pressing **OPTION** or entering **CTRL-O.** It can also be directly accessed from PRINT and DISKIO menus by pressing **OPTION.**

The MAIN MENU is displayed in a special text window of a different color and brightness than the main display. This window uses the last four lines of the screen. (This will be referred to as the "command window"). The MAIN MENU displays the names of the five functions: SEARCH, DISKIO, PRINT, CLEAR, and EDIT.

The first character of each function name is highlighted in inverse video representation. Pressing a letter key matching the first letter of the function name will activate the corresponding function.

To leave the MAIN MENU and return to EDIT, press **E.**

When the Main Menu is displayed, you can access the External Function Menu by pressing **X** (see page R-56 for a description of external functions).

4.2 THE SEARCH SYSTEM

Most search functions are accessed from the MAIN MENU (pressing **OPTION** or **CTRL-O** will display the main menu).

When the MAIN MENU is displayed, pressing **S** first clears the menu, then produces the prompt "ENTER /Old/New/**RETURN** or /Old/**RETURN**".

SEARCH AND REPLACE WITH VERIFY

Old string and replacement string are entered in the form '/old/new/',**RETURN.** Here '/' is the delimiter character, which can be any character not present in the strings themselves. Starting at the cursor position, the search routine will find the first occurrence of 'old', display it on the screen (with the cursor flashing at the first character), sound an alert beep, and then prompt for Replace, Skip, or Quit.

If the 'S' key is pressed, no replacement will be made and the search will continue to the next occurrence. If the 'R' key is pressed, the new text will replace the old text, and the routine will search for the next occurrence of 'old'. If the 'Q' key is pressed, the routine will abort and return to EDIT without making a replacement. If some other key besides 'R', 'S', or 'Q' is pressed, the prompt will be repeated.

The third delimiter is only needed if the last character of the 'new string' is a space. If the 'new' string is empty (no characters between 2nd and 3rd delimiters) then the replacement function will have the effect of deleting the 'old' string. If the 'old string' is not found after the cursor, then a higher pitched beep will sound, the message "NOT FOUND" will be displayed briefly in the command window, and, after a short pause, control will return to EDIT.

Pressing **RETURN** without specifying a search target will return control to EDIT.

SEARCH ONLY

If the initial prompt is answered in the form '/string/'**RETURN,** then the program will look for 'string' starting at the cursor position. If 'string' is found, the cursor will be placed on the first letter of the 'string', a low-pitched beep will sound, and control will return directly to EDIT. If 'string' cannot be found after the cursor, then a higher pitched beep will sound, the message "NOT FOUND" will appear briefly, and control will return to EDIT. In this mode of operation the 2nd delimiter is only needed if 'string' ends with a space.

CONTINUE SEARCH

Once the SEARCH function is used to define a search target (the old string), it is possible to continue searching for the same target directly from the EDIT system. Pressing **CTRL-S** will move the cursor to the next occurrence of the target string, sound the low-pitched beep, and return to EDIT. If no further occurrence is found, then the higher pitched beep will sound, and the "NOT FOUND" message will appear briefly on the status line. If no target string has been defined, this command will have no effect.

WILD CHARACTERS

The '?' character used within an 'old string' is treated as a wildcard character on search operations. For example, to replace both 'Figure' and 'figure' with 'Fig.', use '/?igure/Fig./',**RETURN.**

COMMAND WINDOW EDITING

The command window makes use of the built-in screen editor of the ATARI operating system. When this window is present, you cannot use most editing commands of The Writer's Tool. However, you can move the cursor with **CTRL-arrow** and use single-character **INSERT** and **DELETE** functions. Another difference in text entry is that the command window does not support some of the special character entry keystrokes. For example, **SHFT-CTRL-U** will be ignored, and **CTRL-M** will not insert a block marker. Most important is that **RETURN** terminates string entry and begins the search process, rather than inserting a left arrow as it would in the EDIT system.

SEARCHING FOR SPECIAL CHARACTERS

Because of the different behavior of the command window noted above, special procedures are required to search for some of the special characters used by The Writer's Tool. It is sometimes desirable to search for format lines, which begin with a period following a carriage return. To search for the RETURN - PERIOD sequence, enter '/',**ESC,CTRL-[←],**'./' as the '/old string/' part of the response (this should get displayed as **/←./**). To delete a block marker, enter '/**INV-CTRL-R**//'.

4.3 THE DISK I/O SYSTEM

When the main menu is displayed, pressing **'D'** will produce a new screen display: the main DISK SYSTEM screen. Commands available from the Disk I/O System are DIRECTORY (for drive 1 or 2), LOAD, SAVE, DELETE, INITIALIZE, and EDIT. These are briefly explained at the top of the screen, and are also named in the command window at the bottom of the screen. Commands are activated by pressing a letter key matching the letter highlighted in the command window (usually the first letter of the command). You can return to EDIT by pressing E, or to the Main Menu by pressing **OPTION.**

THE DIRECTORY COMMAND [1 2 (DIR)]

This will clear the screen and display a double-column disk directory for the selected drive. Press **'1'** for a directory of the disk in drive 1, or press **'2'** for a directory of the disk in drive 2. If the directory has more than 36 filenames, the display will stop after the first 36 entries, and the prompt "PRESS ANY KEY TO CONTINUE" will appear. Pressing any key will then cause the remaining filenames to be listed.

A PRINTED COPY of the directory can be produced by holding down the **SELECT** switch before pressing **'1'** or **'2'.** Hold down **SELECT** until printing begins.

THE LOAD COMMAND [LOAD]

The LOAD command is started from the Disk I/O System by pressing **'L'.** This is used to load text into the text memory buffer from a named disk file.

The first prompt displayed when this command is started depends on the position of the cursor. Since text files are loaded into the buffer starting at the cursor position, this routine first checks to see if the text currently in the buffer will be overwritten. If the cursor is not past the end of the current text, then the message "WARNING: Load will overwrite current text (cursor thru textend)" will be followed by the prompt "OK TO CONTINUE (Y/N)?". If you want the file loaded anyway, respond with **'Y',RETURN.** Any other response will restore the Disk System command window, at which point you can go back to EDIT and reposition the cursor.

If no warning was presented, or you ignored the warning, the next prompt will be "LOAD WHICH FILE?". To load file XXX from drive #1, you should respond by entering either 'XXX',**RETURN** or 'D1:XXX',**RETURN.** If the file is to be loaded from drive 2, then you must use the form D2:XXX. Failure to follow these conventions will

result in error responses. If you enter a filename with the wrong format, a buzzer will sound and the prompt will be redisplayed. If the form is valid, but the file doesn't exist on the specified drive, the error will not be discovered until the load is attempted.

After a filename in the proper format is entered, the prompt "ARE YOU SURE (Y/N)?" will appear. If you respond **'Y', RETURN**, the routine will try to find the specified file. If the file is found, it will be loaded into memory starting at the cursor position. When the load is completed, a beep will sound and program control will return to EDIT.

If the file cannot be found, a buzzer will sound, and an error message will be displayed briefly in the Disk System command window. If the file is found, but too large to fit into the available memory space, a warning message will be displayed in the command window.

THE SAVE COMMAND [SAVE]

This function will copy text from the memory buffer to a named diskfile. Only text from the cursor to the TEXTEND pointer will be saved. If the cursor is not at the beginning of the text buffer, the first action of this routine will be to sound a buzzer, display the message "WARNING: only part of the text buffer will be saved...", then present the prompt "OK TO CONTINUE (Y/N)?". If you really want to save only part of the text buffer, then answer **'Y', RETURN.** Otherwise, enter **'N', RETURN** or just **RETURN;** this will produce the prompt "SAVE ALL TEXT (Y/N)?" Answer with **Y,RETURN** to proceed with the save operation.

If you have not previously loaded or saved a file, the next prompt will be "SAVE TO WHICH FILE?". If you intend to save using the name ZZZ on drive #1, you can enter either **'ZZZ', RETURN** or **'D1:ZZZ', RETURN.** If you want the file saved on drive #2, you must use the form D2:ZZZ. If an invalid format is used for the filename, a buzzer will sound, and the prompt will be displayed again.

After a valid filename is entered, the prompt "ARE YOU SURE (Y/N)?" will be displayed. This gives you a chance to change your mind; this is useful if you have made an erroneous entry in typing the filename. If you answer **'Y', RETURN,** the save will be attempted. If the save fails, a buzzer will sound, and an error message will be displayed in the command window. Explanation of possible disk errors is provided in Reference Section 7.

While the save is proceeding, the message "SAVING to---Dn:fname" will be displayed in the command window. A successful save will be followed by a display of the directory of the disk on which the file was saved.

If you have previously loaded or saved a file, the the most recently used filename will also be used at the start of the save process. The message "Saving to---Dn:fname" will be displayed, followed by the prompt "OK to continue (Y/N)?". Answer 'Y',**RETURN** if the displayed name is appropriate. If you enter a negative response, you will be asked to provide a filename as described in the previous paragraphs.

THE DELETE COMMAND [DEL]

This command will delete a specified file from a disk. This function will first produce the prompt "DELETE WHICH FILE?". To delete file EEE from drive #1 you can enter either 'EEE', **RETURN** or 'D1:EEE', **RETURN.** To delete file FFF from drive 2, you must use the form 'D2:FFF'. After a valid filename is entered, the prompt "ARE YOU SURE (Y/N)?" will appear. If you answer with **'Y', RETURN,** the deletion wil be attempted. If it fails, a buzzer will sound, and an error message will be displayed in the command window.

THE INITIALIZE COMMAND [INIT]

This dual purpose command is used either to initialize a disk drive (to set density), or to initialize a diskette (prepare a blank disk for data storage or erase an old disk).

To initialize a diskette press **I** and then answer the prompt "WHICH DISK DRIVE (1 or 2)?" by entering the appropriate drive number, and pressing **RETURN.** The next display will present the options of setting density or initializing a diskette. Press **I** again, and verify your intention by answering the prompt "ARE YOU SURE (Y/N)?" (this function will completely erase the disk). If you answer **'Y',RETURN,** the message "INITIALIZING DISK" will be displayed and the process will begin. After about 40 seconds, the directory of the initialized disk will be displayed and control will return to the disk system command window. If you answered the previous prompt in any other way, the initialization operation would have aborted.

To set the drive density, press **I,** select the drive number, then press **D** to set the drive to double density or **S** to set single density. (Please read Appendix 3 before attempting to use multiple disk densities.) If the density set command is successful, the message "Density Set OK" will appear. Otherwise you may see "Can't Set Density" (your drive won't respond to the command) or "Drive not available" (your drive

may be turned off or not connected).

EXITING THE DISK I/O SYSTEM

When the normal Disk System command window is displayed, you can return control to EDIT by pressing **E**, or return to the Main Menu by pressing **OPTION**.

RULES FOR NAMING DISK FILES

Disk files can be completely identified using the form

Dn:FNAME.EXT

where **FNAME.EXT** specifies the name of the file, and **Dn:** specifies which disk drive is to be used (n stands for 1, or 2). The drive specifier is not needed to address drive number 1. **D1:FNAME.EXT** and **FNAME.EXT** are equivalent forms. **The only exception to this rule occurs in linked printing which always requires a device specifier (D:** or **Dn:**, for example).

The period separates the filename into a primary name and an extender. When you create a file, you do not need to specify an extender. If no extender is used, the period is also not needed.

Rules for defining a filename are as follows:

The maximum length of the primary name is 8 characters.

The maximum length of an extender is 3 characters.

Only upper case letters (A-Z) and numbers can be used.

The first character must be a letter.

DO NOT use the names DOS.SYS or AUTORUN.SYS. These are reserved for use by the disk operating system.

When you identify an existing file, you can use wild card characters. A ? denotes any character, and a * denotes any combination of characters. For example, **LETTER.*** identifies the first file that has a primary name of **LETTER**, while ***.BAS** identifies the first file with the extender **BAS**. The name **L????R** identifies the first file with a 6-character name beginning with **L** and ending with **R**.

4.4 THE PRINT SYSTEM

The Print System is selected from the main menu by pressing **'P'**. This produces the Print System Screen which displays a two-column listing of the print format parameters, the word count between the cursor and the end of the text buffer, and the name of the current printer. At the bottom of the screen is the Print System Menu, displaying six commands: FMAT, PRINT, LINK, MERGE, CHNGE, and EDIT.

Pressing **'P'** at this point will print all text from the cursor forward, using the format parameters shown at the top of the screen (unless these are overridden by format commands imbedded in the text).

Pressing **'E'** will return control to the EDIT system.

THE FORMAT COMMAND (FMAT)

The print process will always begin using the format parameters displayed at the top of the Print System screen. These parameters can be changed within the Print System by pressing **'F'** and responding to the prompt "ENTER FORMAT LINE". This prompt should be answered by entering an initial period followed by a sequence of letter codes and numbers, and terminated with a carriage return (**RETURN**). This is the same form as the imbedded format line described in Reference Section 5.

If you answer the "ENTER FORMAT LINE" prompt with a valid format line, the Print System screen will be redisplayed with the format parameters changed to the values specified in the format line. If you enter an incorrect format line, a buzzer will sound and the prompt will be repeated. Entering just **RETURN** will return to the main Print System Menu.

Examples of legal format lines and their functions:

.S2 m10 L40 Sets up double-spaced printing with a left margin of 10 spaces and a print line of 40 characters.

.T15,30,45j0 Sets first three tabs at 15, 30, and 45 spaces from the left most print position and turns off justification.

.N10 Sets page numbering to start at 10 instead of the default value of 1 (printing a page number requires header or footer with # symbol imbedded in text).

Complete explanations of the format command structure and the meaning of the format parameters are presented in Reference Section 5.

NOTE: Format parameters entered from the PRINT subsystem only affect the initial format used by the print formatting routine. When this routine encounters a format line imbedded in the text, the imbedded command will take precedence. At the end of any print operation most format parameters will return to their original default values. Exceptions are the wait option (.Wn) and the view-to-print transition page setting (.Vn).

THE PRINT COMMAND (PRINT)

The PRINT command is activated by pressing **'P'**. Printing will begin at the cursor position and end with the last character in the text buffer.

Before printing anything, the print routine checks for the presence of initial format commands in the text. If format lines are found before any printable text, the format commands will be interpreted first. The format of the header (if defined) and subsequent text will then be determined by the parameters (if any) present in these initial format commands. If the header or footer blocks themselves contain format lines, then their formats can be different from that of the main text.

If the printer or interface module is not turned on, or if the printer is on but not selected (on line), a buzzer will sound and the message "DEVICE DOES NOT RESPOND, **START** TO RETRY or **OPTION** to QUIT" will be displayed. After correcting the printer or interface module conditions, press **START** to try again.

Printing can be aborted after any print line by pressing **Q** (to return to the print subsystem) or **E** (to return directly to EDIT). Printing can be paused after any print line by pressing **'P'** and resumed by pressing **'P'** a second time. A message reminding you of these opportunities will be displayed in the command window during printing.

When printing is completed, most format parameters will be returned to the default values. The page number will also be reset.

THE LINKPRINT COMMAND (LINK)

An unlimited number of disk files can be continuously printed as one document by linking them together with specially formatted imbedded link names which point to the next file to be printed. To achieve this, each linkname must be part of a one-line paragraph of the form '<Dn:NEXT.PRN', **RETURN,** where n refers to the disk drive number (1 or 2), and 'NEXT.PRN' should be replaced with the name of the next file in the print sequence. (The "<" character must immediately follow a carriage return.) An additional step is needed: in the Print System 'L' should be pressed to activate "LINKPRINT" rather than "PRINT".

When LINKPRINT is started, the message "WARNING: Text now in memory may be modified by linked printing" will be followed by the prompt "OK TO CONTINUE (Y/N)?". If the buffer has already been saved, you can safely respond with **Y,RETURN** and printing will proceed. When all the text before the link line is printed, the named link file will be loaded from disk and inserted just before the remaining text. The text already printed will be erased (except for the most recently defined header and footer blocks) and printing will continue from the beginning of this new text until another link is found, or until all the text currently in memory is printed.

As each new file is linked, the message "Printing Link file xxx" will be displayed (xxx denotes the name of the file being printed). If the linked file YYY cannot be found on the designated drive, the message "Cannot find---YYY, **START** --> RETRY or **OPTION** --> QUIT" will be displayed. At this point you should insert the disk containing the file YYY, then press **START** to continue the printing process.

Linked printing can be paused by pressing **'P'**, resumed by pressing **'P'** again, or aborted by pressing **'Q'**. These are the same controls used to control normal printing.

Linkprint Example: the text

<D1:ARTH.ADR**RETURN**
RETURN
Dear Arthur,**RETURN**
<D1:XMAS.LET**RETURN**
RETURN
Sincerely,**RETURN**

will print the address stored in disk file "D1:ARTH.ADR", then print the salutation, followed by whatever text is stored in the diskfile "D1:XMAS.LET", and will finally print the closing.

THE MERGE COMMAND (MERGE)

Pressing **'M'** from the Print System will activate the Merge System. This is a sub-program which handles the creation of data base files, and the merged printing of these files with template documents.

Before the Merge System can be loaded into memory, you must respond to the prompt "WARNING: Text in Memory will be Erased, OK to CONTINUE (Y/N)?" If nothing of value is in the text buffer, then answer **'Y'RETURN.** This will produce another prompt: "Insert Master Disk in Drive 1, **START** when READY or **OPTION** to QUIT." Pressing **OPTION** will return control to the Print System with the text buffer still intact.

After inserting The Writer's Tool disk as instructed, pressing **START** will load and run the MERGE.OBJ program.

To use any Merge function, you must first identify a **template** file name. The template file contains text with data labels inserted at places where variables are to be used. (A **data label** is any contiguous string of inverse video characters.)

A template is used to match data input to the data requirements. Data input must be in units of records (a text string terminated by a carriage return). The **data source** can be keyboard input or a named disk file. The **data destination** can be the printer (as a merged document), or a disk file (a series of records forming a data base).

During data input it is possible to verify each complete group of data items before they are merge printed or added to a data base. During keyboard input the data labels are used to prompt for the desired data item in the order required. During disk input, the labels are shown next to the input data items to aid in verifying (it is possible to skip complete groups, if desired).

Whenever a filename is requested by the Merge System, pressing **RETURN** can be used to get a disk directory or to quit the Merge System. Quitting the Merge System will return control to the Print System.

Template files can only be created or edited using the EDIT System. Data base files are most easily created with the Merge System, but can be created and edited with the EDIT system (as long as the order and number of each group of data items matches the requirements of the template).

Tutorial Section 9 provides detailed examples of how to use the Merge System.

THE CHANGE COMMAND (CHNGE)

This command is used to install a new printer driver or a custom file defining a new set of default formats. Pressing **'C'** will produce the prompt "INSERT DISK WITH CUSTOM FILE IN D1, **START** WHEN READY or **OPTION** to QUIT. Pressing **OPTION** at this point will return to the Print System menu. Pressing **START** will display the directory of the disk in Drive 1, then ask "LOAD WHICH FILE?". There are two kinds of files which can be loaded: (1) Printer Data files (one of the files with a ".PPP" extender on The Writer's Tool disk), or (2) a custom format file written by the customizer program (these files have ".FFF" extenders). Entering an invalid filename will produce the message "BAD FILE NAME", and return to the Print System Menu.

After loading a printer data file or custom format file, program control will return to the Print System, with a new printer name or new default format parameters displayed, depending on which type of file is loaded.

PREVIEW PRINTING TO THE SCREEN

Before printing text on paper, you may want to check the position of page breaks, and verify that the format commands imbedded in the text are valid. This can be done by previewing text on the screen.

To direct print output to the screen, set the **'V'** format parameter to a value greater than the number of the last page to be printed. After this is done, executing either "PRINT" or "LINKPRINT" will display the first 38 characters of each print line. The number of leading blanks on each line will equal the difference between the left margin and the first tab column.

The screen preview scrolls rather rapidly. You can pause the preview by pressing **'P'** and resume the preview by pressing **'P'** a second time. The preview can be aborted by pressing either **'Q'** (to return to the Print System) or **'E'** (to return to EDIT with the cursor positioned at the beginning of the last line previewed.

The preview does not always display the correct top and bottom margins unless you have the GENERIC printer selected. Fractional line feeds, proportional spacing, superscripts and subscripts, font modifications, and other special print format functions are also not shown as they would appear on paper. In spite of these limitations, you will find the screen preview a useful and paper-saving function.

If the **'W'** format parameter is set to 1 (the single sheet option), the preview will scroll through one page at a time, waiting for **START** after each page before proceeding with the next page.

PRINTING FROM THE MIDDLE OF A DOCUMENT

Printing starts from wherever the cursor is positioned. This allows printing from any point in the text. However, this may not produce the desired print format. Proper page breaks and format changes cannot be accounted for unless the print routines have read through the document from the beginning.

To print a properly formatted page in the middle of the document, set the cursor to the beginning of the text, go to the print system, set the **'V'** format parameter to the number of the first page you want printed, then execute "PRINT". All formatted output prior to this page will be directed to the screen, and all formatted output from the selected page forward will be sent to the printer.

MIXING GRAPHIC IMAGES WITH TEXT

This is possible during a LINKPRINT operation. If a linked filename has the extender ".GGG", all but the first byte of the file will be transmitted directly to the printer without modification or formatting. The first byte is used to tell The Writer's Tool how many half-line feeds are used in printing the graphic image. The remaining bytes in the file must be whatever codes your printer needs to print the image, and must be created by a special-purpose program, not currently provided with The Writer's Tool.

4.5 THE CLEAR SYSTEM

The CLEAR system can be activated from the Main Menu by pressing **C.** It's function is rapid clearing of large amounts of text from the text buffer.

This command first produces the prompt "CLEAR BEFORE OR AFTER CURSOR (B/A)?". Here "AFTER" actually means "at and after". Respond by answering **'B',RETURN** or **'A',RETURN.** This will produce the prompt "ARE YOU SURE (Y/N)?". If you answer **'Y',RETURN,** then the appropriate part of the text will be cleared. After the clear is finished, control will return to EDIT. Otherwise, clear will be aborted and control will return to EDIT.

5.0 GUIDE TO PRINT FORMAT

This section describes how to control the appearance of printed text using The Writer's Tool format commands. These commands are only needed if you want to deviate from the default format.

Format control is achieved through two primary mechanisms: **format lines** (usually imbedded in the text), which affect the appearance of at least one line of printout, and imbedded **format characters**, which may affect as little as one character in a line of printout. These two methods can be combined to provide an almost unlimited variety of printed formats.

5.1 THE STRUCTURE OF THE PRINTED PAGE

Within each page, The Writer's Tool recognizes three sub-units, each of which may have its own unique format. These sub-units are (1) the header block, (2) the main text block, and (3) the footer block. The header and footer blocks generally contain the same text on each page (except for the page number), while the content of the main text block changes from page to page.

VERTICAL PAGE FORMAT

The Writer's Tool begins its page wherever the print head happens to be when the print command is issued. This may or may not be at the top edge of the paper. It all depends on how you adjusted the paper and how your printer works (some printers can't print at the top of the paper). Thus, The Writer's Tool page may be vertically offset from the 'physical' page (the boundaries of the sheets of paper, or the perforations). As long as you are aware of this offset, it shouldn't present any problems. Just remember to start printing with the paper adjusted to a convenient and consistent position.

If a header has been defined, the first thing printed will be the header block. If a header has not been defined, the main text block will be printed first.

The number of print lines used by the header block is determined by the header block itself.

When the header is done printing, The Writer's Tool will begin printing the main text block. This will continue until the footer line is reached (this is line 56 in the default format). At this point, if a footer has been defined, The Writer's Tool will begin printing a footer. After the footer is printed, the paper will be advanced to the beginning of the next page. If the footer is longer than the space between the footer line and the beginning of the next page, then only part of the footer

will be printed. If no footer is defined, the footer line will be blank, and the paper will be advanced to the beginning of the next page.

The most important factors controlling the vertical page format are these:

(1) the starting position of the print head;

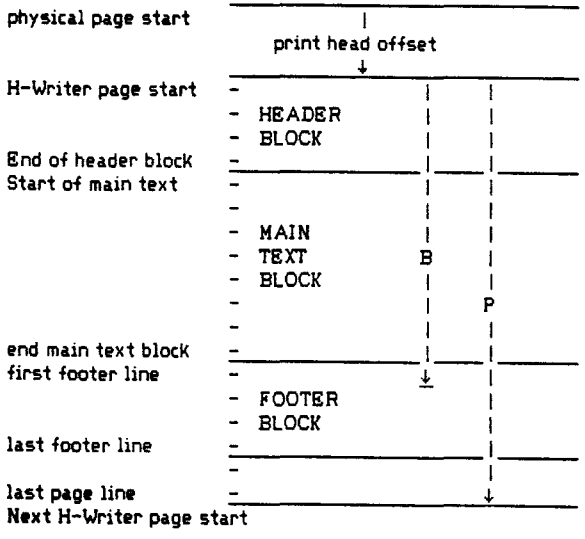
(2) the number of lines used by the header;

(3) the starting line of the footer (the value of the **B** format parameter);

(4) the page length (the value of the **P** format parameter); and

(5) the line spacing used within each sub-unit (the value of the **S** format parameter set within each block).

DIAGRAM OF VERTICAL PAGE FORMAT



To produce blank lines between the header text and the first line of the main text, these blank lines must be specified within the header block. In similar fashion, initial blank lines in the footer block will produce a margin between the end of the main text and the footer text.

NOTE: If you start with a large vertical offset of the print head, you should make sure that the footer block is short enough (or starts high enough) to avoid printing across the end of the physical page.

HORIZONTAL PAGE FORMAT

The left edge of The Writer's Tool page is determined by the leftmost position of the print head (for most printers this is usually 1/4 inch from the left edge of the physical page). This leftmost print position corresponds to a left margin of zero (set by the M format parameter). This is also the point from which tab columns are measured. The main factors controlling the horizontal format are the following:

(1) The left margin setting (M) which is the number of spaces before the first printed character in a line.

(2) The line length (set by the L parameter) which is the number of spaces beyond the left margin which are alloted for printed characters.

(3) The indent (set by the I format parameter). This is a relative shift of the left margin from the margin set by the **M** parameter. It does not affect the right margin.

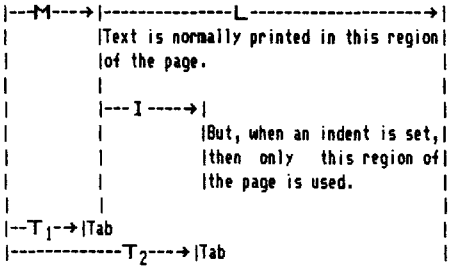
(4) The font setting (determined by the **F** format parameter). Since different fonts take up different horizontal space, the physical length of the margin, print line, and indent can vary considerably even when their numeric values expressed in spaces is fixed.

(5) Tab settings and imbedded tab characters. Tab columns are measured in spaces from the leftmost print position. When a tab character is encountered in the text, the print head will be moved to the next tab column before printing the next character.

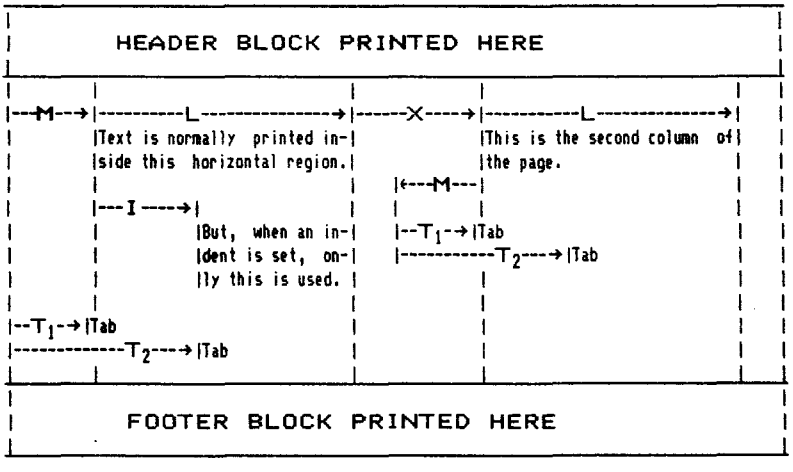
(6) Double-column spacing (controlled by the **X** format parameter). If the double-column spacing is set to zero, printout will be in single column mode. If the spacing is non-zero, two columns will be printed, each column having the same line length set by the **L** parameter. Thus, double column printing requires smaller line lengths than single column printing.

The factors controlling horizontal page format are illustrated in the following two diagrams.

FORMAT DIAGRAM FOR SINGLE-COLUMN PRINTING



FORMAT DIAGRAM FOR DOUBLE-COLUMN PRINTING



5.2 FORMAT COMMANDS

Two kinds of commands are used: **format lines** and **special characters.** Format lines are usually imbedded within the text, but may also be entered from the Print System. Imbedded commands always take precedence over entered commands.

FORMAT LINE COMMAND STRUCTURE

An imbedded format command line is a separate paragraph preceded by a carriage return and terminated by a carriage return. The first character of the format line must be a period. The rest of the line consists of letters, spaces, and one to three digit numbers (and commas in the case of the tab command). Either upper or lower case letters can be used, and spaces are ignored.

EXAMPLE: '.M8L55 p66 **RETURN**' sets the left margin to 8 spaces, line length to 55 characters, and page length to 66 single-spaced lines.

Format lines entered from the Print System should have the same form as imbedded format lines.

Unless noted otherwise, the following format commands can be combined into one format line (using only one period at the beginning of the line).

COMMANDS AFFECTING VERTICAL FORMAT

.Pnnn Sets page length to nnn (the number of single-spaced lines per page) The default value is 66, corresponding to an 11-inch page with six single-spaced lines per inch. The page length should always be less than 127.

.Sn Sets line spacing parameter to n (n=1 for single spacing, n=2 for double spacing, n=3 for 1.5 spaces per line, n=4 for half-spacing). The default value is n=1.

.Bnnn Sets beginning line of the footer block to the nnnth line of the page, where nnn is a number less than 127. The default value is 56. If you try to set B to a value greater than P, The Writer's Tool will force B to equal P.

.Gnn Groups next nn print lines on a single page. This must be isolated on a separate format line. On printout, the text following this command will be printed on the next page, unless there are nn single-spaced lines left on the current page.

.E Page Eject (use by itself on a separate line). This advances the paper to the footer line, prints the footer, then advances the paper to the beginning of the following page.

COMMANDS AFFECTING HORIZONTAL FORMAT

.Mnn Sets left margin to nn spaces (number of blank spaces between the left edge of the page and the first printed character of normal print line). The size of the margin in inches depends on the character font selected.

.Inn Sets the indent to nn spaces beyond the left margin. Text following this command will start printing nn spaces to the right of the left margin, and will have print lines which are nn spaces shorter than the normal line length. This command is not used for indenting the first line of a paragraph; it is used to indent the whole paragraph. Indenting the first line of a paragraph can be done by entering leading spaces or by using an imbedded tab as the first character of the paragraph. In this case, the first line will start one space after the column set by the first tab. The indent **I** must always be less than the line length **L.**

.Lnnn Sets the length of the print line to nnn spaces. In fonts 1, 2, and 3, there will be nnn characters printed on a standard line. In the proportional fonts (usually 4 and 5), there will generally be considerably fewer than nnn characters, since most characters are wider than spaces, but the physical length will stay constant for a given font, regardless of the variation in character width.

.Jn Sets right justification control to n (n=0 turns off justification but retains word wrap, n=1 produces right justification, n=2 turns off word wrap, and n=3 produces microspaced justification for some printers).

.Tnn,mm Sets first two tabs at nn and mm spaces from the left-most print position. Up to eight tabs can be set, with a maximum column of 240. When a tab character is encountered during printout, the printer will space to the next tab column before printing the character following the tab symbol.

.Xnn Sets spacing between first and second columns to nn spaces. If nn is zero (the default value), only single-column printing will be performed. If nn is non-zero, double-column printing will be activated (in this case L should be readjusted so both columns will fit on the page).

.C Centers next text line. This must be used as a separate format line immediately preceding the text to be centered. The total length of the paragraph following the center command must not exceed the line length defined by L. The text will be centered between the left margin (set by **Mnn**) and the right margin (determined by M+L), without regard to the indent setting (I).

.A Alternate command. When placed above a split-justified line, the left and right parts of that line will be reversed on even pages.

OTHER FORMAT COMMANDS

.D Sets print format parameters to default values. This must be used as a separate format line without any numbers. This command does not reset the page number (N), the view-to-print transition (V), nor the wait option (W).

.Fn Sets font parameter to n. Five fonts are supported. Although **these vary from printer to printer,** the most common assignments are these:

1= Pica (standard 10 characters/inch)
2= Elite (12 characters/inch)
3= Compressed (this varies between 16.7 and 17 CPI, depending on the printer)
4= Proportional Characters (at 22.86 spaces per inch)
5= Proportional Characters (at 20 spaces per inch)

Reference Section 6 describes the specific font assignments for each of the supported printers. If a font is selected for a printer which cannot support it, the font setting will be ignored during printout.

.Nnnn Sets page numbers to start with nnn on first page printed. This parameter has a default value of 1 and is incremented by one after each page is completed. The page number is inserted within header or footer blocks wherever a '#' symbol appears. The **.Nnnn** command itself should only be used in the main text. If placed within a header or footer block it will be ignored. If N is set to zero, it will not be incremented after every page.

.Vnn Sets the view-to-print transition page. Pages with numbers less than nn will be printed to the screen. Pages with numbers greater than or equal to nn will be printed on the printer.

.Wn Sets wait option (1=stop at end of each page, 0=no stop). This is useful while printing single sheets in a printer with friction drive. When a page is completed, the prompt "PAGE DONE, PRESS START TO CONTINUE" will appear. After you have positioned the new page in the printer, press START to print the next page. The default setting is 0 (no waiting at page ends). If W is set to one while double-column printing, then the program will wait at the end of each column, even if your printer has reverse line-feed capability and thus wouldn't need to wait.

@ This character followed by a **RETURN** constitutes the only format line that does not use a leading period. It's function is to temporarily stop printout in the middle of a page. This is primarily useful for letter quality printers: it allows changing fonts by changing daisy wheels. Once printing is paused, you are prompted to press any key to resume printing.

IMBEDDED FORMAT CHARACTERS

In addition to the font modifier characters described on Page R-15, there are five other characters which have special formatting functions:

\ (BACKSLASH) Split justification character. When this character is placed within a paragraph shorter than a print line, text to the left of the character will be justified against the left margin, while text to the right of the character will be justified against the right margin. This is especially useful in header and footer blocks.

▶ (TAB CHARACTER) When this is imbedded in the text, the print head will be advanced to the next tab column before the text following this character is printed. If the print head is already past the last tab column a single space will be inserted.

(NUMBER SYMBOL) Has special significance only in headers and footers, where it will be replaced by the current page number on printout.

! (EXCLAMATION POINT) Comment Character. When this is the first character of a paragraph, the paragraph will be interpreted as a comment and will not be printed. Comment paragraphs should not exceed 250 characters.

| (VERTICAL BAR) Soft Hyphen. This character marks possible break points where hyphens can be inserted in long words. Hyphenation will only occur if the word can be split between two print lines. To avoid double hyphens when used with a hard hyphen, it should be inserted **after** a hard hyphen.

▮ (INVERSE-VIDEO SPACE) Hard space character. This character is printed as a space but will not be treated as a space during word wrap or justification functions. It is used to prevent constructs like 'Figure 5', or '10 acres' from being split between lines.

FONT MODIFIERS

Special characters called "FONT MODIFIERS" can be used in pairs to start and stop underlining, emphasized print, double-strike print, italics, super and subscripts,and double width print. Key-ins needed to insert these characters are listed on page R-15. The meaning of some of these modifiers vary from printer to printer (see Reference Section 6).

HEADER AND FOOTER MARKER LINES

Header and footer text must be imbedded within the text file and identified by special marker lines:

:H <RET> Should be placed just before the header text block.

:F <RET> Should be placed just before the footer text block.

: <RET> Should be placed just after a header or footer block.

If beginning and end markers are not used in pairs, a "FORMAT ERROR" message will be displayed when you try to print the file.

A simple header, with appropriate marker lines, can be inserted using the EDIT command **SHFT-CTRL-H.** This produces the header block

```
:H←
TITLE\Page #←
←
←
:←
```

which will print a left-justified TITLE and a right-justified page number on the first line, followed by two blank lines before the main text block is printed.

6.0 GUIDE TO PRINTERS

This section describes important factors to be aware of in setting up your printer, describes the special capabilities of the supported printers, and describes how to control unsupported printers. You should note that some format parameters have different meanings for different printers.

6.1 GENERAL GUIDELINES

Many printers have internal switches which provide special operating characteristics. These are described in the owner's manual for each printer. The most important options to watch for are discussed below.

LINE FEED AFTER CARRIAGE RETURN

Some printers allow two options in their response to a carriage return code: one which executes a line feed, and one without a line feed. Make sure that the line feed option is selected, or all printed text may appear on the same line!

AUTOMATIC PERFORATION SKIPOVER

This option, available on some MX-80's, is a great convenience for making listings of BASIC programs but is not needed by The Writer's Tool, since it controls margins independently. If you have an unsupported printer which does automatic perforation skip-over, it should be turned off (if possible) since it will add extra line feeds at the end of a page and thus interfere with the page formatting performed by The Writer's Tool. Alternatively, you could set the page length parameter to compensate for these extra line feeds.

If you have an MX-80 with perforation skipover, you should leave this function turned on, since The Writer's Tool has built in software to control this function as needed. This is a convenience because you won't have to change the switch settings when you change from word-processing to BASIC programming.

PAPER OUT SENSING

Printers with friction drive can be used with single sheets (using The Writer's Tool wait option makes single-sheet printing especially convenient). However, some of these printers have a paper-out sensor which activates before the end of the page is reached. In this case, successful printing of single sheets will require disabling the sensor. In some cases this can be done by sending a control code to the printer (consult your owner's manual). **[You don't need to worry about this if you have an MX-80, since The Writer's Tool has built in software control of this function.]** In other cases, it will be necessary to physically disable the sensor by some means (usually wedging a piece of cardboard to hold a microswitch closed will do the job). Before you attempt any fix, try single-sheet printing and see what happens.

SELECT AT TURN-ON

Some printers have a front panel SELECT button. When the printer is SELECTed, it is "on-line" and ready to receive data from your computer. When the printer is de-SELECTed, it is "off-line" and cannot receive print data, but will respond to other front panel switches (line feed or "LF", and top-of-form or "TOF"). Printers with a front panel SELECT switch usually have an internal switch determining whether the printer will come up in the SELECT or de-SELECT state when the printer is first turned on. You will find it a great convenience to set this switch so that your printer will come up selected.

All of the supported printers can also be used in the GENERIC mode, although this eliminates support of the special features. [If you use your printer in two different modes, turn it off and on to reset it between modes.]

6.2 ATARI PRINTERS

In the following tables **CPI** denotes characters per inch, **SPI** denotes spaces per inch, and **LPI** denotes lines per inch.

AVAILABLE FONTS AND LINE SPACINGS

	ATARI 825	ATARI 1025	ATARI 1027
F1	Pica (10CPI)	Pica (10CPI)	Elite (12CPI)
F2	--	--	--
F3	Compressed (16.7CPI)	Comp. (16.7CPI)	--
F4	Proportional (21.4SPI)	WIDE (5CPI)	--
F5	Spaced Prop.(18.75SPI)	--	--
S1	Single SP. (6LPI)	Single (6LPI)	Single (6LPI)
S2	Double Sp. (3LPI)	Double (3LPI)	Double (3LPI)
S3	1.5 Spaced (4LPI)	--	--
S4	1/2 Spaced (12LPI)	--	--

AVAILABLE FONT MODIFICATIONS

	ATARI 825	ATARI 1025	ATARI 1027
SHFT-CTRL-E	--	--	--
SHFT-CTRL-D	--	--	--
SHFT-CTRL-I	--	--	International Chars.
SHFT-CTRL-W	Double Wide	--	--
SHFT-CTRL-U	Underline	Underline	Underline
SHFT-CTRL-↑	Superscript	--	--
SHFT-CTRL-↓	Subscript	--	--

PRINTER FILES TO USE WITH ATARI PRINTERS

Printer	Printer Data File Name
ATARI 825	AT825.PPP
ATARI 1025	AT1025.PPP
ATARI 1027	AT1027.PPP

ADDITIONAL ATARI NOTES

Micro-spaced justification (**.J3**) cannot be used.

6.3 EPSON AND GEMINI PRINTERS

AVAILABLE FONTS AND LINE SPACINGS

	MX-80	MX-80 w/G+	GEMINI 10X FX-80, RX-80
F1	Pica (10CPI)	Pica (10CPI)	Pica (10CPI)
F2	--	--	Elite (12 CPI)
F3	Cond. (16.5CPI)	Cond. (16.5CPI)	Cond. (16.5CPI)
F4	--	--	Prop. (20SPI) [only FX]
F5	--	--	--
S1	Single Sp. (6LPI)	Single (6LPI)	Single (6LPI)
S2	Double Sp. (3LPI)	Double (3LPI)	Double (3LPI)
S3	1.5 Sp. (4LPI)	1.5 Sp. (4LPI)	1.5 Sp. (4LPI)
S4	1/2 Sp. (12LPI)	1/2 Sp. (12LPI)	1/2 Sp. (12LPI)

AVAILABLE FONT MODIFICATIONS

	MX-80	MX-80 w/G+	GEMINI 10X FX-80, RX-80
SHFT-CTRL-E	Emphasized	Emphasized	Emphasized
SHFT-CTRL-D	--	Double strike	Double strike
SHFT-CTRL-I	--	Italics	Italics
SHFT-CTRL-W	Double Wide	Double Wide	Double Wide
SHFT-CTRL-U	Underline	Underline	Underline
SHFT-CTRL-↑	Superscript	Superscript	Superscript
SHFT-CTRL-↓	Subscript	Subscript	Subscript

Although the MX-80 with Graftrax-Plus supports all of the font modifications listed, there are some restrictions on their use. For example, compressed and elite fonts cannot be printed with an emphasized modifier (although they can be printed with the double-strike modification).

PRINTER FILES TO USE WITH EPSON OR GEMINI PRINTERS

Printer	Printer Data File Name(s)
EPSON MX-80	MX80.PPP
FX-80	FX80.PPP
	FX80M.PPP*
MX-80 w/G+	MX80G.PPP
	MX80GS.PPP**
RX-80	RX80.PPP
GEMINI 10X	GEM10X.PPP

* This data file has two major changes (relative to the FX80.PPP file) which may be useful for handling equations and special symbols: (1) unlimited levels of superscripts and subscripts are supported at the expense of changing the interpretation of imbedded arrows (in this case an up arrow starts a superscript, but a down arrow stops it); (2) the ROM character set of the FX80 is not copied into the RAM character set (thus any special characters which you have down-loaded to the printer will not be erased by The Writer's Tool when you select font 4 (however, this makes you responsible for putting characters into the FX80 RAM character set, without which font 4 will print as blank characters).

** Uses half height characters for super and subscripts and prints faster than the method used with the other two drivers.

ADDITIONAL EPSON AND GEMINI NOTES

Micro-spaced justification (**.J3**) cannot be used.

The FX-80 always prints proportional characters with an emphasized modification and will not respond to a backspace (CTRL-H) in this font.

6.4 PROWRITER AND NEC 8023 PRINTERS

AVAILABLE FONTS AND LINE SPACINGS

	PROWRITER 8510 or NEC 8023
F1	Pica (10CPI)
F2	Elite (12CPI)
F3	Condensed (16.7CPI)
F4	Proportional (22.86SPI)
F5	Spaced Proportional (20SPI)
S1	Single Spaced (6LPI)
S2	Double Spaced (3LPI)
S3	1.5 Spaced (4LPI)
S4	1/2 Spaced (12LPI)

AVAILABLE FONT MODIFICATIONS

	PROWRITER 8510 or NEC 8023
SHFT-CTRL-E	Emphasized (Bold)
SHFT-CTRL-D	--
SHFT-CTRL-I	--
SHFT-CTRL-W	Double Wide
SHFT-CTRL-U	Underline
SHFT-CTRL-↑	Superscript
SHFT-CTRL-↓	Subscript

PRINTER FILES TO USE WITH PROWRITER OR NEC PRINTERS

Printer	Printer Data File Name
PROWRITER	PROWRTR.PPP
NEC 8023	NEC8023.PPP

6.5 OKIDATA PRINTERS

AVAILABLE FONTS AND LINE SPACINGS

	Micro-Line 82A	**Micro-Line 92**
F1	Pica (10CPI)	Pica (10CPI)
F2	--	Elite (12 CPI)
F3	Condensed (16.5CPI)	Condensed (16.5CPI))
F4	WIDE (5CPI)	Corresp. Qual. Pica (10 CPI)
F5	--	Corresp. Qual. Elite (12 CPI)
S1	Single Sp. (6LPI)	Single Sp. (6LPI)
S2	Double Sp. (3LPI)	Double Sp. (3LPI)
S3	--	1.5 Sp. (4LPI)
S4	--	1/2 Spaced (12LPI)

AVAILABLE FONT MODIFICATIONS

	ML-82A	**ML-92**
SHFT-CTRL-E	--	Emphasized
SHFT-CTRL-D	--	Double-strike
SHFT-CTRL-I	--	--
SHFT-CTRL-W	--	Double Wide
SHFT-CTRL-U	--	Underline
SHFT-CTRL-↑	--	Superscript
SHFT-CTRL-↓	--	Subscript

NOTE: The ML-92 does not allow emphasized or double-strike printing of correspondence quality fonts (4 and 5).

PRINTER DATA FILES TO USE WITH OKIDATA PRINTERS

Printer	Printer Data File Name
MicroLine 82A	ML82A.PPP
MicroLine 92	ML92.PPP

MICROSPACED JUSTIFICATION

The correspondence quality fonts (4 and 5) of the ML-92 can take advantage of the microspaced justification option (.J3). See Tutorial Section 6.10 for an example.

6.6 THE COMREX CR-II PRINTER

The Writer's Tool supports most functions of the COMREX CR-II Daisy-Wheel Printer, including four character pitches, two-color printing, proportional and microspaced justification, and automatic sheet feeder operation.

The switching of fonts on daisy-wheel printers may involve changes of character pitch and/or a change in print wheels. Changing print wheels can be done after a pause command line (**@RETURN**), or within a line, using **SHFT-CTRL-W** before and after the text to be printed with a different wheel. When you change a print wheel, make sure that you lift the entire printer cover, not just the clear plastic part; this will insure proper wheel alignment when printing resumes. Since lifting the cover will always leave the printer deSELECTED, you will have to press the printer's SELECT button after closing the cover.

AVAILABLE PITCHES AND LINE SPACINGS

	COMRITER CR-II
F1	Pica (10CPI)
F2	Elite (12CPI)
F3	Condensed (15CPI)
F4	Proportional (20SPI)
F5	--
S1	Single Spaced (6LPI)
S2	Double Spaced (3LPI)
S3	1.5 Spaced (4LPI)
S4	1/2 Spaced (12LPI)

NOTE: Do not use a zero margin with the proportional font. It is not possible to align proportional characters against a zero left margin. If you try it, the first character of each line will be dropped.

AVAILABLE FONT MODIFICATIONS

	COMRITER CR-II
SHFT-CTRL-E	Emphasized (Shadow Print)
SHFT-CTRL-D	Double Strike (without moving print head)
SHFT-CTRL-I	Red Print
SHFT-CTRL-W	Pause for wheel change
SHFT-CTRL-U	Underline
SHFT-CTRL-↑	Superscript
SHFT-CTRL-↓	Subscript

The modifiers supported by the CR-II printer can be used with any font and in any combination.

As with all font modifiers, the wheel change pause is an on/off toggle and should be used in pairs. If you insert only one wheel change pause, a pause will occur at the beginning of every succeeding print line.

Each wheel change pause character will cause a DESELECT code to be sent to the printer. Using a format line pause (@**RETURN**) will not deselect the printer. In either case, after changing the wheel you will need to press the printer SELECT button to resume printing.

There is one problem with wheel changes within a line: you may only have about 30 seconds to do the change. This happens because the ATARI operating system will not wait indefinitely for the printer to come back to the SELECTED state. This problem can be avoided by inserting a format line pause (@**RETURN**) at the end of the paragraph within which the change is to be made (provided that the paragraph is shorter than 500 words). If you intend to print an entire paragraph with a different wheel, it is most convenient to use the @**RETURN** pause just before and after the paragraph; this method will give you all the time you want for wheel changes.

PRINTER FILES TO USE WITH THE CR-II PRINTER

Printer	Printer Data File Name
COMREX CR-II	CRII.PPP

CUT SHEET FEEDER OPERATION

To obtain automatic sheet feeding, you must use the default page length of 66 single spaced lines per page. Also remember to keep the acoustic cover in place; otherwise finished pages won't be pulled into the tray.

PAPER ALIGNMENT

The print head starting position is software-controlled to match the paper alignment forced by the cut sheet feeder. If you aren't using a sheet feeder, you should insert paper so that the left edge of the paper is 1/2 inch to the right of the zero mark on the paper bar (the left edge should line up with the 5th tic mark and the right edge should line up with the 90th tic mark). This alignment should also be used with a tractor drive.

6.7 USING OTHER PRINTERS

Many printers not mentioned in the previous subsections are similar to those that were mentioned:

CENTRONICS 737 This printer is supposed to be functionally equivalent to the ATARI 825 printer and should therefore be used in the AT825 mode.

OTHER EPSON COMPATIBLE PRINTERS The Mannesman Tally Spirit 80 is one of several printers which claim to be compatible with EPSON printer control codes. These printers should work with one of the MX, RX, or FX data files.

OTHER LETTER QUALITY PRINTERS Many parallel-interface daisy-wheel printers use control codes similar to those of the Comrex, and may be directly useable with the CRII printer data file.

If you have problems operating your printer using on of the Printer Data Files provided with The Writer's Tool, then you may need to edit one of the provided files, or create your own customized data file, using the Printer Data File Editor described in Section 9.4 of the Reference Guide.

6.8 USING PRINTER CONTROL CODES

If your printer has a function which is not supported by The Writer's Tool, and you don't want to create a specialized printer data file, you can insert control codes directly into the text.

PROTECTING THE CONTROL CODES

The recommended procedure for sending direct commands to your printer is as follows: (1) look up in your printer manual the decimal values of the codes you want to send; (2) look up in Appendix 1 the keystrokes needed to insert those codes in the text; and (3) surround the control codes with special marker characters produced by entering **SHIFT-CTRL-P.** The marker characters serve to fence off the code sequence and protect them from being treated as normal words by The Writer's Tool print formatting routines. Whatever is between these markers will be sent directly to the printer without modification. If these codes are not marked, The Writer's Tool may interpret them as font modifiers, may insert spacing commands between the control codes, and may erroneously allocate space for them on the print line, all undesirable effects.

USING THE NULL CHARACTER

The null character has a numeric value of zero, is produced by entering **CTRL-[,]** and is displayed as a heart. Printers usually ignore the null character completely; sending a null character to a printer usually is the same as not sending anything. The Writer's Tool, on the other hand, treats the null character as just another printable character. This difference in interpretation can be used to advantage.

When a protected control code sequence actually does use some space on the print line, The Writer's Tool will not properly account for it, since it ignores such protected sequences. To force The Writer's Tool to allocate space, you can just insert one or more null characters next to, but not between, the protected codes. The Writer's Tool will think that the null characters are using space while it is actually the control sequence which is using it. The net effect is an agreement between program and printer on how much space is needed.

A GRAPHIC EXAMPLE

Printer graphics can be inserted into printed text without greatly disturbing print formatting, provided proper use is made of the protection markers (**SHIFT-CTRL-P**), and the null character (**CTRL-[,]**). The following example shows how to create and print a special graphic symbol.

First, a bit map of the desired graphic symbol must be defined. The table below presents the bit map for the copyright symbol:

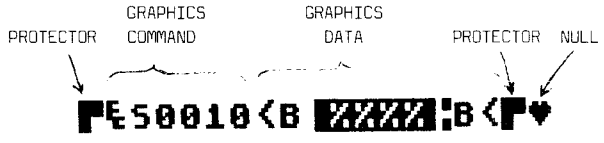
Pin #		Bit value
7	oo**1111111**oo	128
6	o**l**oooooo**l**o	64
5	**l**oo**1111**oo**1**	32
4	**l**o**l**oooooo**1**	16
3	**l**o**l**oooooo**1**	8
2	**l**oo**1111**oo**1**	4
1	o**l**oooooo**l**o	2
0	oo**1111111**oo	1
	12345678910	

In this diagram, the '**1**' symbols indicate dots to be printed (pins to be 'fired'). The ten-byte sequence of numbers corresponding to this bit map is 60, 66, 153, 165, 165, 165, 165, 129, 66, 60. The first byte is the sum of 4+8+16+32, corresponding to the binary number 00111100, which

results in the firing of print-head pins 2,3,4, and 5, needed to print the first column of the graphics symbol.

To create this symbol on the NEC 8023 or PROWRITER requires the following sequence of keystrokes: ESC, **ESC**, **'S'**, **'0'**, **'0'**, **'1'**, **'0'**, **'<'**, **'B'**, INVERSE **CTRL-Y**, INVERSE **'%'** (four times), INVERSE **CTRL-A**, **'B'**, **'<'**. This sequence should be surrounded by **SHFT-CTRL-P** symbols to keep the print formatting routine from treating it as a sequence of 16 printable characters. Since the graphics character will actually take about the space of one normal character, the last thing to do is insert a **CTRL-[,]** adjacent to the protected graphics sequence to tell the print formatter to count just one space.

The above keystrokes should produce the following screen display:



The six characters following the first protector correspond to the codes which tell the printer to print ten graphics columns. The next ten characters represent the number codes of the ten columns of the bit map.

When this graphics code sequence is mixed with normal text, and printed on a PROWRITER, the result can look like this:

```
This is a demonstration of mixed text
and graphics. The '©' symbol is supposed
to be used on copyrighted material. But
'©' is not usually available on
printers.
```

NOTE: More permanent, and easier to use, graphic character definitions can be inserted into a translate table within a printer data file, as described in Section 9.4.

7.0 GUIDE TO ERROR MESSAGES

This section on errors is arranged for trouble shooting. The error messages and warnings are grouped according to which function was active when the error occurred.

7.1 ERROR MESSAGES DURING EDITING

"CURSOR PAST END OF TEXT" (accompanied by a buzzer). This message is displayed when you try to activate commands which do not operate when the cursor is past the end of the entered text. These commands are page forward, insert a block marker, copy a block, and convert between upper and lower case.

"CURSOR ERROR" (accompanied by a buzzer) This error signal is produced during block copy if the cursor is between the block markers. (A block cannot be copied to a location within itself.) It is also produced if you try to type past the end of the text memory buffer.

"MARKER ERROR" (accompanied by a buzzer) This error can occur during block copy or block delete operations if there are less than two block markers imbedded in the text.

"OUT OF MEMORY" (accompanied by a buzzer) This message can result when an attempt is made to insert an amount of text which exceeds the available memory space. This might happen during block copy, during text entry in the INSERT mode, or in the execution of **SHIFT-INSERT** or **CTRL-INSERT.** When this happens it is time to consider splitting the text file into two smaller files. The simplest procedure is to move the cursor to a natural breakpoint (say the start of a major section), then save all text after that point using a new filename. Once this is done the text just saved can be CLEARED from the text buffer and the remaining text saved under the old file name.

7.2 ERROR MESSAGES DURING DISK OPERATIONS

"BAD FILE NAME" The file name may have lower case letters or other illegal characters. It should start with upper case letters and have only upper case letters and numbers (except for the period separating primary and secondary names).

"DATA FRAME ERROR" The diskette may be faulty.

"DEVICE MALFUNCTION" The disk drive may be in need of repair.

"DEVICE NOT RESPONDING" The disk drive may have been turned off or disconnected.

"DIRECTORY FULL" You can have a maximum of 64 different files on a disk.

"DISK ERROR" The disk may have a write protect tab, or the disk directory may be scrambled.

"DISK FULL" There is no room on the disk to save the file.

"FILE LOCKED" You cannot write over or erase a locked disk file.

"FILE NOT FOUND" This means that the disk drive specified does not have the file specified. You may have used the wrong drive number or forgotten to insert the right disk.

"WARNING: Entire file not loaded" (accompanied by a buzzer). This will appear if the memory space between the cursor and the end of the memory buffer is not large enough to hold the file you attempted to load.

"WARNING: Load will overwrite current text (from cursor thru txtend)" (accompanied by a high-pitched beep). This will appear if you attempt a LOAD operation which might destroy part of the text currently in the text buffer. If you want to add the loaded file to the current text, move the cursor one character past the end of the current text before proceeding. Otherwise, just ignore the warning.

"WARNING: Only part of text buffer will be saved (cursor thru txtend)" (accompanied by a high-pitched beep). This may occur during a SAVE operation if the cursor is not at the beginning of the text buffer. This may not be an error if you really intend to save just part of the file.

7.3 ERROR MESSAGES DURING PRINTOUT

"BAD FORMAT LINE" (accompanied by a buzzer) If a bad imbedded format line is encountered during printout, a buzzer is sounded, control returns to the EDIT mode, the error message is displayed on the status line, and the cursor is left flashing at the beginning of the bad format line positioned at the top of the screen. This usually means that the format line contains a typographical error.

"DEVICE NOT RESPONDING" (accompanied by a buzzer) This usually means that either the 850 Interface Module or the printer is not turned on, or that the printer is not on line (or not 'selected').

"Cannot find---Dn:LINKNAME" (accompanied by a buzzer and a disk error message) This message can only occur during linked printing, if the named linkfile cannot by found on the designated disk. If you insert a disk which does have the designated file, you can continue the linked print by pressing **START.** Otherwise you can abort the printout by pressing **OPTION.**

"WARNING:Current text buffer contents will be erased during linked printing" (accompanied by a high-pitched beep) Since linked printing does not preserve the initial contents of the text buffer, this message is provided as a reminder to save the text on disk if you want to preserve it. The prompt "OK to continue (Y/N)?" should be answered in the negative if you want to return to the EDIT mode and save the text before proceeding with the linked printing operation.

7.4 ERRORS IN RESPONDING TO PROMPTS

When prompts are presented in the command window, certain conventions are expected. If you don't follow them, a buzzer will sound and the prompt may be redisplayed without a specific error message telling you what you did wrong. In such cases it is best to consult the Reference Guide for a description of the function which you were attempting to activate at the time the error signal was produced. Most often this means that you answered a prompt with an inappropriate response.

8.0 CUSTOMIZING PROCEDURES

This section summarizes the procedure for customizing the operation of The Writer's Tool.

PRINTER CUSTOMIZATION

Information needed to control special printer capabilities is contained in each of the ".PPP" files provided on The Writer's Tool disk. Any one of these files can be manually installed using the **CHNGE** function available from the Print System.

You can also create your own printer data file using the **PRDAT.EXT** program accessible from the External Menu. The use of this Printer Data File Editor is discussed in Reference Section 9.4.

FORMAT AND DISPLAY CUSTOMIZATION

Changing the default settings of the format parameters and the characteristics of the screen display can be accomplished by creating and installing a custom format file (distinguished by the ".FFF" extender).

A custom format file can only be created using the CUSTM.EXT program accessible from the External Menu.

A custom format file can be manually installed using the **CHNGE** function available through the Print System.

AUTOMATIC INSTALLATION DURING BOOT LOAD

The Writer's Tool can be made to load automatically the customized printer and format information at start-up. This requires that (1) a copy of the desired printer file and a copy of the desired format file must be present on The Writer's Tool disk, and (2) the extender name of the copy must be ".PDF" for the printer file and ".FDF" for the format file.

For example, to create a customized version of the program which automatically loads the MX-80 printer information, and also loads format information stored in a file named "FORM.FFF", you should perform the following steps:

Create the file FORM.FFF

(1) Go to the Main Menu of The Writer's Tool, then press **X** to bring up the External Menu.

(2) With The Writer's Tool Master disk (or your working copy) in drive 1, press **C** to activate the customizer program.

(3) Use the Customizer program to modify format defaults and display characteristics, then save the changes in a file named "FORM.FFF" (the ".FFF" extender will be automatically added by the Customizer program).

Install printer and format files on The Writer's Tool disk

(4) Return from the customizer program to the word processing environment.

(5) Use the Disk I/O System to load the file "MX80.PPP" into memory **(be careful not to modify this file in any way).**

(6) Save the file on your working copy of The Writer's Tool disk using a different filename: "MX80.PDF" (this completes the printer installation).

(7) Next load the file "FORM.FFF" from whatever disk it was saved to **(be careful not modify this file in any way).**

(8) Re-insert your working copy of The Writer's Tool disk and save the memory contents using the name "FORM.FDF" (this completes the format file installation).

From now on, both "MX80.PDF" and "FORM.FDF" will be automatically loaded as part of The Writer's Tool boot process.

9.0 EXTERNAL MENU FUNCTIONS

The External Menu can be accessed from the Main Menu by Pressing **X.** The first character of each selection is highlighted in inverse video to indicate which key should be pressed to activate a desired function. Pressing **OPTION** returns to the Main Menu. The other selections (**S, D, C,** and **P**) load and run support programs provided on The Writer's Tool Master disk:

SPELL.EXT (a spelling checker)

DICTM.EXT (a dictionary manager)

CUSTM.EXT (a format/display customizer)

PRDAT.EXT (a printer data file editor)

These programs are activated from the word-processing environment, and return to the word processing environment when their specialized tasks are completed. However, while they are active, they make use of memory normally reserved for the text buffer. For this reason, any attempt to start an external function while text is in the buffer will produce a warning message reminding you to save the text first. When an external function returns to the word processing environment, the text buffer will be completely empty.

9.1 THE SPELLING CHECKER

This external function is activated from the External Menu by pressing **S**. This will load and run the SPELL.EXT program on The Writer's Tool Master Disk. The use of this program is described briefly here, and in greater detail in Tutorial Section 11.

PROOFREADING

The spelling checker first asks for the name of diskfile to proofread. [Whenever the spelling checker requests a filename, pressing just **RETURN** will provide access to a disk directory display.] Once the filename is entered, the program will read the file, separate words from formatting commands and punctuation marks, and compile an alphabetical list of distinct words used in the specified file. While this is happening a running count of the total number of words and the number of distinct words will be displayed on the screen.

After all words in the named file are read, the program will offer two opportunities to save the wordlist: first as an alphabetical list in which words are separated by carriage returns, and second, as a compressed list (in dictionary format). [A compressed list can later be modified by the dictionary management program.] Normally, both of these save opportunities should be bypassed by answering the prompts with **N**.

The next step in the checking process is the comparison of the list of distinct words with words in the master dictionary. You will be prompted to put the master dictionary in drive #1 and to press **START** when ready. The comparison process takes a little more than two minutes, mainly determined by the length of the master dictionary (about 20,000 words). During the comparison, any word in memory which isn't in the master dictionary will be displayed when it is identified.

When the comparison with the master dictionary is completed, the total number of unmatched words will be displayed and the prompt "CHECK Another Dictionary (Y/N)?" will be presented. This is the point at which a customized user dictionary can be compared with words not matching the master dictionary. Answering **Y** to this prompt will elicit a request for a dictionary filename. [The file identified must be in compressed format.] This can be repeated for any number of user dictionaries.

When the comparison with all dictionaries is completed (by answering **N** to the "CHECK Another..." prompt), you will be given an opportunity to save the final list of words not found in any of the dictionaries searched. After the error list is saved, or the save opportunity is bypassed, the program proceeds to the correction phase.

CORRECTING ERRORS

The correction process consists of reading the original text file a second time, identifying each word in the file which matches a word on the error list, then marking, correcting, or skipping the matched words.

The correction routine first asks for insertion of the disk containing the original text file, then asks for a choice of one of three options: MARK, CORRECT, or QUIT.

Pressing **M** initiates an automatic marking process which first requests the name of a destination file in which the marked copy will be saved. [The marked copy will have potentially erroneous words highlighted in inverse video.] Since automatic marking does not change the length of a file, you can use the same file (and filename) for both source and destination. Once a file is marked, you can return to the word-processing environment, load the file, and make corrections using editing commands. The inverse video marking makes the possible errors easy to find while scrolling through the document.

Pressing **C**, instead of **M**, initiates the interactive correction routine. This routine first requests the name of a destination file in which a corrected version will be saved. [The destination file should not be the same as the original file because correction may alter the length of words.] Once a destination file is identified the original file will be read, and displayed on the screen, word by word. When a possibly erroneous word is encountered, it is displayed in inverse video and you are given a chance to skip this occurrence of the word (Ignore), skip all occurrences of the word (Omit), mark the word (in the destination file), or correct the word immediately. If you choose correction, you will be prompted to enter the correct spelling (and case), which will then be displayed and written to the destination file.

Pressing **Q**, instead of **M** or **C**, will skip the marking/correction procedures and allow you to re-start the spelling checker (by pressing **START**, or to return to the word-processing environment (by pressing **OPTION**).

9.2 THE DICTIONARY MANAGER

This external function is activated from the External Menu by pressing **D.** This will load and run the DICTM.EXT program on The Writer's Tool Master Disk. The use of this program is also discussed in Tutorial Section 11.

Before you can use the dictionary manager you need to create two files: (1) a dictionary file, and (2) a file containing words which are all to be added to the dictionary, or all to be subtracted from the dictionary. The dictionary file must be in the compressed format which the spelling checker uses to save a compressed wordlist. The second file must contain words separated by spaces or carriage returns (do not use a list in compressed format). The words in this list do not need to be in any particular order or case. Multiple entries of the same word are also acceptable.

MODIFICATION PROCEDURE

The dictionary manager first asks for the filename specifying where new words (or words to be deleted) are to be found. The dictionary manager will read this file, compile a list of distinct words, and display a running sum of total and distinct words read. When this read process is finished, the program will ask for the name of an existing dictionary to modify. [Answering either of the filename requests with just a **RETURN** provides access to a disk directory display.]

As soon as the dictionary name is specified, the original dictionary file will be renamed by changing the extender to ".OOO", and the original name will be used for the modified dictionary.

When the renaming procedure is finished, you will be prompted to specify whether words currently in memory are to be added to, or subtracted from the dictionary. If addition is selected, only new words will be added to the dictionary; words you attempt to add, but are already present, will be displayed as they are found in the dictionary. If subtraction is selected, only words which are removed from the dictionary will be displayed. In either case the total number of words in the revised dictionary will be displayed when the modification is completed.

SPACE LIMITATIONS

Because both original and modified dictionaries must reside on the same disk, the dictionary size is limited to about half the available disk space. Because of limited memory space, any list of modifications should be no greater than about 1,000 words.

9.3 THE CUSTOMIZER

This external function is activated from the External Menu by pressing **C**. This will load and run the CUSTM.EXT program on The Writer's Tool Master Disk.

PURPOSE

The purpose of the CUSTM.EXT program is to edit or create custom format files. These files specify certain default print format parameters and default screen display characteristics. When named with the ".FFF" extender, these files can be loaded into the default parameter memory using the **CHNGE** selection available on the Print System Menu. When named with the ".FDF" extender, they can be automatically loaded as part of The Writer's Tool startup process (provided the file is present on the working copy of The Writer's Tool Master Disk).

OPERATION

The operation of the CUSTM.EXT program the procedures for loading the custom default files are both described in Tutorial Section 12. An overview of customizing procedures is also presented in Reference Section 8.

9.4 THE PRINTER DATA FILE EDITOR

This external function (filed under the name **PRDAT.EXT**) is activated from the main menu by pressing **P.**

OVERVIEW

The Writer's Tool reads a printer data file to find out what a given printer can do, and what commands are needed to control it. PRDAT.EXT is a utility program for advanced users who want to create or modify a printer data file. You might want to use PRDAT to support a currently unsupported printer, to change the style of printer support, or to add a translation table which allows convenient printing of customized characters.

The printer data file editor allows you to control all print features supported by The Writer's Tool, except for those related to proportional and micro-spaced justification. In particular, you may set those strings which control fonts, underlining, emphasized print, italics, line spacing, and more, so long as the printer supports those features.

Before getting into the details of how to customize a printer data file for your printer, you will need some background information on how the Writer's Tool tells a printer what to do. The following brief description is only a beginning. Ultimately, you should consult the detailed description of the file format and the listing example, both provided in Appendix 4, and also make a careful study of your printer manual. If you are already familiar with controlling your printer, you may skip to the next subsection.

BACKGROUND ON PRINTER CONTROLS

All programs within your Atari computer, including the Writer's Tool, send data to the printer one character at a time. Most characters get printed on the page, but others (called "control codes") instead effect some change in the operating mode of the printer. These special character codes are often used in combination with other characters as well. Such sequences are sometimes called "escape sequences" since the escape character (27 decimal, $1B hex) frequently begins the control sequences on many printers. The Writer's Tool sends many of these printer control codes during normal printing in order to control such features as underlining, italics, different fonts, and emphasized printing. For each printer that the Writer's Tool supports, there is a printer data file, which contains, as you may have guessed, all the relevant printer control codes. The role of the Printer Data File Editor, then, is to allow you to create or modify new printer data files to better support your printer.

RUNNING THE PRINTER DATA FILE EDITOR

In order to run PRDAT.EXT from The Writer's Tool, press **OPTION** to access the main menu, then **X** to access the "eXternal function" menu. At that point insert your Writer's Tool Disk into drive 1 and type P to select the Printer Data Editor. When PRDAT finishes loading, you will be presented with a main menu containing seven options. You may user either the **SELECT** key or the up/down arrow keys to move a pointer beside one of the options. Then press the **START** key to invoke that option. The PRDAT options are briefly described as follows:

Change Line Spacing - This section of the program allows you to customize those printer control strings that set the spacing between lines of text in single, double, one and one half, and half spacing modes. In addition, you can set the control strings to perform reverse paper motion, which is used for double-column printing.

Change Font Information - This section of the program allows you to set up the printer controls and other information required to put your printer into different fonts.

Change Font Modifiers - In this program section, you can change those printer controls that cause italic printing, emphasized printing, and other font modifiers.

Change Translation Table - If you want to change or add to the "translations" that occur for your printer data file, choose this section.

Read Printer File - This program section allows you to load a custom printer data file from disk into the printer customizer program.

Write Printer File - In this section, you can write out your new custom printer data file to disk in the form of a .PPP or .PDF file.

Return to Writer's Tool - This option allows you to return to text processing with The Writer's Tool. Remember that you should first write out any changes you have made to a custom printer file, as these changes will be lost after returning to The Writer's Tool.

MAKING CHANGES TO FIELDS IN THE PRINTER DATA FILE

Several conventions have been followed throughout the printer data customizer concerning selecting items and entering new data. In general, you should use the up and down arrow keys to select an item within one of the program sections (you can also use the **SELECT** key). Then, to indicate you want to make a change, press the **RETURN** key (the **START** key also works). Depending on the type of the field you are changing, you will be prompted for new information for the field. Flag fields (YES or NO value), however, toggle automatically when you press **RETURN.** Whenever you move the cursor to the left of a field which contains printer controls, the hexadecimal representation of that field appears at the bottom of your screen.

When you are entering new data, the printer data customizer will limit your response to the available space for the field you are changing. When you have entered the characters or value you want in that particular field, hit **RETURN.** If you wish to edit your response (before you hit the **RETURN** key), you can use the **BACKSPACE** key to delete one character, or the **SHIFT-CLEAR** key to delete your entire response and start over.

If you wish to enter characters with particular values (i.e., non-character data for printer control codes), please refer to the table in Appendix 1 of your Writer's Tool manual for the proper characters. Also, you will undoubtedly wish to insert a **RETURN** character ($9B) into some fields. In order to get a **RETURN** character without having that terminate the field, type **ESC-RETURN.** The **RETURN** character will be displayed as an inverse-video left arrow (instead of the normal left arrow as in The Writer's Tool), which is actually a $9D character. These two characters can, however, be distinguished in the hexadecimal display at the bottom of your screen.

NOTE: All numbers within this section are in decimal unless preceded by a dollar sign ($), signifying hexadecimal (base 16) notation. Whenever an EOM byte is referred to in the following documentation, that means a byte with the value 255 ($FF).

CHANGING LINE SPACING

This option of PRDAT.EXT allows you to change the printer controls which affect the line-per-inch spacing. The following eight controls must be specified:

1. **Sgl spaced return** - This string, 8 characters long (including the EOM character), tells the printer to enter single spacing mode and advance one line (i.e., it should have an Atari RETURN character, $9B, at the end).

2. **Dbl spaced return** - This string, also 8 characters, tells the printer to enter double spacing mode and advance two lines. If your printer does not have a double spacing mode, you should put two Atari **RETURN** characters ($9B, entered with **ESC-RETURN**). However, some printers may respond better to a **RETURN, LINEFEED** sequence (a "linefeed" is $0A, or **CNTRL-J**), which can cause less print-head movement. Be careful though, as some printers (the Atari 1027, in particular) entirely ignore line feeds; others ignore linefeeds immediately before a carriage return, but not afterwards.

3. **1 1/2 spaced return** - This field does the same as above, but for 1 1/2 spacing. If your printer does not have line spacing control (e.g., Atari 825, 1027) but does have commands for advancing 1/2 line (for the purpose of performing subscripts), you can fill in this field with a **RETURN**, followed by the codes to advance the paper 1/2 line. In this case, you should set the "Spacing each line" flag (see below).

4. **1/2 spaced return** - Again the same, but for 1/2 spacing. If your printer does not allow line spacing control but does have a control code to advance the paper 1/2 line for subscripts, you can use that code here and set the "Spacing each line" flag.

5. **1 Line Feed** - This printer control generates one line feed (a 1/6 inch paper advance). If the printer is always in 1/6 inch spacing mode (like the Atari 825), this string should either be a RETURN ($9B) or a line feed ($0A) character (don't use the $0A with the Atari 1027 or the Smith Corona typewriters). If the printer can change spacing modes, the string should put the printer into single spacing and generate one line feed ($0A).

6. **1/2 Line Feed** - Same as above, but generates a 1/12 inch paper advance. If your printer doesn't have line spacing control but does have a control code to advance the paper 1/2 line for subscripts, you can use that code here.

7. **Reverse 1/2 LF** - Generates a reverse paper advance of 1/12 inch. This control is only used for double column printing to move the paper back for the second column. If your printer does not support reverse line feeds, you will instead have to manually reverse the paper. This control should also contain the characters to resume forward line feeding, if necessary.

8. **Spacing each line** - This field is a flag which tells the Writer's Tool whether to send spacing codes to the printer at the conclusion of each line. If the value is "NO", then spacing information in fields 1-4 will be sent only when the spacing is changed with the ".Sn" command. For all other lines, only an ATASCII return character ($9B) will be sent.

CHANGING FONT INFORMATION

This option of PRDAT allows you to set up those printer control strings necessary to change the print font. The Writer's Tool supports up to five different fonts within a single printer. You need not use all five fonts, for most printers don't have that many. One font, however, **must** exist (font 1). For each font, the following three pieces of information are needed:

1. **Begin Font** - This field, which has a maximum of 16 characters, is the printer control string which puts the printer into the desired font.

2. **Width of Space** - This numeric value is the width of the space character for a given font (usually in units of 1/120 inch). For example, for a "pica" font (10 characters per inch), this value would be 12.

3. **Triple Print OK** - This flag field tells the Writer's Tool whether to "triple print" 'superscript and subscript lines (see explanation in Appendix 4.

Any unused fonts should have the value zero in the "Width of Space" field. This tells the Writer's Tool that the font does not exist.

Some other considerations exist for the "Begin font" string. If your printer supports both logic-seeking and incremental print modes, your "Begin font" strings should set the proper mode (see, for example, the PROWRTR.PPP file in Appendix 4, in which proportional fonts set incremental print, and non-proportional fonts set logic seeking).

CHANGING FONT MODIFIERS

This option in the printer data customizer allows you to set those strings which modify the way in which characters are printed. Examples of these modifiers include italics, emphasized print, underline, etc. Each of these modifiers has two fields associated with it, one for turning the option ON, and one for turning it OFF. The length of each of these options is 8 bytes (including the EOM character), except for superscript and subscript fields, which are each 16 bytes long. If any of the font modifiers are not used on your printer, they may be put to another purpose. For example, on the Comrex CR-II printer (compatible with the Brother HR-15), the italics ON and OFF sequences are used to switch between red and black ink on a two color ribbon, and the Wide modifier is used to deselect the printer (to wait for a wheel change).

As in the other sections of the printer data customizer, you should select the field you wish to change using the arrow keys, then push **RETURN** to make the change. Three fields in this section have special uses, however. If you just hit **RETURN** when prompted for "New characters:" in the "Underlined ON", "Dbl Strike ON", or "Emphasized ON" fields, you will be prompted for an overstrike character (for performing underlining by backspacing and printing an underline character) in the first two cases, or the number of times to emphasize (1 to 3) in the last case. See the description in Appendix 4 for more details.

CHANGING THE TRANSLATION TABLE

As described earlier, The Writer's Tool supports a "translation table" within the custom printer data file. This table tells the word processor that certain characters are to be replace with a sequence of one or more different characters just before outputting to the printer. For example, in the PROWRTR.PPP file, a "hard space" character ($A0), is a translated to a space character ($20), since the former character comes out as a Greek alpha on a Prowriter (this translation makes hard spaces work properly).

Each translation field begins with the character to to be translated, followed by the translation, and terminated by an EOM byte. Since two EOM bytes in a row mark the end of the table, all translations should be grouped together (if you remove one translation from the table, all following translations must be moved up). You should be aware that the print formatting routines of The Writer's Tool do not interpret the translation. Thus, if you translate 'a' to 'America', no extra space will be allocated, for the six extra characters.

READING PRINTER DATA FILES

This section of PRDAT is used to read into memory an existing printer data file. You might select this option, for example, if you wish to modify one of the .PPP files supplied with The Writer's Tool. After choosing this option from the main PRDAT menu, you will be asked to insert the disk containing the file you wish to read. You can read a printer file from any disk drive, not just drive 1. When you have inserted the desired disk, press the **START** key, or the **OPTION** key to cancel the read operation. At this point, you will be asked for the name of the printer data file. If you don't remember the name, just hit **RETURN** and you PRDAT will allow you to get a directory of printer data files (they have extensions of .PPP or .PDF) by typing the number of the drive (to access more than two drives see Appendix 2). When you know the name of the file you want to read in, type the name in response to the "File to read:" prompt. If you omit the drive number, D1: is assumed. Also, the extension '.PPP' will be added if you don't specify an extension. As usual, you can use the backspace key and the clear key to edit your response. When you give the file name and type **RETURN**, PRDAT will begin reading the file into memory. A successful read operation will be indicated by the message "Done...-Returning to MAIN MENU". If any errors occur, they will be reported instead. After reading a printer data file, you can examine its fields by using the other sections of the printer data file editor.

WRITING PRINTER DATA FILES

When you have finished modifying your custom printer data file, you can write it out to disk by selecting the "Write printer data file" option from the main menu. At that point you will be asked to insert the disk to which you want to write the file. Any disk drive in your system can be used for this purpose. Then hit **START** to continue with the write operation, or **OPTION** if you don't really want to write the file just yet. PRDAT will then ask you for the name of the printer your data file is designed for. This name may be up to 7 characters long and will appear at the bottom of the "Print System" menu in The Writer's Tool when the printer data file is loaded. After typing the printer name and **RETURN**, you will be prompted for the name of the file to write. If you want to get a disk directory at this point, just type **RETURN** (see the READING PRINTER DATA FILES section for more details). When you specify the file name, the extension .PPP will be added if you omit it, and D1: will be assumed if you don't give a drive number. In particular, if you want your printer data file to load automatically upon boot with The Writer's Tool, you should write the file to your Writer's Tool disk and specify and extension of '.PDF'. As soon as you have input the file name, the PRDAT will begin writing out the file. The message "Done...Returning to main menu" indicates a successful write. Otherwise, an error message will appear.

APPENDIX 1. CHARACTER CODES AND KEY-STROKES

There are 256 possible characters which are recognized by your ATARI computer. These are numbered from 0 to 255 (decimal) or 0 to FF (hexadecimal).

The following three pages contain tabulated decimal and hex values of the character codes, the appearance of the characters, and the key-strokes needed to produce them.

USING THE TABLE

To insert printer control codes for an unsupported printer or graphics controls for any printer, you will need to translate decimal or hex codes listed in your printer manual into key-strokes. For example, to start double-width print mode on an Integral Data Systems 560 printer, you must send the printer a decimal 1 (same as hex 1). In the following table you will find that the character corresponding to this code is produced by entering ESC CTRL-A. By inserting this character in the text and surrounding it with SHFT-CTRL-P characters, the appropriate code will be sent to the printer when the text is printed.

Producing Greek characters with a PROWRITER is another task which the table can support. For example, consulting the PROWRITER manual shows that the greek character OMEGA (Ω) can be printed by sending the hex code BC (188 decimal). This corresponds to the ATARI inverse video '<' character. Wherever this character appears in the file, the Ω character will appear in the printout. In this case, since the character does use a known space on the paper, you do not need to surround it with protector characters.

RESERVED CODES

The Writer's Tool uses some characters for formatting functions (such as inverse exclamation point, backslash, and a few others). To send the corresponding character codes to a printer requires surrounding them with SHFT-CTRL-P protector codes. There are only two codes which cannot be sent to a printer: a decimal 155 (translated to 13 by the interface module), and decimal 137 (the SHFT-CTRL-P character code).

DEC	HEX	CHAR	KEYSTROKES
0	0	♥	ESC CTRL-,
1	1	├	ESC CTRL-A
2	2	┘	ESC CTRL-B
3	3	┘	ESC CTRL-C
4	4	┤	CTRL-D
5	5	┐	ESC CTRL-E
6	6	╱	ESC CTRL-F
7	7	╲	CTRL-G
8	8	◢	CTRL-H
9	9	▪	ESC CTRL-I
10	A	◣	ESC CTRL-J
11	B	▪	CTRL-K
12	C	▪	CTRL-L
13	D	▔	ESC CTRL-M
14	E	▁	CTRL-N
15	F	▪	ESC CTRL-O
16	10	┼	CTRL-P
17	11	┌	CTRL-Q
18	12	▔	ESC CTRL-R
19	13	┼	ESC CTRL-S
20	14	●	ESC CTRL-T
21	15	▬	ESC CTRL-U
22	16	▏	CTRL-V
23	17	┬	ESC CTRL-W
24	18	┴	ESC CTRL-X
25	19	▌	CTRL-Y
26	1A	└	ESC CTRL-Z
27	1B	←	ESC ESC
28	1C	↑	ESC CTRL-↑
29	1D	↓	ESC CTRL-↓
30	1E	←	ESC CTRL-←
31	1F	→	ESC CTRL-→
32	20	space	SPACE BAR
33	21	!	SHIFT-1
34	22	"	SHIFT-2
35	23	#	SHIFT-3
36	24	$	SHIFT-4
37	25	%	SHIFT-5
38	26	&	SHIFT-6
39	27	'	SHIFT-7
40	28	(	SHIFT-9
41	29	)	SHIFT-0
42	2A	*	*
43	2B	+	+
44	2C	,	,
45	2D	-	-
46	2E	.	.
47	2F	/	/
48	30	0	0
49	31	1	1
50	32	2	2
51	33	3	3
52	34	4	4
53	35	5	5
54	36	6	6
55	37	7	7
56	38	8	8
57	39	9	9
58	3A	:	SHIFT-:
59	3B	;	;
60	3C	<	<
61	3D	=	=
62	3E	>	>
63	3F	?	SHIFT-?

DEC	HEX	CHAR	KEYSTROKES
64	40	@	SHIFT-8
65	41	A	A
66	42	B	B
67	43	C	C
68	44	D	D
69	45	E	E
70	46	F	F
71	47	G	G
72	48	H	H
73	49	I	I
74	4A	J	J
75	4B	K	K
76	4C	L	L
77	4D	M	M
78	4E	N	N
79	4F	O	O
80	50	P	P
81	51	Q	Q
82	52	R	R
83	53	S	S
84	54	T	T
85	55	U	U
86	56	V	V
87	57	W	W
88	58	X	X
89	59	Y	Y
90	5A	Z	Z
91	5B	[	SHIFT-[
92	5C	\	SHIFT-\
93	5D	]	SHIFT-]
94	5E	^	SHIFT-^
95	5F	_	SHIFT-_
96	60	♦	CTRL-.
97	61	a	a
98	62	b	b
99	63	c	c
100	64	d	d
101	65	e	e
102	66	f	f
103	67	g	g
104	68	h	h
105	69	i	i
106	6A	j	j
107	6B	k	k
108	6C	l	l
109	6D	m	m
110	6E	n	n
111	6F	o	o
112	70	p	p
113	71	q	q
114	72	r	r
115	73	s	s
116	74	t	t
117	75	u	u
118	76	v	v
119	77	w	w
120	78	x	x
121	79	y	y
122	7A	z	z
123	7B	♠	CTRL-;
124	7C	|	SHIFT-|
125	7D	▚	ESC SHIFT-CLEAR
126	7E	◀	ESC BACK S
127	7F	▶	ESC TAB

DEC	HEX	CHAR	KEYSTROKES
128	80		INV CTRL-,
129	81		INV ESC CTRL-A
130	82		INV ESC CTRL-B
131	83		INV ESC CTRL-C
132	84		INV CTRL-D
133	85		INV ESC CTRL-E
134	86		INV ESC CTRL-F
135	87		INV CTRL-G
136	88		INV CTRL-H
137	89		INV ESC CTRL-I
138	8A		INV ESC CTRL-J
139	8B		INV CTRL-K
140	8C		INV CTRL-L
141	8D		INV ESC CTRL-M
142	8E		INV CTRL-N
143	8F		INV ESC CTRL-O
144	90		INV CTRL-P
145	91		INV CTRL-Q
146	92		INV ESC CTRL-R
147	93		INV ESC CTRL-S
148	94		INV ESC CTRL-T
149	95		INV ESC CTRL-U
150	96		INV CTRL-V
151	97		INV ESC CTRL-W
152	98		INV ESC CTRL-X
153	99		INV CTRL-Y
154	9A		INV ESC CTRL-Z
155	9B	EOL(←)	RETURN
156	9C		ESC SHIFT-DELETE
157	9D		ESC SHIFT-INSERT
158	9E		ESC CTRL-TAB
159	9F		ESC SHIFT-TAB
160	A0		INV SPACE BAR
161	A1	!	INV SHIFT-1
162	A2	"	INV SHIFT-2
163	A3	#	INV SHIFT-3
164	A4	$	INV SHIFT-4
165	A5	%	INV SHIFT-5
166	A6	&	INV SHIFT-6
167	A7	'	INV SHIFT-7
168	A8	(	INV SHIFT-9
169	A9	)	INV SHIFT-0
170	AA	*	INV ESC *
171	AB	+	INV ESC +
172	AC	,	INV ,
173	AD	-	INV ESC -
174	AE	.	INV .
175	AF	/	INV /
176	B0	0	INV 0
177	B1	1	INV 1
178	B2	2	INV 2
179	B3	3	INV 3
180	B4	4	INV 4
181	B5	5	INV 5
182	B6	6	INV 6
183	B7	7	INV 7
184	B8	8	INV 8
185	B9	9	INV 9
186	BA	:	INV SHIFT-:
187	BB	;	INV ;
188	BC	<	INV <
189	BD	=	INV ESC =
190	BE	>	INV >
191	BF	?	INV SHIFT-?
192	C0	@	INV SHIFT-@
193	C1	A	INV A
194	C2	B	INV B
195	C3	C	INV C
196	C4	D	INV D
197	C5	E	INV E
198	C6	F	INV F
199	C7	G	INV G
200	C8	H	INV H
201	C9	I	INV I
202	CA	J	INV J
203	CB	K	INV K
204	CC	L	INV L
205	CD	M	INV M
206	CE	N	INV N
207	CF	O	INV O
208	D0	P	INV P
209	D1	Q	INV Q
210	D2	R	INV R
211	D3	S	INV S
212	D4	T	INV T
213	D5	U	INV U
214	D6	V	INV V
215	D7	W	INV W
216	D8	X	INV X
217	D9	Y	INV Y
218	DA	Z	INV Z
219	DB	[	INV SHIFT-[
220	DC	\	INV SHIFT-\
221	DD	]	INV SHIFT-]
222	DE	^	INV SHIFT-^
223	DF	_	INV SHIFT-_
224	E0	♦	INV CTRL-.
225	E1	a	INV a
226	E2	b	INV b
227	E3	c	INV c
228	E4	d	INV d
229	E5	e	INV e
230	E6	f	INV f
231	E7	g	INV g
232	E8	h	INV h
233	E9	i	INV i
234	EA	j	INV j
235	EB	k	INV k
236	EC	l	INV l
237	ED	m	INV m
238	EE	n	INV n
239	EF	o	INV o
240	F0	p	INV p
241	F1	q	INV q
242	F2	r	INV r
243	F3	s	INV s
244	F4	t	INV t
245	F5	u	INV u
246	F6	v	INV v
247	F7	w	INV w
248	F8	x	INV x
249	F9	y	INV y
250	FA	z	INV z
251	FB	♠	INV CTRL-;
252	FC	|	INV SHIFT-|
253	FD	↖	ESC CTRL-2
254	FE	↓	INV ESC CTRL-DELETE
255	FF	→	INV ESC CTRL-INSERT

APPENDIX 2. USING MORE THAN TWO DISK DRIVES

The Disk Operating System provided with The Writer's Tool is configured to support only two disk drives. The Writer's Tool itself can support up to four disk drives, if provided with an appropriately configured DOS.

RE-CONFIGURING DOS

If you own a copy of DOS XL, you can create a reconfigured DOS.SYS, following the instructions on Page 106 of the DOS XL Manual (if you will be using double-density drives make sure that the SABYTE variable is at least 4). Use this version of DOS.SYS to replace the DOS.SYS provided with The Writer's Tool.

If you are using ATARI DOS 3, you should follow the instructions on pages 42-43 of your DOS 3 manual. The standard configuration of DOS 3 supports 5 system buffers (number of drives + number of open disk files at any one time). If you are using three drives, no reconfiguration is necessary. For four drives you will need 6 system buffers.

If you want to use ATARI DOS 2, follow the re-configuration instructions in Appendix G of the DOS 2 Manual. In brief, you should boot DOS with BASIC, poke a new value into decimal location 1802, hit SYSTEM RESET, go back to DOS, then save DOS, using the "H" option. The appropriate value to poke is 7 for three drives, and 15 (decimal) for four drives.

SIDE EFFECTS

Since a sector buffer is required for each disk drive or open file, supporting more disk drives uses more memory (128 bytes per single density drive, 256 bytes per full double density drive). This means that somewhat less space will be available for the text buffer.

ACCESSING DRIVES 3 AND 4

To load, save, or delete files on these drives, use filenames with device specifiers D3: or D4:. To access disk directories for these drives, press **3** or **4** at any point where **1** or **2** are valid directory commands.

APPENDIX 3. USING DIFFERENT DISK DENSITIES

The number of disk drives available for use with ATARI home computers has grown enormously in just the past year. Many of these drives are capable of operation at two or three different densities, and offer much greater storage capacity than the old Atari 810.

COMMON DISK DENSITIES

There are basically three distinct disk formats in common use by Atari-compatible disk drives:

1. Single Density (Atari DOS 2.0s format) which uses 128-byte sectors, 18 sectors per track, and 40 tracks, for a total unformatted capacity of 92,160 bytes.

2. Medium Density (Atari DOS 3 format) which uses 128-byte sectors, 26 sectors per track, and 40 tracks, for a total unformatted capacity of 133,120 bytes (although this is a bit less than 1.5 times the single-density capacity, the Atari DOS 3 manual refers to this as "double density").

3. Double Density (generally observed industry standard format) which uses 256-byte sectors, 18 sectors per track and 40 tracks (for a total unformatted capacity of 184,320 bytes, exactly twice the single-density capacity).

USING SINGLE AND DOUBLE DENSITIES

Although The Writer's Tool is provided in single density disk format, it is supplied with a disk operating system capable of writing and reading either single or full double-density disks. This disk operating system (DOS) is called DOS XL, product of OSS, Inc. (it's stored in the disk file DOS.SYS). When The Writer's Tool boots from its master disk, this will leave the disk drive set in single density format. To change the disk density, use the initialization command available from the Disk I/O system (provided, of course, that you are using a disk drive which supports dual densities). You can then read, write, and initialize disks in double-density format. If you have two disk drives, you have the flexibility to operate them in any combination of single and double density. You can read a single density disk, then write a double density disk, or vice versa.

USING DOS3 (MEDIUM DENSITY)

The Writer's Tool can be used to read and write medium density formats only if you copy the disk part of the program to a DOS 3-formatted disk on which you have installed the DOS-3 version of the

DOS.SYS file (called FMS.SYS). The copying procedure is described in the DOS 3 users guide. This DOS-3 version of the program will only be able to read and write DOS-3 formatted disks, and will not be able to properly switch between medium and single density formats.

MAKING A DOUBLE-DENSITY COPY OF THE PROGRAM

To make a working double-density copy of The Writer's Tool, you must copy the disk-based portion of the software to a double-density formatted disk which contains your double-density disk operating system (usually a DOS.SYS file). Your disk operating system manual should describe how to make a double density copy of a single density disk. You can also do this just by using The Writer's Tool to load each file from the single density master, then save it to a double density disk. The only files you really need to copy are DOS.SYS, AUTORUN.SYS, MERGE.OBJ, and whichever printer data files you may need. For most purposes, there is no need to make a double-density copy of the program, since you can reconfigure most dual density drives from within the program itself.

COPYRIGHT REMINDER

The disk-based portion of The Writer's Tool program is copyrighted material, as is the ROM portion. You are authorized to make copies for backup and extended use capabilities described above, but you are not authorized to sell or distribute such copies.

APPENDIX 4. THE WRITER'S TOOL PRINTER DATA FILE FORMAT

Disclaimer

The following summary of information within The Writer's Tool printer data files is provided by OSS to users for the purpose of creating printer files for their own printers. Although OSS has taken care to ensure the accuracy of the enclosed information, no warranty of its correctness is expressed or implied. OSS bears no liability for costs incurred which result from errors in this documentation.

Important Notes

1. All numbers within the following table are in decimal unless preceded by a dollar sign ($). Numbers preceded by a dollar sign are in hexadecimal (base 16) notation. In the sample listing these conventions are not followed.

2. Whenever an EOM byte is referred to in the following documentation, that means a byte with the value 255 ($FF).

3. Each sequence of data within a printer file must start at the given offset from the beginning of the file. All offsets are given in decimal notation, except in the sample listing, where decimal and hex values are both given.

4. The length of each data sequence must not exceed the length specified. Some sequences are followed by a separate field which indicates their length, while others are terminated by an EOM byte. In the latter case, the entire sequence, including the EOM, must fit within the specified length. Those fields which require a terminating EOM byte are marked by a preceding asterisk (*) in the tables that follow.

Cautions

1. In order to retain flexibility for change in future versions of the Writer's Tool, the contents of the custom printer files are subject to modification without prior notice from OSS.

2. Printers do not always do what their instruction manuals claim. Special features may also have restrictions on their use which may not be documented.

3. A printer file must be tested extensively on the printer for which it was designed in order to make sure it works properly.

4. There exist printers whose control philosophies cannot be fully accommodated. For example, some printers do not provide a method for external control of fractional spacing necessary for proportional font justification, tab alignments, and centering.

Special Features

Fonts. The Writer's Tool allows up to five fonts on each printer to be supported. Whenever a set of flags for each font occur in the custom printer file (5 bytes, one for each font), those flags should be set appropriately for each font that is supported. If your printer supports fewer than five fonts, unused flag bytes should be zero.

Triple printing. If specified within the custom printer file, the Writer's Tool provides a feature whereby a line is printed in three passes. First, the paper is advanced by 1/2 line and any superscripts are printed. Second, the paper is advanced an additional 1/2 line and everything but subscripts are printed. Last, the paper is again advanced and all subscripts for the line are printed. This method is more accurate than moving the paper back and forward for each instance of a superscript and subscript. In general, if your printer does logic seeking, you should use triple print mode. However, if your printer is in "incremental print" mode, you may use either value for the triple print flag. For a proportional font you should not use triple printing, since your super- and subscripts will not be aligned properly with your text. Also, triple will not work for all printers or all fonts on a single printer so some experimentation may be needed. If your printer supports half line feeds in a font, and that font is not proportional, we recommend you set the "triple print" flag for that font.

Micro justification. Some printers allow for single dot head movement (usually 1/120") in order to better justify text. If your printer supports this feature, you may set the corresponding flag for that font within the custom printer file. This flag should only be set for proportional fonts.

CONTENTS OF THE PRINTER FILE

With the above considerations in mind, here is the format for the Writer's Tool printer files. Each field has an associated offset from the beginning of the file and a maximum length. Be sure to note whether a field should be terminated by an EOM or not. In order to aid in understanding the printer file format, a listing of the contents of the Prowriter format file has been included with this document.

OFFSET	SIZE	CONTENTS
0	6	Header for printer file. To ensure compatibility with future versions, these bytes should contain $FF, $FF, $00, $00, $00, $00.
* 6	8	Printer name in ATASCII. The name plus the terminating EOM ($FF) must fit in the 8 bytes.
14	8	Printer control codes to advance the paper 1/6". This sequence should include codes to force the printer into single spacing mode, if necessary.
22	1	Length of printer control sequence in previous field.
23	8	Printer control codes to advance the paper 1/12", if the printer supports half spacing.
31	1	Length of sequence in previous field.
32	5	Flag to indicate whether "triple print" is allowed for a particular font. A one implies yes, zero implies no. The following three fields need data only if "triple print" is allowed for at least one font. In most cases, the fields for 1/2 and full line feeds will contain the same data as corresponding fields above.
37	8	Printer control sequence to advance the paper 1/4 line feed.
45	1	Length of sequence in previous field.
46	8	Printer control sequence to advance the paper 1/2 line feed.
54	1	Length of sequence in previous field.

OFFSET	SIZE	CONTENTS
55	8	Printer control sequence to advance the paper 1 line feed.
63	1	Length of sequence in previous field.
64	12	Printer control sequence to cause a reverse half line feed, if the printer supports it. This sequence is used to perform double-column printing.
76	1	Length of sequence in previous field.
77	5	Flags to indicate which fonts are to be "micro-justified" by single dot spaces. This only applies to non-proportional fonts which can make use of horizontal motion index commands (see offset 409).
82	5	Width of the space character in each font, usually in 1/120 inch units (e.g., pica fonts are 12). A zero in a cell here implies that a font is not available. A non-zero value need not be in 1/120 inch units, but, for proportional fonts the units must agree with units specifying the proportional character widths.
87	5	Flags to indicate whether it is allowed to insert an extra dot space between proportional characters in a particular font. A one implies yes, zero implies no.
* 92	8	Printer control codes to insert zero dot spaces (1/120") into the current line. This is a pattern for fractional space insertions between 1 dot and s-1 dots, where s denotes the dot-width of the space character. A designated variable byte in this pattern will be replaced by n+byte value to achieve a fractional space of n dots.
100	16	Printer control sequence to insert n dot spaces into the current line. This sequence should contain one byte to which n is added that tells the printer how many dot spaces to advance.
116	1	Length of sequence in previous field.

OFFSET	SIZE	CONTENTS
117	1	The offset in bytes of the variable part to which n will be added to produce a fractional space of n dots in the previous sequence.
118	32	Printer control sequence to insert n dot spaces into the current line while continuing to underline text. This sequence should contain one byte to which n is added that tells the printer how many dot spaces to advance[3].
150	1	Length of sequence in previous field.
151	1	The offset in bytes of the variable part to which to add n in the previous sequence.
152	256	The width of each character in the proportional font, if any, in units of 1/120 inch (or whatever units your printer uses to specify dot widths). Each character's width takes up one byte in this table.
408	1	Flag to indicate that horizontal motion index command is present in the following three fields. This is currently used for the Comrex CR-II (for proportional fonts and micro-justification of non-proportional fonts) and Microline 92 printers (for micro-justification of correspondence quality fonts).
409	3	Horizontal motion index command sequence for zero motion. This is a pattern for the actual command sent to the printer. The third byte designates an offset which will be subtracted from n to set the horizontal motion index to n units per character (usually in units of 1/120 inch).
* 412	8	Printer control codes to begin italic printing. The sequence in this and the following 13 fields must be terminated by an EOM character.
* 420	8	Printer codes to cease italic mode printing.
* 428	8	Printer codes to begin emphasized printing.
* 436	8	End emphasized printing (see Additional Features Note 1, at the end of the table).

OFFSET	SIZE	CONTENTS
* 444	8	Begin double strike printing. This is the same sequence as emphasized printing on some printers (see Additional Features Note 2).
* 452	8	End double strike printing.
* 460	8	Begin double width printing.
* 468	8	End double width printing. If the eighth byte in this field is and upper case 'F' the font command issued before double width was selected will be re-issued immediately after double wide printing is completed. This feature is provided for printers that have no "double wide OFF command".
* 476	8	Begin underlined printing. (See Additional Features Note 2.)
* 484	8	End underlined printing.
* 492	16	Begin superscript printing. This sequence and the following 3 may advance or reverse the paper or not, depending on how the printer performs superscript and subscript printing. Superscript and subscript commands are not used when the triple print flag is set.
* 508	16	End superscript printing.
* 524	16	Begin subscript printing.
* 540	16	End subscript printing.
556	1	Flag which, when one, indicates that one of the following line spacing sequences must be sent at the end of every line, instead of just when changing line spacing (e.g., .s2 format command). If this flag is zero, just ATASCII end of lines ($9B) will be sent unless the line spacing is changed. On some printers, this flag needs to be one, but on others it may then disable bi-directional printing.
* 557	8	Printer sequence to begin single spacing and advance the paper one line (i.e., sequence includes an end of line ($9B) character).

	OFFSET	SIZE	CONTENTS
*	565	8	Begin double spacing and advance the paper one double space.
*	573	8	Begin 1 1/2 line spacing and advance 1 1/2 lines.
*	581	8	Begin 1/2 line spacing and advance 1/2 line.
*	589	8	Turn on paper end detection within printer. This sequence, which must be terminated by an EOM, is only used to control paper end detection when switching between sheet feed and continuous feed paper.
	597	1	Length of data in previous field, excluding the EOM character.
*	598	8	Turn off paper end detection within the printer to perform single sheet feeding. This sequence must be terminated by an EOM character.
	606	1	Length of data in previous field, excluding the EOM character.
	607	5	Flags to indicate which fonts are proportional. A one indicates a proportional font, zero non-proportional.
*	612	16	Printer control sequence to begin printing in font 1. All the font sequences must be terminated by EOM characters.
*	628	16	Printer control sequence to begin printing in font 2.
*	644	16	Printer control sequence to begin printing in font 3.
*	660	16	Printer control sequence to begin printing in font 4.
*	676	<256	Printer control sequence to begin printing in font 5. This sequence must be terminated by and EOM character and may be longer or shorter than the previous font sequences.

The following fields are optional and form the character translation table for the printer file. This table must immediately follow the EOM character which terminates the font 5 sequence. The total length of the translation table must not exceed 255 bytes, regardless of the length of font 5.

FIELD	CONTENTS
Start of table	This EOM byte ($FF) indicates that a translation table is present. If a table is used, this byte must immediately follow the EOM which terminates the font 5 sequence above. In other words, two EOM bytes in a row must be present.
* Table entries	Any number of table entries which consist of 1) a byte containing the character to be translated, 2) any number of bytes which form the translation of that character, and 3) a terminating EOM character.
End of table	An additional EOM byte which terminates the entire translation table.

Additional Features of Printer Files

Three fields within the custom printer file can perform special functions designed to widen the variety of printers supported. These features are invoked by putting an EOM terminating byte as the first byte of the field's printer control sequence. When this is performed, the second byte of the sequence has special significance. The following fields have special capabilities:

1. If the first byte in the field which causes emphasized print to begin is an EOM, emphasized printing will be performed by sending the printer a backspace character, followed by the character to be emphasized. The second byte in the field specifies how many times to overprint characters in addition to the first printing. This byte must be less than 4. This feature will not operate on some printers which use logic seeking, but is instead designed for daisy wheel and other letter quality printers.

2. If the first byte in the field which causes double strike printing to begin is an EOM, double striking will not be performed. Instead, the double strike mode causes The Writer's Tool to send the printer a backspace followed by an additional character. The second byte is the character which will be printed on top of the original character. For example, if the second byte were the slash character, "/", the double strike format would cause slashes to be printed on top of text. This feature is mainly useful in indicating proposed deletions of text, say in a contract, in which case SHIFT-CTRL-D takes on the meaning of "delete" rather than "double strike."

3. If the first byte in the field which causes underlining to begin is an EOM, underlining will be performed by sending the printer a backspace followed by an additional character. The second byte is the character which will be printed on top of the original character. For example, if the second character is an underline character, "_", underlining will be performed by sending a backspace and then an underline. This feature is very useful on some daisy wheel and typewriter printers which do not have an underlining "mode."

EXAMPLE LISTING OF A PRINTER DATA FILE

The following commented listing of the PROWRTR.PPP file is mainly in ATARI Macro Assembler format. The occasional numbers in the leftmost column are decimal offsets corresponding to those identified in the previous table. The next column displays the hexadecimal offsets. The third column displays the data (in 2-digit hex format) generated by the defining statements of the source code on the right. Data following the DB Pseudo Op are in decimal format, or literal strings (when surrounded by single quotes).

```
                                ;D1:PROWRTR.ASM   4-1-84
                                ;
                                ; Copyright 1984  Madison Micro
                                ;
                                ; Printer Data Tables for the
                                ; C.ITOH PROWRITER 8510 DOT MATRIX Printer
                                ;
                                ;Symbol Definitions
                                ;
                 = 00FF         EOM = 255 ;End Of Message
                 = 001B         ESC = 27  ;ESCAPE
                 = 000A         LF = 10   ;LINE FEED
                 = 009B         RET = 155 ;Carriage Return (ATASCII)
                                ;          (translated to 13 by I/F)
                                 LIST D
      0000       = 0000         ORG $00
                                ;
  0   0000   FFFF000000  HEADER DB EOM,EOM,0,0,0,0
               00
                                ;
  6   0006   50524F5752  PRNAME DB 'PROWRTR',EOM
               5452FF
                                ;
 14   000E   1B5432340A  LFDAT   DB ESC,'T24',LF  ;1/6 inch feed
      0013   000000              DB 0,0,0
      0016   05          LFLEN   DB 5     ;no of bytes in LFDAT
                                ;
 23   0017   1B5431320A  HLFDAT DB ESC,'T12',LF  ; 1/2 LF
      001C   000000              DB 0,0,0
      001F   05          HLFLEN DB 5
                                ;
                                ;        F1,2,3,4,5
 32   0020   0101010000  TRPOK DB  1,1,1,0,0 ;Triple Print Flags
      0025   1B5430360A  LF4     DB ESC,'T06',LF ;1/4 LF
      002A   000000              DB 0,0,0
      002D   05          LF4LEN DB 5
```

```
      002E  1B5431320A  LF2     DB ESC,'T12',LF ;1/2 LF
      0033  000000              DB 0,0,0
      0036  05          LF2LEN  DB 5
      0037  1B5432340A  LF1     DB ESC,'T24',LF ;FULL LF
      003C  000000              DB 0,0,0
      003F  05          LF1LEN  DB 5
                        ;
 64   0040              REVDAT  ;codes for reverse 1/2 lnfeed
      0040  1B5431321B   DB ESC,'T12',ESC,'r',LF,ESC,'f'
            720A1B66
      0049  000000       DB 0,0,0
      004C  09          REVLEN  DB 9
                        ;
                        ;         F1,2,3,4,5  MICRO JUSTIFY
      004D  0000000000  JUS3OK  DB 0,0,0,0,0 ; OK IF 1
                        ;
 82   0052              FONWID   ;Size of space in 1/120 in
 82   0052  0C             DB 12 ;FONT 1 (10CPI)
      0053  0A             DB 10 ;FONT 2 (12CPI)
      0054  07             DB 7  ;FONT 3 (17CPI)
      0055  07             DB 7  ;FONT 4 (22.86sp/in)
      0056  08             DB 8  ;FONT 5 (20sp/in)
                        ;
                        ;        F1,2,3,4,5
 87   0057  0000000001  SP1OK  DB 0,0,0,0,1
 92   005C  1B31FF0000  SP1DAT  DB ESC,'1',EOM,0,0,0,0,0
            000000
                        ;
100   0064              FSPDAT  ;codes for zero fractional space
      0064  1B30000000   DB ESC,'0',0,0,0,0,0,0
            000000
      006C  0000000000   DB 0,0,0,0,0,0,0,0
            000000
      0074  02          FSPLEN  DB 2
      0075  01          FSPVP   DB 1   ;offset to variable byte
118   0076              FSUDAT         ;zero fract space with underline
      0076  1B30         DB ESC,'0'       ;95=UNDERLINE CHAR
      0078  0000000000   DB 0,0,0,0,0,0
            00
      007E  0000000000   DB 0,0,0,0,0,0,0,0
            000000
      0086  0000000000   DB 0,0,0,0,0,0,0,0
            000000
      008E  0000000000   DB 0,0,0,0,0,0,0,0
            000000
      0096  02          FSULEN  DB 2
      0097  01          FSUPVP  DB 1  ;OFFSET OF VARIABLE PART
                        ;
```

```
152  0098                    DOTTBL  ;Proportional character widths (1/120 in)
     0098  0606060606         DB 6,6,6,6,6,6,6,6
           060606
     00A0  0606060606         DB 6,6,6,6,6,6,6,6
           060606
     00A8  0606060606         DB 6,6,6,6,6,6,6,6
           060606
     00B0  0606060606         DB 6,6,6,6,6,6,6,6
           060606
                            ;
     00B8  07070A0E0C         DB 7,7,10,14,12,16,13,7  ;space to '
           100D07
     00C0  07070C0C07         DB 7,7,12,12,7,12,7,12     ;( to /
           0C070C
     00C8  0C0C0C0C0C         DB 12,12,12,12,12,12,12,12 ;0 TO 7
           0C0C0C
     00D0  0C0C07070C         DB 12,12,7,7,12,12,12,12   ;8 TO ?
           0C0C0C
     00D8  0E100F0E0F         DB 14,16,15,14,15,15,15,14 ;@ TO G
           0F0F0E
     00E0  0F090D0C0D         DB 15,9,13,12,13,17,16,15  ;H TO O
           11100F
     00E8  0D100F0C0E         DB 13,16,15,12,14,15,16,17 ;P TO W
           0F1011
     00F0  0B0E0B0C0C         DB 11,14,11,12,12,12,12,12 ;X TO UND
           0C0C0C
     00F8  070C0C0A0C         DB 7,12,12,10,12,12,10,12  ;SQ TO g
           0C0A0C
     0100  0C08070A08         DB 12,8,7,10,8,16,12,12    ;h TO o
           100C0C
     0108  0C0C0A0C0A         DB 12,12,10,12,10,12,12,16 ;p TO w
           0C0C10
     0110  0C0C0A0A07         DB 12,12,10,10,7,10,12     ;x TO tilda
           0A0C
     0117  01                 DB 1
                            ;       Beginning of 2nd 128 Codes
                            ;
     0118  1010101010         DB 16,16,16,16,16,16,16,16
           101010
     0120  1010101010         DB 16,16,16,16,16,16,16,16
           101010
     0128  1010101010         DB 16,16,16,16,16,16,16,16
           101010
     0130  1010101010         DB 16,16,16,16,16,16,16,16
           101010
                            ;
     0138  1010101010         DB 16,16,16,16,16,16,16,16
           101010
```

```
                0140  1010101010      DB 16,16,16,16,16,16,16,16
                      101010
                0148  1010101010      DB 16,16,16,16,16,16,16,16
                      101010
                0150  1010101010      DB 16,16,16,16,16,16,16,16
                      101010
                0158  1010101010      DB 16,16,16,16,16,16,16,16
                      101010
                0160  1010101010      DB 16,16,16,16,16,16,16,16
                      101010
                0168  1010101010      DB 16,16,16,16,16,16,16,16
                      101010
                0170  1010101010      DB 16,16,16,16,16,16,16,16
                      101010
                0178  1010101010      DB 16,16,16,16,16,16,16,16
                      101010
                0180  1010101010      DB 16,16,16,16,16,16,16,16
                      101010
                0188  1010101010      DB 16,16,16,16,16,16,16,16
                      101010
                0190  1010101010      DB 16,16,16,16,16,16,16,16
                      101010
                                      ;
408             0198  00              WHEEL DB 0 ;This produces horizontal
                                      ;          motion control on each char
                0199  00              HMI1 DB 0   ; Horizontal motion index
                019A  00              HMI2 DB 0   ; control codes 1 and 2
                019B  01              HMIOFF DB 1 ; 3rd is subtracted from
                                      ;             N, where motion will be
                                      ;             N/120 inch.
                                      ;
                                      ; Font Modifier Translations for PROWRTR
                                      ;
412             019C  1B58FF          ITON   DB ESC,'X',EOM  ;USE UNDERLINE
                019F  0000000000              DB 0,0,0,0,0    ;for italics
                01A4  1B59FF          ITOFF  DB ESC,'Y',EOM
                01A7  0000000000              DB 0,0,0,0,0
                01AC  1B21FF          BLDON  DB ESC,'!',EOM  ;Bold print on
                01AF  0000000000              DB 0,0,0,0,0
                01B4  1B22FF          BLDOFF DB ESC,'"',EOM
                01B7  0000000000              DB 0,0,0,0,0
                01BC  1B21FF          DSTON  DB ESC,'!',EOM  ;USE BOLD FOR
                01BF  0000000000              DB 0,0,0,0,0    ;double strike
                01C4  1B22FF          DSTOFF DB ESC,'"',EOM
                01C7  0000000000              DB 0,0,0,0,0
```

```
                                ;
                                ; Double Width
                                ;
      01CC  0EFF          DBLON   DB 14,EOM
      01CE  0000000000            DB 0,0,0,0,0,0
            00
      01D4  0FFF          DBLOFF  DB 15,EOM
      01D6  0000000000            DB 0,0,0,0,0,0
            00
      01DC  1B58FF        UNDON   DB ESC,'X',EOM
      01DF  0000000000            DB 0,0,0,0,0
      01E4  1B59FF        UNDOFF  DB ESC,'Y',EOM
      01E7  0000000000            DB 0,0,0,0,0
492   01EC  1B5431301B    UPON    DB ESC,'T10',ESC,'r',LF
            720A
      01F3  1B543132FF            DB ESC,'T12',EOM   ;5/12 LF UP
      01F8  00000000              DB 0,0,0,0
      01FC  1B5431301B    UPOFF   DB ESC,'T10',ESC,'f',LF
            660A
      0203  1B543132FF            DB ESC,'T12',EOM   ;5/12 LF DWN
      0208  00000000              DB 0,0,0,0
      020C  1B5431300A    DWNON   DB ESC,'T10',LF,ESC,'r'
            1B72
      0213  1B543132FF            DB ESC,'T12',EOM   ;5/12 LF DWN
      0218  00000000              DB 0,0,0,0
      021C  1B5431300A    DWNOFF  DB ESC,'T10',LF,ESC,'f'
            1B66
      0223  1B543132FF            DB ESC,'T12',EOM   ;5/12 LF UP
      0228  00000000              DB 0,0,0,0
                                ;
                                ; Line Spacing Data for PROWRTR
                                ;
556   022C  00            LSCFLG  DB 0 ;NOT NEEDED EVERY LINE
                                ;
      022D  1B5432349B    LSCOD1  DB ESC,'T24',RET ;Single space
      0232  FF0000                DB EOM,0,0
      0235  1B5434389B    LSCOD2  DB ESC,'T48',RET ;Double space
      023A  FF0000                DB EOM,0,0
      023D  1B5433369B    LSCOD3  DB ESC,'T36',RET ;1+1/2 SPACE
      0242  FF0000                DB EOM,0,0
      0245  1B5431329B    LSCOD4  DB ESC,'T12',RET  ;1/2 SPACE
      024A  FF0000                DB EOM,0,0
                                ;
                                ; PAPER END DETECTOR HANDLING CODES
                                ;
589   024D  FF            PEDON   DB EOM  ;DUMMY FOR PROWRTR
      024E  0000000000            DB 0,0,0,0,0,0,0
            0000
```

```
        0255  00              PEDONL DB 0
        0256  FF              PEDOFF DB EOM
        0257  0000000000              DB 0,0,0,0,0,0,0
              0000
        025E  00              PEDOFL DB 0
                              ;
                              ;
                              ; FONT DATA FOR PROWRTR
                              ;
                              ;       F1,2,3,4,5
        025F  0000000101      PRPF DB 0,0,0,1,1 ; 1=PRPORTNL FONT
                              ;
                              ;   ESC ']' SETS LOGIC SEEKING
                              ;   ESC '[' sets incremental print
                              ;   ESC '&' Selects Greek/Graph
                              ;           over Japanese chars.
                              ;
612     0264  1B261B4E1B      FONT1 DB ESC,'&',ESC,'N',ESC,']',EOM
              5DFF
        026B  0000000000            DB 0,0,0,0,0,0,0,0
              000000
        0273  00                    DB 0
        0274  1B261B451B      FONT2 DB ESC,'&',ESC,'E',ESC,']',EOM
              5DFF
        027B  0000000000            DB 0,0,0,0,0,0,0,0
              000000
        0283  00                    DB 0
        0284  1B261B511B      FONT3 DB ESC,'&',ESC,'Q',ESC,']',EOM
              5DFF
        028B  0000000000            DB 0,0,0,0,0,0,0,0
              000000
        0293  00                    DB 0
        0294  1B261B501B      FONT4 DB ESC,'&',ESC,'P',ESC,'[',EOM
              5BFF
        029B  0000000000            DB 0,0,0,0,0,0,0,0
              000000
        02A3  00                    DB 0
        02A4  1B261B501B      FONT5 DB ESC,'&',ESC,'P',ESC,'['
              5B
        02AA  1B543132FF            DB ESC,'T12',EOM
                              ;
                              ; Translation Table
                              ;
        02AF  FFA020FFFF            DB EOM,160,32,EOM,EOM
                              ;
                              ;Translates the inverse video space
                              ;to a normal space
```

APPENDIX 5. COMMON PROBLEMS

UNEXPECTED BLANK SPACE ON PRINTOUT

You inserted a screen line of blank space, typed a few words, then forgot to use **CTRL-J** to close the gap, or forgot to press **RETURN.**

UNEXPECTED LINE FEEDS

If you use a left margin and line length which, in total, equal or exceed the number of printer columns, the printer will probably execute an automatic line feed in addition to the line feed provided by The Writer's Tool (readjust M or L to avoid this).

BASIC PROGRAM LISTING IS GARBAGE

A BASIC program that is written to disk using the BASIC SAVE command has a tokenized form from which The Writer's Tool cannot produce a readable listing. You first need to LIST the program to disk from BASIC. This produces a text file which The Writer's Tool can then format and print (or edit).

PROGRAM APPEARS TO BE LOCKED-UP

It's probably waiting for a response. Most prompts require a RETURN at the end of your answer. You may have used the screen stop (**CTRL-1**) and didn't restart it (with a second **CTRL-1**). With some printer interfaces, a long pause can result when you try printing with the printer turned off. If all else fails, you can always press **SYSTEM RESET** (this won't damage your current text in memory).

UNEXPECTED PRINT FORMAT

Check Reference Section 6 to make sure that you haven't tried to use a format your printer can't print.

STRANGE CHARACTERS DISPLAYED NEAR END OF TEXT

This can happen when you are working on a text file that comes close to filling the entire text buffer. In this case, when you move the cursor near the end of the text buffer, the display will show not only the end of the text, but also the memory contents just past the end of the text buffer. This is not a problem, just a reminder that you are running out of room. Don't worry about damaging the program by typing characters into the non-text area; The Writer's Tool won't let it happen.

INDEX

G

H

I

J

K

L

R

S

THE WRITER'S TOOL
REFERENCE CARD

TO START THE WRITER'S TOOL

(1) Computer OFF, Disk Drive ON
(2) Insert The Writer's Tool Cartridge
(3) Insert The Writer's Tool Disk
(4) Turn Computer ON
(5) Wait 30 seconds for Initialization
(6) Begin Text Entry and Editing

TO SAVE TEXT TO DISK

Press **OPTION** then **D**
then **S** then **Filename**

TO LOAD TEXT FROM DISK

Press **OPTION** then **D**
then **L** then **Filename**

TO PRINT TEXT

Press **OPTION** then **P**
then **P** again

TO SEARCH FOR **phrase**

Press **OPTION** then **S**
then **/**
then type **phrase**
then **RETURN**

(**CTRL-S** to continue search)

EDIT COMMANDS

CURSOR MOVEMENT

CTRL-←,→	Cursor Left, Right
CTRL-↑,↓	Cursor Up, Down
TAB	Cursor to next 5th column
CTRL-A,Z	Beginning, End of line
CTRL-W	Next Word
CTRL-B,E	Beginning, End of text
CTRL-F,R	Page Forward, Reverse (text up, down)
CTRL-S	Continue Search

CTRL-RETURN - GOTO NEXT LINE COLUMN 1

INSERT COMMANDS

CTRL-INS	Insert a blank space
SHFT-INS	Insert a blank line
SHFT-CTRL-INS	Insert available space
SHFT-CTRL-H	Insert a header block
CTRL-U	Insert last deleted line

DELETE COMMANDS

DEL(BACKS)	Backspace & delete char.
CTRL-DEL	Delete char. at cursor
SHFT-DEL	Delete screen line
SHFT-CTRL-DEL	Delete line w/o beep
CTRL-U	Undelete last deleted line
CTRL-J	Delete space to next character
CTRL-X	Delete marked block

FONT MODIFIER INSERTION

SHFT-CTRL-E	Emphasized print on/off
SHFT-CTRL-D	Double-strike on/off
SHFT-CTRL-I	Italics on/off
SHFT-CTRL-W	Double-Wide on/off
SHFT-CTRL-U	Underline on/off
SHFT-CTRL-↑	Superscript on/off
SHFT-CTRL-↓	Subscript on/off

BLOCK COMMANDS

CTRL-M	Mark a block (insert marker)
CTRL-C	Copy a marked block
CTRL-X	Delete a marked block

CTRL-CLEAR - ERASE BLOCK MARKERS

TEXT ENTRY MODES

CTRL-T	Typeover mode on
CTRL-I	Insert mode on

CASE CONVERSION
(of previously entered text)

CTRL-L	Convert to lower case
CTRL-K	Convert to upper case

KEYBOARD COMMANDS

SHFT-CAPS	Caps-lock
CAPS/LOWR	Caps-Unlock
CTRL-CAPS	Cursor exchange on/off
◪ or ⅄	Inverse video on/off

DISPLAY COMMANDS

START	Toggle word-wrap on/off
SELECT	Change typeover cursor
SHFT-CLR	Erase status line display
CTRL-?	Restore status line

MAIN MENU FUNCTIONS

MAIN MENU ACCESS

OPTION or CTRL-O displays Main Menu

FUNCTION NAMES

SEARCH	Search, Search & Replace
DISKIO	Load, Save, Delete disk files
PRINT	Print, Link, Merge, Change
CLEAR	Erase text before/after cursor

PRINT PREVIEW

[.]Vnn[RET]	Prints to screen pages with numbers less than nn. Pages with numbers equal to, or greater than, nn are printed on the printer (enter from Print System)

PRINT FORMAT

CHARACTERS WITH FORMAT FUNCTIONS

<RET>	End of paragraph or blank line
ESC TAB	Print head to next tab column
\	Split justify
|	Soft Hyphen
■	Hard space
#	Replaced by page number when present in header or footer

COMMAND LINES(brackets excluded)

[.][format command(s)][RET]
[!][non-printing comment][RET]

[:]H[RET]	Marks start of header
[:]F[RET]	Marks start of footer
[:][RET]	End of header or footer
[@][RET]	Print pause

VERTICAL FORMAT COMMANDS

Pnn	Set page length to nn/6 inches
Sn	Set line spacing option (1=single, 2=double, 3=1.5, 4=1/2)
Bnn	Begin footer on line nn
E	Eject page
Gnn	Group next nn print lines (conditional page eject)

HORIZONTAL FORMAT COMMANDS

Fn	Select Font n (1=Pica, 2=Elite, 3=Compressed, 4,5=variable)
Mnn	Set left margin to nn spaces
Inn	Indent nn spaces
Lnn	Set line length to nn spaces
Tn,m	Set tabs at print columns n,m
Xnn	Set space between double-columns to nn spaces (nn=0 selects single-column printing)
C	Center next text line
A	Alternate sides of following split-justified line on even pgs.

OTHER FORMAT COMMANDS

D	Reset to default format
Nnn	Set first page number to nn
Wn	Set wait option to n (0=continuous, 1=single sheet)

ERRATA

This page summarizes the last minute changes and fixes which have been applied to **The Writer's Tool**.

At the start of each change description is a reference to one or more pages in the manual which may require corrections and/or additions. As with the manual, page numbers preceded by R apply to the reference manual; those preceded by T refer to the tutorial. Other letters refer to appendices.

1. R-16 In the description of the **OPTION** key, strike the first two words, "In EDIT". The **OPTION** key will now cause the MAIN MENU to be displayed even if it is pressed while in another sub-menu.

2. R-10 Add a new command which affects block commands:
   CTRL-CLEAR Erases all block markers.

3. T-24, R-20, X-2 Since the **OPTION** key is now always used to enter the MAIN MENU, Directory Printing is accomplished with the SELECT key instead. Replace the appropriate references to **OPTION** with **SELECT**.

4. T-27, R-21 Normally, the **SAVE** command (from the DISKIO menu) will only save text from the current cursor position to the end of the file. The manual implies that you must return to **EDIT** mode to move the cursor if you start a **SAVE** and then discover that the cursor is not at the beginning. **Not so!** If you answer 'N' to the "Continue" query (after the "Warning: only part ..." message), you will be presented with another option: "SAVE ALL TEXT (Y/N)?". If you answer 'Y', the cursor is automatically moved to the beginning and all text is SAVEd. (If you answer 'N', you are returned to EDIT.)

5. R-6 Another cursor movement command is available:
   CTRL-RETURN moves the cursor to the beginning of the next text line. It is equivalent to using CTRL-A followed by CTRL-↓.

6. R-20, T-25 Although the menu displays only choices 1 and 2 as requests for a directory listing, **The Writer's Tool** will recognize drives 3 and 4 (for SAVE and LOAD as well as directory). However, the disk as shipped from OSS is configured for only two drives. If you have 3 or 4 drives, you will have to copy all files to a disk already initialized and configured with a DOS which supports the additional drives. Presumably, if you have 3 or 4 drives you have a copy of such a DOS already. Be assured that **The Writer's Tool** works with virtually any DOS.

7. T-27, R-21,22 The manual implies that only the last file name by which you **LOAD**ed a file is remembered for use by a subsequent **SAVE**. Not true. If you **SAVE** a file using a different (or new) name, that name is remembered instead. This makes editing new files, etc., easier.

8. T-76 to T-82 The scheme used my The Writer's Tool for merge printing is compatible with that used by SYNFILE (from Synapse Software). To prepare a file for use with **MERGE**, follow the instructions in the SYNFILE manual for doing the same thing with AtariWriter. Sophisticated SYNFILE users may also opt to use the repeated items capability of **MERGE** (section 9.8) and/or the non-printing comments feature (section 9.6). Better documentation of this feature will be available soon.

BUGS

There is only one known significant bug in **The Writer's Tool** at this time: Double Wide printing is not properly supported only when a line is justified. In particular, whenever a double-wide space is inserted in the justification process, its size is miscalculated. Double Wide does work when the format specifies .J0, when a headline is centered (.C), or when a header/footer is split (.A). We are working on a fix for .J1 mode.

FUTURE FEATURES

Actually, these features are already present in **The Writer's Tool**, itself, but the support programs needed to easily use them are not yet ready. If you are reading this ERRATA sheet, you are entitled to one **FREE UPDATE.** Simply send back in your warranty registration card, and we will send you the update (with new manual pages) when it is ready. Anyway, here is what you can look forward to:

1. For printers with graphic capabilities, you will be able to define certain characters to be translated to graphic sequences. For example, you might have CTRL-C print out a copyright symbol. This is simpler than the method outlined in the manual.

2. You may "linkprint" graphics files. Any file with an extender of ".GGG" is assumed to be a "graphics" file which consists of (1) a single byte (which specifies the number of half-lines occupied by the graphics "picture") and (2) an undefined number and type of bytes which will be sent, as is, to the printer. If you are able to write a program to produce such a file, you may experiment with this capability now. If not, we will have conversion programs available to translate from MicroPainter, Koala Pad, Atari Artist Pad, etc., files to the more popular printer formats.

THE WRITER'S TOOL

The most natural and complete word processor available for ATARI Home Computers!

Optimized Systems Software Inc.

1221 B Kentwood Avenue San Jose, CA 95129