

By Mitchell White and Julie Arca

- Microcomputer and Personal Computer Oriented
- Mitch's Market Sample of Word Processing Programs
- FRED—FiRst EDitor Program Listing in Microsoft BASIC
- Spelling, Word Index, Mail Merge, and Grammar Programs

WORD PROCESSING PRIMER

by

Mitchell Waite and Julie Arca

BYTE/McGraw-Hill
70 Main Street
Peterborough, NH 03458

Copyright © 1982 by Mitchell Waite and Julie Arca. All rights reserved. No part of this book may be translated or reproduced in any form without the prior written consent of the authors and McGraw-Hill, Inc.

The authors of the program provided with this book (FRED) have carefully reviewed it to insure its performance in accordance with the specifications described in the book. Neither the authors nor BYTE/McGraw-Hill, however, makes any warranties whatever concerning the program. They assume no responsibility or liability of any kind for errors in the program or for the consequences of any such errors. The authors have generously released all rights to this program.

Edited by Bruce Roberts.

Design and Production Supervision by Ellen Klempner.

Drawings by Bob Johnson.

Production Editing by Tom McMillan.

Typesetting and Production by LeWay Composing Service, Fort Worth, Texas.

Printed and Bound by Halliday Lithograph Corp., Arcata Company, North Quincy, Massachusetts.

23456789 HDHD 89876543

Library of Congress Cataloging in Publication Data

Waite, Mitchell.

Word processing primer.

Includes index.

1. Word processing (Office practice)

I. Arca, Julie. II. Title.

HF5548.115.W34 1982

652

82-9534

ISBN 0-07-067761-1

AACR2

To Jim Madden

. . . who showed me I could write

Mitchell Waite

ACKNOWLEDGEMENTS

Any book worth its stuff rides on the shoulders of many people. This book is no exception. One person of extreme importance to this work is Annie Fox. She turned an interesting but somewhat rambling manuscript filled with perhaps too much detail into a book that was engaging, personal, and really exciting to read. Her way with wordsmithing is extraordinary and I am deeply grateful for her efforts. I would also like to thank Scot Kamins for his revealing, frank, and perceptive review of the first draft. It caused me to rethink the organization of the book and led to a smoother and much more practical presentation. I would also like to thank David Fox for his valuable comments on the manuscript, but especially for his support in areas one would not think necessary for finishing a book. I also would like to thank Henry Dakin for his continual support in all my writing efforts. Finally the authors would like to thank the following individuals and the companies they represent for the assistance, reviews, materials, equipment, and software they provided.

Mitchell Waite

Fred "Chip" Pood, MicroPro International
Seymour Rubenstein, MicroPro International
Martin Dean, Select Information Systems
Perry Gee, Lexisoft, Inc.
Charles P. Rosenzweig, Exxon Office Systems Company
Paul W. Harris, NBI, Inc.
Susan M. Storey, Lanier Business Products, Inc.
Richard Frank, Akins Office Machines
Joanne Abbot Green, Geltzer & Company, Inc. (for Sony)
Don Novak, AM Jacquard Systems
Bill Baker, Information Unlimited Software
Dennis Coleman, Innovative Software Applications
Bruce Wampler, Aspen Software Company

TABLE OF CONTENTS

1. PERSPECTIVES 1

Low Cost Word Processing Is Here 5

What Are They? 5

What Can They Do? 6

A Little History 7

Where Are They Headed? 18

How To Use This Book 19

2. BASIC CONCEPTS: EQUIPMENT AND PROGRAMS 21

The Parts Of A Word Processor 21

The Character and the Byte 24

Memory 26

Screen Display 27

Keyboard 30

Permanent Disk Storage 31

Printer 33

Microprocessor 36

Interfaces 37

Word-Processing Program 38

Dedicated vs. Micro-Based Word Processing 40

Summary 43

3. BASIC EDITING 45

Cursor Control 45

Screen Scrolling 49

Insertion 53

Deletion Commands 54

Paragraph Reform 56

4. TEXT FORMATTING 57

The Word Processor Screen Layout 58

Word Wrap 60

On-Screen Reformatting 62

Justification 65

Hyphenation 67

Embedded Printing Commands — Off Screen Formatting 68

Summary 78

5. ADVANCED EDITING 79

<i>Search and Replace</i>	79
<i>Global Search-and-Replace Operations</i>	81
<i>Block Operations</i>	82
<i>Boilerplate Techniques</i>	84
<i>Summary</i>	85

6. WORD PROCESSING SOFTWARE TOOLS 87

<i>Spelling Correctors</i>	88
<i>Indexers</i>	95
<i>Table of Contents Generators</i>	96
<i>Mail Merge</i>	97
<i>Automatic Grammar Checking</i>	101
<i>Word Profiler</i>	106
<i>Graphics Figure Creation</i>	107
<i>Summation</i>	114

7. HOW TO SELECT A WORD PROCESSOR 115

<i>Attorneys</i>	117
<i>Secretaries</i>	117
<i>Mass Mailers</i>	118
<i>Authors and Technical Writers</i>	120
<i>The Home User</i>	121
<i>Seven Laws of Word Processor Purchasing</i>	122
<i>Memory-Based vs. Disk-Based</i>	123
<i>On-Screen Formatting vs. Off-Screen Formatting</i>	125
<i>Display Size</i>	126
<i>Disk Capacity and Number of Drives</i>	127
<i>Quality and Quantity of Editing Commands</i>	128
<i>Help Menus</i>	129
<i>Documentation</i>	130
<i>Know Your Bottom Line</i>	131

8. HOW TO BUY A WORD PROCESSOR 133

<i>Financing Your Word Processor</i>	133
<i>Where To Buy Your Word Processor</i>	137

**9. MITCH'S MARKET SAMPLE OF
WORD PROCESSING PROGRAMS 143**

<i>Introduction</i>	143
<i>Definitions</i>	144
<i>Product Overviews and General Information</i>	146
<i>Word Processing Program Comparison Charts</i>	151

Appendices**A. Word Processing Buzz Word Dictionary 155****B. Record Processing 165***What Is It?* 165*An Application* 165*The Market* 166*A Typical Design* 167*So What Is A Record Processor?* 169*Word + Record Processor + Mail Merge + Sorter* 169**C. FRED: Your FiRst EDitor Program 171***Hardware Requirement* 171*Program Design* 171*Operation of FRED* 172*Summary* 175**Index 185**

PREFACE

A new breed of inexpensive, microcomputer-based word-processing program has recently appeared that was previously unavailable to the general public. These programs transform any microcomputer into a powerful text-editing device that allows precision control over the creation and alteration of text documents. Many of these new products outperform dedicated word processors costing ten times as much. This book is all about these revolutionary, inexpensive microcomputer-based writing tools and how to use them for both fun and profit.

Word processing is the New Age method of generating, correcting, and managing text. This includes documents of all kinds, such as theses, articles, tables, manuscripts, reports, business letters, brochures, and newsletters. Word processing allows you to manipulate text electronically in ways which are impossible with a typewriter. Imagine being able to insert or delete text in the middle of a paragraph, then instantly transport the entire paragraph from one page to another. Word processing can do this and more.

This book, which is carefully organized into nine chapters and three appendices, was written on a word processor. It will teach you everything you need to know about this exciting field.

Chapter 1, "Perspectives," presents a survey of home computer word processing. It explains, in a simple dialogue between two friends, the incredible things that people are doing at home with their word processors. Included here is a comprehensive history of word processing, tracing the roots from which today's powerful machines have evolved.

Chapter 2, "Basic Concepts: Equipment and Programs," introduces the general hardware and software concepts behind word processing on a microcomputer. Described and explained are the components of a word processor, the different types of programs available, and the kinds of features that various word-processing peripherals, such as printers and terminals, offer the user.

Chapter 3, "Basic Editing," begins the exploration of the actual operation of the word-processing program — how it does what it does. Taught here are the basic concepts of cursor control, scrolling, insertion, deletion, and reformatting. These are your first flight instructions, and when you've finished this chapter you'll be well on the way to earning your word-processing wings.

Chapter 4, "Text Formatting," teaches the subject of *formatting* — controlling how your final document looks on paper. Covered are on-screen and off-screen formatting, screen layout, justification, proportional spacing, paragraph reform, word wrap, hyphenation, and embedded print commands. We'll clear up the mystery of how the word processor achieves its fine quality printed appearance.

Chapter 5, "Advanced Editing," presents the more powerful editing commands available to the word processor user. We'll teach you everything you ever wanted to know about character and word search, replace, global search and replace, block operations, and boilerplate techniques.

Chapter 6, "Word-Processing Software Tools," covers power-enhancing tools you can buy for your word processor. Discussed are programs that automatically proofread documents for spelling errors, create

tables of contents and indexes, and generate personalized form letters from a mailing list. You'll learn about graphics-figure creation (drawing on the word processor) and grammar-checking programs that can actually improve your style as you use them.

Chapter 7, "How To Select A Word Processor," helps you pinpoint features that are important in a word processor for your particular profession. We discuss the individual word-processing needs of secretaries, attorneys, authors, mass mailers, and office managers. You will learn the seven most important features to look for when considering a word processor and what today's two most popular types of programs are.

Chapter 8, "How To Purchase A Word Processor," explains how to buy a word processor and includes many tips and tricks for making a safe purchase as well as a sound investment. A true-life fable about a person who bought a word processor the wrong way is presented just to keep you sober and alert.

Chapter 9, "Mitch's Market Sample of Word-Processing Products," presents a low-calorie selection of personal computer-based word-processing programs that are on the market today. The products are arranged by price and represent a wide range of capabilities. A detailed chart is provided for quick feature comparison, as well as a summary sheet of general information pertaining to each program.

The appendices contain a glossary of word-processing buzz words and a detailed account of record processing. (Record processing is one of the fastest growing applications for harnessing the word processor in the office of the future.) In addition, there is a complete word-processing program called FRED that you can type into any personal computer or microcomputer that uses a version of Microsoft BASIC. FRED affords you the unique opportunity to "get inside" a word-processing program and take a look around.

There is an impressively active climate in the personal computer marketplace today. The use of personal computers for word processing is perhaps the most powerful application of the computer to date. There now exists a huge selection of highly sophisticated inexpensive text-editing tools for the home and small business user. We hope that through this book the reader will come to harness these tools for his or her own enjoyment and profit.

Mitchell Waite
February 9, 1982

WORD PROCESSING PRIMER

Perspectives

Bob's fingers danced across a typewriter keyboard as if he were playing a piano. In front of him, a computer's video screen glowed a bright electric green as it filled with letters.

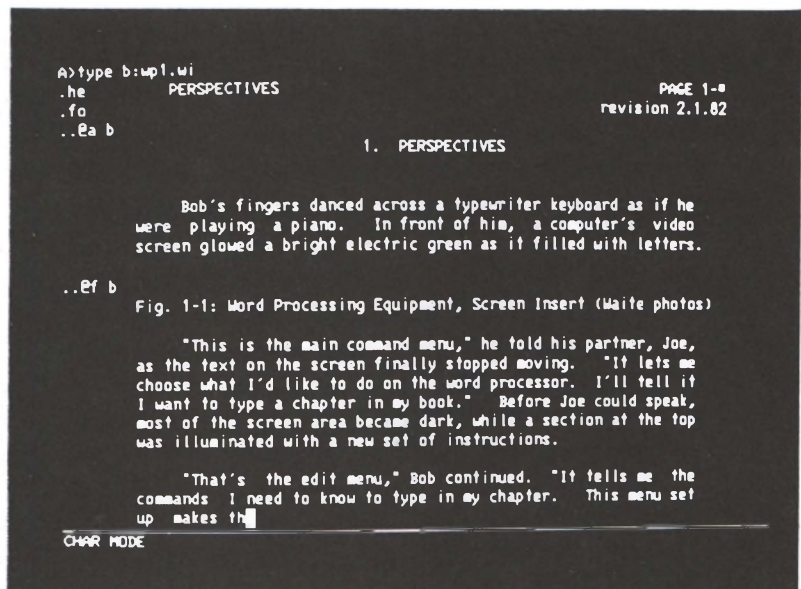


Figure 1.1: Screen of word processor at work.

"This is the main command menu," he told his partner, Joe, as the text on the screen finally stopped moving. "It lets me choose what I'd like to do on the word processor. I'll tell it I want to type a chapter in my book." Before Joe could speak, most of the screen area became dark, while a section at the top was illuminated with a new set of instructions.

"That's the edit menu," Bob continued. "It tells me the commands I need to know to type in my chapter. This menu utility makes this a user-friendly system, but it still took me some time to learn to operate."

Bob started typing on the keyboard and his letters instantly appeared in the screen's dark area. As he typed, the letters seemed to

march eastward, a tiny box of light preceding them. When the letters reached the screen's right edge, Bob confidently kept on typing. As if by magic, the letters suddenly appeared on the left side of the screen, one line down. Bob never pressed the carriage return key!

"That's called word wrap. I can type continuously without looking at the screen and the program will take care of everything."

"Amazing," Joe admitted.

After typing in a few sentences, Bob stopped. "See all those errors I've made in spelling and typing?" he asked. "This is where the word processor really shines. Watch how easily I can fix those mistakes."

Before Joe's eyes, Bob began to use the keyboard in a new way, sending the little box of light gliding backwards through the sentences. Finally, the box stopped at a word that was missing a letter. Bob typed the letter and the entire text moved over to accomodate it! Then he moved the box between two sentences and added a new sentence. Once more everything shifted to the right and down to make room for the new text.

Bob continued to type and edit in a furious fashion. Words spilled from his fingers and poured onto the screen like he was painting them in place. He would reverse the path of the box of light and start typing new text, then turn it around and delete a letter, word, or entire sentence in seconds.

"Looks like you spent \$3,000 for a very fancy eraser," said Joe, who was really at a loss for words.

"Not really, Joe. It's so much more than that. Watch this. See how my paragraph is uneven from all my corrections and insertions? Now if I simply press the REFORMAT key. . ." The disorganized paragraph suddenly shimmered and reorganized itself on the screen — right and left margins were perfectly straight, like a newspaper column. "Incredible!" exclaimed Joe.

```

B:WP1.WI PAGE 3 LINE 24 COL 33          INSERT ON
L-=====R
"As a matter of fact, yes. This is the piece de
resistance." Bob reached down under the computer and placed
his telephone receiver in a special cradle. "With this device
I'm plugging the computer into the telephone line, and I can
send my text over the phone lines to another computer. Now I'm
sending it to my coauthor, who lives 48 miles from here. By
mail it takes about two to three days, not counting the time to
package the chapter properly,
pay the postage and go to the
post office. This way my text goes over the phone and is
stored on his disk. I pay for the transmission like it was a
conversation and it appears on my telephone bill. My co-author
can print it out on his printer or edits it while it's in the
computer.
He can send it back to me the same way. We have
given up buying stamps and insurance, weighing packages,
storing packing materials, shipping magnetically sensitive
floppies, etc. Still have to send the artwork by mail, but
that soon will change too."

Welcome to the office of the future in the home of the
CHAR MODE

```

(a)

```

B:WP1.WI PAGE 3 LINE 33 COL 01          INSERT ON
L-=====R
"As a matter of fact, yes. This is the piece de
resistance." Bob reached down under the computer and placed
his telephone receiver in a special cradle. "With this device
I'm plugging the computer into the telephone line, and I can
send my text over the phone lines to another computer. Now I'm
sending it to my coauthor, who lives 48 miles from here. By
mail it takes about two to three days, not counting the time to
package the chapter properly, pay the postage and go to the
post office. This way my text goes over the phone and is
stored on his disk. I pay for the transmission like it was a
conversation and it appears on my telephone bill. My co-author
can print it out on his printer or edits it while it's in the
computer. He can send it back to me the same way. We have
given up buying stamps and insurance, weighing packages,
storing packing materials, shipping magnetically sensitive
floppies, etc. Still have to send the artwork by mail, but
that soon will change too."

Welcome to the office of the future in the home of the
present.
CHAR MODE

```

(b)

Figure 1.2: Screen before (a) and after (b) reformatting.

"Want to see more?" Bob acted like a used-car salesman ready for the kill. "Earlier today I worked on a ten-page section of a manual I'm writing. My mind was elsewhere and I made a lot of spelling errors. But now watch how I deal with those mistakes." He typed the word "SPELL" followed by the name of the chapter of the manual.

“Spell is a program that will automatically check the spelling in my chapter for me. Not only will it show me what it thinks are misspelled words, it will also guess at the proper spelling, then let me alter these words as I see fit. It can proofread 100 pages in under three minutes! Any word that it doesn’t already have in its built-in dictionary can be added and will be remembered for future use. It learns words that I frequently misspell and automatically corrects them without even asking me!”

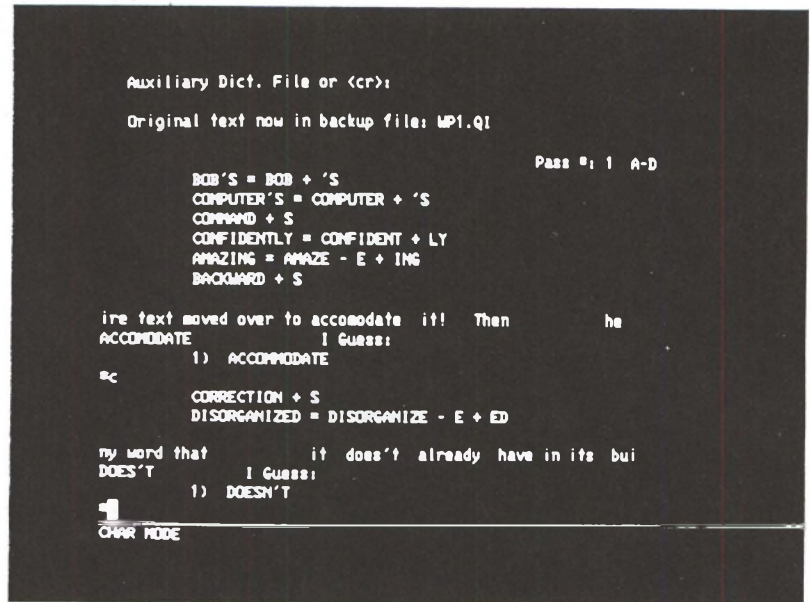


Figure 1.3: Screen during spelling check.

Joe was entranced as Bob instructed the Spell program to check his manuscript. The results had an uncanny accuracy.

“Now for a real treat I’ll print out the section on my daisy-wheel printer,” Bob explained. “See how the text looks like it’s typeset.”

Bob pushed a few keys and before you could say “punch that printer,” a typewriter device next to the terminal began to spit out the same text that was on the screen. But the copy on the printer was beautiful and loaded with professional touches: underlining, bold fonts, superscripts, and page numbers appeared automatically.

“Notice how balanced the text looks,” said Bob. “That’s because the printed information is microjustified. The spacing between letters is controlled to 1/120 of an inch so that it appears much like the proportionally spaced text you find in books, magazines, and newspapers. This gives a professional appearance.”

“And that is just the beginning,” Bob stated with the confidence of someone who knows he has a captive audience. “Now I’m ready to create my index, automatic table of contents, and heading numbering. I have an Index program that takes all the pain out of renumbering manuals every time I write a new section. It also generates an automatic index for me to send along with the manuscript. My publisher and the company that commissions me to write manuals are extremely pleased that I can work faster and more accurately.”

“Anything else you can show me?” Joe timidly asked.

“As a matter of fact, yes. This is the *piece de resistance*.” Bob reached down under the computer and placed his telephone receiver in a special cradle. “With this device, I’m plugging the computer into the telephone line. Now I can send my text over the phone lines to another computer. I’m sending it to my coauthor, who lives 40 miles from here. By mail, it takes two or three days, not counting the time to package the chapter properly, pay the postage, and go to the post office. This way, my text goes over the phone and is stored on his disk. I pay for the transmission like it was a conversation, and it appears on my telephone bill. My coauthor can print it out on his printer or edit it while it’s in the computer. He can send it back to me the same way. We have given up buying stamps and insurance, weighing packages, storing packing materials, and shipping magnetically sensitive floppies. Still have to send the artwork by mail, but that soon will change, too.”

Welcome to the office of the future in the home of the present.

LOW COST WORD PROCESSING IS HERE

If this kind of talk strikes you as too strange to be of this world, then guess again. Conversations like this take place at the author’s home regularly. The word-processing system described above can be purchased today for under \$3000, and you can learn to use it in a few hours. A word processor can completely transform the way you relate to the printed word. It can help you become a full-fledged member of the New Information Age.

This book is about the new wave of word processors, particularly those that utilize the new, inexpensive microcomputers. *Word processing* is the manipulation, alteration, and creation (processing) of text (words) by electronic, mechanical, and other computer techniques. It’s about to rock the world like an explosion. If you are interested in utilizing this powerhouse technology, then this book is for you.

WHAT ARE THEY?

If you have ever slaved over a typewriter correcting a sentence until it was almost but not quite perfect, then you will want to explore the word processor. Word processors are nothing more than intelligent,

computerized typewriters. They have been around for years, but because of the recent developments in computer-on-a-chip technology, they have made quantum leaps in price and performance. Word processors allow people to control communications in new and powerful ways, to manipulate words and text quickly and efficiently, and to increase written output many times over. This book is about these new, powerful tools — what they are, how they work, how to tell if you need one, and how to buy one.

Today's word processors come in many shapes and sizes. At first glance, the field might even seem overwhelming. There are a few major types, however, and being familiar with them will greatly simplify the selection. In this book we will discuss four kinds of word processors, with the main emphasis on the inexpensive, microcomputer-based word processors that are sweeping the country.

In all cases you will discover that there is a general range of universal functions that all word processors perform. Our introductory scenario mentioned a few of the word processor's functions without really explaining how they are performed. We'll get to all of that later. For now, you should understand that most word processors have a common repertoire of operations and a number of ways to perform them. It is more important to be aware of the range of possibilities than of the specifics of "how does it do that?" Like automobiles, all word processors accomplish the same goal, but unlike the automobile, they may use several different approaches to do it. For example, one word processor (like the one at Bob's house) may use a cathode ray tube (CRT) screen to display text, and another may use a single line of solid-state displays. Or, while one word processor might use an expensive daisy-wheel printer for printing your text, another may use an inexpensive dot-matrix printer. On the operation side, various key sequences and buttons are defined differently for each of the word processors, so knowing one system doesn't necessarily help you to use another. The point is that this is a dynamic and growing field, and you are going to need to do some homework to travel confidently in the land of word processing. This book is designed as a starting point for your journey.

WHAT CAN THEY DO?

If you have a word processor, you can throw away the liquid correction fluid, correction tape, scissors, erasers, paper clips, scotch tape, and all the old tools for dealing with printed words on the paper. You now deal with words inside an electronic memory. They can be moved and changed easily, quickly, and efficiently, allowing you to type in your words without thinking about the appearance of your finished product. It is as if an assistant were at the other end of your

typewriter keyboard, collecting the letters and words as they come in. This assistant corrects the words as needed, stores them in a safe place, and simultaneously displays them for you on a glass screen.

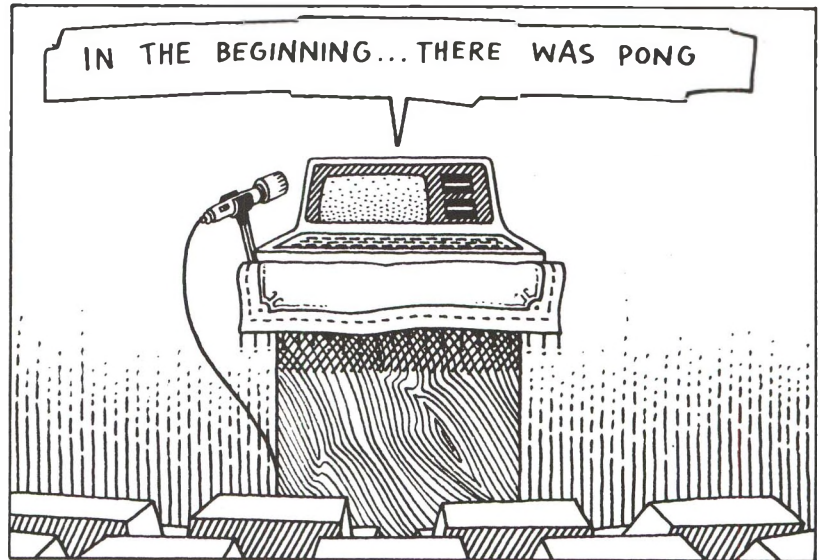
A computer-controlled word processor permits you to turn a ragged right margin into a beautiful, professional-looking typeset format with the push of a button. Or, suppose you want to add a new sentence in the middle of your many-paged document. No problem. Simply enter the Insert mode and start typing away. Because a word processor has an electronic computer brain on a chip, it can instantly shift all the text in its memory forward to make room for new words and do it without losing its paragraph structure. You spelled the same word incorrectly in 15 places. That's quick work for your word processor. Merely enter the Search-and-Replace mode, and the computer will automatically change all instances of the misspelling. You just sit back comfortably and sip on your *café mocha*. Other things you can do with one-key operations include moving paragraphs and blocks of text, creating boldface, underlining, and inserting superscripts and subscripts.

Today's word processors allow you to do things that would require days of work using a standard typewriter: reformatting an entire document from double-spaced to a single-spaced narrow column suitable for publication (30 seconds); printing 25 individually addressed copies of the same letter (10 minutes); generating an index (30 minutes); sorting and merging 1000 mailing labels (5 minutes); and proofreading and correcting a 100-page document (60 minutes).

This kind of power means that you will never experience the written word in the same way again. With very little effort you can drastically change the way words appear on paper. Your text has a new malleability. It molds like wet clay. Over and over you rework the same sentence, deleting old words with hardly a care, until the words say exactly what you mean. Because of the fluid nature of the work, you become braver. Your creativity tends to flow more freely because you're not concerned with how the typing looks. Nor are you hampered by the fear of putting an idea in the wrong place. You can allow the ideas to emanate freely, for there is no effort involved in rearranging them later. You do the creating and let the machine manage the words.

A LITTLE HISTORY

The history of the word-processor market is unique. No other product on the market today can boast of such a rapid increase in performance over the last few years. Yet the arrival of word processors and text-editing machines is long overdue. Although great strides have been made to increase productivity in blue-collar industry, the white-collar office environment has seen little improvement in



efficiency over the past 100 years. The introduction of the word processor is creating major changes in productivity in offices everywhere.

The roots of the word processor can be traced back to a comparatively crude mechanical invention called the typewriter. The typewriter revolutionized the office in the early 1900s, because until that point, all correspondence was handled by manual pen-and-ink copying. A pen-and-ink copier would take hours to produce a single letter, and even then copy might be very difficult to read as the individual was usually overworked. The mechanical typewriter allowed the average typist to out-perform the most talented copier.

But as far as quality is concerned, early typewriters were barely an improvement over hand-written documents. Letters were fuzzy and frequently smudged since characters were often poorly aligned, and the keys frequently jammed. Better quality typewriters existed, but these were hopelessly large and expensive and required extraordinary skills to operate. (See Figure 1.4.)

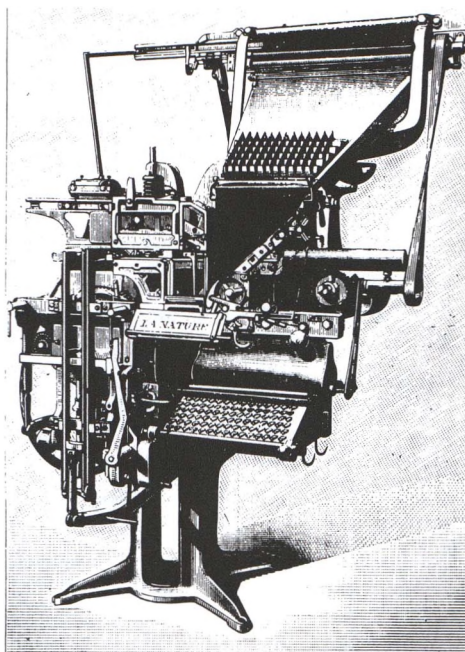


Figure 1.4: *One of the first word processors (c. 1880).*

The introduction of the automatic typewriter in the 1930s created a major increase in letter processing. These beasts worked somewhat like old player pianos. They used as a storage medium a roll of punched paper that contained marks indicating what letters to type. The machines could repetitively type form letters, contracts, and journals, making the copying process much more accurate. In the 1950s, the same automatic typewriters were improved by replacing the punched paper roll with narrower paper tape and adding electrically driven mechanical logic that could not only punch and read these tapes, but also operate more than one reader at a time. This latter feature allowed the accomplishment of tasks involving the selecting and merging of text previously stored on paper tape.

The real jump in performance, however, came with the invention of the inexpensive transistor and the introduction of magnetic recording materials in the 1950s and 1960s. Solid-state devices were developed to perfection during the race to land a man on the moon. Transistors, integrated circuits, tantalum capacitors, and magnetic tapes soon replaced large relays, delicate vacuum tubes, and bulky paper tapes in the new word-processing devices. Yes, they were still expensive, but the performance of this new technology drove thousands of businesses to invest heavily in word-processing equipment in hopes of raising efficiency and increasing profits.

In 1964, there was a stunned silence all over the world when IBM introduced the first Magnetic Tape Selectric Typewriter (MT/ST), a device that utilized a tape cartridge for its storage medium. (See Figure 1.5.) We call these “semi-intelligent typewriters”. In many cases, depending on variables such as the volume of work to be done and the time available to train operators, the MT/ST makes more sense than the microcomputer-based system. This single product was responsible for a monumental jump into the age of sophisticated word-processing equipment. The magnetic tape of the MT/ST was superior to paper tape in many ways. It could hold more information per unit of length (25 characters per inch versus 11 for the paper tape of that era). Magnetic tape was also capable of being erased and re-recorded, whereas punched paper was not. Also, the magnetic tape moved considerably faster than paper tape and therefore the word processor could search more quickly for information on it. Finally, you could handle and store the cartridges much more easily than paper tape, resulting in less potential damage to them.



Figure 1.5: *First Magnetic Tape Selectric typewriter (courtesy IBM).*

In 1969, IBM introduced its Magnetic Card Selectric Typewriter (MC/ST). In typical conservative style, IBM clung to the word typewriter. They didn't want to frighten the office manager out of purchasing their new-fangled contraption. The Mag Card Selectric

Typewriter featured a reusable magnetic card that could record a single page of text. This one-card-for-one-page arrangement made it easy for secretaries to relate to the device and gave IBM a great boost in profits. Suddenly, IBM had a market entirely to itself. American business saw a challenge. Two years later, several other companies introduced similar machines. By then, however, IBM had such a grasp on the market that it was impossible for another single company to outdo them.

Although the correcting Selectric could store text on tape or magnetic card and allowed insertion and deletion of text, its editing intelligence was severely limited by the transistor technology of its electronics. The amount of text that could be stored in memory was limited, speed was poor, and operation was a bit too complex for most users. Yet fortunately for the competition, the semiconductor technology on which these word-processing machines was based was rapidly decreasing in cost and growing in capability. It was becoming easier to give better performance in a smaller volume. The next move was to endow the Selectric typewriter with greater digital intelligence. The result was the true “intelligent typewriter” shown in Figure 1.6. This product had increased memory, more editing features, greater speed, and better user interface. It was easier to use, more reliable, and less expensive than the early semi-intelligent typewriter!



Figure 1.6: *Intelligent typewriter (courtesy Exxon).*

The Seventies began with another round of corporate battles as companies such as Lexitron and 3M introduced the first video display systems. The video display offered much faster editing than was ever possible on the standard paper and printer-ball technology of the

Selectric typewriter. In 1973, Vydec introduced the first floppy diskettes for use with a word processor, further increasing the performance of these products. With the floppy diskettes replacing the tape cartridge, vast amounts of text could be saved almost instantly, and information could be searched for and located within seconds. (See Figure 1.7.)

The video word processors were rather large affairs with price tags to match. They had limited line width (only 40 to 64 characters on a line) or limited line capacity (perhaps only 16 lines on the screen at once). The early video units were difficult to master, somewhat slow to respond, and hard to set up. Once they were running, however, they could edit circles around the intelligent typewriters. The screen allowed instant user response, whereas the intelligent typewriter or mag card made you wait for the typewriter to type the new text out on paper before you could verify it. Furthermore, it was possible for screen-based word processors to provide users with *on-screen help menus* — instructions that appeared next to, but not in the way of, the text you were typing. On-screen menus certainly made it easier to master the word processor.



Figure 1.7: Early video-based word processor (courtesy Vydex).

Screen-based word processors proved to be superior text-editing products compared to the intelligent typewriters. The market for them increased. During this decade, most of the major dedicated word-processing companies which still exist got their starts.

Word processors were becoming more compact and easier to use and sported the look of a modern appliance, as seen in Figure 1.8. It was clear that transistor technology was improving the word processor at a faster rate than any product before.



Figure 1.8: *More recent video-based word processor (courtesy Wang).*

In the mid-seventies, the world was presented with an extraordinary gift from the transistor wizards — the microprocessor. This amazing device was a computer on a chip — a complete “brain” in a small, inexpensively packaged electronic module. The microprocessor was quickly incorporated into word-processing products and the result was a higher level of performance at a much lower price.

The new dedicated word processors based on this microprocessor appeared in the late Seventies. They were designed on the premise that more integration and more compatibility was needed. The integrated word processor would have the printer, keyboard, screen, and storage devices all in one cabinet. (See Figure 1.9.) These devices eliminated the need for cables between the equipment modules. The result was decreased cost (fewer cabinets), increased reliability, and higher performance.



Figure 1.9: *Dedicated word processor, late 70's (courtesy Lanier).*

The computer on a chip and the low cost random-access-memory (RAM) devices led to the development of the personal, or home, computer. Although these devices were somewhat slow and possessed less capacity than the big computers of the day, they were fully capable of processing words just like the dedicated word processors. It was therefore only a matter of time before independent software companies began to offer text-editing programs that would turn your home computer into a full-blown word processor. The ED editor program (Digital Research) and the Electric Pencil (Michael Shraye Software), for example, were among the first word-processing programs to hit the market. This caught the large word-processing firms off guard. Their first response was, “Personal computers — ha! Why, they’re nothing more than toys. They can’t do what our big, expensive word processors can do.” This complacency encouraged smaller companies to produce superior software products for the growing number of personal computers and get the jump on the large, established corporations.

Finally, the personal computer manufacturers themselves got into the act and began producing their own word-processing software. But several companies had a foot-hold in the market by then, including MicroPro (WordStar) and Radio Shack (Scripsit).



Figure 1.10: *Personal computer word processor.*

Almost every major personal computer manufacturer offers a word-processing program for its computers today. This book focuses on this part of the word-processing industry. However, it would be negligent to stop here in our history. There is more. As one might expect, the dedicated word-processor companies didn't take the loss of sales lightly.

In the early 1980s, we began to see a surge of dedicated word-processing equipment with increased power and capability. This equipment featured ideas developed in the home computer markets and utilized the latest microprocessor technology. Some of these devices were designed so an office could simply plug a Correcting Selectric into the word processor and use its keyboard!

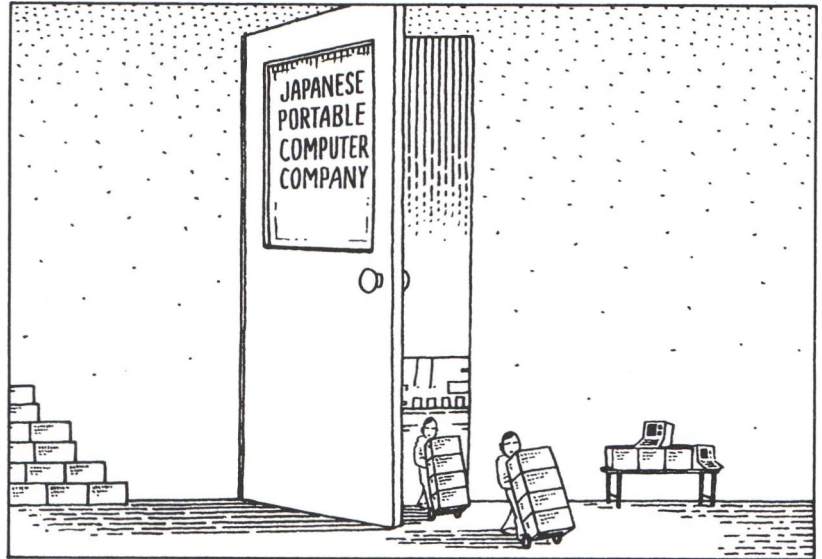
As you can see in Figure 1.11 these new products were becoming compact, streamlined, and conveniently designed to reduce strain on the user.



Figure 1.11: *Contemporary dedicated word processor (courtesy Syntrex).*

Today's major word-processing manufacturers are playing a catch-up game to bring out products that can compete with the less costly personal and microcomputer-based products. Most of the new word-processing systems are being offered with CP/M (the most popular disk operating system for microcomputers). It seems that we are at the point where the large computer corporations are finally jumping in, floppy disks first. Obviously, a major battle is brewing between the large, dominant companies and the smaller personal computer and microcomputer companies. Who will win? Everyone. For the consumer, this will mean a heyday in equipment deals. Equipment will keep getting better and better for the money. Should you simply wait until tomorrow for the ultimate word processor? You can, but while you're waiting your friends will already be doing unprecedented things with word processors.

Things don't end quite yet. During the period that the word-processing equipment was breaking records in sales, the office-copier market was being lost to the Japanese (along, of course, with the television, automobile, cotton, and steel markets). Everyone knew it was just a matter of time before Japan entered the word-processing equipment race. Major Japanese companies watched the U.S. market and drew up their plans for increased performance, smaller packaging, and greater portability — a consistently winning strategy for them. They spent research money on the miniaturization of



the various components of a word processor and it paid off. As Figure 1.12 illustrates, they have come up with a very small system.

One new Japanese word processor uses 3-inch floppy wafers rather than 5-inch floppy disks. These store the same amount of information as the larger units, but are more reliable and less likely to be damaged. The word processor itself is also very small. Next to the location for the two disk wafers is a spot where a microcassette plugs in. The microcassette, in turn, plugs into a battery-operated, hand-held keystroke-capture device. The screen for the unit shows an entire page of text in black letters on a white background.



Figure 1.12 Japanese word processor (courtesy Sony).

WHERE ARE THEY HEADED?

There are many more changes in store for the users of word processors. In terms of capability, let's take the word processor to its logical evolutionary conclusion. Elimination of the keyboard is the next step. Voice-recognition technology is advancing rapidly at this time, and it is quite likely that one day we will simply talk into a word processor and have our speech converted into text on a screen. Such a device will probably also be able to recite a manuscript too, as speech generation is already a reality. (See the Texas Instruments Speak and Spell toy for an example of current speech-synthesis technology.)

The word processor of the future will probably still have a screen, but it will be a thin, folding affair. You will be able to simply touch the screen to indicate changes you want in your text. We will probably see more of these keystroke-capture devices on the market, followed by personal computers which can accept the captured keystrokes and process them more efficiently.

We can safely assume that American households will become equipped with enough computers to rival the number of televisions and telephones. There are presently about 600,000 personal computers (under \$2,000) sold every year as opposed to 10 million televisions. Sound like computers still have a ways to go to catch up? Yes, but maybe not too far, because the personal computer market is doubling and tripling almost every year, while the television market grows at a rate of only about 10 percent per year.

Such a large user base will encourage immense services based on the transmission of text information. For example, soon an entire encyclopedia will be available to those with computers in their homes. At that point, your word processor will begin to tie into the world data base in marvelous ways. Since the word processor will allow instant access to words written by millions of other sources, you will be able to move vast quantities of information into your personal data base with ease. It is likely that your computer will be smart enough to generate a personal data base according to your wishes. It will tie into the world data base, searching for just the information you want. You can sit in your armchair and fly through a cosmos of ideas and information — a data-base orbit.

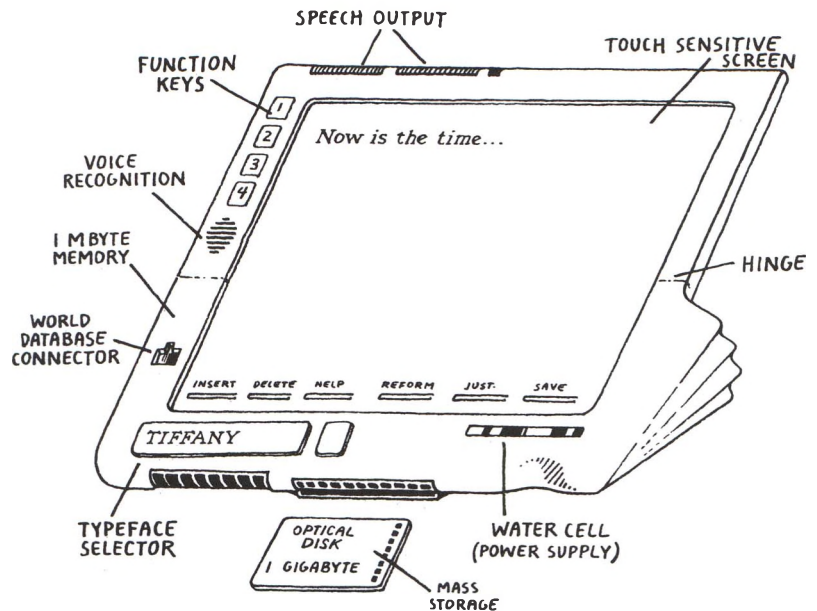


Figure 1.13: *The word processor of the future.*

HOW TO USE THIS BOOK

The best way to use this book is to curl up in front of a fire or the equivalent creature comfort, put on Vivaldi, pour yourself a glass of orange juice, and begin at the beginning. Reading from start to finish, you will glide through the book's topics and learn about the world of word processing in a proper sequence. Or, if you like, you can take the "shotgun" approach and simply pick the chapters which appeal to you most. You can start with the catalog section and see if a particular program sounds promising. Whichever method you choose, you should definitely read the section at the end of the book about defining your word-processing needs. This section is essential in order to avoid falling prey to overkill or underestimation when it comes to the purchase of a word processor.

Now that you have a general feeling for the current market and know the reasons why word processing is such an invaluable tool, you are ready to learn about the kinds of equipment and types of programs that are available to consumers today.

Basic Concepts: Equipment and Programs

There are dozens of personal computers on the market today, but not all of them are truly adequate for word processing. The manufacturer's goal is to offer a computer that can be configured from a host of available peripherals and software and can be used for word processing, business and educational applications, games, and so on. This approach allows for broad advertising statements like: "The Venus Moon Beamer III can do accounts payable, wind-power generation, prayer management, word processing, and back alley adventure games." There is, however, a wide range of variables the user or buyer must contend with to make sure the product can accomplish what is really desired.

We assume that the reader is extremely interested in using the personal microcomputer for word processing. Features such as graphics, sound, and game paddles may not matter as much as the ability to craft words quickly and effortlessly. This chapter will arm you with some technical knowledge of the components of a typical personal computer-based word processor. We want you to understand the state-of-the-art components and their contribution to the entire system. We'll tell the story of the word processor's external components (peripherals, disks, and terminals) as well as its insides (bytes, RAM, and ASCII code). This will help you to better understand and judge the merits and limitations of one word processing system over another *for your unique needs*. This chapter can be skipped if you like, but it would be enlightening even for professionals who could use a refresher course in how the word processor processes its words.

So let's get going.

THE PARTS OF A WORD PROCESSOR

A word processor is similar to a stereo system. Like a stereo, the word processor has a number of components that work together to make up the complete system. These components may be housed separately or combined together in one complete box. It is important to be able to identify the parts of the word processor, regardless of their physical locations. You don't need to know how they work on the inside, but you should be familiar with their functions and capabilities. Why? Because if you want to get the most out of your word-processing dollar,

you will most likely be configuring your system from components offered by different manufacturers. In order to do this effectively, you must know what you need and which product will be most likely to give it to you.

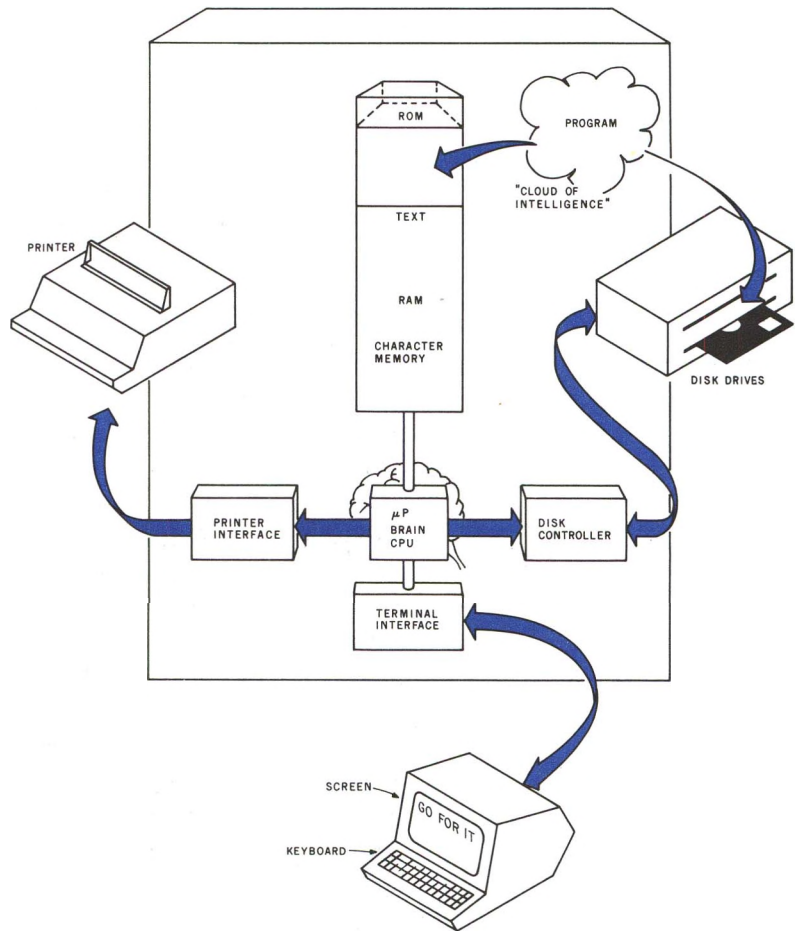


Figure 2.1: *Components of the word-processing system.*

Figure 2.1 shows the individual components of the kind of word-processing system (microcomputer-based or personal computer-based) which we focus on in this book. This is the most common type of word processor available today. There are also large, expensive multi-user word processors and portable word processors (like the Sony Typecorder), which use display devices different than the one pictured. Neither of these will enter into our discussion because they make up only a small portion of today's market.

In Figure 2.1 the items outside the colored box are the *peripherals* of the word processor. Peripherals are input and output devices. The turntable in a stereo system is an input device because the amplifier *receives* signals from it. The speakers are output devices because they *send* music into the room. In the word processor, we have three basic peripherals: the terminal, the printer, and the floppy-disk drive.

The most visible of the peripherals is the *terminal*. This consists of a keyboard similar to the kind found on electric typewriters and a television-like screen. The *keyboard* is the place where you type text into the word processor, and the *screen* is where the text is displayed. Herein lies some of the magic of the word processor. Because the letters are displayed electronically, they can be moved around the screen with incredible speed — literally painted with an electric paintbrush. A good word processor can usually show about 24 lines of 80 characters each — about a third of a regular $8\frac{1}{2} \times 11$ -inch page. A better word processor can show an entire page. Smaller displays (24 lines $\times$ 40 characters) are popular also.

The next type of basic peripheral in a word-processing system is the *floppy-disk drive* (shown in Figure 2.1 on the right). It is analogous to the turntable in a stereo system. You place *floppy diskettes* (circular pieces of magnetic storage material) in the disk drives, and they will play back the information on them. Unlike a record, the floppy diskette (also known as a *disk* or a *floppy*) is reusable. Information can be stored, erased, and restored. Today's best 8-inch floppy can hold up to a million characters of text, which translates to more than 200 pages. Less expensive $5\frac{1}{4}$ -inch floppies can store about 30 pages. Besides holding text, the diskette often contains the word processor's most important program — an operational set of instructions which the machine must have in its memory before it can do its work. This disk program is like a cloud of intelligence, ethereal in quality because it easily can be moved around, altered, updated, and enhanced by the company that created it. We'll say more about this later.

The final peripheral in our component system is the *printer*, which serves a function similar to the typewriter in that it puts words on paper. Unlike the typewriter, however, the printer is usually used only after your document is completed, rather than throughout the composition process. There are several types of add-on devices that make your printer more versatile. A single-sheet feeder that allows the printer to type many similar letters on a company's letterhead without operator intervention is available. Often a tractor mechanism is added for moving perforated paper through the printer. Printers for word processors come in many types, and if high quality print is required, they can be the single most expensive peripheral in your system.

These three peripherals — terminal, floppy-disk drive, and printer — are present in almost all word processors. Although in some systems the terminal may not be a single, isolated component, there will always be a keyboard and a display of some type.

One final note about peripherals in general. You should be aware of the fact that when you buy a word-processing system you may need to purchase some additional parts (cables or interface boards) that will enable you to connect your peripherals to your computer.

While the internal parts (shown surrounded by a box in the figure) are not as apparent as the peripherals, they are vital to the proper functioning of the “organism.” These insides are composed of electronic components and are located on a circuit board. The most important of these charged tidbits is the microprocessor, also known as a computer on a silicon chip, or the “brain.” The microprocessor is what orchestrates all the phases of your word processor’s operations, including text manipulation. All personal computer-based word processors contain microprocessors, and even the larger, more expensive word processors are coming around to using this marvel of silicon.

Where does the microprocessor store your text while it’s being manipulated? The first holding area for text is in the word processor’s *memory*. All characters you type are stored temporarily in the memory as electronic numbers that can be easily moved about by the microprocessor. Residence in the memory is temporary until you permanently save the text onto the disk.

Now that you have a clear picture of the components of the word processor, it is time to zoom in and examine the hardware and software in more detail. In this way, you will better understand the very nature of the word-processing beast.

THE CHARACTER AND THE BYTE

For the typist, the smallest piece of information is the alphanumeric character — a letter, number, or mark of punctuation. In the word processor, each alphanumeric character is made up of smaller pieces of information called *bits*. A bit is like a switch that can be only in the on position or the off position (an on switch is represented by a 1 and an off switch by a 0). Any pattern of 8 bits is called a *byte*. All the circuits of the word processor will operate on bytes of information. For this discussion, consider that a byte and a character are one and the same. In our figures, we show these characters and bytes as being made up of 1s and 0s. If you put eight switches side by side, as we have done below, you can set them up in different on/off patterns to indicate different numbers.

For example, with eight switches you can represent the numbers 0 to 255:

SWITCHES

off	off	off	off	off	off	off	off	= 0
off	off	off	off	off	off	off	on	= 1
off	off	off	off	off	off	on	off	= 2
off	off	off	off	off	off	on	on	= 3
•								
•								
on	on	on	on	on	on	on	on	= 255

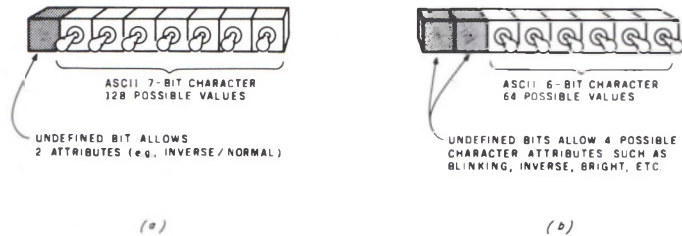


Figure 2.2: Eight switches represent one byte.

In the word processor, all characters are assigned one of these numbers according to the American Standard Code for Information Interchange (ASCII). This standard code is adhered to by most of the western world and makes it possible for computers to “talk” to each other. Figure 2.3 gives a picture of the ASCII code numbers and the symbols that they represent. As you can see, the ASCII value of “A” is 65, “B” equals 66, and so on.

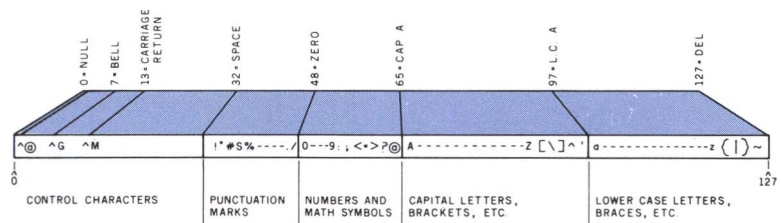


Figure 2.3: The ASCII codes for word processing.

In actuality, there are only seven switches (bits) used in the ASCII code, leaving one of them free for whatever function the computer manufacturer decides upon. Seven switches still allows 128 characters, which is more than enough. A few manufacturers have taken some short cuts in order to lower costs, and consequently their computers represent a single character with only six switches (bits). The resulting system is not very useful for word processing because it offers only an upper case display. Some of these products do have conversion kits available that convert the display and keyboard to allow lower case.

MEMORY

We have mentioned that our word processor contains a memory that is used for holding our text. The memory is made up of many bytes (characters) that can be imagined in a vertical stack. As you type at the keyboard, the characters are distributed upwards in the memory and moved about at breakneck speed. Word-processor memories come in several sizes, as shown in Figure 2.4. The "K" represents 1024 characters. Thus 64K (the most common size) is actually 64×1024 or 65,536. (Other popular sizes are 56K, 48K, 32K, and 16K.)

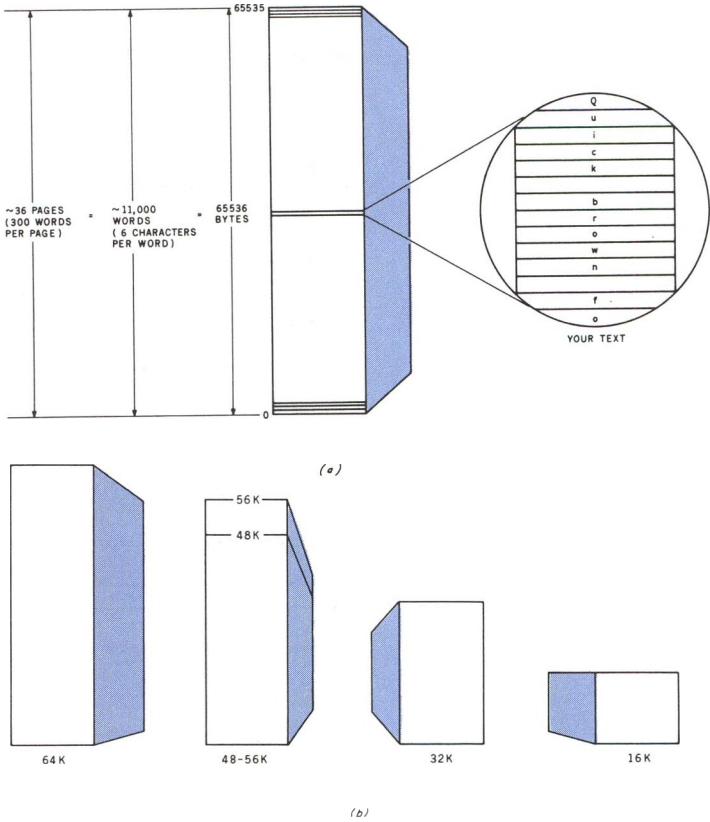


Figure 2.4: Word-processor memory layout.

If all 64K bytes were available in the memory for holding characters, this would be about 64,000 characters of text. Since the average word is made up of five characters, that much memory could hold about 13,000 words. At 250 words per page (double-spaced text), this translates into approximately 50 pages of text. Unfortunately, the entire memory of the word processor rarely is available for holding just text. Part of the memory is devoted to storing the word-processing program, which must be there for the word processor to operate. Both text and instructions share this space through a special feature that allows the memory to hold them both simultaneously.

Another problem regarding this memory is that information storage depends upon electric power. If the power to the word processor is removed, the information in the memory is lost. In fact, memory is so sensitive that even a power “brownout” will usually scramble the contents of the memory. This is an excellent reason for routinely saving your text on the floppy disk. Some of the new portable word processors have a special type of battery-powered memory whose information is *not* lost when you turn it off.

Before we complete this section, we would be remiss in our responsibilities if we forgot to mention another kind of memory that you might encounter when dealing with word processing on personal computers. This type is called read-only memory, or ROM. Read-only memory has information permanently etched into it, almost like having its switches stuck. And, unlike the memory used for holding text, ROM *never* loses its information when power is removed. It's a good thing, too, since ROM is used for holding special programs and information that the word processor must have in order to operate.

SCREEN DISPLAY

As Figure 2.5 shows, the information you type into the computer appears on the screen.

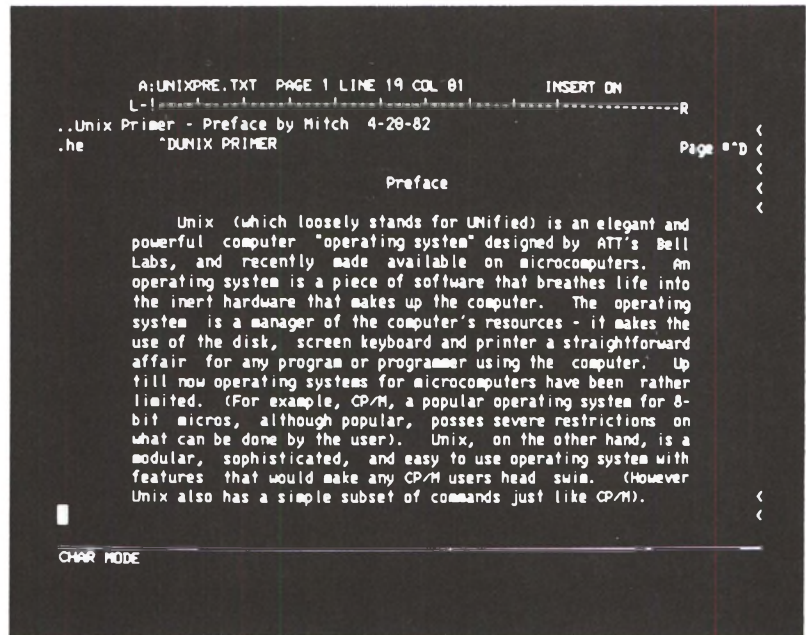


Figure 2.5: *Output on a CRT screen.*

The operation of the screen is quite simple to understand. Within the word processor's memory there is a special area devoted to holding information that is to be displayed on the screen. (The total number of characters in this display will be between 1024 and 1920 characters, depending on the format of the screen itself.) This area is referred to as the *screen memory*, and it is mapped into physical screen positions, as Figure 2.6 shows. A group of circuits constantly scans the screen memory, and the ASCII codes that are stored in it are converted to electronic signals that modulate an electronic gun in the CRT (screen).

If we change the screen memory (we'll discuss later how this is done), these changes will appear on the screen. Great. The scanning circuits do all the work for us, and since we are working with electrons whose velocity almost equals the speed of light, the changes appear to happen instantly.

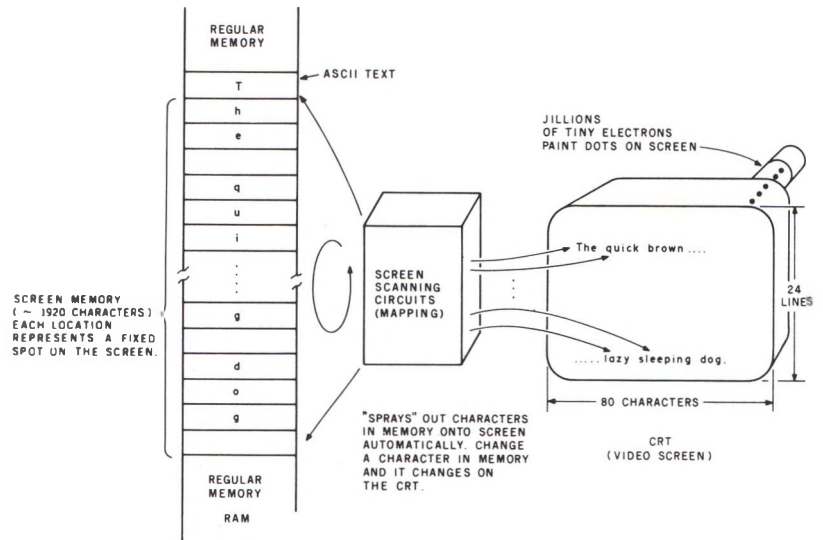


Figure 2.6: Screen memory scanning.

How are these characters put on the screen to begin with? The secret is the *dot matrix*, which is a format of pinpoints of light that are close enough together to look like a single letter. (Dot matrices come in various sizes. 5×7 is popular in inexpensive, upper-case only word processors. In higher performance units 7×9 is common, and 9×12 is available on the most expensive systems.) As Figure 2.7 shows, all the dots in a line of characters are sent out to the CRT at one time. It takes nine scans of the first line to get all the characters painted out, nine for the second line, and so on.

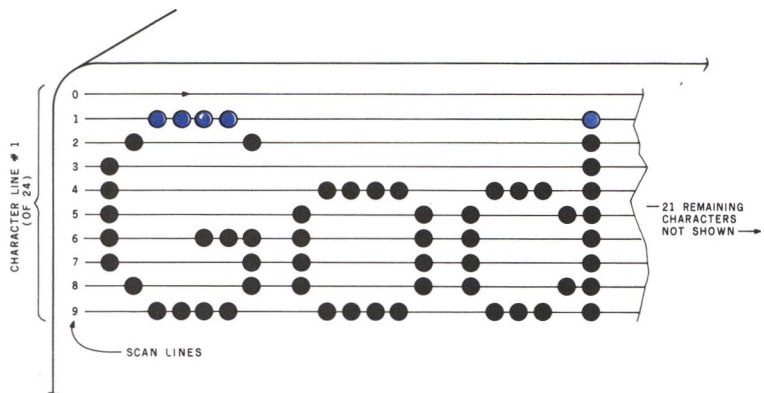


Figure 2.7: Dot-matrix characters.

Remember your eyes when evaluating a word-processor display. Screen display color is important. Soft green is much easier on the eyes than black and white. Recent studies show that amber (yellow-orange) displays are perhaps the most gentle of all. Character resolution is a consideration as well. The characters on the screen should be sharp and clear. Low dot resolution (i.e., characters that are obviously composed of an insufficient number of dots) can cause eye strain. When you go shopping, notice whether the display can be adjusted to prevent reflection and glare. A polarized-screen visor for the CRT can be a great help. (For more details on displays see *Computer Graphics Primer*, by Mitchell Waite.)

KEYBOARD

We enter our characters into the word processor via the keyboard. The word processor's keyboard is like a standard typewriter keyboard except that each key has a tiny switch under it which is wired to an *encoder* circuit. The encoder converts each key to a distinct 8-bit ASCII code value which is then sent to the computer. This is how the keys become ASCII bytes. It is up to the word processor to make sure all key presses are interpreted correctly. The human operator notices any malfunction of the keyboard since each character typed immediately appears on the screen for all to see.

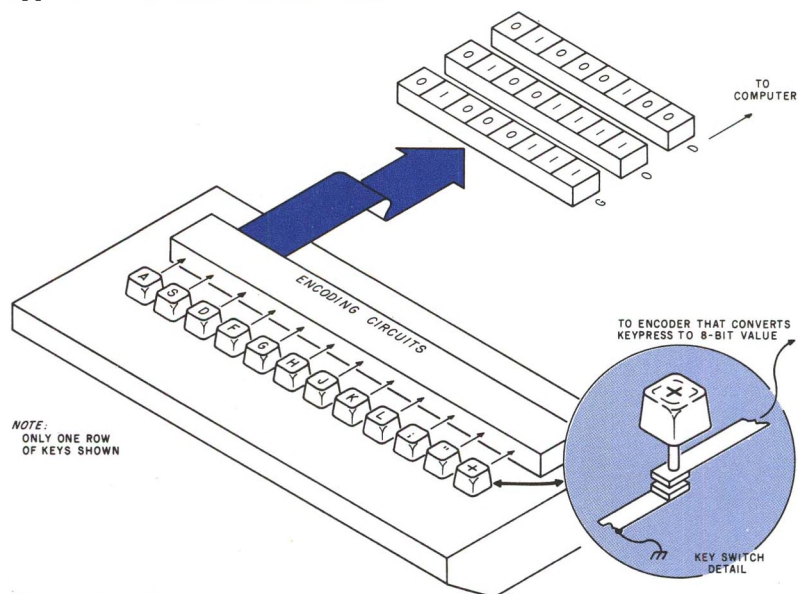


Figure 2.8: Keyboard revealed.

You should know that the word-processing keyboard has a few keys with unusual names. Let's look at the functions of some of them. The RETURN key (sometimes called ENTER or CR) is reminiscent of

the carriage return lever on a typewriter. Pressing RETURN signals the end of a line of text you have entered. The CONTROL key (sometimes labeled CTRL) is frequently used as a leading key to distinguish a command to the word processor. For example, to tell the word processor to display the next page, you might type CONTROL-C. Another often used key is called ESCAPE (sometimes labeled just ESC). ESCAPE is similar to CONTROL, as it is often used as a lead key in combination with other keys to create a command. You may also run across RESET (restarts the computer — usually it is wise to leave this one alone). Some keyboards are equipped with a set of *function keys* which might be labeled with the symbols F1 through F11, or with words like INSERT, SEARCH, or DEL CHAR. Function keys tell the word processor to perform a specific word-processing function and can save you from having to press several keys to get the same job done.

Another important characteristic of the keyboard is the “feel” or tactile feedback of the keys. A good keyboard is one with true mechanically clicking keys. This provides very clear auditory and tactile feedback which makes it easy to know when the key has been pressed properly. The absence of the click means that the key did not register.

Keyboard preference is highly subjective. Make sure you carefully test the keyboard of every word-processing system you are considering.

PERMANENT DISK STORAGE

Although the character memory mentioned earlier can store large amounts of text, its dependence upon electricity makes it less than an ideal storage place. We need a method to store text permanently so it can be retrieved and used again. There are two popular permanent memory-storage media used in word processing today — the magnetic tape cassette and the floppy diskette. (Because tape is slow and not extremely reliable, we will not be covering it in our discussion. However, it is coming back in new keystroke-capture devices.)

The most popular form of permanent storage today is the *floppy diskette* used in conjunction with a disk drive. A floppy diskette is a thin disk made of mylar and coated with a thin layer of magnetic material. (It is called “floppy” because it is flexible.) The magnetic material on the surface stores the bits that make up your text. The disk drive contains a recording head and a motor that spins the disk at a rate of 360 RPM. To protect it at this speed, the disk is encased in a cardboard envelope that is specially treated for low friction. There is a single slot or window in the envelope that gives the recording head access to the spinning surface. The recording head reads information from (or sends information to) concentric tracks on the disk. (Most

disks contain between 40 and 77 tracks.) The word processor labels these tracks so it always knows the location of all information.

Figure 2.9 shows the internal structure of a disk drive. The disk controller converts your text bytes into a storable form and tells the recording head where on the disk to “write” each byte. The information is located by placing identifying impulses on the disk which mark the beginning of separate disk tracks. In this way, characters can always be located accurately and read back into the computer’s character memory.

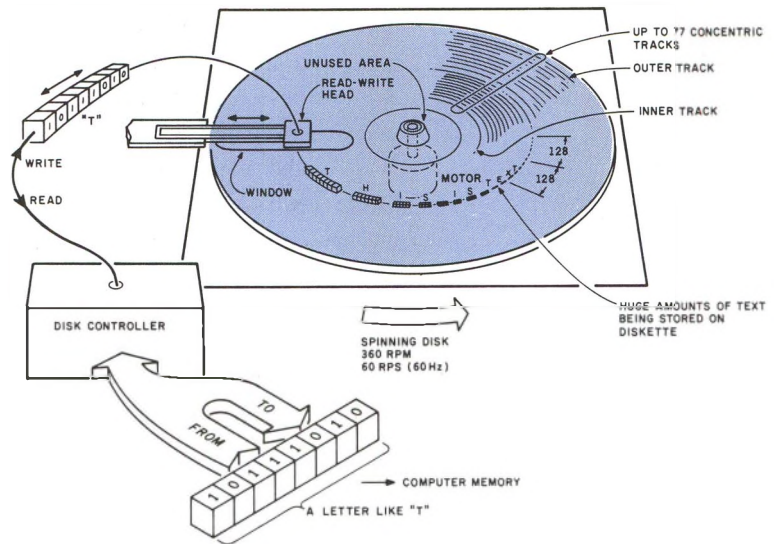


Figure 2.9: *Floppy diskette and disk drive exposed.*

Floppies must be always treated with extreme care. Touching the magnetic surface or getting it dirty will destroy the data on the disk and may even ruin the disk. Heat can warp the envelope and mylar disk, so keep diskettes out of sun and off heaters. Always return the floppy diskette to its envelope when you finish using it. Never turn the word processor off with a diskette still in the drive! In many cases you will destroy the disk if you do.

Today’s floppy disks come in two sizes, but there’s a third one on the way. The largest size is 8 inches, which is the diameter of the disk. The 8-inch disk can usually store from 256,000 to more than 1,000,000 characters, depending on such factors as the disk’s density (how tightly packed the pieces of information are) and whether both sides are used. These diskettes cost about \$10 (\$3.50 in quantity), and their drives range in cost from \$750 to \$1000.

The next popular size in floppies is 5¼ inches. These are called mini-diskettes and hold 80,000 to 500,000 characters. 5¼-inch diskettes sell for about \$8 with corresponding drives for \$400 to \$750 each.

A new Japanese diskette coming on the market is the micro-diskette. It is about 3¼ inches square, holds about 300,000 characters and costs about \$15. The micro-diskette is housed in a sealed, solid plastic container with a door that the disk drive opens. Although its diminutive size might make it easy to misplace, its reliability and high storage capacity give it a promising future among word-processing products.

Another important concept to be concerned with is disk compatibility. Not only are there two popular sizes of floppies (5¼ and 8-inch), there are also many different ways the information is stored on the diskette. Each computer has its own operating system. This is a “mother” program that tells the word-processing program where everything in the computer is located and how to make it work. If different operating systems are used, a disk from one computer cannot be used on another. Sometimes even computers from the same manufacturer can’t read each other’s disks!

There is some hope of compatibility if your computer uses the same operating system as your friend’s. For example, if you both use CP/M (a popular operating system for microcomputers), have 8-inch drives, use single-density disks, and use the same word processing program, then there is a fairly good chance that your diskettes will be interchangeable. We must qualify this because if there are any speed variations between your disk drives, then they would not be compatible.

PRINTER

Printers are perhaps the most ancient part of the word processing system because most of them are still lowly mechanical devices rather than electronic wonders like the rest of the system. But, in this day and age most people still use paper for communicating, so it is necessary to have some sort of printer device to put text on the paper. (In the future we will see situations where no paper is needed and all of our text flies back and forth via telephone cables and satellites, back and forth between floppy disks.)

We’ll be looking at two types of printers used today for word processing — the *formed character impact* printer and the *dot-matrix* printer. (In actuality, an impact printer is any printer that uses force to strike a ribbon and create an image on paper. For our discussion, however, we will be using the term impact printer to refer only to the printers that have formed characters.) There are also new laser printers and even “spray-painting” printers, but these are still quite expensive.

A typical impact printer is illustrated in Figure 2.10. It prints a raised symbol on paper. The print-head mechanism consists of a solenoid hammer, an inked ribbon, and a print element. It all moves horizontally under the control of a motor. The print element is usually a wheel or disk cluster that has all the letters or symbols on its spokes. Between the print element and the paper is a ribbon, and behind the print element is a hammer. The hammer strikes the letter, which in turn strikes the ribbon and makes an impression on the paper. The printer has enough intelligence to move the desired symbol under the hammer in response to the signal being sent from the computer. So, for example, when the word processor sends the letter “T” to the printer, the “T” in the print element is rotated to a position under the printer hammer, the hammer strikes, and the letter appears on paper.

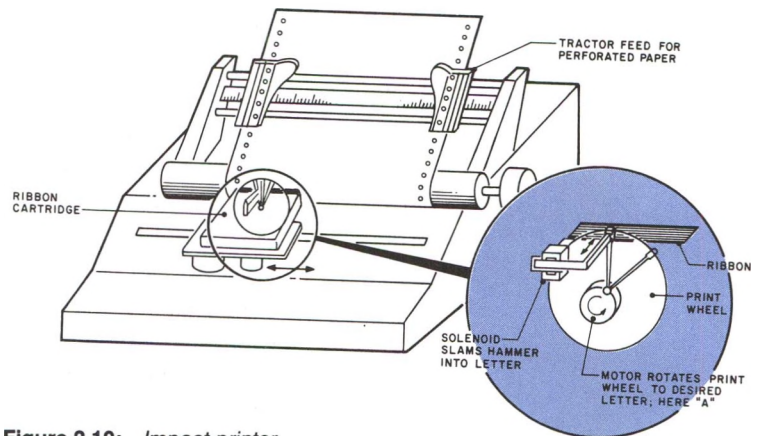


Figure 2.10: *Impact printer.*

Impact printers produce beautiful documents. Their letters are clearly printed and very easy to read. Impact printers usually allow *proportional spacing*, which means that the space between letters can be controlled for professional-looking copy. Let it not be overlooked that impact printers are expensive, typically costing between \$2000 and \$3500. Price, as always, depends on factors such as the type of tractor feed, the size of the carriage, and whether there is a keyboard on the printer. (Keyboards are not necessary on printers, but they certainly come in handy when you want to use them like a typewriter.) The speed of impact printers ranges from 30 to 80 characters per second. Most of them print at a rate of 45 characters per second. At this rate it takes approximately two minutes to print out an entire single-spaced page (500 words).

The matrix printer uses a different mechanical approach than the impact printer. It “forms” each letter as the printing head is moving across the paper. A matrix printer uses a single column of tiny wires.

Each wire is connected to a solenoid which fires it, forcing the ribbon onto paper, and then retracts it. The printer knows which pins to fire in order to create the desired character. The matrix printer, given a dense enough matrix, can form characters with nearly the quality of an impact printer. Using a dense matrix of wires, it may be possible to reprogram the printer so you can create custom character fonts and even graphics characters of your own. Some matrix printers even have built-in ROMs containing graphics-character sets for certain computers. This added benefit is not available on impact printers without expensive retooling of the print element.

Matrix printers can take advantage of several technologies beyond paper and ink ribbon for printing the dots. There are matrix printers that use a thermal print head and special heat sensitive paper. The printer burns the letters onto the paper. These are extremely quiet devices. Electrosensitive matrix printers use a special aluminum-coated paper with a dark oxide undercoating. The print head gives off an electrical discharge that burns off the upper aluminum layer and reveals a dark dot. You can recognize this type of printer by the peculiar smell it gives off.

On the negative side, matrix printers are not as reliable as impact printers. Their characters may not be printed as darkly, and therefore their documents are harder to read and duplicate. But matrix printers can be much cheaper than impact printers; costs start at \$500. Most people on a budget who don't need professional looking copy get along fine with a matrix printer. Matrix printers usually print at approximately the same speed as impact printers. When it comes to transporting these printers, matrix printers are as easy to move as a large toaster, while impact printers are undeniably cumbersome.

Generally speaking, if you can't easily carry the printer, it is probably an impact. If it's easy to pick up, it's probably a matrix. If it smells funny when running, it's electrosensitive.

As an operator of a word processor, your principle interaction with the printer will be replacing the ribbon and installing the paper. The ribbons on most new printers are quite easy to replace since they are packaged in some sort of cartridge. A typical cartridge ribbon for an impact printer costs about \$5.00. As we mentioned earlier, paper for the printer may be either single sheets that are inserted one at a time or a continuous sheet. Single sheets are useful for multiple letters on company stationery. Continuous paper has the edges perforated every $\frac{1}{2}$ inch so it can be moved continuously by the tractor mechanism on the printer. Occasionally, you may have to change the print wheel — if you wish to switch type fonts, for example. This usually involves removing the ribbon, leaning the print head back, and carefully pulling the print wheel off its axle.

Many impact printers can only use plastic fonts rather than the more durable metal ones. Even though plastic fonts are less expensive, you will find yourself replacing them much more frequently because they can wear out after about a month of heavy use. The savings in the long run are negligible. The ideal would be to have a printer which can use either plastic or metal fonts. The advantage of using plastic is the cost and the wider range of fonts available. You could use plastic for occasionally used fonts and metal for everyday use.

Because printer technology is in such a state of rapid expansion, much of the information about it is out of date as soon as it's printed (no pun intended). The only way to really keep up with printer products is to examine the various models at the nearest computer store, talk to friends, and read about them in magazine reviews.

MICROPROCESSOR

The *microprocessor* is the "brain" of the word processor. A microprocessor is a tiny electronic circuit that contains thousands of transistor devices. It is the logic that this brain is endowed with that makes it possible for it to process the words that are entered into your word processor.

Until recently, most microprocessors were equivalent in intelligence, and it didn't matter much which one was in your word processor. Today, some microprocessors are smarter, faster, and more accurate than others. The 16-bit units, like the ones used in the new IBM Personal Computer, are more powerful than the others. In Figure 2.11 we show a typical microprocessor found in a word processor.

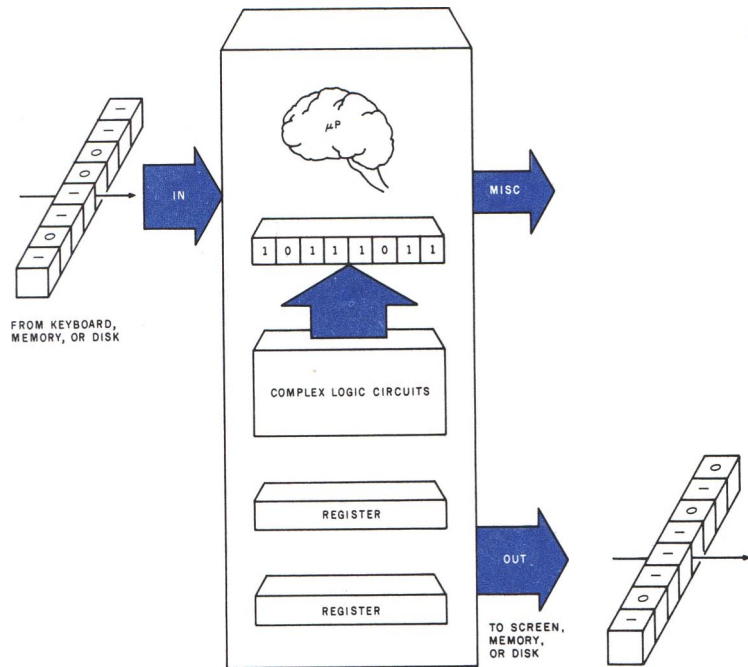


Figure 2.11: *Microprocessor unit.*

As you can see, there is a side where bytes go in and a side where they come out. The microprocessor, being brain-like, “talks” to all the peripherals and controls much of their activity through outside wires. There are registers inside the microprocessor for holding information while the microprocessor works on it.

The microprocessor in itself is not adequate for doing anything. It is basically a device that is programmable. That is, the microprocessor simply reacts in logical ways to instructions that are given to it. These instructions arrive as patterns of bytes that have special meaning to the microprocessor. It reads these instruction patterns and responds in logical ways to them. Often these instructions share space with your text. By carefully arranging these instructional bytes, we can put together a word-processing program that tells the microprocessor what to do.

INTERFACES

The printer, terminal (or video screen and keyboard), and floppy disk all require an additional component to operate. This is the *interface*. The interface is usually an electronic circuit board located inside the word processor (or sometimes in a peripheral). You will not usually have to worry about it. The interface allows the peripheral to communicate with the word processor and allows the word processor to command the peripheral to do things.

Usually when you purchase a microcomputer or personal computer for word processing, the accompanying manual will provide you with details about the interfaces. In all cases, the sales person should be able to explain if the interface is required and, if so, install it for you.

WORD-PROCESSING PROGRAM

As we have said, the word-processing program tells the microprocessor how to handle all aspects of text manipulation and text storage. It also provides instructions for searching for text and printing it out. This word processing program is powerful and complex, and Figure 2.12 may help explain it.

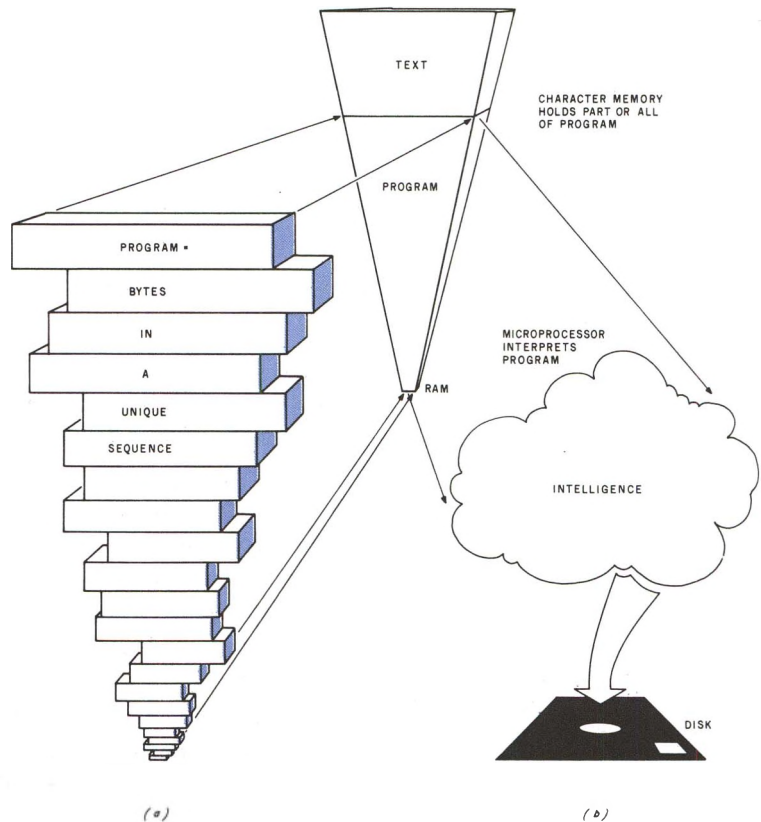
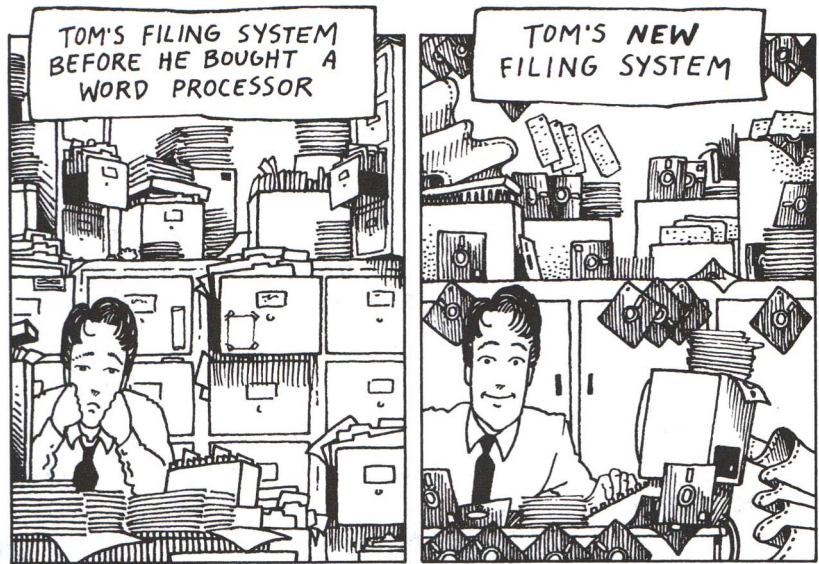


Figure 2.12: *The word-processing program.*

The cloud-like word-processing program sometimes resides permanently in a unique sequence of bytes on a diskette. Sometimes the same sequence of bytes may be stored in read-only memory (ROM). When the word-processing program is stored on a disk it is read into memory when you start it.



There are many kinds of programs, and some are better than others. In this book we have a mini-catalog section that reviews a number of popular word-processing programs on the market today. This will certainly help, but even so, you must get involved in the word-processing program before you can tell what word processing is all about.

There are several features you can examine that will enable you to distinguish important differences between word-processing programs. One of the first features is how the program handles the *text files*. A text file is the way the word processor organizes your typed text on the disk. Many files can exist on a disk, and a text file is just one of the types. Some word-processing programs are *RAM-based*. This means that your text is always kept in random-access memory (RAM) and cannot exceed the amount of space available there. RAM-based programs are fine for short documents and letters and are usually lower in cost because they are easier for the programmer to create. Moving and inserting text, however, is slow work since every letter in the document may have to be moved to accommodate changes.

A second way of handling text files is with a disk-based program, in which the file is written and read from the disk in such a manner that only a small amount of text (a few pages) is stored in memory at one time. This type of program is capable of storing files as large as the disk's capacity and is generally faster than a RAM-based system in doing text insertions and block moves. Because of its increased complexity, a disk-based program is usually more expensive than its RAM-based counterpart. Since only small sections of the text are stored in the character memory while most of it is stored on the disk, moving

through large portions of the text may be slower because of all the required disk activity. Another distinguishing feature of the word-processor program is its length (in bytes). If it's very large, you won't have much room for your text files and may find yourself in a corner when you're working with a lengthy document.

Perhaps this will be less of a problem in the future. There are now very large RAM devices that simulate the operation of a disk drive. They have the same capacity as a disk but are 20 to 50 times faster. This device used in a disk-based system would make it an extremely fast word processor.

DEDICATED VERSUS MICRO-BASED WORD PROCESSING

It is quite important to draw a distinction between the traditional, stand-alone, dedicated word processor that has been on the market for some time and the newer, cheaper word processors built around microcomputers. In the past, the word-processing market was dominated by expensive and well-supported dedicated systems — machines that were non-programmable and could perform only word-processing functions. With the advent of the versatile personal and small business microcomputer, a new type of word processor emerged. This word processor is set up with a home computer using a special word-processing or editing program and some type of printer. Since the number of personal computers has swelled to the millions, the cost of these programs is relatively low.

Many of the personal computers lacked functions that a die-hard traditional word-processor user would deem absolutely essential. The screens could only display 40 columns and 24 lines. The keyboards generated only upper case and needed a special add-on device to generate lower case. Yet, to average home users who had never seen or heard of a word processor before, the programs that transformed their computers into word processors without costing tens of thousands of dollars were wonderful.

The development of the small business computer further changed and eroded the traditional word-processing market. Here was a personal computer designed especially for business, without any of the drawbacks mentioned before. Because it uses the CP/M operating system, hosts higher capacity disk drives, and employs sophisticated serial terminals with good displays and keyboards, the small business microcomputer presents a second large market for word-processing programs. At the present time, it is estimated that there are over 300,000 S-100 microcomputers that use CP/M. Incredible programs of all types soon appeared for these small business microcomputers, among them several powerful word-processing programs, such as Electric Pencil, WordMaster, and Easywriter. Soon these were followed

by another host of word-processing software, including Applewriter, Magic Wand, Spellbinder, WordStar, Select, and Text Wizard. When these machines, which cost under \$3000, were coupled with letter-quality printers (under \$3000), a word-processing system emerged that rivaled units from the dedicated vendors in performance, but cost one third to one half the price!

Today we are at a crossroads where the vendors of traditional dedicated products are well aware of the threat from the microcomputer market and have begun their own onslaught of competitive products. For example, Xerox has brought to market the Xerox 820 SAM (Simply Amazing Machine). It is based on a microprocessor, runs on CP/M, and has a powerful word-processing program. Other companies that are offering microcomputer-based word processors with CP/M include Wang, DEC, IBM, and Lanier.

In the following paragraphs we will discuss in more detail the personal computer-based word processor and how it compares to the dedicated word processor.

Display Problems

The differences between the personal computer and the dedicated word processor are obvious ones. The lack of an 80-column display in some personal computers is a serious problem for word processing. (If you buy a hardware device that converts your computer's display to 80 column, you can expect to pay \$300 to \$400.) This is not usually a problem on the small business microcomputer because an 80-column serial terminal is usually provided with the original system. The terminal is usually capable of inverse video, half intensity, blinking, etc. This means the word-processing program can easily do sophisticated things with the text on the screen. Dedicated word processors have fine displays, often showing a complete page of 66 lines of 80 characters. Some, like the Xerox Star, can show two 66 lines pages side by side!

No Lower Case

Some personal computers (the Apple II, for example) lack the ability to display and accept lower case characters. A hardware option may be purchased that allows the computer to display the lower case, but this increases the price by \$100. With both the small business microcomputer-based word processor and the dedicated unit, this is never a problem. Furthermore, the keyboards on personal computers are not particularly designed for ease of typing. The slope is usually wrong, and often a numeric pad and function keys are unavailable.

Construction

Quality of construction on the personal computer and small business microcomputer is usually lower than on the dedicated word processor. Although this is not really critical, it can lead to more frequent problems with the equipment. For example, the contacts on the various bus connectors inside a dedicated word processor are usually gold-plated and trouble-free, while on the personal and even small business microcomputer they are often solder-plated and subject to problems from oxidation. (I have to clean connectors with an eraser about twice a year.)

Sales Support

Sales support may not be as good on the personal or small business microcomputer as it is with the purchase of a dedicated word processor. One reason the price of the personal unit is so much lower is because the expensive after-sale support is not as intense. This is beginning to change with the entry of companies like IBM and Xerox into the personal computer arena. The new IBM Personal Computer and the Xerox 820 Simply Amazing Machine have ironclad service contracts that you can purchase and an extensive network of service centers across the country.

Not Locked In

On the plus side, there is a major advantage to the personal or small business computer — *you are not locked into the vendor's mandated configuration*. It is possible to purchase a large range of equipment and programs from suppliers other than the original equipment manufacturer, and you can fulfill your needs in customized ways. You can equip your microcomputer with the most extensive hardware — color graphics, hard disks, plotters, printers of all types, special sound effect, or whatever. At the same time, a huge range of non-word-processing programs are available for the microcomputer. These include education, games, industrial control, a variety of computer languages, and general business. It is also possible to configure a system especially for a certain type of business by selecting from the software shelf.

Although most necessary hardware changes can be added on by just plugging them in, installing this extra hardware usually takes technical knowledge beyond the ken of the typical home user. With a little luck, you can usually find a dealer or a consultant who will do it for you. Software installations on a personal or small business microcomputer are usually simpler.

On the other hand, dedicated word processors are most often *turnkey* systems, meaning they are not meant to be expanded, modified,

or customized in any way, except by the manufacturer or authorized service center. This means that when you consider the dedicated word processor, you should understand that growth and alteration must be done in a manner established by the manufacturer. Planning should be done around the features available in the equipment today. Don't count on features coming or promised, as a shake-down period is usually required for all equipment.

In many cases, the personal computer and small business microcomputer may be the best choice for a word processor. They offer (at a lower cost) a wider variety of programs, more hardware options, more instructional information, and more documentation. The drawbacks are lower quality, weaker service, sometimes lower resolution displays, and smaller capacity disk drives.

The dedicated machine may be a better choice when service support, human engineering, display and keyboard quality, high disk capacity, and training are important. If the dedicated unit offers a service contract, it usually means that if the terminal should go out, a technician will come within 24 hours and fix it on the spot. If you own a personal computer or a small business microcomputer, a failure may mean that you must bring the terminal to the original vendor and wait several weeks for repair. You may be able to get a local ComputerLand store or Radio Shack dealer to fix your system. If the drives have problems, it's off to the manufacturer of the personal computer. In either case, it's bad news. So if more than 24 hours of down-time is out of the question for you, then look more towards those products with strong service contracts.

SUMMARY

In this chapter you have become familiar with the various parts of a word processor and the function of those parts. You were also given a picture of the way the market is moving.

Remember that the most distinguishing aspect of the word processor is the way the program and machine presents itself to the human user — the operation of the keys, the way the text is displayed, how the instructions (if any) are given and how you manipulate the controls (i.e., the editing features). The editing features are perhaps the most difficult of all to pin down. These are functions that must be understood to make use of these incredible devices. In the next chapters you will learn about basic editing, text formatting, and advanced editing features.

Basic Editing

Once you have reached the point where your word-processing program is “up and running” you are ready for the real fun — editing. Editing involves both entering text and changing existing text. The skills required to edit your text are fairly easy to learn once you have a basic understanding of the movement of the magic cursor and the concept of scrolling.

Before you can do any editing there are several procedures you must complete first. These include “booting” the system, starting the word-processing program, choosing the document file you wish to edit, telling the word processor to put it into memory, and setting any help levels. (Each manufacturer has its own way of implementing these steps, therefore we direct you to the manual that comes with the word-processor program for specific instructions.) For the sake of simplicity, let’s assume that you have gone through all the steps and are ready to edit.

Basically, there are four features that make text editing possible: cursor control, scrolling, insertion, and deletion. With just these functions you can efficiently alter any document.

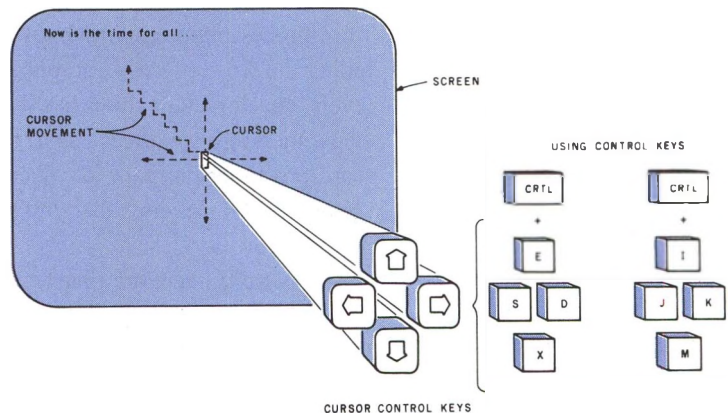
CURSOR CONTROL

When you see the word *cursor*, you should immediately imagine yourself moving through the lines or characters of your text. The cursor (from the Latin *cursorius*, meaning “of running”) comes in many forms. There are blinking rectangles, non-blinking rectangles, underlines, and blinking underlines. One important function of the cursor is to show you where the next character would appear if you typed anything on the keyboard. In addition, it allows you to move to any character on the screen and alter it. As shown in Figure 3.1 the cursor can be thought of as a boat that floats on the surface of the display screen. It can be “steered” with cursor control keys and moved about the screen in a left-right-up-down fashion.

The cursor moves to the right as you type. The cursor is like the notch on a typewriter that sits over the paper in the location where the next key will strike. Since the notch moves over the paper, it is easy for the user to see where the next letter will appear. However, on a video display screen, also known as a cathode ray tube (CRT), there are no

carriage, no notch, and no print hammers, so we need another way to see where the next character will appear, hence the cursor.

Suppose you want to change a letter somewhere on the CRT and your cursor is currently at the bottom of the screen. To change the letter you simply put the cursor over it and type the new letter. But how do you get the cursor to the new location? With the cursor control keys. There are usually four types of cursor control keys as shown in Figure 3.1: up, down, left and right. For example, if you wish to reverse the cursor's direction to correct a previously typed letter on the same line, you would press the left arrow cursor control key as many times as necessary to move it over the mistyped letter. It's not necessary to count as you do this — simply watch the cursor move as you press the key and stop when it's where you want it to be. To move the cursor up to a different line and make a change you would naturally use the up-arrow cursor control key.



(a)



(b)

Figure 3.1: Cursor movement (a) and cursor appearance (b).

If the cursor is a solid block of light, how can you see what character is underneath it? Usually when a solid-block cursor covers an existing character, it reverses the character's video color so it shows, for example, a black character on a white background. Since all other characters are white on a black background, the character that is under the cursor is easily identifiable. See Figure 3.1b.

While you use a word processor, you spend a great deal of time positioning the cursor on the display. You move the cursor from the bottom of the screen to the middle to get to a particular place in the paragraph, and then perhaps you move the cursor over to the right, to the beginning of a particular word you want to change. A good measure of the capability of an individual word processor is the ease with which you can manipulate and move the cursor.

There are a number of basic types of cursor movements, as follows:

- up line
- down line
- right character
- left character
- right word
- left word
- top screen
- bottom of screen
- left margin
- right margin
- to marker n
- up n lines (n = 1 to 65535)
- down n lines
- forward n characters
- back n characters
- forward n words
- back n words
- forward n pages
- back n pages

Up line and *down line* move the cursor either up a line or down a line from its present location. If it is at the top or bottom of the screen, moving it up or down will cause the screen to advance one line at a time. *Right character* and *left character* are used when you want to move backwards or forwards a few characters either way. *Right word* and *left word* move the cursor backwards or forwards through the text

one word at a time. *Top of screen* and *bottom of screen* put the cursor in the top or bottom left-hand corner of the screen. *Right margin* and *left margin* put the cursor on the right or left edge of the text window.

Up and down n lines means you can move the cursor an exact number of lines (n) up or down in the file. This is very useful when you know the exact distance (in lines) to a specific part of the text. Forward and backward by n characters or n words means you can tell the cursor precisely how far to travel in distances of words or single characters. *Forward and backward n pages* allows you to move a certain number of pages (a page is defined by your word processor's format) forward or backward from the cursor's present position.

Displayed in a status line on the screen of some word processing programs is a direction symbol, such as the "greater than" or "less than" symbols ($>$ and $<$). The direction this symbol displays tells whether any subsequent cursor operation will be forward or backward in the document ($>$ means forward and $<$ backward).

On the newest terminals, a special diagonal cursor movement is beginning to appear as a hardware feature (not included in the program). When you press one of four keys, the cursor moves in one of four diagonal paths across the screen on a per line basis.

Remember that the overall purpose of these cursor movements is to allow you to get to the place you want to change as quickly as possible. On some of the more sophisticated word processors, a mouse is used to move the cursor. A *mouse* is a device that moves the cursor on a display screen as it is rolled on a flat surface. A small ball under the mouse follows the movements in the X and Y directions and controls the cursor movement in the X-Y plane on the screen. Such a hardware device is expensive and consequently not too popular, yet it imaginatively solves the problem of locating the cursor. In the future we will probably see screens that are touch sensitive. With such a device you could simply touch the screen and the cursor would make an instant journey to where you pointed.

Not all keyboards contain a complete set of separate arrow keys necessary to fully move the cursor. Some just have right and left keys. Others have no cursor keys at all. Usually the less expensive microcomputer-based word processors require a keystroke or set of keystrokes to move the cursor. In an attempt to standardize things, manufacturers have designated the CONTROL key to be used along with various letter keys for cursor control. The two most popular arrangements are the E S D X and the I J K M configurations. Each of these letter sets forms a crude diamond that is easy to remember and easy to locate with the fingers. See Figure 3.1.

The diamond for cursor control may be arranged so that the cursor movements "left word" and "right word" correspond to the sides of

the diamond. For example, A is “left word” and F is “right word.” There are many possibilities for commands that take advantage of this four-sided figure.

Understand that there is no standard way to lay out a keyboard for all the various functions a word processor may perform, but it makes sense to use an arrangement that is logical to most people. (This is a tall order for some programmers of word-processing software, as many of them are exclusively machine oriented and don’t know much about the psychology of interaction between human and computer.) It helps if the standard keys are close to the CONTROL key. Then you can easily move across the keyboard and give a command by pressing the CONTROL key and a letter at the same time.

Other Cursor Control Functions

Several other cursor movements may be implemented on a word processor. The *block* and *marker* commands are used to move the cursor to a spot in the text that may not be currently displayed on the screen. For example, suppose you wish to move to the start of a particular heading that is somewhere in the middle of your many-paged document. You would mark this heading earlier by pressing a key that puts a special non-printing character in the text called a marker. By pressing the right keys, you can make the cursor go to this marker. The text on either side of it will appear on the screen. Scrolling, which we will discuss next, is another way to locate a block of text quickly. Block markers and cursor movement aren’t the most exciting things in the world, but understanding them is essential to appreciating the word processor fully.

SCREEN SCROLLING

Scrolling allows the upward or downward movement of the text lines on your display screen, thus revealing formerly unseen parts of the text. For the purpose of understanding the ideas we are about to introduce, imagine that your text is like many pages stacked one above the other and that only a portion of it appears on the screen at any particular time. Your CRT screen is like a window into the text, usually showing 24 text lines of 80 characters per line. See Figure 3.2. Imagine that the text moves while the window remains stationary.

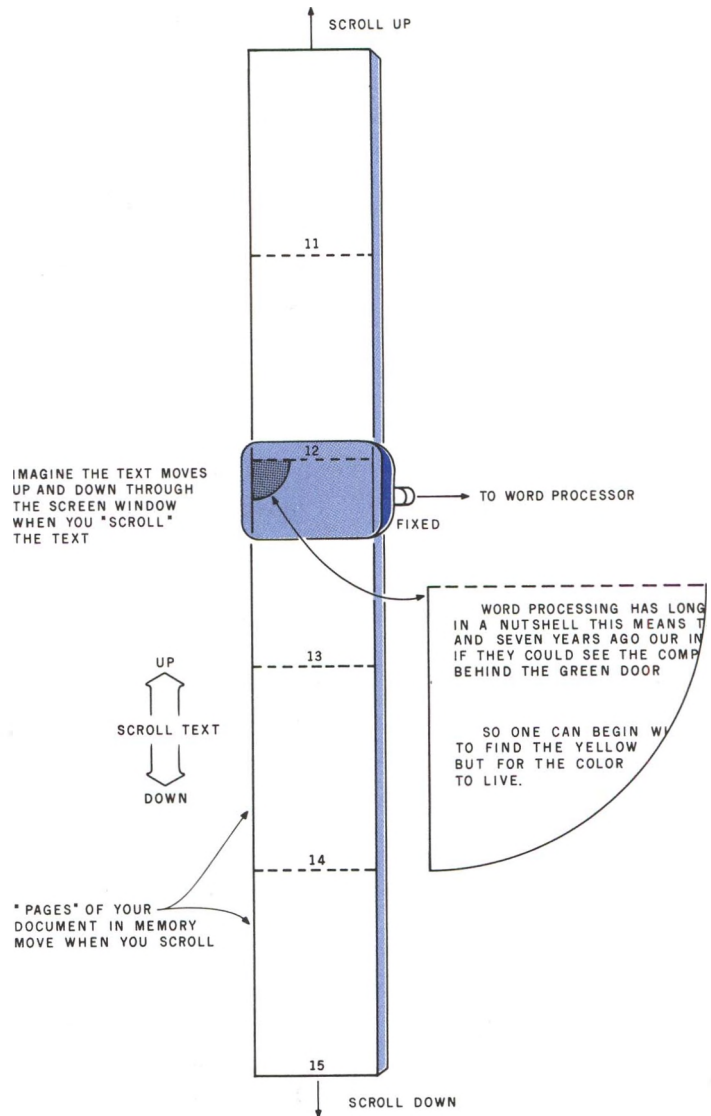
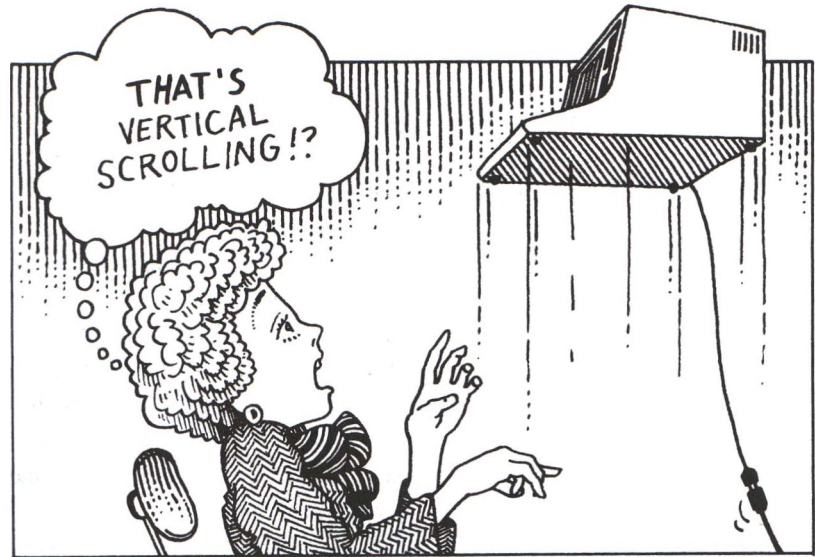


Figure 3.2: *Scrolling.*

Vertical Scrolling

The direction of scrolling usually refers to the direction in which the text moves past the window. For example, when we give the command to *scroll up*, the text of our document moves upwards and the text below the current window moves up into the window. When you *scroll down* you push the text down and reveal preceding portions of it.



Scrolling by Line or Screenful

We can scroll the text in several manners. One way is *line scrolling*, where pressing one key moves the text up or down by one line. This might be done to bring a line into view that is just outside the screen edge. We can scroll by a whole screenful if we desire. This moves the text up or down a number of lines at a time, so that a large section of the text moves into or out of view. Since there are 24 lines visible on most terminals, a scroll-screen up or down command can move you through a document rather quickly. This command is frequently used when you are reviewing or reading through an entire document.

Since a typical page of text in a book contains about 66 lines, we find in word processing on a terminal with 24 or fewer lines that we have to turn many more pages (press the SCROLL SCREEN key) than we would with a book or a typewritten document. Some word processors have the ability to display an entire page of text and can scroll an entire page at a time.

Most word processors cannot scroll if there is no text in the character memory. In fact, on most word processors you can't even move the cursor around if there is no text on the screen to move to. This is why when you first bring up a word processor and start a new document, you won't be able to activate the cursor or scrolling keys until either you read in a pre-existing text file or enter new text from the keyboard.

Horizontal Scrolling

A rare form of scrolling that is becoming popular among the more expensive word-processing software packages is *horizontal scrolling* or horizontal screen flip. Horizontal scrolling lets you move into view columns of text that are to the right of the 80 column screen edge. When you are trying out a word processor with horizontal scrolling, make sure that there is a cursor command to move to the extreme left and right margins no matter where you are on a line.

This feature helps you work with tables and documents that are wider than the actual screen. For example, suppose you wish to produce a 132-column table. On an 80 column screen without horizontal scrolling, the lines past the eightieth column would wrap under the previous line, making it difficult to visualize and align the table. With horizontal scrolling, all the text on the screen is moved over to the left by a certain number of columns when the cursor reaches the right screen edge. (See Figure 3.3.) A typical operation would make the middle of the line move to the left edge of the screen so that, for instance, column 40 becomes like column 0. The middle of the screen is column 80 and the right edge becomes column 120. In a properly implemented horizontal scroll, you should be able to get up to 256 columns, and in some of the very best, 32,000 columns is attainable (but what use would anyone have for 32,000 columns — perhaps to get in the Guinness Book Of World Records).

In addition to being software-controlled, scrolling can also be handled through hardware. On some of the newer terminals the manufacturers have implemented a second character font that allows 132 columns. This font is coarser and cruder than the 80-column font, but it allows an entire 132 columns to be viewed at once without flipping the screen. This may become a standard in future terminals as the added cost is negligible (just an additional \$1.25 for a ROM chip!)

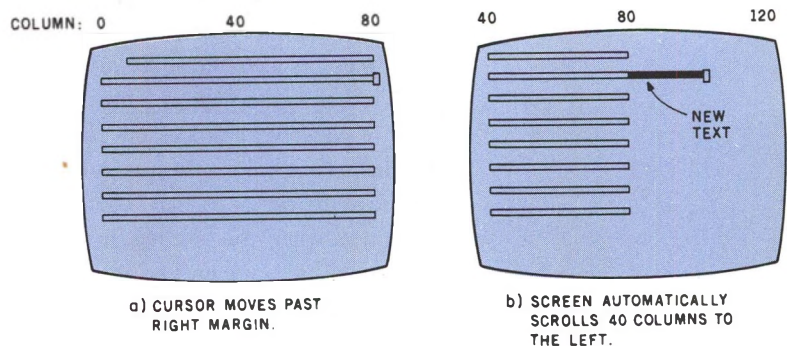


Figure 3.3: Horizontal scrolling.

With a properly implemented horizontal scroll, you should be able to move the cursor and type without even thinking about what the screen is doing. The word processor moves the text as the cursor changes its position.

Continuous Scrolling

Another important scrolling feature to look for is *continuous scrolling*, which scrolls the text automatically up and down on a line-by-line basis. This allows you to sit back and read your text as it smoothly flows up the screen. Some of the better packages allow you to control the speed of the continuous scrolling, thus accommodating slower and faster readers.

INSERTION

Insertion is the feature that adds new words in between other words or text. We can insert everything from a single character to an entire paragraph neatly between two sentences! (The latter is done quite often by authors, the former quite often by proofreaders and secretaries.) When you insert new text, what is “in front” of the inserted text is “pushed” forward to make room for it. This can result in rather messy looking screens with lines sticking out in various places. A second more advanced editing command called reformatting is available for cleaning up the paragraphs. This will be discussed later.

Here’s how you might use the insert feature for editing. Let’s say you have typed a paragraph. After reviewing it, you decide to add a few new sentences to it, in the middle. First you move the cursor on the screen to where you want to insert the new sentence. Then you press the INSERT command key to turn on the insert mode, and start typing. (The word processor may normally be in the insert mode, in which case turning it on is unnecessary.) The new text you type moves to the right as you would expect. However, the old text that is to the right of the cursor is automatically pushed forward (to the right) to make room. Each time you type a character, the text to the right moves over one space. (See Figure 3.4.) When the right margin has been reached, the text automatically wraps around to the beginning of the new line. This last feature is called word wrap and will be discussed later. Without word wrap, there would be no way to know when a line was too long to fit on the screen.

You have seen how text is inserted. This is truly a sight to behold, and most people get their first hint of the power of the word processor when they see insertion in action. It is something you should try as soon as you get a hold of a word processor. Insertion is rather difficult to draw in a book, and you must actually see it to appreciate the power and flexibility of this feature. Figure 3.4 reveals the process.

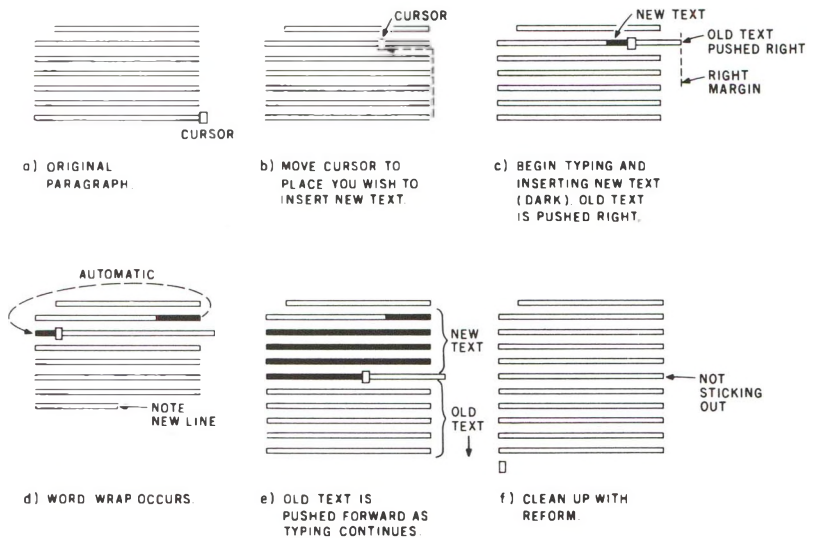


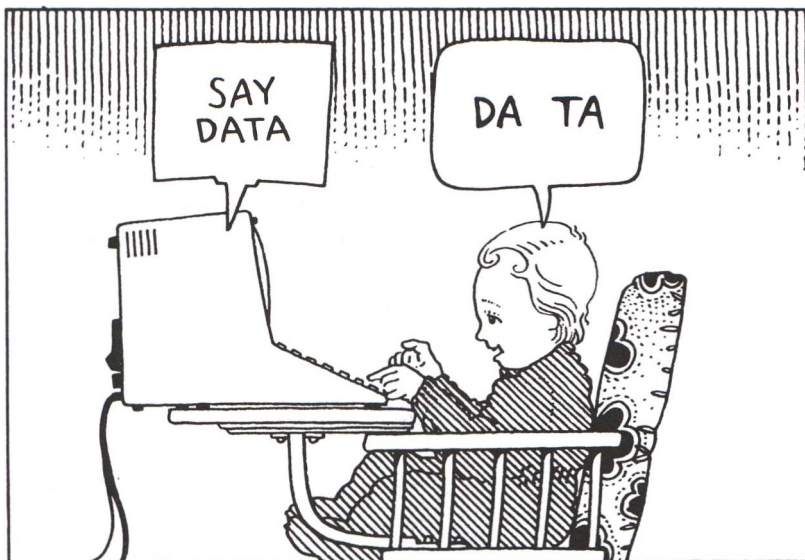
Figure 3.4: *Insertion in action.*

DELETION COMMANDS

Insertion adds text and deletion gets rid of it. *Deletion* is the removal of letters, words, sentences, and paragraphs from your document. There are several forms of deletion:

- delete character right
- delete character left
- delete word right
- delete word left
- delete line
- delete to end of line
- delete to beginning of line
- delete marked block

The most often used of these deletion commands depends on the needs of the individual user, but delete character right, delete word right, and delete to end of the line are the ones most used by this author. *Delete character right* is used when you have typed an incorrect letter. You would back the cursor up to the left of the letter and then delete it with delete character right. Then you would type the new correct character with INSERT on. If your terminal has a DEL key, you could move the cursor to the right of the bad character and delete it. *Delete word right* is often used when you have misspelled a word. You would back up the cursor then type the new word with INSERT on. Now the misspelled word is to the right of the cursor. *Delete word right* gets rid of it for you. *Delete word left* removes word to the left of the



cursor. Use it when the word you just typed is totally wrong and you want to remove it.

Delete line is used when you wish to remove an entire line, or group of lines, from the screen. You use *delete to end of line* occasionally to clean up unnecessary characters and words that have accumulated to the right of the cursor as a result of insertions performed without deletions. The ability to remove entire lines is useful when a complete thought deserves to be edited out of existence. *Delete to beginning of line* is reserved for times when you deduce that the previous line has gotten off to a poor start.

Delete marked block is a “block buster” of a deletion command. It removes (erases) all text between a special beginning and ending *block marker* that you have entered. It gives you the freedom to annihilate large sections of text. After you have deleted a block, the two pieces of adjacent text are instantly linked as one, without the benefit of glue, tape, or a marriage license. This feature alone is evidence of the fantastic utility of the word processor.

It is important to use *extreme* caution when playing with the delete keys. Unlike those inserted, characters removed usually cannot be recovered. Unfortunately, it is far too easy to accidentally remove important ideas from your text. An ideal word processor would have a “recover the last ‘n’ commands” option that you could use when you mistakenly deleted 5 lines of precious text. It could restore the lost text (which would have been held in some buffer).

PARAGRAPH REFORM

Now here is a truly amazing feature for those of you who might still need some convincing. Remember our paragraph that got scrambled due to the insertion of new words? How can we get it back to a normally formed paragraph? Enter the *paragraph reform* or *reformat* command! On those word processors that feature on-screen formatting, the paragraph reform feature reworks lines that stick out to form a smooth paragraph that fits perfectly between two margins. In the better programs, you can instruct the word processor to control the right margin of the text to be either straight, like a newspaper column, or ragged, as in normal typewriter output.

We will cover the subject of reform in more detail in the next chapter on text formatting. You'll also learn how to distinguish on-screen formatting from off-screen formatting. We'll also give you the inside story on hard and soft spaces, carriage returns, paragraph justification, word wrapping, and embedded print commands.

Text Formatting

In the previous chapter, you learned the fundamentals of editing, but this is only the beginning of utilizing the full potential of your word processor. The main function of a word processor is to output text which has a professionally typeset quality — clean, bold, and proportionally spaced. *Text formatting* pertains to the control of all aspects of text appearance. It is a rather complex subject as it deals with line length and spacing, margins and tabs, the darkness and pitch of the letters, centering, justification, and page numbering.

Since there are two different approaches to designing the formatting part of a word processor, it is difficult to make universal statements about the subject. One approach requires the insertion of special print-format commands in the text as you type. When you are finished with your document, it is sent to a print formatter program, which prints it out according to your specifications.

This pure command approach has one major drawback. It is difficult for the user to visualize exactly how the text will look, since he or she can see it only after it has been printed. For simple documents that don't require a great deal of margin changing or line spacing, this approach is adequate. Several popular word processors on the market today, such as AppleWriter and Magic Wand, utilize the print formatter approach. We refer to these pure, format-command-based word processors as having an *off-screen* print formatter.

The second tactical approach to formatting utilizes what is known as *on-screen* formatting. On-screen formatting means that the way the text appears on your screen is almost exactly the way it will appear on paper. There is rarely a one-to-one correspondence between the two because many print commands cannot be shown on today's inexpensive screen displays (although higher cost displays can simulate them). Secretaries and authors prefer on-screen formatting since it does not require them to imagine the way the final text will look before it is printed.

In reality, on-screen formatting usually borrows from the pure print-control approach and uses embedded print-control commands as well. But since the on-screen word processor manipulates text on the screen in more complex ways, it requires knowledge of additional subjects, such as on-screen rulers, word wrap, justification, and page-break displays.

In order to better understand on- and off-screen text formatting, we will start with a basic description of how a word-processor screen might be organized and what kinds of things might appear on it.

THE WORD PROCESSOR SCREEN LAYOUT

There are no real standards for word processor screen organization; however, most microcomputer-based word processors work with a display screen that is 80 columns wide and 24 lines long. Since the trend seems to be towards 80-column screens, we will focus our attention on these. Keep in mind that various products may in fact have 40-column screen versions but that these usually operate at reduced performance levels.

Basically, there are two kinds of information that can appear on the display screen: things that you type, such as text and instructions to the word processor, and things that the word processor types all by itself. Characters that the word processor originates and displays on the screen are translated information that would be taken for granted on the typewriter. It may include *status information* (the name of your document or the page you are on), *ruler lines* (which indicate margin and tab settings), and *flag characters* (which provide information about the way a line ends).

One possible approach to arranging information on a word-processor display screen is shown in Figure 4.1. As we have mentioned, since there are no standards in setting up these screens, there are also no easy rules for evaluating them. For example, some of the less expensive word-processing programs do not provide page information and others may lack margin indicators. Many do not have a ruler on the screen because their margins and format are set by separate embedded-format commands. Some word-processing programs even have a separate screen called the format screen that is activated with a special key. The format screen reveals detailed information about the default settings of the document.

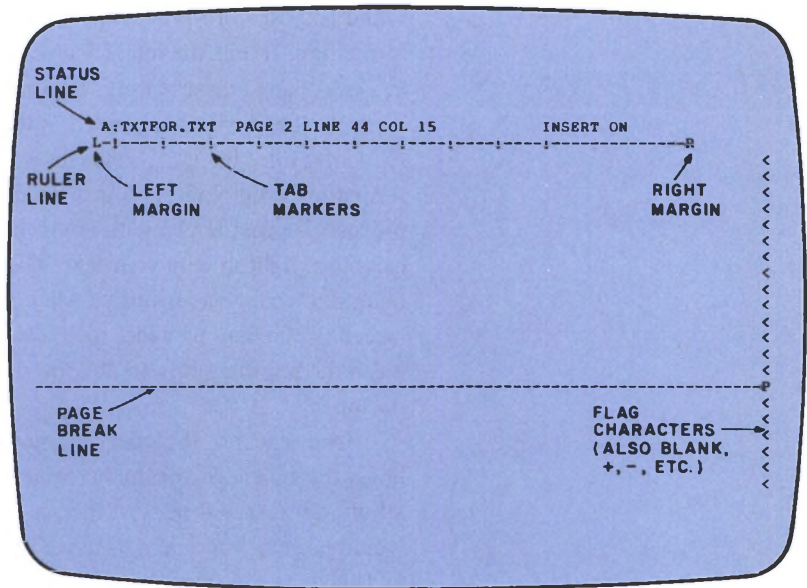


Figure 4.1: Typical word-processor display.

Let's dispense with the possible non-standard formats (however many there may be) and discuss the typical screen in Figure 4.1 so you have an idea of what to expect.

Status Line

The most obvious feature of the figure's screen display is the status line that appears at the top of the screen. Its purpose is to show the name of the file we are editing so we can remember it when we wish to save it. This one tells us the page we are on and then the column and row the cursor is in. This information helps us keep track of where we are in the document. It is helpful to know what line (row) we are on since only one-third of a page can be seen at one time through the 24-line window that is our screen. The column number is useful for helping us set margins and tabs and perform column moves. Finally, we see the words "INSERT ON," which remind us we are in the insert mode. Turning off the insert mode removes the message from the screen.

The status line may include more or fewer items than those we've mentioned above. The information may not appear in the same order as in the figure and the status line itself may not always be at the top of the screen.

Ruler Line

Immediately below the status line is an on-screen ruler. It resembles the markings on a typewriter carriage. The ruler is usually

found only in word-processing programs that allow on-screen formatting. It tells the left (L) and right (R) margins of the document. We can change these settings to control the width of our document, and instantly the change is reflected in the length of the ruler. In this particular ruler line, “!” represents tab positions. In an off-screen formatting program a ruler is unnecessary, since off-screen word processors provide you with non-printing margin commands that you can place right in with your text. These commands force the left or right margin to change as specified when the document is printed. If the screen is 80 columns wide, then LM 10 - RM 71, for example, would mean the left margin is 10 and the right margin is 71 for the printed document.

One aspect of the ruler that may not be obvious is that its setting holds for the entire document. So if you wish to do a global reformat, all of your text will be formatted to the same margin settings. This would be inappropriate if you had paragraphs of various widths in your text. Having the ruler change when embedded margin commands are encountered would be ideal.

The ruler and status lines remain on the screen as you type. They may occasionally be written over for special user messages, but will reappear when the message has been responded to.

Flag Characters

If you look closely at Figure 4.1, you will note that there is a vertical line of “less than” symbols (<) in the extreme right column of the screen. These are called flag characters. In some word processors, the extreme right column of the screen is reserved for the word processor and typing is not allowed here. The word processor uses this column for special characters that tell something about the preceding line. For example, a “<” might mean a carriage return ends the line and a “+” may indicate the line exceeds the width of the screen and proceeds to the right. Since manufacturers create their own symbolic vocabularies, you will have to check the manual that comes with your system for the precise meaning of each flag character.

Now that you understand the way that a word-processor screen can look and the kinds of things the program will display on it (status line, rulers, flag characters, and *of course* your great text), you are ready to take a look at some of the features that control the shape of your finished text and the ways it can be entered into the word processor.

WORD WRAP

Word wrap makes it possible for you to type continuously on a word processor without having to lift your fingers from the keyboard to press a carriage return key. (See Figure 4.2.) When the cursor reaches

the right edge of the screen or the end of the line (as specified by the right margin), it is sent automatically to the beginning of the next line without the loss of any of your letters. This makes it possible for you to type at the keyboard and not even watch what is happening on the screen. The word-processing program “keeps an eye” on the cursor and the characters, too. Since word wrap is controlled by this program, your text is constantly being formatted, line by line. Word wrap would make Gutenberg roll over.

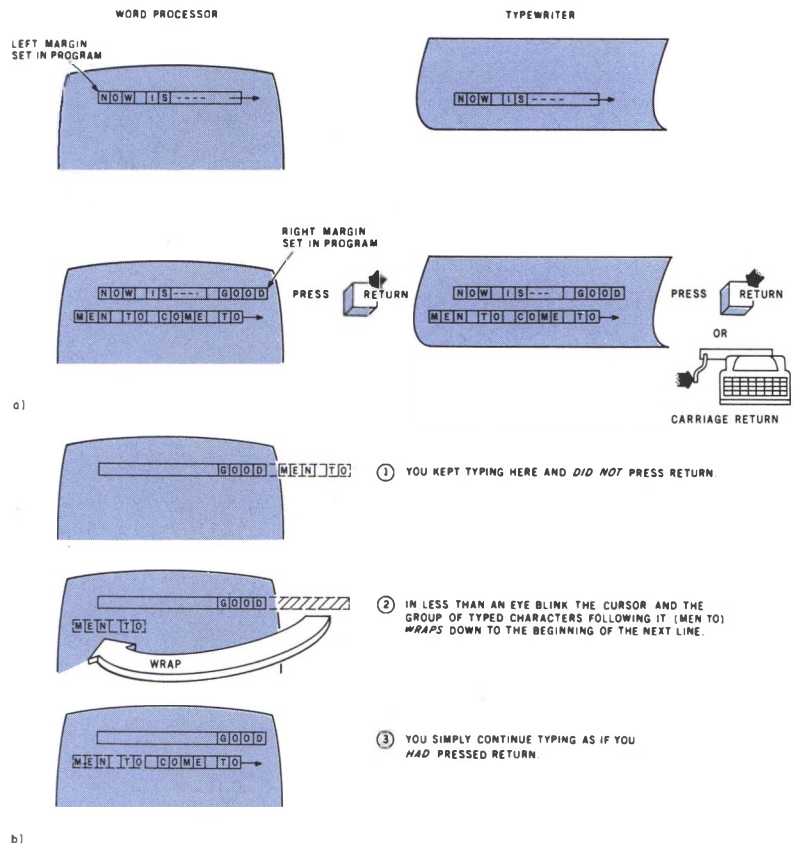
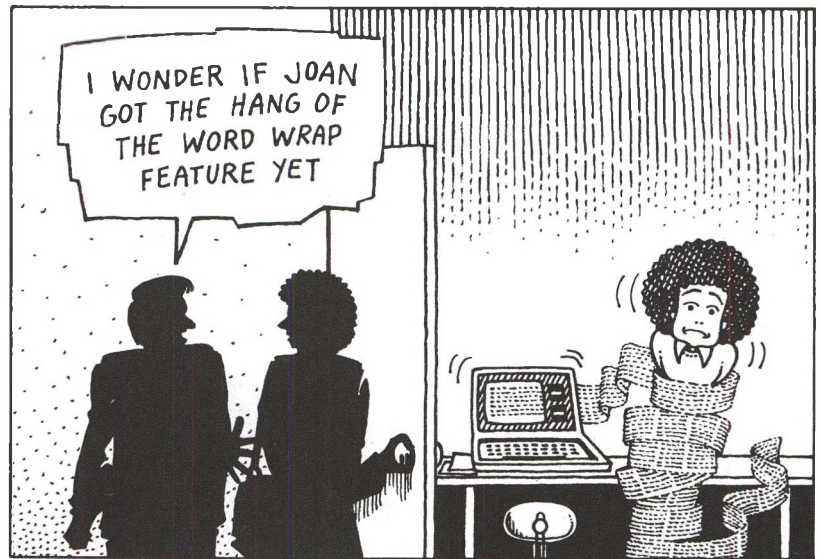


Figure 4.2: Word wrap in action.

There are cases where you will want to disable word wrap because you want to control exactly the way the line breaks. In some word-processing programs you can start up a file in a non-document mode. In this mode, the word wrap is disabled and you must press the RETURN key to end each line you type. Typical non-documents are tables, forms, addresses, and computer programs — anywhere you require lines that break in definite locations.



IDEA BOX

An interesting feature for a word processor would be keeping track of the amount of garbage (wasted keystrokes) that builds up to the right of the cursor as you type (this assumes that the insert mode is always on). When you made a mistake, you would backspace the cursor to the left of the mistake and the program would count how far you had to go. The number of spaces it counts would represent unusable keystrokes which you typed or lost. The average number of disposable characters typed per paragraph would tell you roughly how many mistakes you had made. This could indicate many different problems, such as which of your ideas were especially difficult to put into words. On the other hand, maybe this would not be such a desirable feature.

ON-SCREEN REFORMATTING

On-screen reformatting is used to reshape a paragraph or section of text directly on the screen of the word processor. For example, as shown in Figure 4.3, after you have deleted or inserted a sentence somewhere, there may be protruding pieces of text on the right side of the screen. In a word processor with on-screen reformatting you will be able to have the paragraph return to its correctly formatted condition. On some word processors, the reformatting occurs automatically after any editing. On others, you would place the cursor at the beginning of the paragraph and press the REFORMAT key. The result is the same for both methods. The sentences of the paragraph shift east and west, split at certain points, and, once again, everything becomes perfectly aligned.

```

B:WP1.W1 PAGE 3 LINE 24 COL 33          INSERT ON
L-!-----R
"As a matter of fact, yes. This is the piece de
resistance." Bob reached down under the computer and placed
his telephone receiver in a special cradle. "With this device
I'm plugging the computer into the telephone line, and I can
send my text over the phone lines to another computer. Now I'm
sending it to my coauthor, who lives 48 miles from here. By
mail it takes about two to three days, not counting the time to
package the chapter properly,
pay the postage and go to the
post office. This way my text goes over the phone and is
stored on his disk. I pay for the transmission like it was a
conversation and it appears on my telephone bill. My co-author
can print it out on his printer or edits it while it's in the
computer.
He can send it back to me the same way. We have
given up buying stamps and insurance, weighing packages,
storing packing materials, shipping magnetically sensitive
floppies, etc. Still have to send the artwork by mail, but
that soon will change too."
Welcome to the office of the future in the home of the
CHAR MODE

```

(a)

```

B:WP1.W1 PAGE 3 LINE 33 COL 01          INSERT ON
L-!-----R
"As a matter of fact, yes. This is the piece de
resistance." Bob reached down under the computer and placed
his telephone receiver in a special cradle. "With this device
I'm plugging the computer into the telephone line, and I can
send my text over the phone lines to another computer. Now I'm
sending it to my coauthor, who lives 48 miles from here. By
mail it takes about two to three days, not counting the time to
package the chapter properly, pay the postage and go to the
post office. This way my text goes over the phone and is
stored on his disk. I pay for the transmission like it was a
conversation and it appears on my telephone bill. My co-author
can print it out on his printer or edits it while it's in the
computer. He can send it back to me the same way. We have
given up buying stamps and insurance, weighing packages,
storing packing materials, shipping magnetically sensitive
floppies, etc. Still have to send the artwork by mail, but
that soon will change too."
Welcome to the office of the future in the home of the
present.
CHAR MODE

```

(b)

Figure 4.3: On-screen reformatting of a paragraph, before (a) and after (b).

Another use of on-screen reformatting is reshaping paragraphs with different margin settings. This is only possible on word processors that have on-screen rulers and on-screen paragraph-width control. (As we mentioned earlier, there is another way to control line width, called

off-screen formatting. For word processors employing this approach, the following explanation does not apply.)

Suppose you typed a paragraph with margin settings of 9 and 71 (a line width of 63) and now wish to make the paragraph narrower, changing the margins to 18 and 60 (a line width of 43). To do this with on-screen reformatting, you first reset the margins. With the cursor placed at the beginning of the paragraph, press the REFORMAT key. Your paragraph instantly reforms with the new margin setting.

During the split second between command and performance, the word-processor program went through your paragraph and laid out the words on the screen all over again, as if you were typing them anew. It placed the very first word at the beginning of the new left margin and then placed the next words after it. When it came to the right margin, it did a carriage return and started laying out the remaining words on the beginning of the next line. The program kept doing this until it reached the end of the paragraph. (For the sake of simplicity, we omitted mentioning the line feed, which always follows the carriage return and brings the cursor down to the beginning of the next line.) If you are interested in the technical side of this process, see the box called "Soft and Hard Things in a File."

SOFT AND HARD THINGS IN A FILE

The way a word processor changes the shape of a paragraph or section of text on screen is simple. When you type with word wrap on, the word processor puts special invisible characters into your text. For instance, since you do not press RETURN, it must put in a special "soft" carriage return for you. A soft carriage return is just like a regular carriage return, except the word processor can process it and move it around to new places, thus getting the text to appear correctly on the screen. When you reach the end of a paragraph, you press the carriage return and a "hard" carriage return is entered. Actually, the word processor treats your paragraph as one long continuous line of characters separated by soft carriage returns at every line, and a single hard carriage return at the end of the paragraph.

In addition to carriage returns, the word processor puts special spaces in your text called soft spaces. These are the spaces between the left edge of the screen and the beginning of the text. If your left margin is set at 9, then the word processor inserts nine special soft spaces for you every time you begin typing a new line. (The right margin is set by the soft carriage return.)

The word-processing program is able to distinguish the soft spaces from the actual spaces between words. Its soft spaces are there to be deleted or expanded should you change the left margin. If you change the margin settings and reform the paragraph, the word processor goes into your text and adds the required soft spaces in the left margin, inserts soft carriage returns at the end of lines, and puts the text back on the screen in the new format. These soft and hard characters are portable so they can fit the new format. The word

processor simply replaces the positions of soft spaces and carriage returns, and you get a new, perfectly shaped paragraph.

When you save a file on a disk, the soft and hard spaces are retained so you may reformat later. But, when you print the file on a printer, the soft spaces and carriage returns are treated like hard ones, and the document is printed normally.

In some word-processing programs, you can look in the right column and tell whether there are hard or soft carriage returns. A "less than" symbol in the flag column might tell us that there is a hard carriage return at the end of the line and a blank might indicate that there is a soft carriage return at the end of the line.

JUSTIFICATION

The dictionary defines *justification* as "the spacing of words and letters within a line of type to make it meet both margins of a column." It is possible to create spacing for a document in which the left and right edges of the text are perfectly straight. (See Figure 4.4.) In word processing, *right justification* (also known as the "plain vanilla" approach) is the insertion of spaces between words on a line to make the last word always end in the same column.

A non-justified document is one in which the line length changes according to the number of words in the line and the setting of the right margin. We call this *ragged right*. Correspondence and most typewritten documents are ragged right.

a) Ragged Right

There are two basic differing philosophical approaches to the design of the formatting part of a word processor, and this makes it difficult to say universal things about the subject. One approach requires that you insert special print formatting commands in the text itself as you type. Then when you are finished with your document, it is sent to a Print Formatter program. The Print Formatter program then prints out the document according to the specifications given by your formatting commands.

b) Straight Right Edge

There are two basic differing philosophical approaches to the design of the formatting part of a word processor, and this makes it difficult to say universal things about the subject. One approach requires that you insert special print formatting commands in the text itself as you type. Then when you are finished with your document, it is sent to a Print Formatter program. The Print Formatter program then prints out the document according to the specifications given by your formatting commands.

Figure 4.4: *Ragged right vs. right justification.*

On-screen formatting word processors enable you to use justification. When the paragraph is formed or reformed, spaces are added between words to align all the right margins. With an off-screen formatting word processor you can include a special embedded print command to print the document in a right-justified format.

Some word processors allow left justification, where the right edge of the paragraph is straight and the left edge is ragged. Some even allow centered justification, where each line is centered on a line printed down the middle of the page.

Microjustification (Microspacing)

While we are on the subject of justification, we might mention *microjustification* (also known as microspacing). This term was coined by the word-processing people to describe the control of fine spacing between characters on the printed page. We can usually insert between words only spaces which are one character wide, but on some new printers the spacing between characters can be as fine as 1/120 inch. With this kind of character spacing possible, the new word-processing programs offer a feature called microjustification in which the spacing on a line is regulated by expanding and shrinking the spaces between characters as well as the spaces between words. Microjustification produces a visually impressive effect and makes the reading of a document more pleasant.

No Microjustification

This is an example of text that does not use microjustification. Note that the spacing between words is based on one character width which sometimes results in large gaps between words. This is also how the text will appear on the screen of your terminal or CRT. Someday technology will reach the point where spacing between words on the CRT will appear microjustified, as in the example below.

Microjustification

This is an example of text that does use microjustification. Note that the spacing between words is based on 1/120th of an inch spacing. This tends to spread out the large gaps between words, since now we can control the spacing between individual characters in the words, something we couldn't do in the above example.

Figure 4.5: *Microjustified text example.*

Proportional Spacing

One might conclude that microjustified text looks like the kind found in newspaper and professionally typeset materials. Sorry, that's not true yet. Newspaper text has an additional feature that makes it more balanced than microjustified text. It is proportionally spaced; the characters are not all the same width, as in the case of normal type, but

rather come in widths proportional to their sizes. A capital “M” is wider than a lower case “i,” and the word processor takes this into account when spacing the printed characters on the line. Proportional spacing is available now on several word processors, and should eventually find its way to all of them. (Remember, to achieve proportional spacing, it is necessary for the print element of a printer to be equipped with proportionally spaced characters as well.)

This is an example of proportionally spaced text. Notice how the widths of different letters vary. The “N” in “Notice” is wider, for instance, than the “i” in “instance.” This subtle characteristic of proportional spacing is now becoming popular on inexpensive word processors. Until now, only typeset text (magazines, books, newspapers, etc.) was proportionally spaced.

Figure 4.6: *Proportionally spaced text example.*

HYPHENATION

Hyphenation is another formatting feature available on some word processors. We hyphenate words that are too long to fit entirely at the end of a line by splitting the word with a hyphen between syllables. Hyphenation makes the right edge of a document appear more balanced because the individual lines can be equal in length. Hyphenation becomes particularly important as you begin to narrow your paragraphs. As you make the line width smaller, the word processor needs to break words more frequently to maintain a good appearance and avoid wasting space.

Hyphenation can be a very time-consuming process when you have to choose the correct place to break a word. Today’s word processors with on-screen formatting have a feature called *hyphen help* that eliminates much of the tediousness of this chore. When you use hyphen help on a paragraph and then rejustify it, the word processor will scan your text as it is reshaping it and consider words that are candidates for hyphens. When it finds a word that can be broken up into two syllables, it will automatically break it and move on, or give you the option of changing the word or leaving it as is. An off-screen formatting word processor will add the hyphens as it sees fit during the printout without offering you a choice in the matter. (Hyphen help is a great time saver for the hyphen-hungry person. If your documents don’t need to be printed in narrow columns, however, you may not want to bother with the hyphen feature at all.)

Be aware that there are soft and hard hyphens. Soft hyphens are created by the word processor when hyphen help is on. When the text is reformatted, they can be ignored if the word falls in the middle of a

EMBEDDED PRINTING COMMANDS — OFF-SCREEN FORMATTING

sentence. Soft hyphens are printed in a lower intensity on new serial terminals so you can distinguish them from regular hyphens.

You now understand the text-formatting features of word wrap, on-screen reformatting, justification, and proportional spacing. Earlier we discussed on-screen and off-screen formatting for word processors. We learned that on-screen formatting is done with the aid of on-screen rulers. In order to fully appreciate text formatting we also need to learn about the way off-screen formatting works.

Off-screen formatting programs use *embedded printing commands*, of which there are two kinds: formatting commands and printer-control commands. As you recall, the formatting commands are those that control the final format of the printed text — the line spacing, margin settings, characters per inch (pitch), type of justification, etc. Printer control commands determine attributes of the individual characters that are printed such as whether a character is printed in bold, double struck, underlined, and so on. (The on-screen formatting word processor is also likely to use embedded printer control commands to manipulate some of its print formatting.) Embedded printer control commands are found in both on-screen and off-screen formatting word processors, so let's cover them later. For now, we'll begin with off-screen formatting commands in general.

Embedded Formatting Commands

Embedded formatting commands are those that affect the shape and size of the final printout and include such items as number of lines, margin sizes and positions, footings, headings, page numbering, and character pitch.

Off-screen formatting is commonly done by embedding (typing) special printout commands in the beginning of your text file. These commands will appear in the text on the screen but *will not appear on paper*. When the document is printed out, these commands are interpreted by the word-processing program as instructions for setting printout features.

Suppose you were typing a letter. In most off-screen formatting word processors you first would specify the left and right margins for printout by typing a command, such as LM 9 — RM 71 (left margin 9, right margin 71). LM and RM would be preceded by a special character, such as a backslash (\), an ampersand (&), or a period (.). Usually, the special character embedded for the command must begin in column 1. Thus your document could begin with:

```
\LM 9
\RM 71
```

As with all embedded commands, these would appear on the screen but not be printed. When the letter is printed, it will have a left margin of 9 and a right margin of 71. There are many other commands that affect the format of your printed text. Here are some others:

Line Spacing (single, double, etc.)

Justify (Center, Left, Right, None)

(See Figure 4.7 for an example of these commands in a document.)

Left Margin and Right Margin As discussed above, these commands control which column the line begins and ends in. The left and right margin settings also regulate your line width. In most word processors, the size of the right margin is limited to 132 columns. If you wish to have a certain block of text appear in a new margin setting, you would embed a new set of margin commands at the beginning of the block and then embed the original ones again at the end of the block. In an on-screen formatting word processor, right and left margins are set by the ruler on the screen.

```

\CW 10
\LH 12
\IG Letter - John Romington, Jan 3, 1986
\OP
\FO      FILE NUMBER 45739                                BOX 77
\UJ 0
\LM 9
\RM 71

                                THE WAITE GROUP, INC.
                                1111 SPACE WARP DRIVE
                                PHASER CITY, CA 95000

```

Jan 3, 1986

John Romington
1302 Bitmap Way
San Frandiskco, CA 94116

Dear John:

It's been a long time since we talked, old buddy. How are you? How are the kids, the wife, the dog, and of course, your twin-engine time-traveling program? I'll never forget that last trip to the Russian Revolution! I thought we would never get out of there in one piece. Remember the big guy in the fur coat that wanted to hang us by the toes? Thank God we had a subroutine for dealing with him. Anyway let's get together soon for another trip. How about the American Civil War? Let me know what you think. Send messages to my login via the Lectric World Network. Take care, old friend.

Love,

Mitch

Explanations of Embedded Print-Formatting Commands

All print commands in this word processor are preceded by a backslash character (\). Other word processors may use a different character, like a period (.) or ampersand (&). Above is how the document, in this case a letter, appeared on the CRT screen. Below is how the output appeared when printed.

\CW 10 sets the character pitch in this example. In this word processor a 10 means 12 characters per inch, which is the pica setting in the

typesetting world. We could have left the \CW command out and the word processor would have defaulted to 10 characters per inch, which is standard elite spacing. This writer prefers the pica because it allows more words per line and reduces the amount of paper consumed.

\LH 12 sets the line height in this example. Here 12 means 4 lines per inch. Normal single spacing (the default if we leave the \LH command out) is 6 lines per inch. This writer prefers lots of space between lines, but not a double spacing. The \LH 12 gives 1½ line spacing.

\IG is the command for a comment line. A comment line is text that will appear on the screen but won't get printed out. It is useful for notes to the writer. In this case it reminds us who the letter is to and the date it was written. The writer owns a neat utility program that can sift through all his files searching for such comment lines and present a summary of them on the screen along with the file name. Thus the comment serves as an extension to the file name, giving a summary of what's in the file without having read the whole thing.

\OP means omit page numbering. Since this is a one-page letter, there is no reason to number the pages. This word processor automatically puts the page number at the bottom of the sheet of paper unless we tell it not to.

\FO means footing—a line of text that will appear at the bottom of all successive pages of this document when it is printed. Here it includes reference information for the sender that tells the file that the letter is stored under should he wish to reference it at a later date. As explained in the text, footings are very useful for indicating chapter contents, holding page numbers, etc.

\UJ 0 turns off the automatic microjustification so that the letter doesn't appear to be produced on a word processor. This is a matter of taste and was done only because the writer wants the addressee to think he labored hard to produce the letter.

\LM 9 sets the left margin of the printed letter to be 9 spaces from the left.

\RM 71 sets the right margin of the printer letter to be 71 spaces from the left. As you can see above, the screen used the full 80 columns to hold the letter. When it was printed out it took more vertical space because the lines held fewer characters.

THE WAITE GROUP, INC.
1111 SPACE WARP DRIVE
PHASER CITY, CA 95000

Jan 3, 1986

John Romington
1302 Bitmap Way
San Francisco, CA 94116

Dear John:

It's been a long time since we talked, old buddy. How are you? How are the kids, the wife, the dog, and of course, your

twin-engine time-traveling program? I'll never forget that last trip to the Russian Revolution! I thought we would never get out of there in one piece. Remember the big guy in the fur coat that wanted hang us by the toes? Thank God we had a subroutine for dealing with him. Anyway let's get together soon for another trip. How about the American Civil War? Let me know what you think. Send messages to my login via the Electric World Network. Take care, old friend.

Love,

Mitch

Figure 4.7: *Example of embedded print-formatting commands.*

Line Spacing Different documents may require different line spacing. Manuscripts and legal documents are usually double spaced (one space line between text lines). Some documents may even be triple spaced, and it is not unusual to have as many as 8 spaces between lines. All word processors have commands for setting the line spacing. (If you are using an on-screen formatting word processor, you will set the line spacing on the screen rather than with an embedded command.)

Justify There is a command for specifying whether or not you want the final text to be right-, or left- or center-justified. (You also have the option of specifying no justification and your text will be printed as is.) Refer back to the previous section on justification if you need a review of these terms. The on-screen word processor will do any justification on the screen, without embedded commands.

The formatting commands discussed above are found in nearly all word processors. The next group of commands represents features which may not all be available on any one particular word-processing program:

- Line Height
- Paper Length
- Margin at top
- Margin at bottom
- Heading
- Footing
- Page # column
- Page Offset

New page
 Conditional page
 Omit Page numbers
 Page Number
 Character width
 Subscript roll
 Microjustify on off
 Bi-directional print on off

Most of these features require a quality impact printer with a horizontal resolution of 1/120 inch and a vertical resolution of 1/48 inch. The horizontal axis of an 8½-inch page is divided into 1020 columns, each of which is 1/120 inch wide, and the vertical axis is divided into 528 rows, each of which is 1/48 inch high. The word-processing software takes advantage of this fine resolution to perform these commands. Some of the newer dot-matrix printers do offer matrices that can, with the proper software, reproduce all of these features.

WORD-PROCESSING IMPACT PRINTER MAKES DIGITAL PLOTTER

The resolution on an impact printer is extremely high. In fact, by printing periods (".") it is possible to turn the printer into a high-resolution plotter. The resolution will be 1020 dots by 528 dots, which is horizontally about 3.6 times more resolution than the Apple II computer high-resolution screen! Before you get too excited, you should know that this function will require special software and will also slow down the printer considerably.

Let's take a look at the meaning of these print-formatting commands and the way they affect document appearance.

Line Height The line height command tells the printer the number of lines per inch, i.e., the vertical spacing of the text (not to be confused with line spacing). Normally, this defaults to some standard such as 6 lines per inch, which is 66 lines on an 11 inch page at 1/48 inch per line. (Note: All print-formatting commands have defaults which enable you to leave them out entirely and still print your document.)

Paper Length The command for paper length sets the total number of lines (as defined by the line height command) on a page. Usually, but not always, it represents the number of 1/48-inch increments on the standard length of the page. It may or may not include the size of the top and bottom margins, which increase as you reduce the setting for paper length.

Top and Bottom Margins These commands are used to set off the text and can contain page numbers, section numbers, and special messages called headings and footings. The top margin command sets the number of lines between the top of the page and the beginning of the main text. The bottom margin command controls the number of lines between the bottom of the text and the bottom of the page.

Headings and Footings Among the most useful of embedded formatting commands are ones for creating automatic headings or footings to be inserted on every page. As you may have deduced, the heading is a line at the top of the page, printed in the top margin, and the footing is a line printed at the bottom of the page, in the bottom margin. The information after the command is interpreted as a *string* (a collection of characters of some predefined maximum length) to be printed on this special line. The heading may contain words like “Chapter One — Perspectives, revision 1.2.” A footing might contain similar information.

Page Number Column Most good word-processing packages will print the page number automatically on all pages but will let you specify what column the number should appear in. Some will also let you indicate the type of heading to precede the page number, and some will even permit you to put the page information inside the heading or footing. You can, for example, have a heading like: “Chapter Two — Basic Concepts, revision 2.1 Page 9-*x*”. When the word processor prints the document it will place the message “Chapter Two — Basic Concepts, revision 2.1 Page 9-*x*” (where *x* equals the correct page number) on each page.

While we are on the subject of page numbering, there ought to be an option for beginning page numbers at some arbitrary value (conditional page numbering), and one to disable (omit) page numbering entirely. This is necessary when a document is split into several text files. After printing one section and determining the page on which it ends, you can enter the correct page number to start the next file. Disabling page numbering entirely is necessary for letters and other business documents.

Page Offset Sometimes you want to have *all* the text in your document shift uniformly to the right by a certain number of spaces, because the alignment of the paper in the printer has been set to the extreme left. This is often done to print directly from the operating system at the maximum width of the page. Otherwise, when we print text files, everything is shifted too far to the left. An offset command can correct this problem.

New Page With on-screen formatting, a page break is a line of dashes across the screen that indicates the beginning of the next page. It occurs whenever the bottom of one page and the top of another are visible

simultaneously. The place at which the break occurs is controlled by the setting of the page length command.

Often you wish to create a new page before you reach the end of the current page. It might be desirable to start a new section heading on its own page. You would use a new page command for this purpose. With an on-screen word processor, a new line of dashes will appear across the screen and a new page is formed. The printer will advance the paper when it sees the new page command and move to the top of the next page.

Character Width An embedded format command should be available that will allow you to set the character width, i.e., the amount of space per character on the printer (which is in increments of 1/120 inch on a daisy-wheel printer). This allows you to set the pitch from narrow to wide and produces text that is stretched or condensed. (See Figure 4.8.) You can change the spacing within the document several times by simply altering this character width command. You might want to print the contents of a sidebar (a boxed area of text in an article or book) in narrower spacing.

```
M i n e   e y e s   h a v e   s e e n   t h e   g l o r y .   ( 5 / i n c h )
M i n e   e y e s   h a v e   s e e n   t h e   g l o r y .   ( 6 / i n c h )
M i n e   e y e s   h a v e   s e e n   t h e   g l o r y .   ( 7 / i n c h )
M i n e   e y e s   h a v e   s e e n   t h e   g l o r y .   ( 8 / i n c h )
M i n e   e y e s   h a v e   s e e n   t h e   g l o r y .   ( 1 0 / i n c h )
M i n e   e y e s   h a v e   s e e n   t h e   g l o r y .   ( 1 2 / i n c h )
M i n e   e y e s   h a v e   s e e n   t h e   g l o r y .   ( 1 5 / i n c h )
```

Figure 4.8: *Various pitch spacings.*

Miscellaneous Embedded Format Commands There is usually a command to turn microspacing on and off, one to control how much the carriage will move up and down for subscripts or superscripts, and one to activate or deactivate bidirectional printing. *Microspacing* allows the fine spacing between letters to be controlled and results in a better looking printout. In some cases, such as tables, you may not want the spacing between letters to be adjusted. So by turning microspacing off, then on again, you can detour for the table.

Now you understand the kinds of text formatting that can be accomplished using embedded print-formatting commands on a word processor. Keep in mind your word processor might not have all of these features.

Embedded Printer-Control Commands

The next type of text-formatting commands are the printer control commands. These are commands which are embedded right along with

the words of your document. (See Figure 4.9.) Features usually controlled by embedded print commands are:

- bold strike
- double strike
- underlining
- subscripts and superscripts
- strikeout
- overprint
- ribbon color
- alternate pitch

Unfortunately, most word-processing displays will not show bold strike or underlining, so manufacturers use a special character (or characters) to stand for these printout features that can't be displayed. For example, to indicate that you want a word underlined when it's printed, you put an underline character in front of that word — like “_this_” will be underlined. When this symbol (_) is encountered, it turns on the printer's underliner. Since the (_) symbols always come in a pair, the second occurrence of it deactivates the underlining mode.

Suppose you wanted to print a word or group of words in bold strike. You would type the special bold command symbol (the one chosen by the manufacturer) before and after the word or words to be printed. Suppose the special bold command character key was CONTROL-B, (the CONTROL key followed by the B). If you wanted to make the word “awful” appear in bold, you would type “CONTROL-B-a-w-f-u-l-CONTROL-B”. It would appear on the display as ^Bawful^B and would print out on the printer as **awful**. *Awfully* simple, right?

Figure 4.9 shows examples of using embedded printer control commands in a document.

-
- It is important to realize that the use of ^Bprinter control commands^B can be overdone. This example illustrates several uses of these commands, but one would probably not use them like this.
- ① commands^B can be overdone. This example illustrates several uses of these commands, but one would probably not use them like this.
 - ② ^SUnderline is a very important printer control^S. It is not found in all word processors, and not all printers can perform it.
 - ③ ^DAnother important printer control is double strike. Double strike lets you highlight certain words, sentences, or paragraphs. It is not as dark as bold.^D
 - ④ ^BBold, as this example shows, is darker then double strike.^B
 - ⑤ ^AAlternate pitch may be another printer control. It changes the pitch from 10 pitch to 12 pitch in this program.^A
 - ⑥ ^A^D^SYou can gang up printer controls as this example of double strike, underlining and alternate pitch shows.^S^D^A
-

Explanations of Embedded Printer-Control Commands

- ① In this example the user presses the control key followed by a P (for print) followed by another key to signify the desired printer control. In this case we pressed control P, followed by a B to cause the bold command to be inserted in the file. We must "turn off" the bold command by repeating the control P-B keystrokes a second time. On the screen the keystrokes appeared as ^B (the P wasn't displayed). Understand this is not the standard way to display these commands; rather, it is an example of how one manufacturer approaches the problem.
 - ② Here we turned on underlining with control P-S. You must be careful to remember to turn off the command or else the entire document could be underlined!
 - ③ Here the doublestrike printer command was inserted.
 - ④ An example of boldface again.
 - ⑤ This is the alternate-pitch printer command. You might be able to use a format command (such as CW 10) for this function as pointed out in Figure 4.7 if your word processor allows it.
 - ⑥ In this example we use three printer commands: doublestrike, underline, and alternate pitch. Note how the printing is condensed and even the underlines become darker.
-

It is important to realize that the use of printer control commands can be overdone. This example illustrates several uses of these commands, but one would probably not use them like this.

Underline is a very important printer control. It is not found in all word processors, and not all printers can perform it.

Another important printer control is double strike. Double strike lets you highlight certain words, sentences, or paragraphs. It is not as dark as bold.

Bold, as this example shows, is darker then double strike.

Alternate pitch may be another printer control. It changes the pitch from 10 pitch to 12 pitch in this program.

You can gang up printer controls as this example of double strike, underlining and alternate pitch shows.

Figure 4.9: Embedded printer-control commands.

In order to fully appreciate these commands let's look at each of them a little more closely.

Bold Strike Bold strike makes printed letters appear very dark when printed out. When the bold strike command is encountered, the printer moves over a small amount and the letter is printed again. Since the new letter is slightly displaced from the first, the letter appears darker and wider, hence bold. Understand this is not true bold in the typesetter's sense, but it is still quite effective in highlighting words, titles, headers, and so on. (See Figure 4.10.)

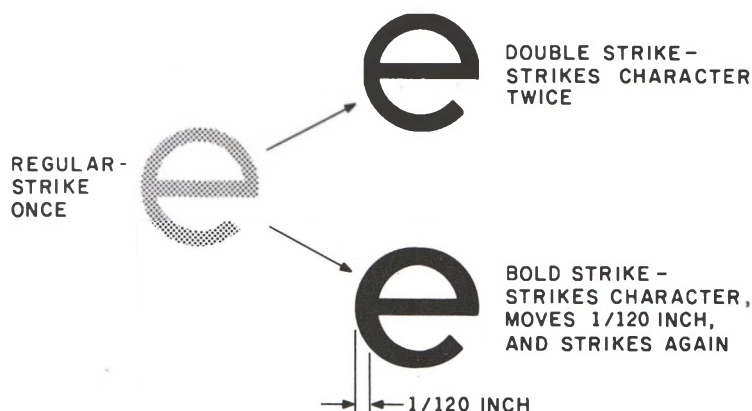


Figure 4.10: *How boldface and doublestrike work.*

Double Strike Double strike is similar to bold strike except the printer doesn't move. A character comes out darker than normal, but not as dark as the bold. Double strike is useful for highlighting words within sentences and tables, or any time you want your hard copy to be extremely dark. Double strike often is used to create camera-ready copy. If all text is printed in double strike, the printout takes twice as long. All printers cannot support bold and double strike. All impact printers surely can, and some matrix printers can, but almost none of the teletype printers can support this feature.

Underlining Underlining is having the printer put a line under any word or sentence you type. Again, an embedded command is used to turn underlining on and off. Occasionally, this author has forgotten to turn underlining off after having turned it on and had 15 pages of underlined text printed out. Moral: one has to be careful with this kind of on/off print command.

Subscripts and Superscripts A subscript is a character, word, or group of words printed a fractional line below the rest of the text in that line. The amount of this roll down varies, but 1/16 inch is popular. This is used for formulas, chemical notations, and so forth.

The superscript command, on the other hand, prints the text enclosed by the control characters a fractional line above the text. Again, we can use the superscript in formulas, fractions, music notation, and other shifting features.

Strikeout The strikeout command gives you the ability to draw a dash character (“—”) or a slash character (“/”) through a word, sentence, phrase, or formula. It is extremely useful to indicate text to be eliminated, such as in a manual update or legal document.

Overprint and Backspace There should be a command for overprinting the next character and giving the effect of backspacing. This is useful when you wish to make up special characters. For example, “a^H` la king” prints as “à la king.”

Ribbon Color There is often an embedded print command that causes the ribbon color to change, assuming you have a dual colored ribbon in your printer. Usually this prints red (not black). This color change is very effective for invoices or important, eye-catching notices (like bills).

Alternate Pitch This is a feature that allows you to change from a standard 10-character-per-inch pitch to something such as 8 or 12 characters per inch.

Pause Often it is necessary to stop the printer in the middle of a document to change something, such as adding a different kind of paper, form, print wheel, or ribbon. When the pause feature is entered into the file, the printer stops, then resumes when a special key is pressed.

SUMMARY

Now you have a good idea of the way that word processors handle the appearance of the printed text. You have learned the concepts of word wrap, justification, on- and off-screen formatting, and embedded printing commands. You are now ready to become familiar with some of the more advanced editing tools that the word processor offers.

Advanced Editing

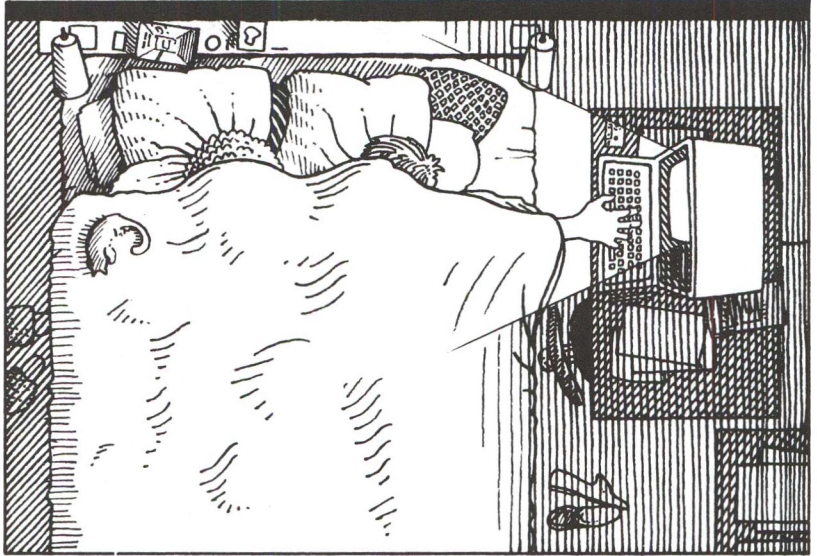
In the last two chapters you learned the basic editing functions of a word processor and the different concepts of printing and formatting text. There are a few additional principles of word processing that are more advanced than the ones discussed so far. In this chapter we will cover these additional features, which include *global* operations such as block-move and search-and-replace functions and boilerplate and glossary operations involving the reading and writing of blocks of text on the disk.

SEARCH AND REPLACE

Suppose you had just finished writing a 25-page document like a legal brief, a long letter, or a speech. Now imagine that you had spelled the same word incorrectly throughout the document. We all know that fixing this kind of error on a typewriter involves searching through every page for each occurrence of the word and correcting it. Not only is this a boring, time-consuming project, it is also inefficient as it is very easy to miss a word. This is where the word processor really shines. If your word processor has a *search-and-replace* feature, and most do, you can have this task performed automatically or semi-automatically under your watchful eye. The word processor “searches” for your misspelled word then “replaces” it (fixes it) with the new word.

Search Only

Most word-processing packages on the market today have the ability to search for (also called find) any word or unique series of text characters in a document. This works by telling the word processor to find a certain *string*. When the string is completely entered and the RETURN key is pressed, the cursor will be moved to the location of the first occurrence of that string within the document, and the text above and below it will be brought into view. With the cursor over the unwanted word, you can delete it and retype the new word. Since the new word might be longer or shorter than the old word, you may need to reformat the paragraph. If, however, the string is not found, the word processor will usually display some message like STRING NOT FOUND — PRESS ESCAPE. You would press the ESCAPE key



(remember we said it would come up), and the word processor would simply return to where it was when it completed the search.

As an example of how the Search function is used, suppose you had typed the word "ITEL" where you had meant "INTEL" in 25 different places throughout your document. To fix this on the word processor, you would simply request the word processor to find "ITEL." The cursor would move to the first occurrence of "ITEL" and give you the opportunity to replace it with "INTEL." To find the second and subsequent occurrences of "ITEL," you would usually press the FIND AGAIN key, causing the word processor to find the string and move the cursor to it. (On some systems you may have to retype the string to be found.) Or suppose you have labeled the chapters in your document Chapter 1 to Chapter 8. You can move the cursor from anywhere to the beginning of one of these chapters by simply doing a search (or a find) for Chapter N, where "N" is a number 1 to 8.

The duration of a search indicates the quality of the package you are using. On a RAM-based word processor, where the entire document is in memory at the same time, a search is quite fast. On a disk-based word processor, where the document can be as long as the disk allows, a search takes longer because frequent reading and writing of the disk is required.

Search and Replace

Searching is powerful, but most word processors have the ability not only to find a word but also to replace that word automatically with a new word! The best part of this is that you don't need to type the new

word more than once. For example, in the case of the misspelled word “INTEL,” we could specify search and replace and be asked only to type in the word to be found and the word with which to replace it. The word processor moves the cursor to the first occurrence of the misspelled word, deletes it and then inserts the replacement word.

GLOBAL SEARCH-AND-REPLACE OPERATIONS

To really enhance the power of the word processor, manufacturers usually add to the search-and-replace function some additional features known as *global* operations. This term describes an operation performed repeatedly throughout the entire document. In our examples so far, we had to press a FIND AGAIN key to search for and replace the next occurrence of the word. In better word processors, there are usually options that allow you to specify an automatic find throughout the document. There is also the option of specifying that an operation occur a certain number of times, or that it be semi-automatic and have a yes/no permission question asked by the program each time the string is located.

With the yes/no-permission option, the user can skip a particular replace by typing “N” for “no” or permit it by typing a “Y” for “yes.” The permission feature can be suppressed if you are confident about the replacement. In these cases the entire operation will occur automatically without human intervention. However, such free-running, unmonitored, automatic replacements have potential dangers. They may change justification and alter the text in unexpected ways. For example, if you asked for all occurrences of the word “HE” to be changed to “SHE,” the word processor might change “HEADING” to “SHEADING.” So beware of unmonitored, automatic, totally global search-and-replace operations.

Most word processors do their searching in one direction in the file, starting at the current cursor position. So to search from top to bottom you must first move the cursor to the beginning of the text (the top). On some word processors there is an option that will allow you to search backwards should you be at the end of a long document. There may also be a command to ignore the distinction between upper and lower case (so you will find both occurrences of “Age” and “AGE,” for example).

It is most important that there be a command to search for whole words only, that is, words that have a non-letter or non-digit character before and after them. This prevents the word processor from replacing strings that are inside other words. For example, to change the word “FORMAT” to “DESIGN” automatically, you would not want the program to stop on the word “INFORMATION” and change it to “INDESIGNION,” would you? Therefore, you would specify *whole words only* in your search-and-replace command.

WORDCRAFTING TIPS:**INSERT ANY PHRASE IN ONE KEYSTROKE**

If you have access to a word processor, try this one. Give the search-and-replace command. Answer "Find?" with a carriage return (find nothing), and answer "Replace?" with a word or phrase you wish to insert. For example, the string could be a company's lengthy name, the name of a type of product, or a legal phrase that is used often. Now, whenever you type REPLACE-AGAIN, your string will be inserted automatically. You type the string once and use it a number of times.

Special Characters for Find/Replace

On some systems there are special characters you can use with a search-and-replace command to make it more powerful. A CONTROL-N in a search string may mean to match a carriage return, so you can replace or find items only at the beginning or end of a line. For example, answering "Find?" with "^NSection 5" will only find a "Section 5" that is preceded by a carriage return line feed, or in other words, one that is at the beginning of a line. There may also be commands to match any single character, match any character not a letter or digit, and match any character other than X. This last command might be used to search for a bad notation in the document. You could also use a beefed-up search-and-replace command to put "|" characters on either side of some text to form a box. All of these options increase the sophistication of the search function. You may never need all this power, but it's nice to know that it's available.

BLOCK OPERATIONS

Block operations are used to manipulate paragraphs and larger sections of text on the word processor. In the world of typewriters, whenever you have to cut out paragraphs with scissors, and paste them in another location in the document, you are performing a block-move operation. In word processing, a block is usually a number of continuous lines of text marked by a beginning and an end marker. The marker may be a special character you enter into the file, or it may be a unique word at the beginning and end of the file. Markers are inserted into the text by putting the cursor where you want the marker to be and then pressing certain keys on the keyboard. A character of some sort will usually appear in that spot. It will be a non-printing character such as the ones we use for embedded printer control. Its only purpose is to show us where the text block begins and ends.

Once you have marked the beginning and ending lines of a block, either by markers or unique beginning and ending words, you are ready to move it. Almost all the less expensive word processors can move the

block to a new location in the document (sometimes called “cut and paste”). You can copy or reprint the block in as many places as you want. You can also delete the block. (Always use extreme caution when deleting because there is usually no simple way to recover a lost block.) There is a certain distinct set of keystrokes for each of these functions, or a menu of special options which enables you to perform the desired function on the block.

The process of moving a block is illustrated in Figure 5.1. On this word processor, we assume that the paragraph to be moved appears on the screen. As we mark the block’s beginning and end, these markers appear on the screen as shown in 5.1a. Then we move the cursor to the place in our document where we want the block to be relocated, as shown in 5.1b. Now we are ready to move the block. We press the BLOCK MOVE key(s) and the block we marked earlier appears under the cursor, as shown in 5.1c. If you look for the old text, you will find that it’s gone. The text below the cursor was pushed down to make room for the block automatically — no scissors! That’s all there is to it. You could just as easily have copied the marked text to the new location so the old copy wasn’t deleted. Or you could delete the text block with the CORRECT key(s). The nice thing about these operations is that you can do them over and over until the ideas flow in the proper sequence.

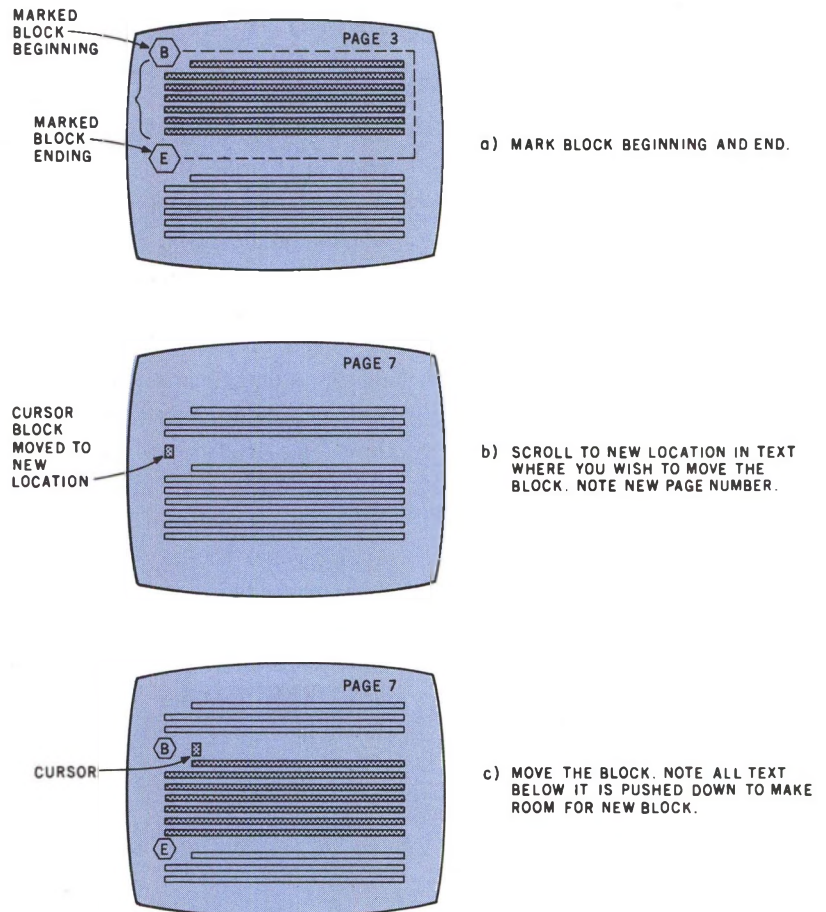
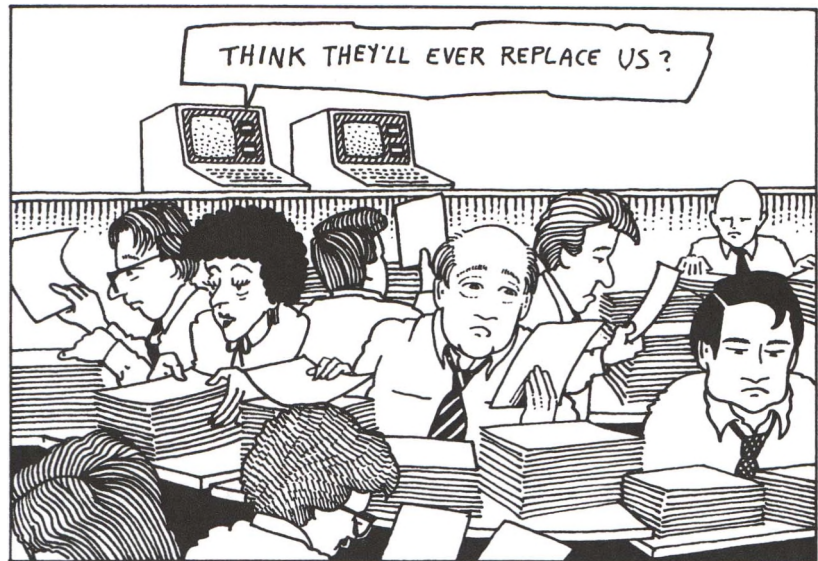


Figure 5.1: Block move operation.

BOILERPLATE TECHNIQUES

More expensive word processors have extra block operations that are extremely helpful. The most unique of these are the *write block to file* and *read block from file* operations. With the former, you can send any paragraph or continuous lines of text to a file on the disk, giving it a distinct name. With the latter, you can enter this paragraph into your text at any time. These functions are very useful when you want to assemble a document from a collection of fixed, standard paragraphs. The technique is called *boilerplate*. The main body of text is the “boiler” and the standard paragraphs are the “plates” used to mend it. Anyone interested in producing contracts, legal agreements, mass mailings, or lists of requirements will find this feature a lifesaver.

In some word processors, you can store a block or blocks of text as special glossary blocks. Then, when certain keys are pressed, one of several fixed blocks of text is brought into your current text file and appended to it permanently.



File block operations are also used for merging different sections of a document into one long piece. You can move to the end of the current working document then read in another section that is stored on the disk under a different name. It will be appended to the end of the current working document.

Another really helpful block operation is called *column move*. Column move allows you to move not just lines of text but a distinct column of text. In column-move operations, the markers define a rectangle enclosing the text you wish to move. The beginning marker locates the upper left corner and the ending marker the lower right corner. This is extremely useful when you want to construct a document with a multi-column format. For example, you would write your paragraphs with normally wide margins and reformat them using 1/2- or 1/3-column margins. Finally, you could move the columns so they are next to each other.

There may be other ways to mark blocks than with beginning and ending markers. For example, some of the more expensive word processors allow you to put as many as ten special numeric markers in the document. These allow you to specify several spots in the text. There are usually commands that cause the cursor to move to any one of these numeric markers with ease.

SUMMARY

These last chapters have focused only on the mechanical text functions the word processor performs. We have covered the use of search-and-replace, block operations, and boilerplate techniques. It would probably be fine with most people if the word processor had no

other tricks in its repertoire, since its advantages over a typewriter are boggling. There is, however, another entire class of operations that supplement the word processor's mechanical advantages. We are referring to add-on software word-processing tools that increase the power of text manipulation. In the next chapter we will dive into these other wonders that allow automatic spelling correction, automatic word indexing, and even grammar correction!

Word Processing Software Tools

Add-on software tools can extend the capability of the word processor beyond mere editing and printing. The market for them is probably even more competitive than the market for microcomputer word-processing software. What kinds of things can a word processor do besides standard editing? For starters, you can buy a spelling-corrector program that automatically checks the spelling of your entire document. It finds all your spelling errors and even corrects them. Some of these spelling-checking programs actually remember the words that you misspell frequently and automatically correct them without your help!

There are powerful indexing programs that read your document looking for special words that you have marked (using embedded commands) to be put in the index. The index program then prints all the indexed words in an alphabetical list, with page numbers next to each entry. Done manually, a job like this could easily take a week. The index program does it in less than 10 minutes. To further simplify the problem of organizing the heading formats for your document, you can purchase a program that generates headings and tables of contents. This little gem will automatically read your document and find the headings that you have marked with special characters. These characters symbolically tell the level of the heading (first, second, third) and enable the program to number the headings in sequence. The program will also generate a complete table of contents with page numbers.

To help the marketing department manager or a newsletter composer, there are mail-merge options that turn your word processor into a mailing list management system. A mail-merge program will accept name and address records (packets of information) created by the word processor in a certain format. It will then merge them into a certain document, such as a letter, to create a completely personalized letter to each person on the list. These mail-merging programs also allow the insertion of a nickname or first name in the middle of the letter! You may have received a letter created by such a program that says "Dear Mr. (Your Name Here)" and repeats the use of your name in the body of the letter.

The joy may at last return to Mudville (the fictional home of Casey and the English teachers) with the invention of the grammar-checking and correction program! A grammar-checking program reads your

document and tells you when it thinks a particular phrase is archaic, informal, or awkward. It will help you clean up your writing style, improve your word usage, and prevent your presentation from becoming overworked, trite, redundant, or vague. Grammar-checking programs border on artificial intelligence in that they can teach you how to improve your writing without ever giving you grades for your efforts.

For a peek around the corner, there is a developing market for picture-drawing programs that can be easily included right in your text file on your word processor! These are called graphics-figure processing programs. With such a program you could throw away, along with the correction fluid and scissors, all your pencils, rulers, triangles, T squares, and compasses.

With so many people beginning to use word-processing programs, it is not surprising that the market for add-on tools is heating up. In the next sections we will examine some of these tools in more detail and see what they mean to the user of a word processor.

SPELLING CORRECTORS

A spelling corrector program is one that goes through your file, looking at the words in it and comparing them with correctly spelled words stored in the spelling program's "dictionary." When a word is found that the speller thinks is misspelled, it usually presents the word to you and lets you decide whether to keep it, correct it, or put it in the dictionary.

Types of Operation

There is much to say about a spelling program. First, in terms of its operation and function, it may be an add-on program that can be used with a range of word-processing software products, an integral part of the word-processor program itself, or an option you can have installed along with the word processor. It all depends on who is marketing the product. Third-party companies do not have easy access to the insides of the word-processor program (the instruction statements that created the original cloud of intelligence), so the original manufacturer is in a better position to get the speller and the word processor working together.

There are quite a few differences in the way spellers work. A speller that is built into the editor can be used while you are in the edit mode — flashing potentially misspelled words in their original context and allowing you to correct them as desired. In this approach, all the word processor's tools for adjusting the new format, inserting, and deleting are right at your fingertips.

In another speller design approach, the spelling program is not integrated into the word processor. This type operates on finished text

files created by any of several word processors. You create your document with the word processor, save the file, and then you submit it to the spelling checker program.

Basically, there are two approaches for treating misspelled words found by this type of spelling program. First, the speller (under your supervision) can actually change the word or mark it with a special character, such as a bracket symbol (“[’”). With the marked approach you would then go into your file again with the editor, search for the “[’” symbol, and change the ones you want to change. If the word gets changed, and you have an on-screen word processor, you may need to go into your file again to clean up formatting problems which may occur when the new word is quite a bit longer or shorter than the old word. With an off-screen word processor, you may never need to re-enter the file to clean up formatting. If you use ragged-right format, you may not need to do any tidying up with an on-screen word processor either.

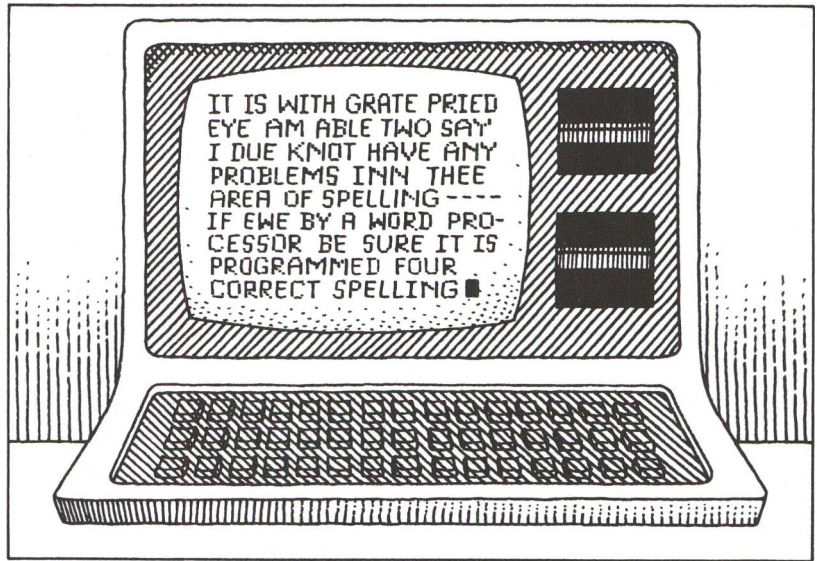
Short comings of Spellers

The spelling-checking programs were developed both to reduce the drudgery and increase the accuracy of proofreading. These programs provide you with a good tool that has tremendous ability for finding misspelled words. They cannot, however, take the place of a good human proofreader, because most of them cannot pick out words that are correctly spelled but incorrectly used in a particular context.

For example, the speller can't tell that the word “you” in “this is designed for you friends” should be “your.” It can't tell that “this is four you” is also incorrect. Most spelling errors are not of this type, but rather are caused by missing a keystroke or adding an extra keystroke at the terminal. Sometimes the operator creates a spelling that is close, but not correct. For these kinds of errors, a speller program does the job very well.

This author is plagued with a mind that rushes ahead of his fingers as he writes on the word processor, so he is very skilled at creating misspelled words. Before the advent of the spelling-checker program, it took approximately 8 hours to completely search through and correct the spelling errors in a twenty-page chapter. It would then take another 8 hours to correct all the errors that other people found. With the spelling checker, this has been reduced to a 10-minute game!

A sample spelling program session is shown in Figure 6.1. It will show you how one of these programs might operate.



Features of Spellers

There are several impressive features available on some of the spelling programs on the market. In the following paragraphs we will describe dictionaries (use of, editing, probing, and maintenance), word count analysis, guessing techniques, and speed.

General Dictionaries The spelling dictionary is simply a file containing lists of correctly spelled words. When the speller is checking your document, it is comparing your words with the words in its dictionary. If a word is not found in the dictionary, the speller assumes the word is misspelled. It is normally up to the user to tell the spelling program if the word is, in fact, misspelled, or if it is correctly spelled but simply not in the dictionary.

Dictionary Maintenance There are several ways in which this right-, wrong-, or missing-word information can be passed to the dictionary in the speller. In some cases, after the program has corrected misspellings, you can run it in a special *learn* mode. This will put into the dictionary all words in the document that were not in the dictionary. In other cases, the words are automatically inserted into a temporary file as you use the program, and when the spelling process is over, the program updates its dictionary automatically.

Dictionaries are rather large files, often containing between 20,000 and 40,000 words, and therefore consume much space on your disk. There are ways to reduce storage requirements and allow larger dictionaries. Some spellers feature auxiliary dictionaries that contain special technical words indigenous to a specific profession. You can specify that the speller use one of these auxiliary dictionaries instead of its general dictionary.

PREFACE

Did you ever wish you could understand how CP/M actually works. Have you ever wanted to install a new printer and been unable because you couldn't modify your own BIOS? Would you like to know the secrets to utilizing CP/M's powerful built-in system calls. Do you wish to write programs that run under ALL versions of CP/M? If the answer to any of these questions is yes then this one-of-a-kind book is for you. In these pages CP/M Application Primer teaches all you'll need to know to finally take control of the most popular personal computer DOS in the world: CP/M-80. This book, a companion to the popular introductory text CP/M Primer, takes you a step beyond tutorial, into the real meat of the CP/M operating system.

Starting at the most basic level the book explains just what makes CP/M so special, the idea of transportability of programs via a constant BIOS. You'll be shown how modification of BIOS allows CP/M to run on any 8080 machine has lead to a great number of CP/M systems on the market (350,000). You'll learn the number one law of CP/M: how a call to BIOS in one system is a call to BIOS in all systems. In the next chapter we reveal the basic organization of CP/M, its BOOT, TPA, CBASE, FBASE, etc and their purpose. You'll learn how memory is divided up, where programs are run, what no-mans land is, where your disk controller might be, the different boot locations, etc. We'll show how the disk surface is laid out, how the directory is stored on the disk, the way transients and utilities are run, and how the FCB is set up.

Next the book takes you by the hand with DDT and through examples teaches all the non-disk CP/M system calls. You actually write short programs from DDT and run and test them. **IN LESS THEN 30 PAGES YOU WILL BE WRITING AND RUNNING ASSEMBLY LANGUAGE PROGRAMS!** This remarkable approach originated by Mr. Waite, (using DDT's mini-assembler and its disassembler) make this fast learning possible. Long editing sessions are avoided this way.

Once an understanding of the non-disk calls is in hand you are ready for the real power of CP/M, its disk calls. The next chapter explains through several examples, how to use the CP/M disk calls in your programs. You will learn how to search the directory, how to open a file and read from it, and how to write data to a file. Applications in this chapter require use of an editor and assembler. We use ED and ASM since they are part of standard CP/M. Programs covered are for determining number of words in a text file, displaying top "n" lines of a text file, using wildcards on both of these applications, comparing two files and displaying any inequalities (as in Unix), and a fast disk to disk copy program.

The last chapter shows you how to modify your own BIOS. We present a simple BIOS listing that has a dumb "no-echo" printer driver. We explain the problem is to hook up a Diablo 630 letter quality impact printer that needs an XON/XOFF protocol to work. We show all the steps to do this on your own system.

It is hoped that through this book the budding software writer can immediately begin to write programs that run under CP/M. BASIC, Pascal, and all assembler language programmers will want to get this book so they can make their programs run under CP/M efficiently and with no bugs.

Mitchell Waite

① MicroSpell, Version 3.2 Copyright (C) 1980, Bob Lucas
Auxiliary Dict. File or <cr>:
Original text now in backup file: EX6-1.QXT

② Pass #: 1 A-D

③ APPLICATION + S

④ could understand how CP/M actually works. Have you ever wa

⑤ ACTUALLY I Guess:

- ⑥ 1) ACTUARY 2) ACTUAL
 3) ACTUALLY
- ⑦ *c3
- ⑧ w printor and been unable becuae you couldn't modify your
 BECUASE I Guess:
 1) BECAUSE
- *c
- ⑨ CALL + S
 COMPUTER = COMPUTE - E + OR
- ion of CP/M, its BOOT, TPA, CBASE, FBASE, etc and there porp
 CBASE
- ⑩ *i
- CALL + S
 DDT'S = DDT + 'S
 CALL + S
 CALL + S
 APPLICATION + S
- and assembler. We use ED and ASM since they are part of sta
- ⑪ ASM I Guess:
- 1) AS 2) ASH
 3) ASK 4) ASS
- *i
- DETERMINING = DETERMINE - E + ING
 APPLICATION + S
 COMPARING = COMPARE - E + ING

Explanations of Spelling-Checker Program Results

- ① The program was started by typing SPELL EX6 - 1.TXT. Three lines of initial information are then printed by the SPELL program. The first line indicates the name of the program, the version number, and the copyright notice and date. The second line asks what dictionary to use. A carriage return here <cr> means use the standard, built-in dictionary. A name and a drive indicate a special user dictionary that may, for example, contain unique technical words. We pressed <cr>. The third line shows the old file is now backed up with the same name, but the first letter of the extension has been changed to a "Q" so the backup is now EX6-1.QXT.
- ② The program runs over the source document in four passes; this is the first pass, covering words beginning with the letters A through D.
- ③ The spell program displays the roots, suffixes, prefixes, and combinations of words that it calculates are legal and displays them for us to check visually. This particular program sometimes makes mistakes, so we must watch for these. You would write them down and go back and fix them later.
- ④ The program prints out one line of text containing a word it thinks is spelled wrong.

- ⑤ Below the context line that the spell program prints out, the misspelled word is displayed in all caps (here the word ACTUALLY is spelled wrong (as "ACTUALY").
- ⑥ The program then analyzes the misspelled word and, if possible, presents a collection of words that it thinks might include the one desired by displaying "I Guess" and the list. In this case three words are presented as possible corrections. One of them (the third) is correct.
- ⑦ The program then prints an asterisk (*) prompt and waits for the user to enter one of several possible commands to indicate what to do. In this case we typed a "c3," which means "choice number 3 is correct." The program will then automatically replace the word ACTUALY in the source file with the word ACTUALLY.
- ⑧ Here the program correctly found the misspelling of the word "because" and presented the correct spelling of it.
- ⑨ This shows the weakness of this type of spelling program. The word computer was spelled wrong as "computer," yet the program decomposed it improperly as being correct! We would have to go back into the file with the word processor and correct it.
- ⑩ Here we use the "i" command for insert, which tells the program the "found" word is actually spelled correctly and simply isn't in the built-in dictionary. The insert command simply says "put this word in a temporary dictionary that exists for the duration of this session and don't call it a misspelling if it occurs again."
- ⑪ Here is another example of the program making a mistake.

Figure 6.1: *Spelling correction session.*

A spelling dictionary can grow rather quickly as you first use it because there will be many words in your writing vocabulary that are not in the dictionary. (It is easy to envision a market for lists of words that can be added to your dictionary to make it smarter.) After using the speller on several of your principle documents, however, you will find a decrease in the number of new words it finds. (This is true because the size of your vocabulary tends to remain constant.)

Dictionary Editing There should be commands in the spelling program for listing and editing the contents of a dictionary. This is important because it is easy to allow misspelled words to creep into your dictionary and wreak havoc with your life. As long as there is a misspelled word residing in the dictionary, the program will not catch it when it is encountered in the text. Some programs display the words that they accept as they are encountered. This gives you a chance to notice any misspelled words in the dictionary so you can remove them.

Dictionary Probe A dictionary-probe command is a feature that searches through the dictionary for the proper spelling of a certain word of which you know only the first few letters. For example, if you enter the probe mode and type "ex*," all the words beginning with "ex" will be displayed by the spelling program. The one you are looking for may appear, saving you from dragging out Webster's.

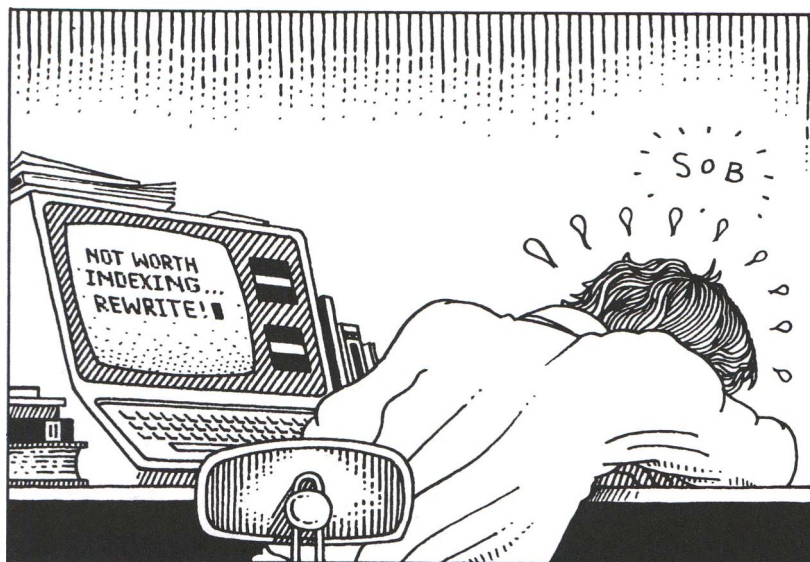
Word Count Analysis After the spelling analysis, most spellers can give you statistics about the words of your text file. Standardly, you are given the total number of words read. Some programs also provide information on the number of different words in the document. If the total words read was 1000 and the number of different words was 500, the percentage of unique words would be 50 percent. This information can help you figure out if the reading level is too high or too low for your audience. Other programs will even provide the number of misspelled words. This is an indicator of how poor a speller you are. Some systems allow you to suppress this figure when your friends are present.

Guessing Misspellings There are many different ways the spelling program guesses that a word is misspelled and several ways it may deal with misspellings. In some cases, the program will create a special misspellings list that contains your frequently misspelled words and their correct spellings. This allows the program to find and correct words on its own. In general, however, you are presented either with a list of misspelled words or a single misspelled word in its context.

Some spellers will present the misspelled word and a selection of correctly spelled words with a similar configuration. You can then indicate which, if any, of these guesses is correct. This feature can make you think you are dealing with a genius computer.

Speed The time that a speller takes to complete a document depends on several factors, including the length of the document, the type of computer used (specifically, its clock speed and size of memory), and the actual correcting technique employed. (This last factor causes speed claims to vary widely.) For example, one manufacturer claims that it takes one minute to proofread a 10,000-word document. By “proofread,” however, the company means only that the document has been marked, and not corrected, by the speller. The time it takes for the editor to go into the file and search for and replace the misspelled words must be included in an accurate estimate of completion time.

Accuracy Not all spellers will catch all misspellings, thus you must watch for potential slip-ups. The way that some spelling programs analyze words makes it possible for misspelled words to be accepted by the speller. For example, one word speller wouldn’t find anything wrong with the word “twoing.” It would think it was a combination of the word “two” and “ing”. It might even gleefully display the acceptance of such a word. But, since these are not typical spelling errors, their occurrence should be infrequent.



INDEXERS

One of the most time-consuming tasks in putting together manuals, technical manuscripts, works of non-fiction, and other long documents is creating the index. Book publishers usually employ individuals whose only job is to go through a manuscript, find key words, note the pages on which they occur, and put them in an alphabetical list. Such a job is not only time consuming, it is also very difficult for someone not familiar with the manuscript's subject to locate the proper words for the index. Wouldn't it be wonderful if we could earmark words for the index during the writing process and the computer could create the index for us? This is precisely what the new word-indexing programs do. Often, they are separate programs designed to work on finished text files. Here's an overview of how it's done.

A typical indexing program uses embedded commands that are recognized during display on the screen, but ignored during the printout cycle. Whenever you finish a paragraph, you examine it for any words you want to include in the index. Type the index command first, then the word you want. That word will later get put into the index. On some word processors you don't have to type the index command, you can just bracket the word with embedded commands as we did to do bold strike and underlining.

When you complete the document, there will be words with index commands marking them. Then you run the index program and submit your document to it. The program searches through your file collecting all the indexable words and noting page numbers. When it is finished, it creates a sorted list of these words and installs the list in a new file, complete with page numbers. Figure 6.2 is an example of the output of such a program.

A

Automatic

- Figure Numbering, 10
- Heading Numbering, 10
- Mail Merge Reformat, 11

B

Boiler Plate

- Mail Merge, 14

C

Chained Printing, 11

Comment line

- Word-Index Command, 10

Comment lines

- use of, 9

Computer Graphics, 15

Control-Z

- End of File, 4

D

Database Program, 1

Database

- Mail Merge, 11
- Mailing, 13
- Suite and Building Names, 13

Dictionary

- Spellguard, 7

E

Ethernet, 16

Figure 6.2: *Indexing program results.*

Most of the indexing programs have a provision that allows you to have multi-leveled index items. For example, you often have one main subject in the index and several variations of it explained in different parts of the document. You may have an index entry called Memories, followed by ROM, RAM, disk, size, and cost. A good index program will differentiate between such levels.

TABLE OF CONTENTS GENERATORS

Table of contents and heading-numbering generators are similar to indexing programs. A table of contents generator liberates you from the chore of actually numbering the headings and chapters in your work. It is common in the engineering, electronic, and computer industry to use a special numbering system for designating the different levels of subjects in a document. For example, the first level may be 1, the next level down would be 1.1, the third level would be 1.1.1, and so on.

This numbering scheme effectively organizes documents but also puts great strain on the writer, who must number the headings as he or she writes. If a new subject must be added later, the entire document may need to be renumbered.

A table of contents (TOC) or heading-numbering program solves this problem by requiring the user to use “symbolic” letters to represent the levels of the various headings. Instead of numbering a heading, like “1.1 Mild Spices,” you would simply enter a code letter that indicates what the level of the heading is, such as “B Mild Spices.” For example, a chapter heading would be typed following the command letter “A,” a second heading level would follow “B,” the third would follow “C,” and so on. These heading level commands are embedded in the file and are only recognized by the specific TOC or heading program.

When your document is finished, it is submitted to the TOC program. The TOC program goes through it and collects the headings, their levels, and the pages on which they are found. The program then creates a second version of your file that contains actual numbers to replace the symbolic commands that you inserted. In addition, it creates a table of contents with level headings and their page numbers. The TOC program usually has the ability to number figures and tables as well. Frequently heading numberings are done in military (MIL) specifications and can be changed if the user is prepared to go into the program and adjust the indexer-overlay files.

MAIL MERGE

Mail merge is the amazing ability to take information stored in a data base and merge it into the printing of standard documents. The data base can contain names, addresses, and other pertinent information for customers, club members, survey participants, magazine subscribers, or political contributors. It is usually created with the same word processor or via a special data entry program.

A company can use mail merge in several ways. For example, in the production of a survey aimed at a group of customers, the survey designer would compose a letter in which the names in the data base would be inserted to make the letters seem more personal. A good mail-merge feature allows you to insert information such as the person’s first name or a company’s name. The word processor usually will automatically reformat the paragraph during printout, so these insertions cannot be detected by the reader. Conveniently, mail merge programs are also used to print mailing labels from names and addresses in a preformed data base.

You may object to using a letter-quality printer for custom letters to all the individuals in your data base. Since the printer is slow, the

cost of each letter is substantially higher than if copies are used and the names typed in later. On the other hand, the personally typed letter has more appeal and results may justify the extra cost. A printer with higher speed and slightly lower quality, such as a band printer, may make this operation more practical.

Another use of the mail merge, which has nothing to do with mail, is chained and nested printing. Chained printing allows you to print out any number of files sequentially. For example, you can create a command sequence that will print out all the chapters of a book in succession. Nested printing allows you to insert the contents of another file in your document as it prints it out. This is useful for inserting boilerplate copy and paragraphs into your documents.

How Mail Merge Is Used

A typical mail-merge program works as follows. We will assume that the task calls for the printing of a letter to several individuals participating in a survey. Each participant in the survey is an expert in some field. Figure 6.3 gives a sample of the document that would be prepared by your word processor.

```
..file SURVEY1.LET      (this is a comment to WordStar)
.OP                     (this says omit page numbering)
.DF ADDRESS1.DAT        (this is the name of the data file)
.RV COMPANY, ADDR1, ADDR2, ADDR3, NAME1, NAME2 (these are variables
                                     that occur in the database file)
```

```
&COMPANY&              (the name between the ampersand is taken
&ADDR1&                 from the database file and printed here)
&ADDR2&
&ADDR3&
```

Dear &NAME1&,

Enclosed is a survey on several book ideas I would like to get your opinion on. This survey is similar to the last one you filled out, &NAME2&. I would like you to rate how you like the book on a 1 to 10 scale, where 1 means you are not interested at all, and 10 means your interest is extremely high. In addition, &NAME2&, I would like your opinion on what you think the market is for such a book idea today, on a 1 to 10 scale, where 1 equals terrible no market and 10 equals very large market. As before I will send you a copy of any books written by me as a result of this survey and in addition I will thank you in the Acknowledgements section of the book. I hope all is going well with you and your work and look forward to receiving your opinion. Thanks in advance.

Sincerely,

Mitch Waite

Figure 6.3: Sample use of mail-merge program.

Most of the features of the letter are fairly obvious. First, the name of the file containing this document is given in a comment line at the beginning of the letter. The next command turns off the automatic page numbering since we're dealing with a single-page letter. The next embedded print command, (\ DF), tells the mail-merge program the

name of the file that contains the data base for this letter, i.e., where the names and addresses will come from. The next embedded print command, (\RV), stands for "Read Variable" and tells the mail-merge program the names of the variables to get from the data-base file and the format in which they will appear. In this case, we expect the file to be ordered by company, three address fields, and an individual name.

Our data-base file "ADDRESS1.DAT" will contain fields separated by commas. Two names and addresses in the file are shown in Figure 6.4.

```
Joint Venture, 6666 Subroutine Way, "Free Space, California",
94911, Doctor Rose, Larry
Solid State Snorkels, 101 Wetsuit Way, "Dry Ocean Park,
Indiana", 46200, Mr. Bitlevel, Barney
```

Figure 6.4: *Typical name-and-address mail merge data base.*

When the mail-merge program is run, it will print out copies of your letter, inserting information from the data base, one record at a time, until every name and address in the data base has been used. The above letter might appear as follows:

```
Joint Venture
6666 Subroutine Way
Free Space, California
94911
```

Dear Doctor Rose,

Enclosed is a survey on several book ideas I would like to get your opinion on. This survey is similar to the last one you filled out, Larry. I would like you to rate how you like the book on a 1 to 10 scale, where 1 means you are not interested at all, and 10 means your interest is extremely high. In addition, Larry, I would like your opinion on what you think the market is for such a book idea today, on a 1 to 10 scale, where 1 equals terrible, no market and 10 equals very large market. As before I will send you a copy of any books written by me as a result of this survey and in addition I will thank you in the Acknowledgements section of the book. I hope all is going well with you and your work and look forward to receiving your opinion. Thanks in advance.

Sincerely,

Mitch Waite

Figure 6.5: *Example of mail-merge output.*

As you can see by the bold-faced words, the mail-merge process produces a letter which seems hand-typed and personally directed by the author. (The addressee's name was made bold to help you find it; in a real letter this would not be the case.)

Operator Data Entry

Company marketing departments often send letters acknowledging the order of a particular product. These letters must mention the

quantity of items ordered as well as other coding numbers that are part of the order. A good mail-merge program should allow you to enter key information into a standard document just before it is printed. In some mail-merge programs, an embedded command standing for Ask For Variable (\AV) can be placed in the body of your form letter. When the mail-merge program is run, the computer will stop and ask the operator to enter certain pieces of information following every \AV command that it encounters. For example, \AV DATE-TODAY? would request the user to type in today's date, \AV COMPANY? prompts for company name, and \AV QUANTITY? would request the number of a certain item sold. Once this information has been entered, the mail merge will install it in the body of the letter in the places that the respective variables appear. If your letter said, "Thank you for ordering { \AV QUANTITY} copies of { \AV PRODUCT}," and the operator types *Word Processing Primer* for PRODUCT and 5000 for QUANTITY, then the mail merge program will print "Thank you for ordering 5,000 copies of *Word Processing Primer*."

Mailing Labels

As mentioned above, a good word processor should allow you to produce mailing labels. Of course, the simplest way to do this is to collect your names and addresses into a format that looks exactly the way you want it printed on the labels. In other words, you could have a continuous strip of labels (one label wide) in the printer and have all your names and addresses take up exactly the right number of lines so that each begins on a separate label. This is quite cumbersome, though. Adding a new item to the data base can mess up the entire format for the printer. It is better to use mail merge and create a document with just the commands for the necessary label variables. (The document would also need commands to deactivate the top and bottom margins for continuous-forms labels.) Your data base would be the same one used in the previous letter, except that only the name and address portion of it would be required. You can usually have the mail-merge function print this same information on your envelopes.

It should be possible to include more detail in the data base so that you can pinpoint an address more accurately. You may wish to include a suite number in the data base, or a building name. Usually, the data base will allow you to enter these items in the form letter and leave them out of the data base if they are not applicable.

Other Features of Mail Mergers

You may need to send messages to the screen in order to help the operator enter information. This display information can contain variables that are part of the data base, such as “printing letter to {NAME}.”

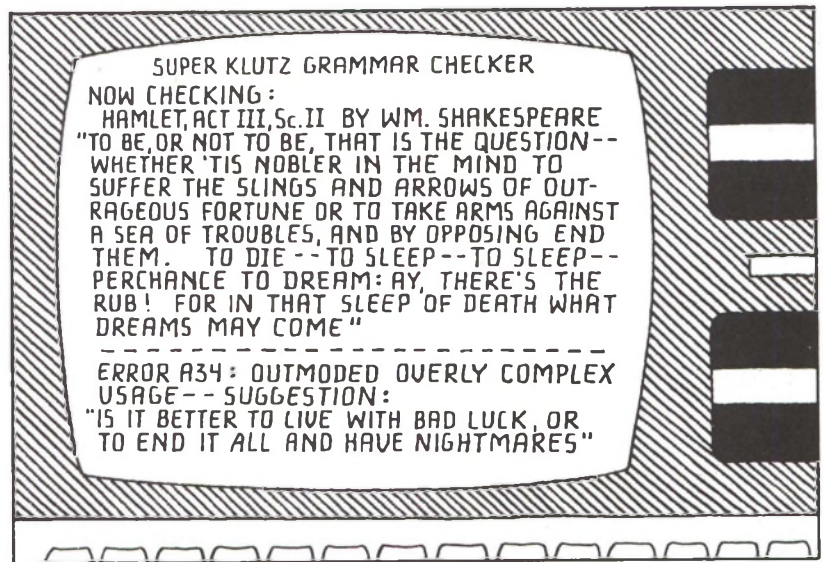
A mail merger should allow for file insertion. This means that the word processor can read standard text from other files and insert it in proper order before printing it out. This feature allows you to automatically create letters through boilerplate text. You can combine operator-data entry with boilerplate for some very sophisticated printing. By first requesting the date from the operator, you should, for example, be able to create a multiple command file that will print form letters and then properly addressed envelopes.

AUTOMATIC GRAMMAR CHECKING

One of the most exciting developments in word-processing add-on tools is a program for checking the grammar, style, and typographical error rate of your text file. A grammar-checking program goes beyond spelling checking in that it looks at phrases in your text and tells you when a phrase is archaic, informal, or uses incorrect gender. It also tells you when a phrase is awkward, commonly misused, overworked or trite, redundant, improper, vague, or wordy. It is as if you had the perfect English teacher watching over your shoulder. Of course, the grammar-checking program assumes that you have run a spelling check on your text.

A typical grammar-checking program works by storing a set of rules for word phrasing within its built-in dictionaries. It scans through your text file and looks for phrases and words that do not adhere to these rules. When such a phrase or word is found, the program presents it to you and suggests a better way to express the thought! In addition, a grammar program will check your punctuation and tell you where periods have been left out or where there are unbalanced parentheses, brackets, quotes, or braces.

The program should find any double words like “the the” and discover capitalization errors like “someTimes.” One grammar program (Grammatic, from Aspen Software in New Mexico) provides interesting statistical information pertaining to your text, such as the number of sentences, number of words, average sentence length, or average word length. The company also provides a program called Profile that will do a word-usage study of your text file. It gives the number of different words used. This bit of information, in conjunction with the total number of words used, can help you evaluate the reading level of your document.



Using A Grammar Checking Program

You can usually set up a program so that it marks your file with a “#?x” where the errors are found. “x” represents a letter A to Z which tells the type of error that has been found. (“G” means Gender specific, “M” commonly misused.) At the same time, more complete information will be printed out describing the error and suggesting corrections. After running the grammar program, you could use the global search routine to find all occurrences of the “?,” then fix each one (if you wanted to).

In Figure 6.6, we have run a chapter through a grammar checker and extracted the most interesting errors for your inspection.

GRAMMATIK TEST SAMPLE

A spelling checker may may not be enough! This paragraph contains several common errors that will be discovered by Grammatik that would seldom ever be caught by a spelling checker. For example, Grammatik checks for improper word usage as identified by a number of writing style manuals (such as "seldom ever"). Grammatik will check for the presence of certain words such as jargon or sexist terms. it also checks for consistent punctuation, capitalizAtion, balanced quotation marks and parentheses, and and repeated words. In addition, it can produce a list of all all unique words found in your document with the number of times each was used. Grammatik comes with a dictionary of commonly misused phrases and a dictionary of sexist terms. It also includes a complete set of utilities to build,, sort, and merge phrase and jargon dictionaries of your own. Works with Scripsit, Electric Pencil, and other standard TRSDOS text files on the TRS-80. Works with all popular CP/M word processors such as Wordstar or Magic Wand.

- ① Aspen Software Grammatik (tm) CP/M-V1.6
(c) (p) 1981 Aspen Software Company

- ② Reading dictionary: PHRASES.GMK
Remaining free chars for dict: 3409

- ③ ***** Enter any command:
- | Command,<=> | optional / Meaning / | Current setting |
|-------------|---------------------------------|----------------------------|
| C<=file> | Read configuration file | CONFIG.GMK |
| D<=file> | Read a dictionary | PHRASES.GMK |
| E<=list> | Select errors to show | ABCDEFGHIJKLMNOPQRSTUVWXYZ |
| F=char | Set format command char | |
| I=file | Set input file to check | EX6-6B.TXT |
| L<=file> | List errors on file | |
| O<=file> | Output with errors marked | EX6-6B.QXT |
| P or NOP | Print errors on printer | NOP |
| S or NOS | Show suggestions | S |
| T or NOT | Show errors on TTY screen | T |
| W or NOW | Wait when error displayed | W |
| // | Done - begin checking | |
| | Use Control-C to abort to CP/M. | |

- ④ ***** Enter any command: //

GRAMMATIK TEST SAMPLE

- ⑤ A spelling checker may may
-----> may
* At sentence 1 - 'D' - Doubled word or punc.
* Press RETURN to mark error
* Press N first to not mark error
- ⑥ not be enough! This paragraph
contains several common errors that will be discovered by
Grammatik that would seldom ever
-----> seldom ever
* At sentence 2 - 'R' - Redundant phrase
* Suggestion: seldom
* Press RETURN to mark error
* Press N first to not mark error
- ⑦ be caught by a spelling
checker. FOR
-----> FOR
* At sentence 3 - 'C' - Capitalization error
* Suggestion: mixed cases
* Press RETURN to mark error
* Press N first to not mark error
- ⑧ example, Grammatik checks for improper word usage
as identified by a number of
-----> a number of
* At sentence 3 - 'W' - Wordy phrase
* Suggestion: several,many,some
* Press RETURN to mark error
* Press N first to not mark error
- writing style manuals (such as
"seldom ever")
-----> seldom ever
* At sentence 3 - 'R' - Redundant phrase
* Suggestion: seldom
* Press RETURN to mark error
* Press N first to not mark error
- ⑨ certain words such as jargon or sexist terms. it
-----> it
* At sentence 5 - 'C' - Capitalization error
* Suggestion: 1st word of sent
* Press RETURN to mark error
* Press N first to not mark error

```

⑩ also checks
    for consistent punctuation,c
----->
* At sentence 5 - 'P' - Punctuation error
* Suggestion: needs separator
* Press RETURN to mark error
* Press N first to not mark error

apitalizAtion,
-----> capitalizAtion
* At sentence 5 - 'C' - Capitalization error
* Suggestion: mixed cases
* Press RETURN to mark error
* Press N first to not mark error

    balanced quotation
        marks and parentheses, and and
-----> and
* At sentence 5 - 'D' - Doubled word or punc.
* Press RETURN to mark error
* Press N first to not mark error

repeated words. In addition, it
    can produce a list of all all
-----> all
* At sentence 6 - 'D' - Doubled word or punc.
* Press RETURN to mark error
* Press N first to not mark error

unique words found in your
    document with the number of times each was used. Grammatik
    comes with a dictionary of commonly misused phrases and a
    dictionary of sexist terms. It also includes a complete set of
    utilities to build,,
----->
⑪
* At sentence 8 - 'D' - Doubled word or punc.
* Press RETURN to mark error
* Press N first to not mark error

    sort, and merge phrase and jargon
        dictionaries of your own. Works with Scripsit, Electric
        Pencil, and other standard TRSDOS text files on the TRS-80.
        Works with all popular CP/M word processors such as Wordstar or
        Magic Wand.

⑫
* At sentence 10 - 'B' - Unbalanced (){}[] or "
* Suggestion: ""
* Press RETURN to mark error
* Press N first to not mark error

**** end of file ****

⑬ Summary for SAMPLE.TXT / Problems detected: 12

# sent: 10 ; # words: 0172
avg sent len: 17.2 ; avg word len: 4.9
# questions: 0 ; # imperatives: 1
short sent (< 14 wds): 1 ; long sent (> 30 wds): 0
longest 28 wds at sent # 9 ; shortest 11 wds at # 1
to be's: 4 ; prepositions: 27
User category totals:
    NONE
-----
Hit ENTER when ready to exit

```

Explanations of Grammar-Checking Program Results

- ① To start the grammar-checking program we typed GMK EX6-6A.TXT. This caused the grammar program (GMK) to automatically read in a configuration file (CONFIG.GMK) that contained information about how we wanted the session to be handled (we will not describe this information here). The name of the program and copyright information were then displayed.
- ② The configuration program caused the dictionary for the grammar program to be read in (PHASES.GMK). This dictionary contains the rules of grammar for the program. Other dictionaries could be created by the manufacturer of the program.

- ③ Next a menu of selections is presented to the user. Here you can select several operating features not enabled by the configuration file. As you can see the menu shows our sample file (EX6-6B.TXT) as the input file to check. We then typed 0 = EX6-6B.QXT to specify that the grammar program mark a different file with errors it finds. That way the original file is still intact. The markings for this program consist of single-letter codes placed next to the mistake. This way we would go through the marked file with our word processor and fix the mistakes. The rest of the menu is self-explanatory. As you can see it is quite flexible.
- ④ We signify for the program to start by pressing //.
- ⑤ The program immediately finds the double words "may may." It shows us the context containing the double words, tells us the mistake is a double word, and then waits for us to press return to mark the new file with a "D" or to press "n" to skip the marking (should the program be wrong). We pressed return.
- ⑥ Here the grammar program found a redundant phrase (seldom ever) and suggests that we simply type "seldom."
- ⑦ Next Grammatik finds a capitalization error "FOr." Such an error arises often when using a sticky keyboard or typing so fast that you don't unshift in time for the next character.
- ⑧ Here the program finds a so-called "wordy phrase." It suggests we replace "a number of" with "many, several, or some."
- ⑨ Note the program can tell that the first word in a sentence is not capitalized.
- ⑩ In this example the grammar program finds that we left out a space before a comma and calls it a punctuation error.
- ⑪ Here the program found a double comma.
- ⑫ This last error shows a weakness in Grammatik. It notices that the initial quote found in the third sentence ("seldom ever) never has a mate to close it. But the program tells us the problem exists in the tenth sentence (probably because this is the sentence that ends the paragraph). It should have alerted us when the third sentence ended. Oh well, programs are not perfect.
- ⑬ The Grammatik program ends with this useful summary of the makeup of the corrected document. It tells us that there were 10 sentences and 172 words. We are informed that the average sentence length is 17.2 words, which is fairly long. With this information we can tell if our writing style is fit for the intended audience. Next we are told the number of questions (none here) and the number of imperatives (commands like "go jump in the lake"). The program then summarizes the sentence lengths, telling us the longest and shortest sentences and where they occurred. It then tells us the number of to be's and the number of prepositions (here 27). When we are done we press return to end the program.

Figure 6.6: *Typical session using Grammatik.*

After running the program, 38 potential problems were uncovered. A common mistake was detected. (Didn't you think the period goes outside of the quotation marks? Wrong — it goes inside.) Right away, the program found an informal usage (the use of "lots of" instead of "many"), a triple ("the the the"), three wordy phrases ("in this case" should be "here"), a vague verb usage (never "quite"), a trademark

error ("Xerox" instead of "photocopy"), another vague verb, two common errors ("further" instead of "farther" and "all ready" instead of "already"), and two redundant phrases in a row.

Thus the grammar-checking program not only cleaned up the style and grammar of the current manuscript, it also cleared up some misunderstandings that this author had had about rules of punctuation. This kind of program can help you improve your writing style.

WORD PROFILER

A program for analyzing word usage in your document is called a word profile program. One such program (also from Aspen Software) provides the information in Figure 6.7.

```
Aspen Software Grammatik(tm) Word Use Profile V1.1
```

```
(c) (p) 1981, Aspen Software Company
```

```
Enter name of file to profile: EX6-7A.TXT
```

```
-- Reading file
```

```
-- Sorting words
```

```
Write profile to a file (y/n)? y
```

```
Print profile on printer (y/n)? n
```

```
Aspen Software Grammatik(tm) Word Use Profile V1
```

```
Total number of different words: 106
```

```
***** 1 *****
```

addition	balanced	build	can	capitalization	caught	certain
check	comes	common	commonly	complete	consistent	contains
cp	dictionaries		discovered	document	each	electric
enough	errors	example	files	found	identified	improper
includes	list	m	magic	manuals	marks	merge
not	on	other	own	paragraph	parentheses	misused
phrase	phrases	popular	presence	processors		pencil
punctuation		quotation	sample	scripsit	set	produce
standard	style	test	text	this	times	several
trsdos	unique	usage	used	utilities	wand	sort
would	writing				was	trs
						wordstar

```
***** 2 *****
```

also	checker	checks	dictionary	ever	in	jargon
may	number	or	seldom	sexist	spelling	terms
will	word	works	your			that

```
***** 3 *****
```

all	be	by	it	such	the	words
-----	----	----	----	------	-----	-------

```
***** 4 *****
```

as	for	with
----	-----	------

```
***** 5 *****
```

```
grammatik
```

```
***** 7 *****
```

```
a and
```

```
***** 8 *****
```

```
of
```

```
Press ENTER/RETURN when ready to exit
```

Explanation of Profile Program Results

This program analyzes the word use in our sample paragraph. It presents lists of words according to how often they occur in the text. In our example most words occur once, as indicated by the ***1*** in the beginning of the figure. You can see that the words "as, for, and with" occur four times and the word "and" occurs 7 times. The most popular word in this sample is "of," which occurs 8 times. Armed with this information from the profile program you could determine whether you use technical terms too often. Note that the word Grammatik occurred 5 times in the text, perhaps indicating overuse.

Figure 6.7: *Example of profile program output.*

The program shows that out of the 4938 words in our text file, 1321 of them were different. This tells us that one out of every 3.7 words was a unique word. As this number gets smaller the reading level becomes more difficult. The reader is encountering many words for the first time, and those new words are not often repeated. In addition, such a program can usually show in what frequency individual words appear. In our example, the word "computer" was used 30 times, "processor" 34 times, and "word" 81 times. The most common word was "the," used 324 times.

Using this data, you could go into the text file and do a global search on the word "computer," for example, and remove it to lower the reading level. Or you could find an important word that was used only twice and use it in several other places. The point is that the profile program reveals how words are distributed throughout your document, and from this you may be better able to characterize your writing style.

GRAPHICS FIGURE CREATION

One of the most intriguing enhancements for word processing is graphics-figure processing. Most writers are familiar with the problems that arise when figures must be generated on paper and managed separately from the rest of a manuscript. There are often problems in keeping the numbering of the drawings coordinated with the numbering in the manuscript. There is also the potential for damaging drawings or losing them entirely. One solution would be a word processor with built-in capability for drawing pictures on the screen of the CRT. Even though the technology which permits graphics and word processing to exist simultaneously has been available quite some time, no traditional word-processor manufacturers have provided graphics capability on their systems. The real impetus for graphics processing has come from the hobbyist and from the microcomputer companies who have provided personal computers with high-resolution color graphics. (This author has developed the basic ingredients of a figure-processing program that runs on the Apple II computer. It allows him to create figures from a menu of standard outline shapes, move them about on the screen, fill them with color, and print them out with a dot matrix printer.)



gutenberg software announcement

The Gutenberg word and print processor is the **ultimate** software package for preparing documents of any kind and complexity on the Apple II personal computer. It promotes creativity in the user. It allows him to combine text and graphics to such an extent, that the document can in fact be mistaken as being typeset. **This news bulletin was produced entirely on the Gutenberg using the Epson MX-100 printer.**

Multi-language documents can now be printed with an unlimited number of user definable floating accents. They can be superimposed onto the printer's standard character set to yield an unexhaustible variety of accented letters.

Science and technical writers need not look any further. The Gutenberg's secondary, user definable character sets can be mixed with text in all line justification modes. True superior or inferior figures are now possible as shown in the attached sample.

Club newsletters can now be enhanced by including large headlines, sketches, graphs, drawings and logos.

A personal letter can now become very entertaining when special affects ♥ are included in the text.

MICROMATION
NEW PRODUCT RELEASE



Figure 6.8: Output from Gutenberg program.

Apple Text and Figure Processor

A company in Canada called Gutenberg Software Limited now provides a complete word-processing and figure-processing program that runs on the popular Apple II computer. Called the GUTENBERG Word and Print Processing Program, it offers the standard text-editing features described in this book while also allowing the user to create customized sets of 120 graphic characters that display along with the standard ASCII text characters on the Apple's graphics screen. These custom characters can be from 2 to 7 dots wide and up to 12 dots high, which permits a very detailed effect. Such a feature means you can finally combine text and graphics characters in the same document. A custom heading, for example, can be added to your letterhead, thereby bypassing the need for separately designed and printed stationery, newsletters, brochures, etc. Apparently an unlimited number of custom characters can be developed, kept on line, and merged into the text when desired. Mathematicians, scientists, and engineers could use such a program for entering formulas and non-standard symbols right into the text of their documents.

The product also allows you to merge standard Apple binary picture files into the word processor's text files. Thus you could use a graphics entry program and graphics tablet to draw a picture on the Apple's high-resolution screen, saving it on disk when it's done. Or you might use a graphics program to output a graph to the high-resolution screen and then save it on disk. Then you could write about the picture using the Gutenberg text processor and finally merge the picture file right in with the text file. When the text file is read on the screen the picture appears with it! The package can then print out the combined text *and* the picture on certain dot-matrix printers that have graphics options, such as the Epson MX-80F/T, MX-100, Qume, and Centronics 739. Figure 6.8 shows an example of output from the Gutenberg program.

Retro-Graphics

What about all those people out there using standard serial terminals for word processing? Is there any way in which they can obtain graphics capability? The answer is yes. Several companies make devices that go inside some standard serial terminals (such as Lear Siegler ADM-3A Dumb Terminal (R)) and turn them into powerful graphics machines with point-plotting ability on a 512×250 grid. Similar devices are also available for the popular DEC VT100 Terminal, giving it a resolution of 640 by 480! (See Figure 6.9.)

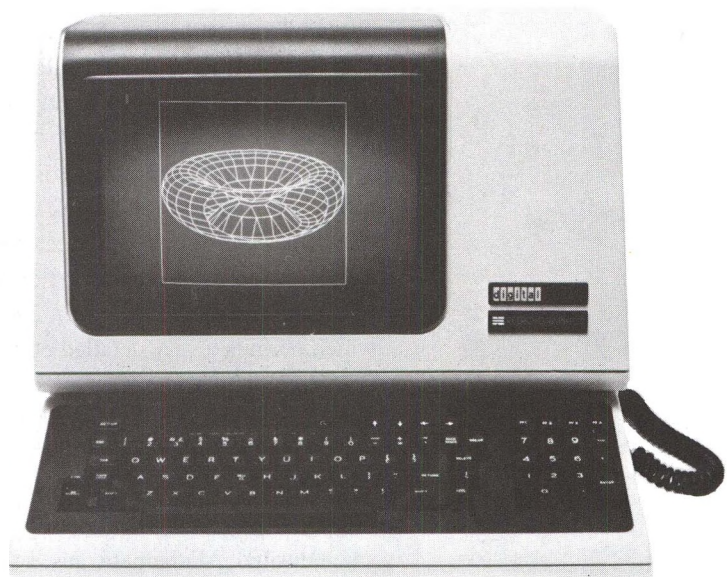


Figure 6.9: *Retro graphics in a terminal.*

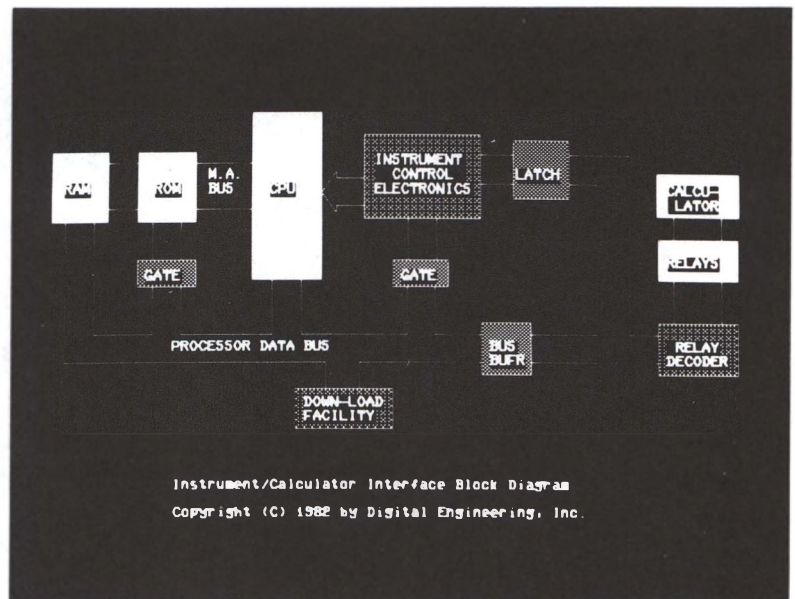
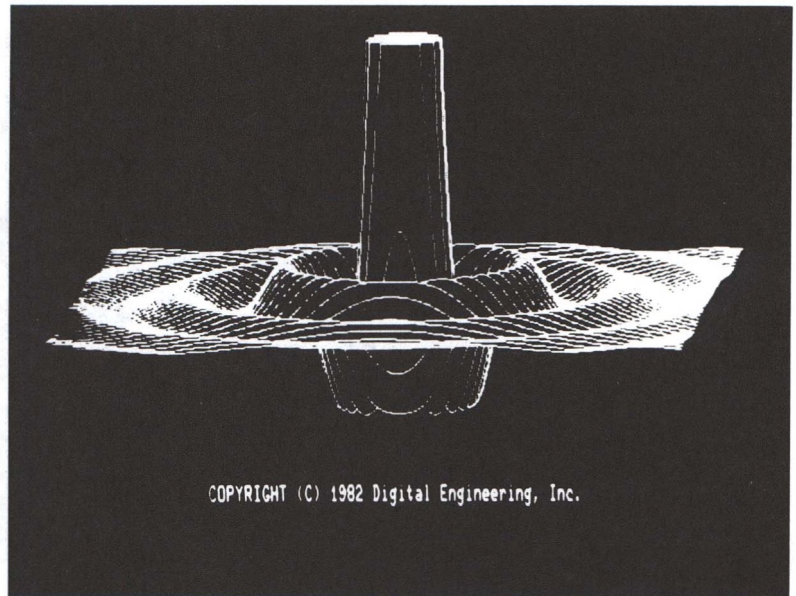
Although no software is provided for graphics-figure processing, there is usually elementary firmware provided to plot points and draw vector lines.

Given adequate software, such a graphics board can allow a very powerful figure processor to exist along with a word-processing package. Figure 6.10 illustrates some of the screen displays produced by such a product.

Figure 6.10



Figure 6.10



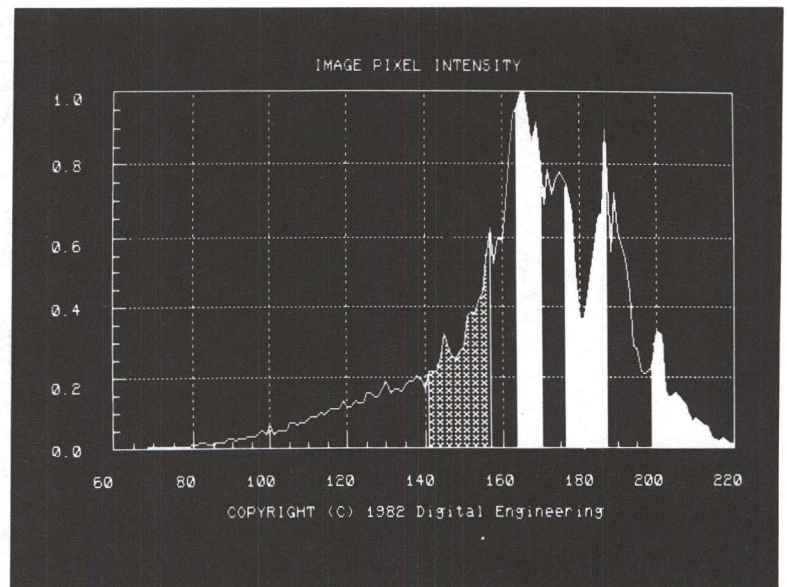
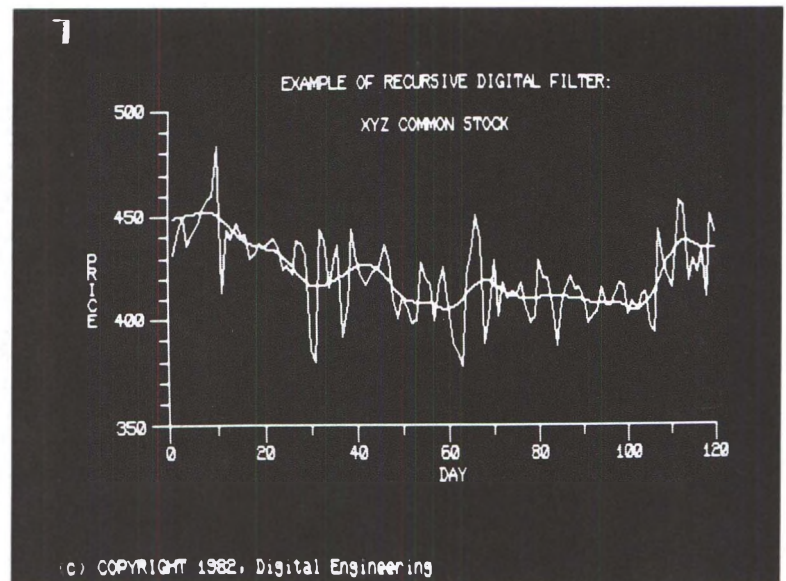


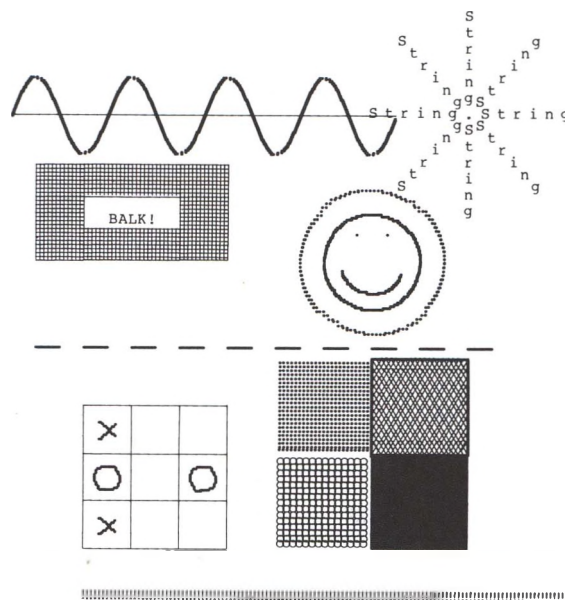
Figure 6.10: Retro graphics examples.

TURN YOUR PRINTER INTO A PLOTTER — DAISYWHEEL GRAPHICS

Since the daisy-wheel printer has such incredible resolution (1/120 inch horizontally by 1/48 inch vertically), you might wonder if it could be used as a plotting device. The answer is a definite yes. A company called Escape, Ltd. (P.O. Box 18797, Atlanta, Georgia 30326) markets a set of FORTRAN subroutines designed to produce graphs, barcharts, pie charts, plots of equations, art, and so forth on a regular daisy-wheel printer such as a Diablo or NEC.

Contained in this package are numerous routines for drawing lines, scaling pictures, and marking off special grids of all shapes and sizes. There are routines for fixed symbols, including upper- and lower-case characters of any size. As time goes on, you will undoubtedly see more programs that use the printer to plot numeric information. Integration of the plotting routines and user-friendly figure-processing software into the word-processing program is the next step in product evolution. Eventually, the user should be able to enter numbers into the word processor, describe the kind of drawing desired, and then simply wait while the printer creates it. Figure 6.11 shows some of the output of the daisy-wheel printer with plotting features.

Figure 6.11



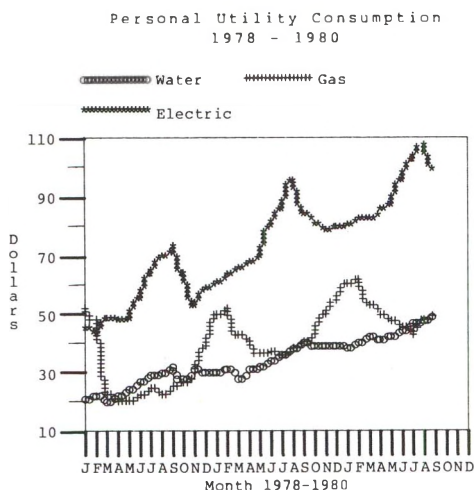
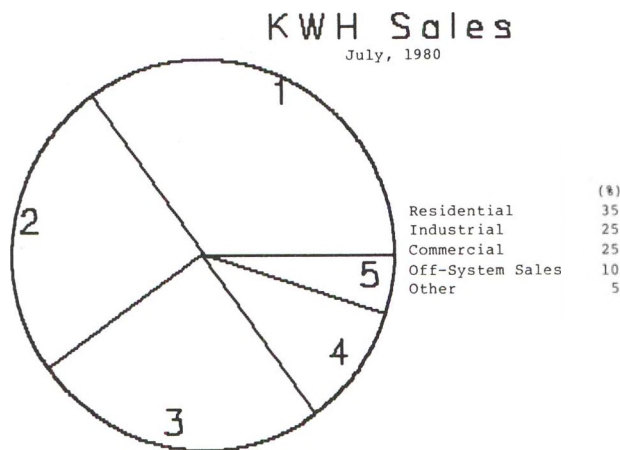


Figure 6.11: *Daisy-wheel printer plot examples.*

SUMMATION

In this chapter, we learned that there are many software tools that can give you more power than editing alone can provide. We saw that there are spelling correctors, index and table of contents generators, mail-merging programs, grammar-checking programs, and even graphics-figure processing packages available for your word processor. Now you should have a good understanding of all the things that make a word processor tick and the wonderful things it allows you to do with words. In the next chapter, you will learn how to select a word processor for a particular application.

How To Select a Word Processor

In the last two chapters, we discussed the ways that add-on software packages can be combined with the main text-editing program to vastly increase the power and flexibility of the word processor. You actually learned how to use the word processor to edit a document, and by now you are aware of the tremendous possibilities that the word processor offers. Even though you are convinced that owning a word processor would be extremely beneficial, you need to know a bit more before you go out and buy one. The first thing you should know is that all word-processing programs are not created equal. (In fact, just like people, you will probably never see two that are identical.)

There are a great many text-editing programs on the market today, and this fact alone makes choosing the best one a difficult task. To further complicate matters, programs stress different features and approaches. Some are geared for light-performance home use and have just a few of the more powerful features. Some are set up for high-powered text editing and possess many advanced features. These are consequently more difficult to learn to use (but are not necessarily impossible to master, given a reasonable amount of time). Still others are designed for specific professional users like attorneys, doctors, secretaries, or authors.

What combination of computer and word-processing software will best suit your particular needs? If you are a home user, what particular things should you look for in a word-processing program? If you are a writer, what is most important?

The answers, it turns out, depend on many factors, some of which are hardware (computer) related and some of which are software (program) related. Since this book discusses word-processing programs more than equipment, we will assume that either you have already decided what particular hardware you want or you are examining word-processing program features and plan to choose your computer after selecting your software.

Before we get bogged down in debating one approach against another, let's clarify one thing. In many ways it doesn't matter which word processor you end up with! The truth is that going from a typewriter to a word processor is such a giant leap that regardless of which particular one you end up with, it will feel like you are way

ahead of where you were before (as indeed, you will be). Of course, there are many levels of performance among products, but whatever decision you make will definitely increase your efficiency and accuracy, and that's what really counts. To date, this author has not discovered any word-processing programs that could be considered "bad" or any that are as inefficient as a typewriter. Still, it would be wise to read this chapter before you rush out to purchase a word-processing program or a personal computer to run it on. Our goal is to help you avoid buying a system that does less than you desire, is more complex than you want, or leaves you in a corner when you're ready to expand.

Word-processing requirements are often similar within the same professional group, and exposure to systems currently used in your profession may help you determine the most important features. To aid you in this determination, this chapter will examine the needs of five different classes of word-processor users:

- Attorneys
- Secretaries
- Mass Mailers
- Authors and Technical Writers
- Home Users

Although many word-processing salespeople will try to dazzle you with the multitude of features available on their word processor, it has been our experience that there are only a few features which should radically affect your choice. Therefore, to make your buying easier, we will introduce seven features which we believe serve to separate one word-processing program from another. These features are:

- Memory-based vs. disk-based
- On-screen formatting vs. off-screen formatting
- Screen display size
- Disk capacity and number of drives
- Quality and quantity of the editing commands
- Help menus
- Documentation

After explaining the individual professional's word-processing needs, we will discuss each of these features and its importance in the choice of a word processor.

ATTORNEYS

Attorneys protect themselves against the use of imprecise legal language by using standard wording in many documents such as wills, dissolutions, or legal briefs. The use of standard sections or paragraphs also saves time. A word processor takes this sensible approach to document preparation one step further. It allows an attorney to build a file of pre-written paragraphs which are used frequently in the preparation of legal documents. The attorney can indicate which paragraphs are to be used in a specific document, use the word processor to electronically "cut and paste," and thus create a customized legal document for each client. Non-standard (unique) information can, of course, be added as necessary to connect these standard paragraphs.

Another feature helpful to attorneys is the search-and-replace function. This makes it possible to change "executor" to "executrix," or "Mr. Jones" to "Mr. Smith" very easily. The program's formatting capability is crucial, since a legal document usually must adhere to strict guidelines. The ability to double- and triple-space without upsetting headings and footings is essential.

Of course, some attorneys may have special word-processing requirements dictated by the type of law they practice. Some may need automatic numbering of each line of a pleading or a table of contents for every legal document they prepare. Some word processors provide these features, and an attorney should definitely bring samples of the types of documents he or she commonly prepares to a word-processor vendor.

Attorneys should investigate the possibility of accessing and merging information from remote law library data bases into the main text on the word processor. A modem will be necessary for this kind of link up, as will software that performs the transmission and reception operations. A word-processing program that is designed to be used in a data-transfer operation would be better than one that only performed stand-alone word processing.

SECRETARIES

The word-processing needs of secretaries depend on the industry in which they work. Regardless of their field, most secretaries are given all of the difficult jobs. This would include frequent transcriptions and correspondence and the preparation of in-house memos and reports from rough notes or sketches. Secretaries spend the majority of their time at the typewriter, typing and retyping short documents and letters that are not standard in nature.

The most important word-processing need of secretaries is efficient editing capability, which allows them to produce typed material with a minimum number of keystrokes. They will want a system that allows them to enter text and print it. This printout will be proofread and then

returned to the word processor for corrections before printing out a final copy. In addition to basic editing features, the system should provide global block-move functions, as text will be moved quite a bit. There may be no need, however, for fancy printing commands such as boldface type, special page number placement, or alternate page numbering for back-to-back printing.

Because of the high rate of turnover in office personnel, secretaries need a program that is easy to learn and easy to use (but then don't we all?). A built-in teaching program or computer-assisted tutorial may accelerate the learning process. Help menus are important. These allow the operator to determine what to do next. For example, by pressing a special "learn" key, like the "?," you can immediately get a complete list of the features found in the system. This eliminates flipping through pages to find how to perform a certain function. Accompanying documentation should be carefully examined before a purchase is made. The manual should be user friendly and a reference card should be provided. An index is an absolute must.

If a word-processing system is being used in an office environment where several departments are spread throughout a building, it may be practical to have an electronic mail facility built into it. With electronic mail, the various users of the system can send and receive messages to and from other terminals. These messages can be sent to a host of terminals, thereby sending ye olde memo in a flash. If electronic mail is desired and is available in a particular program package, make sure you don't need a degree in computer science to operate it.

Although an electronic mail feature may be considered a luxury option, the spelling corrector should, without a doubt, be an integral part of a secretary's word-processing system. Most employers depend upon their secretaries to correct their spelling errors, and an automatic spelling corrector can save secretaries considerable time in proofreading reports and letters. If an employer plans to purchase a word processor, it makes sense to have the secretary "test drive" each system being considered. Although a certain system may have terrific appeal to the employer or office manager, it's the secretary who will have to live with it.

MASS MAILERS

Often a major task of any organization is the maintenance and manipulation of large mailing lists. These mailing lists can contain names and addresses in addition to other vital data on members, customers, or clients. Updating and manipulating such a list manually (i.e., via typewriter, labels, and copier) is extremely time consuming. With the word processor you can keep your mailing list on floppy disks and easily make updates any time you want. For a company specializing

in mailing lists, or an organization where the mailing list function is highly important, there are several features to look for.

If all you expect to do is manipulate a simple mailing list and print out simple four-line mailing labels, then almost any word-processing program will probably be fine. When you want to do more than just create mailing labels, however, things get more complex. For example, suppose you have a group of 20 members to whom you want to send the same letter. In addition, you want to use the heading "Dear _____" and fill in the blank with the first name of each person on the mailing list. Word-processing programs set up to do just this type of operation are referred to as mail-merge programs (see Chapter 6).

A mail-merge feature gives each letter a personal touch despite the automatic printing process. In fact, some word processors have mail-merge features that allow you to print the person's name several places inside the body of the letter.

Mass mailing with a word processor can be quite sophisticated. Suppose you wish to maintain not only a mailing list of names and addresses, but also a file of vital information pertaining to each person on the list. For instance, your organization might wish to identify its members by profession, age, sex, or shoe size. A company might want to create a list that includes all customers and a record of their purchases, or vendors and the things ordered from them. Such a collection of information in conjunction with a simple mailing list is referred to as a data base.

The user of a data base system can say, "Send a letter to all members with experience in programming computers who also live in Third World Countries." The operator creates the actual letter on the word processor. A second program then selects just those members on the list that meet the desired criteria. Finally, the word processor's mail-merge feature is used to create the custom letters. One additional requirement is that the word processor be able to tie into a data base management program. This is a rather complex subject and better left to a future book. For now, suffice it to say that there are many levels and types of data base management programs available. You'll have to examine each program in light of your specific needs.

Powerful editing features are probably not greatly important to an organization whose major word-processing need is assistance with bulk mailings. Most form letters are only a page or two in length, and the simplest word processor available can correct errors and insert changes in the text without too much trouble. Obviously, such an organization requires a program with mail-merge capability and possibly an interface to a data base management program.

AUTHORS AND TECHNICAL WRITERS

Professionals who deal primarily with the written word need a word processor. This includes both authors of books and articles and technical writers who create manuals, brochures, or data sheets. The income of these individuals is directly related to their productivity. They can be hampered by constant re-writes, physical movement of text, and correcting spelling. A good word processor can more than triple an author's production, because time will not be wasted on the tedious aspects of writing.

In searching for the right word-processing program, an author requires maximum ease of editing and flexibility in the appearance of the final printed document. Look for a word processor that provides sophisticated editing capability, such as multiple deletion options (e.g., delete character, word, line, or paragraph), unlimited insertions, and simple block-move operations to move text from one section of a document to another. It may also be useful to find a word processor that can accept command strings, which are instructions to perform certain tasks. For example, you may wish to have the editor find all headings which are initial caps and change them to all upper case. An author or technical writer may wish to automatically extract certain blocks of text that are highlighted with distinctive beginning and end markers and write them to a file. Therefore, writers should look for extensive block-marking capability, especially the ability to simultaneously mark several independent blocks in different spots in the file.

An author's word processor should make it simple to print a document (or selected parts of it) many times during editing. It should not be necessary to answer long questions about the format of the output each time printing is required, unless this is desired by the operator. One way a word processor gets around this problem is by allowing the ESCAPE key to be pressed immediately after the file name has been entered. ESCAPE is interpreted as "print out the document immediately; bypass the question-answer session."

Authors preparing camera-ready copy will need a word processor that provides special printing features such as underlining, boldface, variable pitch, proportional spacing, subscripts and superscripts, and right margin justification. If an author is only required to prepare rough drafts, these specialized printing features are not necessary.

Crucial to the author and writer are spelling-correction programs. Especially useful is one that can count the number of unique words and produce a final readability index. This insures that the document's language meets the needs of the intended audience. If the spelling corrector indicates that 80 percent of the words in a document are not included in its standard dictionary, either the writer is taking the "technical" in technical writing too seriously or the dictionary is too small to be of any use.

Technical writers will also appreciate an optional indexing program that will automatically generate indexes, tables of contents, and lists of tables. Such programs eliminate the headache of renumbering tables or rewriting indexes every time an addition or deletion is made or a chapter is moved to a new location.

Other desirable options are the ability to search for open-ended quotes and control characters that turn on a function but are not followed by the corresponding characters to turn it off. When either of these errors is found, the word-processor editor or add-on tool would correct the omission.

Formatting may be extremely important for a technical writer or producer of manuals. If this is the case, check the output-formatting features of the word processor for the capacity to print boldface, underlining, double strike, superscripts, and so on.

THE HOME USER

The home user may be a student, hobby investor, officer of a small club, housewife, engineer using a computer at home, or even a child. He or she most likely is someone who needs a personal computer that can go beyond word processing and handle home budgeting, stock tracking, education, games, instruction monitoring, etc. Word-processing capability will help such a user with projects like college papers, holiday newsletters, or simple letters to friends. The basic features of editing, insertion, and deletion are probably enough for this type of user, and sophisticated features like global search-and-replace may not be necessary.

An inexpensive word processor (in the \$100 range) with basic editing and printing capabilities will probably suffice for the average home user. A technically-oriented home user who wants to explore the inner workings of word processing and learn about BASIC programming at the same time may want to try FRED, our own word processor (introduced in Appendix C of this book). Because FRED's BASIC program listing is included, anyone whose computer uses the popular BASIC language may enter it into that computer, use it right away, and even experiment with modifying it.

There are many inexpensive word-processing programs on the market for the home user and most are sufficient for most needs. If, however, you plan on using the word processor extensively, then it would be best to investigate the more sophisticated products.

Now you should have some idea of where you fit in as word-processor user. In the next section of this chapter, we will examine the seven main distinguishing features of word processors.

SEVEN LAWS OF WORD PROCESSOR PURCHASING

The previous section highlighted the special word-processing needs of selected users. Here we will describe seven basic features of a word processor (some program related and some hardware related). This will enable you to distinguish one system from another and can help you determine the usefulness of a given product for the particular application you have in mind.

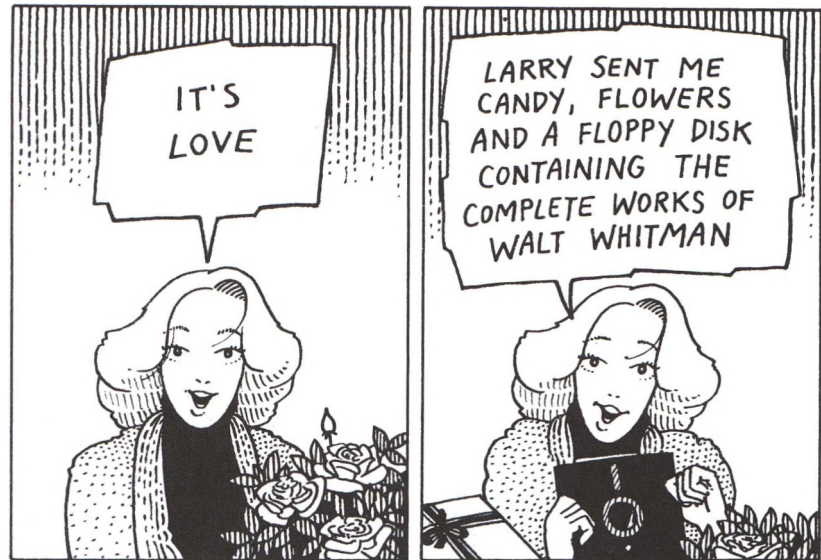
Of course, there is quite a large number of important features (or combinations of features) other than the ones we will introduce. Our feeling is that you should first examine a system in light of the seven features we will outline. This will save time by ruling out many word processors right away. Then, after you have narrowed the list, you can begin to examine the more specialized features.

As we discuss each feature, we will explain how its implementation affects the operation of the word processor. We will follow this by discussing the significance of the feature to the selected professions described earlier. We will explain how to determine if a particular feature is available in a word-processing program and provide the names of some programs that include these specific features. This is not intended to be the definitive word-processing program review but rather a guide to unraveling the large number of products on the market.

The seven features covered are:

- Memory-based versus disk-based
- On-screen versus off-screen formatting
- Screen display size
- Disk capacity and number of drives
- Quality and quantity of editing commands
- Help menus
- Documentation

Of all of these features, the screen display size is probably the most important. Screen display size, however, is solely a function of the hardware rather than the program. Screen resolution has a major effect on price. This is mainly because the hardware for displaying high-resolution text is quite expensive compared to the other parts of the word processor. As the quality of screen-resolution decreases, the price of a word processor drops quickly. The best microcomputer-based word-processing systems allow up to 24 lines of 80 characters, while less expensive ones allow the same number of lines with only 40 characters per line. Recently we have begun to see systems like the Xerox Star, which allows two full 66-line pages side by side on the screen. Today this capability comes with a \$16,000 price tag, although in a few years we may see inexpensive microcomputers and personal computers sporting the same feature. Now let's discuss the seven laws of word-processor purchasing.



MEMORY-BASED VS. DISK-BASED

As a word-processor operator enters text, the word processor temporarily stores it in the computer's random-access memory (RAM). Part or all of the word-processing program is also stored there. The way a word processor handles temporary text storage is important because it may limit the size of the documents that it can create.

Some word processors limit the size of each document to the capacity of the computer's temporary memory minus the amount of RAM occupied by the word-processing program. Such programs are referred to as *memory-based* or *RAM-based* word processors. If a document is larger than that capacity, it must be created in sections that can fit into RAM. These sections are then edited and saved on the disk individually. Most programs can link such files together when they are printed, making them appear to be one document. Deleting text, however, will be much harder, for information must be moved between separate files.

Word-processor salespeople should be able to tell you if a word-processing program is RAM-based. If they're not sure, find out how large each document may be in bytes, words, or pages. If the maximum file size is 15K–30K bytes, 3,000–6,000 words, or 5 to 10 pages, the program is RAM-based. Examples of RAM-based word processors are EasyWriter, Applewriter, and Electric Pencil.

On the other hand, some word processors will allow the creation of documents that are many times the capacity of RAM. With these programs, the document size is limited only by the storage capacity of the accompanying disk drive, which is frequently between 160,000 and 1,000,000 bytes! Such programs are called *disk-based* or *threaded*

because the documents are threaded to and from the disk by the program as necessary to edit them. When the word processor loads a document into RAM from the disk, it will load only a few pages of it. As an operator edits the document, moving through the file, the word processor will take the text that scrolls up off the screen and write it back onto the disk. At the same time, it will automatically load more of the text from the disk into RAM, so that the operator can edit the remainder of the document. It is as if the document were one long string of characters on the disk which you can view only one section at a time.

This disk-based word processor allows the user to create very long documents and eliminates the need to link files together. It makes cut-and-paste operations more practical, since block moves across large amounts of text are possible. In fact, a perfect word processor would allow an entire book to be stored under a single file name (say a 200,000 character file) and still work as efficiently as if it were a file only one page long.

If a word processor can handle single documents more than 15 to 20 pages in length, or if document size is limited only by the storage capacity of the disk, that system is probably a disk-based word processor. Examples of disk-based word processors are WordStar, Magic Wand, and Spellbinder. One of the problems of disk-based word processing programs is that as the file gets longer, it takes more time to read and write to and from the disk. Saving a file which exceeds ten pages can become a lengthy process.

Professional Tips

Attorney An attorney probably needs a disk-based word processor since long pleadings may require large file capacity. In addition, multiple “cut-and-paste” operations also require a lot of memory space.

Secretary A secretary producing one- or two-page letters or memos can get along fine with a RAM-based word processor. If the secretary does mailing lists and the RAM size is small, the work may become difficult due to the number of sections into which the file must be divided.

Mass Mailers Membership organizations or others doing bulk mailings probably need a disk-based word processor, because the mailing list, the form letter, and the word-processing program just won't fit into RAM simultaneously.

Author The requirements of a writer depend on the type of documents being produced. A novelist will require a disk-based word processor to accommodate the length of chapters. A technical writer preparing manuals will also need the storage capacity of a disk-based word processor. On the other hand, the author of short scripts, magazine

articles, news items, or columns may find a RAM-based word processor entirely adequate.

Home User The home user will most likely be satisfied with the capacity of a RAM-based word processor, which is generally less expensive than a disk-based word processor. On those few occasions when correspondence exceeds the size of RAM, linking files together, although time-consuming, should satisfy a home user's requirements.

ON-SCREEN FORMATTING VS. OFF-SCREEN FORMATTING

Almost all word processors allow a high degree of control over the final printed appearance of the text. Text formatting, as we learned in earlier chapters, allows you to control such details as margin placement, text justification, text centering, character spacing, bold effects, and so on. Some of the more expensive word-processing programs display on the screen the format of the document almost exactly as it will appear when printed out, continually updating it as the operator makes changes. We say "almost" because inexpensive display hardware still can't show bold, different character spacing, or underlining. This feature, described earlier as *on-screen formatting*, allows you to see that the printed text will look like before you actually print it.

Other kinds of word processors require that the operator embed within the text special print commands that will control the appearance of the document when it is finally printed. This is called *off-screen* or *print-time* formatting.

On-screen formatting is important if complex placement of text on a page is required. If columns of numbers need to be correctly placed or a section of text is to be shaped to fit around an illustration, on-screen formatting is crucial because it enables you to see in advance what it will look like. WordStar and EasyWriter Professional are two word processors that provide on-screen formatting.

Off-screen formatting is acceptable if the word processor is being used to produce documents requiring little in the way of special layout. Magic Wand, AppleWriter, and Electric Pencil are word processors with off-screen formatting. You usually can tell if a word processor provides on-screen formatting simply by looking at a document displayed on a video screen or a photograph of a screen in a manual. If the text is justified, with straight left and right margins and the distance between words filled with one or more spaces, then you are looking at on-screen formatting.

Professional Tips

Attorney Most attorneys will require on-screen formatting to insure the professional appearance of pleadings, wills, and other legal documents.

Secretary Most secretaries can live with off-screen formatting if their work is generally limited to brief documents. Secretaries who create complex financial reports with multiple columns and sub-headings, however, may need on-screen formatting to save time setting up such documents.

Mass Mailer Organizations doing bulk mailings will not need on-screen formatting because they will constantly use the same form letter, which can be perfected before production of the multiple printing begins.

Author Writers who send manuscripts to publishers for final typesetting are more interested in the text than in its layout and may not need on-screen formatting. Any writer preparing camera-ready copy, such as a manual writer, will need on-screen formatting to insure that the copy is laid out correctly for final printing.

Home User The home user will probably be satisfied with a word processor that formats the text as it is printed (i.e., off-screen formatting).

DISPLAY SIZE

As an operator enters text by typing on the word processor's keyboard, the text will appear on the video display screen. The maximum number of characters that may be displayed on the screen at one time is determined primarily by the video display. The character-display capacity is represented by the number of characters per line by the number of lines per screen (e.g., 80 characters $\times$ 24 lines).

For the professional operator, the display size should be as large as possible to permit the review of the maximum amount of text without continual scrolling. In the microcomputer industry, the standard video terminal is designed to display 80 characters $\times$ 24 lines, so most professional word-processing programs were designed to display the same. For those terminals which are capable of horizontal scrolling, some word-processing programs have a built-in capacity to display the additional characters in increments of 80 characters (a maximum of 32,000 characters per line!). WordStar and EasyWriter Professional provide this option.

Some word processors were designed to be operated on computer systems using a television set for video display. These programs permit a maximum display capacity of 40 to 64 characters per line and 24 lines. This is because of the limited band width of the television's circuits. Of course, even when the screen is only 40 columns wide, the operator may embed commands to print the text with 80 characters per line. On-screen formatting, however, would be extremely awkward on a 40-column screen. AppleWriter and EasyWriter Original are examples

of programs designed to operate with a television set for display (and neither has on-screen formatting). Again, you should be able to determine which type of word processor you are reviewing just by looking at a full page of text on the screen.

Professional Tips

Attorneys, secretaries, and writers should purchase word processors which permit the display of at least 80 characters by 24 lines. Membership organizations may be able to live with a smaller display size if the mail-merge capability of the program is sophisticated enough to satisfy their primary requirement. Home users will probably be quite happy with a smaller display, particularly if they can cut costs by using their own television sets for display purposes.

DISK CAPACITY AND NUMBER OF DRIVES

Whether you purchase a disk-based or RAM-based word processor, the storage capacity of the disk drives you select is important. In a disk-based word processor, the disk capacity will tell you how large a document you can create (deducting the size of the word processing programs that may be stored on the same disk). In a RAM-based word processor, the disk capacity will tell you how many document files you can store on a disk. Of course, each document is limited (by the size of RAM) to only 5 or 10 pages in length.

Technically, an unlimited number of pages may be stored on a word-processor by splitting a document into sections that can be saved on as many disks as necessary. If each disk can only store 30 pages, though, and most of your documents exceed 50 pages in length, it will be time consuming and inconvenient to handle many diskettes for each document you prepare. Also, it will be tedious to move a section of text from one disk to another just to edit a single document too large for one disk. Therefore, look for a word processor that will allow you to store, on one disk, two to three times the amount of text of the longest document you normally create.

The extra space will protect you in case you get carried away from time to time and create an unusually long document. If you work with very small documents (e.g., one-page letters), consider how many of these letters you would like to be able to access at one time. This will tell you how large a disk drive you need. Don't worry about the rare document you may create which is twice or three times the normal size. In that instance, you can probably live with the inconvenience of storing different parts of the document on different disks. Also keep in mind that you may want to have several utility programs, like word indexers and spellers, on the same disk on which the text is stored. These programs will reduce the number of text pages you can store. On this

author's system the disks are 8-inch double-density and can hold 596,000 bytes. However, since many utility programs are on the same disk, the maximum free space is approximately 200,000 bytes.

Professional Tips

Attorneys, secretaries, membership organizations, and writers should not purchase a system which stores less than 50 pages per disk. A single drive may be fine for the home user, but it will severely limit the more serious user.

Most word-processor salespeople will quote disk capacity in number of bytes stored per disk. To help you better use this information, we have provided Table 7.1, which translates byte capacities into numbers of pages which may be stored on a disk.

In addition to disk capacity, the actual number of disk drives is an important consideration when purchasing a word processor. There is more to word processing than creating, printing, and saving text (although those functions are miraculous enough). If you need to make back-up copies of your documents, a two-disk drive system will allow you to make copies more quickly. A single-disk drive system will force you to switch disks in and out of the drive when you make copies, as it loads part of the file from one disk into the computer's memory, and then writes it onto the new disk. A two-disk drive system will automatically transfer information from one disk to another without operator intervention.

If you need to merge names and addresses into form letters, a two-disk drive system will allow you to store the names and addresses on one drive and the form letter on the other. This gives you increased storage on your address disk.

QUALITY AND QUANTITY OF EDITING COMMANDS

Every word processor includes basic editing and printing commands. Some of these commands, however, may be useful only for particular users. When considering a system, you should look at the actual quantity of available commands as well as their quality and usefulness.

Before you purchase a word processor, be sure you try the types of commands you expect to be using. Common editing commands, such as deletions and insertions, should be easy to remember and simple to perform. It should also be simple to move the cursor and scroll backward or forward through the entire document for easy editing. In addition, such commands should take effect quickly, so you are not forced to wait unreasonable amounts of time for the system to perform.

Professional Tips

Attorneys and writers may need sophisticated search-and-replace capabilities to permit location or replacement of both upper- and lower case versions of a search word. Attorneys may wish to change a person's name through a document, and writers may wish to correct a consistent misspelling.

Attorneys and secretaries will need flexible boilerplating capability to allow them to build customized documents or responses to customers. A writer needs easy cut-and-paste capability to mark a section of text for copying, moving, or deletion.

Automatic centering of text, easy changes in margins, and line spacing will be important to attorneys, secretaries, and writers alike. Users interested in form-letter capability will expect their word processor to be able to insert information within the body of a letter and right-justify the resultant text. This is so it won't be obvious that personalized information was just plugged into a form letter.

It is difficult to offer advice on the types of functions each user will need. However, generally speaking, the more sophisticated and complete the available commands, the better constructed the word processor. Balance this against how easy these commands are to remember on the keyboard. A command set should be intuitively obvious. Some systems use color-replacement keycaps to tell you what keys to press to do what.

HELP MENUS

Some word-processing programs will display necessary commands on the screen along with the text you are entering or editing. These lists, which may include commands used for editing, saving files, changing margins, etc., are called *help menus* because they help the operator remember available commands.

These menus are important because they help an operator through the learning stage. Such menus can also save time otherwise spent searching through a manual to find an infrequently used command. Many word-processing programs include different menus for different levels of operator expertise. This permits each operator to display only the amount of information appropriate for his or her level of experience. Some programs provide information normally only found in the program manuals, such as when and how to use the commands listed. WordStar and Select provide such on-screen information.

Infrequent users can quickly become productive if the program provides a great deal of on-screen information regarding commands and their use. Select is one of the first word processors to provide a complete self-training program as part of the package. This allows a new operator to be trained with a minimum of effort.

Professional Tips

The attorney or bulk mailer who hires temporary staff or has employees who use the word processor infrequently will appreciate a program with flexible help levels. Secretaries and authors will find help menus useful in the early stages of learning word processing but will probably want to turn them off later. Mass-mailing organizations may not need them because editing may not be their primary concern. The home user would probably like help menus, but may not wish to spend the extra money to have them.

DOCUMENTATION

The *documentation* provided with the word processor may make the difference between success or failure in a word-processing venture. Regardless of the amount of training provided with a word processor, you will still need to refer to the operator's manual when you attempt a new function or train new operators. Perhaps your first question to a vendor should be, "May I see the manual?" We won't specify what type of documentation each user needs, because we feel that good documentation is important for every user.

Far too often you will find documentation written by a programmer, for a programmer. This will leave the average human completely confused after reading page 2. Assuming you're an average human, you deserve documentation that supports your efforts to use your word processor effectively.

How thick is the operator's manual? Granted, size cannot tell you if the manual is well-written, but it will give you an idea of the amount of time the developer spent producing it. Take time to read the first chapter. Can you understand it? Can your operator understand it? Does it carefully explain the concept of the program and introduce you to basic dos and don'ts of operation? If the manual begins with the assumption that you know more than you do, you may find yourself in trouble before you know it.

Quickly skim through the entire manual. Are new concepts logically introduced, building on information provided in previous chapters? Is the language simple, direct, and friendly? If you find yourself trying to figure out the definition of every other word, the author didn't do a good job and important information may be lost in your struggle to understand the computer jargon. The inclusion of graphics will enliven the text and may assist you in understanding difficult concepts.

Is there an index? There will be many times you will need to look up a specific topic in order to complete a project. In such cases, you don't want to reread the entire manual to figure out how to perform one operation. It would be nice if one page contained a complete command

listing which could be used as a quick reference guide. Surprisingly, both indexes and command listings are conspicuously absent from some documentation which is otherwise quite good.

Does the manual include a step-by-step training section, or is one available separately? You will find it invaluable to practice with such a guide to be sure you understand the fine points of the program. In addition, most vendors will charge for training additional operators. A competent beginner can use a training manual to get up to speed without formal training. Some word processors, such as Select, have built-in tutorials that will introduce a novice to the features and functions of the program. We hope other program developers will recognize this need and soon follow suit.

KNOW YOUR BOTTOM LINE

The intent of this chapter has been to help you figure out which word processing capabilities are essential for your particular application. If you concentrate on evaluating the seven important features we have discussed here, you should be able to make an intelligent and successful purchase.

In many cases, by focusing just on the important choices of RAM-based versus disk-based and on-screen versus off-screen formatting, you can cut your available choices down to a manageable number. At that point, you can begin to evaluate the quality of the programs based on the other criteria we have discussed, as well as any special features your application requires.

Don't let yourself be dazzle-dazzled by a salesperson trying to sell you the only system he's got. Buy the word processor that best does your work, not somebody else's.

How To Buy a Word Processor

This chapter explains the ways you can finance the equipment that makes up a complete word processor. We are discussing microcomputer and personal computer-based word processors here, not dedicated word processors from the traditional word processing manufacturers. (By the way, these companies usually have their own leasing or rental plans available.) The financial concepts of equipment depreciation, investment tax credit, leasing, and rental will be covered.

The second part of the chapter describes the many sources from which you can purchase a word processor, including mail order houses, single product vendors, retail computer stores, company owned stores, and original equipment manufacturers (OEMs). It will be helpful to know the pros and cons of buying from each.

FINANCING YOUR WORD PROCESSOR

There are several ways to purchase the equipment for a word processor. By equipment, we are referring to a personal computer that runs a word-processing program and has a printer and disk storage system. This kind of system costs between \$1500 and \$8000, depending on many factors such as type of printer and computer or size of disk drives. You can assume that the vendor will include the software itself as an equipment add-on or a free extra.

Paying Cash

Of course, you can go out this minute and pay cash for your word processor. That makes sense if you have money, plans to buy an inexpensive system, and/or an accountant that thinks that's the best route for you to take. You can always depreciate the purchase and take the investment tax credit.

Depreciation

Depreciation allows you to deduct a certain portion of the cost of the equipment over a period of time known as the equipment's "useful life" (normally 5 to 7 years for computer equipment). The total depreciation expense is the difference between the initial cost of the equipment and its estimated salvage value after its useful life is over. A portion of this expense can be deducted from your annual income tax each year during the entire lifetime of your word processor.

In other words, each year you can deduct as a “loss” the amount of money the equipment has devalued as a result of use. In effect, the equipment costs you less and less as time goes on. For example, let’s say your accountant determines that the useful life of your word processor is 7 years. You purchased the equipment for \$10,000 and expect its resale value at the end of seven years to be \$3,000. You may deduct \$1,000 per year, for 7 years, from your tax liability.

Investment Tax Credit

The investment tax credit is a one-time credit that is equal to 7 percent of the investment if the depreciation period for the equipment is 7 years or more. The credit is two-thirds of the 7 percent if the lifetime is between 5 and 7 years, and one-third of the 7 percent if the lifetime is between 3 and 4 years. This credit is applied against your tax liability for the year in which the equipment is purchased. For example, if your word processor costs \$10,000, you may deduct \$700 from your tax bill in the year in which the purchase was made.

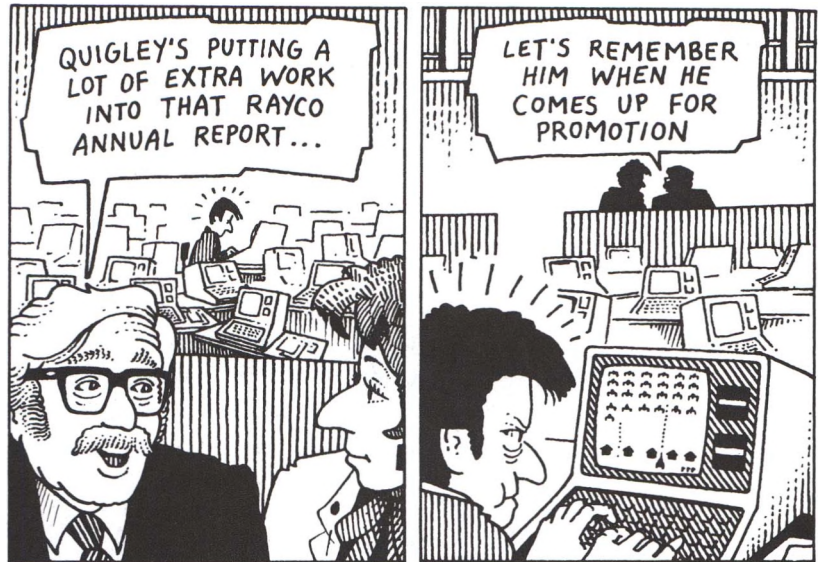
Using both depreciation and the investment tax credit, you may deduct \$1,700 from your tax liability in the first year after purchase! Not bad at all!

In order to depreciate your word processor or use the investment tax credit, you must have purchased the equipment for business purposes. Doctors and lawyers who have gone to the trouble of incorporating themselves can probably safely use both depreciation and the investment tax credit to reduce their word-processing expenses. Because of the subtle variance within the tax laws, you should definitely consult your accountant about how all this might apply directly to you.

Financing

If you do not wish to pay cash for your entire purchase, you can finance a system through a bank or finance company. Financing generally requires a 10 to 20 percent down payment and monthly payments which include interest, of course. In most cases, almost 78 percent of the interest you pay will be due in the first year of the loan, so you will not save much money by paying off the note early. The two main advantages of financing a word processor are being able to depreciate the cost of the equipment and write off the interest payments as a business expense. You may not write off that portion of your monthly payments that is going towards repaying the principal, since you are accruing equity in the equipment.

Another advantage of financing is that many banks will help you negotiate with the company from which you purchase equipment with their funds. In the event that you have serious problems with your



microcomputer, this can provide you with some useful leverage. However, this advantage may cost more than it's worth, given the high interest rates currently being applied by financial institutions. At 18 percent, a 36-month loan of \$10,000 will ultimately cost you \$13,000, with monthly payments of about \$360. On the other hand, you may consider these fixed monthly payments as a hedge against inflation, since most finance contracts do not permit upward adjustments in the interest rate as inflation increases over the term of the contract. Inflation notwithstanding, you may consider the high interest rate to be worth the advantages of owning a word processor.

Leasing

Leasing a system may be a wise move for a business looking for a word-processing system. Most leasing companies require that a lessee be established as a business entity for at least two years before signing a contract. Here we're talking about lease-to-buy arrangements with a third-party leasing company. In such cases, the leasing company holds title to the equipment, since they have actually purchased it from the vendor, and the lessee may buy out the lease at the expiration of the agreement.

In a lease agreement, the lessee agrees to make periodic payments whose total will generally exceed the original cost of the equipment by about 10 percent. Many leasing companies will allow you to add purchases during the term of the lease or upgrade your system to a larger configuration, and some may even pass on the investment tax credit to you. Because the lessee makes a minimal initial investment

(usually only one or two monthly payments in advance), a lease arrangement is a good way to control your cash flow. In addition, since you are effectively renting the equipment, you may deduct all the monthly payments as a business expense. Again, as with financing, the fixed monthly payments can serve as a hedge against inflation.

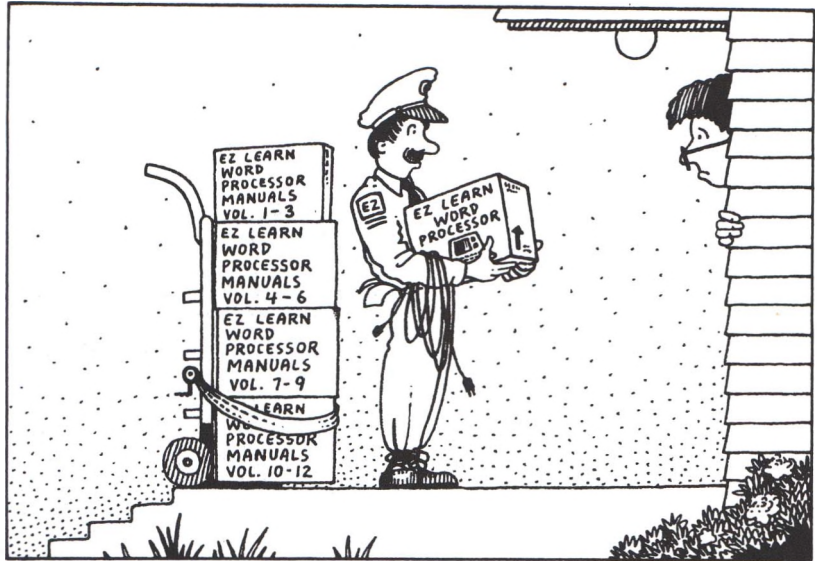
At the expiration of a lease agreement, you are generally offered several choices: purchase of the equipment for a nominal outlay of cash; a radically reduced renewal rate on a new lease agreement (sometimes as low as the total of 4 monthly payments per additional year); trade-in for new leased equipment; or return of the equipment to the leasing company. Leasing probably offers the most flexibility to a business buyer. Unlike finance companies, leasing companies are often willing to modify terms and conditions of a lease to satisfy the particular requirements of a buyer.

Rental

There is one final way to acquire a word-processing system, and that is with a rental agreement. Generally, these are offered by companies as an incentive to try out their word processors. This kind of arrangement is attractive to the uncertain buyer, because he or she feels no real financial or emotional commitment to the system. If the renter is satisfied and decides to purchase the system, it is common practice that a certain percentage of the rental costs may be applied to the purchase price.

Rental contracts are very lucrative arrangements for the companies offering them because very few people will actually get rid of a system with which they have become familiar, even though it may not be the best available system for them. At the time of purchase, the renter will lose a portion of the prepaid rental costs and still have to decide on the best purchase arrangement. If a renter actually does return the equipment, the monthly rental payments can be deducted as a business expense. If he eventually purchases the equipment, he may only be able to deduct a certain portion of the rental costs as interest payments on the purchase.

Buy now or buy later? If you are unsure whether you really need a word processor, you will be wasting your time and money getting accustomed to something you may not want to keep. If, however, you have already decided you need a word processor and have taken the time to define your needs, bite the bullet and buy one. It takes one to three months to get really comfortable with a system, but persevere. After the breaking in period, your word processor will become an invaluable friend and you'll wonder how you ever got along without one.



WHERE TO BUY YOUR WORD PROCESSOR

You know what you need, you've decided how you're going to pay for it, and you're ready to buy. One question: Where do you buy a word processor? It's all a matter of . . .

Whom Do You Trust?

There are several different outlets for word-processing equipment. For example, there are plenty of mail-order houses who will be happy to ship you whatever you like. There are single-product vendors who only sell stand-alone word-processing equipment. There are independently owned retail computer stores that sell several different lines of microcomputer-based word processors. There are also company-owned retail stores who sell to the public. There are original equipment manufacturers who, contrary to what their name implies, purchase other people's hardware and software and put together complete systems for their buyers. Let's discuss the benefits of each.

Mail Order Houses

We would recommend a mail-order house only to those who are comfortable with computers, who really want a discount, and who have a friend who can fix or at least assemble all the components fresh out of the box and get them running. The reason mail-order houses can offer a discount is they don't have to discuss your needs with you or deal with you face-to-face when your equipment needs immediate servicing or you want a refund. In other words, they don't have the same overhead costs as other word-processor vendors have. However, their lack of overhead can translate into headaches for you in the long run. Let me tell you a true story — only the names have been changed to protect the innocent.

HOW MUCH DOES IT REALLY COST?

A poor young writer came into a small sum of money and decided to invest in one of those magical new word processors. He didn't really know where to begin, so he looked up "Word Processing" in the Yellow Pages and randomly picked out a retail computer store.

The salespeople at the store were extremely helpful. He quite frankly told them how much money he had to spend, and they discussed at great length the selection of a system that would fit both his needs and his wallet. When it came to the printer, however, they reached an impasse. This meticulous author just had to have a letter-quality impact printer. But alas, no matter how they juggled the figures, the total cost of the system was undeniably \$1,000 over his budget. The salespeople suggested that maybe he could live with a good-quality dot-matrix printer. But no, the man was adamant. He just had to have the crisp, clean quality of an impact printer.

There seemed no way out of the dilemma. The retail store couldn't drop the price (as much as they wanted to see the young man happy) and the poor young man just couldn't part with his dream of owning an impact printer. Oh, well. Everyone shook hands, the young man sighed, and to ease his pain, he bought a few magazines on his way out the door.

Back in his humble abode, the young man glanced lethargically through one of the magazines. Lo! Much to his astonishment, he spied a notice from a mail-order house which was advertising the very system the computer store had picked out for him. And, miracle of miracles, the price was a full \$1,500 less than what they had quoted to him! He was saved! Without a moment's hesitation, he picked up the phone, placed his order, and sent a check.

After a week of anxious anticipation, his whole system arrived at his door, all antiseptically packed in factory-sealed cartons. Being the intelligent young man that he was, he carefully followed the step-by-step instructions for assembling his system.

It took him a week to get his word-processing software to talk to his computer. It took another month after that to accept the fact that his dream printer really didn't work. After numerous long-distance phone calls to the mail-order house, he finally reached the individual who was their technical staff. This person agreed that it certainly sounded like the printer might not be working. With heavy heart, the young man packed it all up, sent it back, and waited the arrival of a new one. It came, and "Eureka," it worked, sort of. It did its job admirably, but after a page or two, it jammed. Meanwhile, back in the real world, the young writer had completely missed three deadlines, much to the dismay of his disgruntled publishers.

Finally, in desperation, he called the nice people at the computer store, confessed what he had done, and begged for salvation. They decided to forgive him and solved his printer problem that same day. He swore his undying allegiance to them and can sometimes be seen tending the flowers in front of the store. His publishers are delighted because he hasn't missed a deadline in more than a year, and he and his printer are living happily in a houseboat.

What's the moral of this story? Actually there are several: There is no such thing as a free lunch; time is money, but saving money doesn't necessarily save you time; and you get what you pay for. Pick one — they may all apply.

Let's note that the mail-order house was not at fault in this case. They shipped as promised and rectified the problem as well as they could, considering the distance between the purchaser and themselves. The real blame (and he suffered for it) belongs with the young man, because he was unwilling to pay the price for the support he needed.

Before you order your word processor from a mail-order house, ask yourself two questions. Are you completely comfortable troubleshooting your own equipment (or finding someone to do it for you)? Do you trust yourself to learn how to operate it just by reading the documentation which is provided? If you can honestly answer "yes" to both questions, maybe you can afford to buy a word processor from a mail-order house. Most people cannot.

Single Product Vendors

Single product vendors can be very attractive. They know their own product's capabilities and limitations very well. Their service staff is generally large, knows the hardware and software, and can be expected to fix the equipment without too much trouble. Such companies generally have very sophisticated and liberal training procedures and policies. You will pay plenty for this kind of quality service, but you will definitely get your money's worth.

The major drawback of a company like this is that you are "locked into" their software and hardware. Tomorrow, if someone else develops the latest and greatest new word-processing software, you will not be able to use it because it won't fit the non-standard computer system that the single-product vendor offers. And, if you do become dissatisfied with that company's service or support, you will be unable to find anyone else willing or able to support you. However, if you choose a reputable company that will allow you to contact their other accounts for reference, you may have a very comfortable, long-term relationship with a single-product vendor.

Retail Computer Stores

Privately owned retail computer stores like ComputerLand or On-Line will offer you several different systems from which to choose. They generally offer a number of options in choosing the components to make up a word-processing system that suits your needs. You should get more personalized service in such a store, because retail computer stores are smaller and more attuned to the needs of a small business than single-product vendors. You should also be able to try several different word processors on several different systems to see which combination you like best. Because these stores are selling standard microcomputers, you should be able to find someone else to service your equipment if you relocate. If a new word-processing program

becomes available, it is more likely that it will be transferable to your standard computer system.

There are some drawbacks to purchasing from this type of vendor. Their service and support staffs are small, and they may not be able to instantly answer your call for service. Although most stores do have an in-store service staff, they may not provide on-site service. If this is the case, they may have to send your printer, terminal, or disk drives out for service. Also, you cannot expect to send all your employees to a retail store for training without additional expense. Retail store owners always have to consider overhead when making commitments to their customers.

If you find the right store, however, you can have a very satisfying relationship and may find that “your” salesperson will bend over backwards to support you.

Company-Owned Stores

Company-owned stores offer many of the benefits of the privately owned retail store and the single-product vendor. A few years ago, big-name computer manufacturers like Digital Equipment Corporation and IBM realized that they were missing out on the whole small business market by not developing and marketing low-cost computer systems. That’s when they began to introduce competitively priced microcomputer-based systems sold through national chains of retail stores.

Because there is a very wealthy national corporation backing such a store, you can expect to find the same sort of service and support staff as you’ll find with a single-product vendor. Such stores may also offer in-house financing, rental, and lease agreements. In addition, because the stores are small, you should be able to develop a personal relationship with a salesperson who really understands your needs.

You may, however, experience the same problems with a company-owned store as you would with a single-product vendor, because both sell only one product line. In general, these big-name stores are selling hardware and software which was developed on their own mainframe systems. This being the case, you won’t find a wide range of third-party software available for their microcomputers. You will only find that company’s hardware, which may be more or less than you need. If their product and your needs do coincide, then, naturally, third-party software is not an issue.

Original Equipment Manufacturers

An original equipment manufacturer (OEM) does not manufacture computer components. It buys equipment directly from a manufacturer and sometimes slaps its own name on it. Sometimes, the OEM will

purchase someone else's software and modify it to fit your needs or even contract with you to write custom software. In any case, the OEM will select the hardware and software you will purchase, preferably after carefully discussing your requirements with you.

It is difficult to generalize about OEM's as a group. Most are very small companies, some comprised of a single individual. An OEM is only as reliable as its commitment to you and your hardware and software. Very carefully check the credentials of an OEM if you plan to go this route. Insist on references. Some OEM's can be a dream come true, with on-call service and training. They can provide the most personalized and complete support available. Or, they can deliver equipment that never quite works and then become unavailable after your check is cashed.

I know of one OEM who is currently being hunted by buyers throughout California. He delivered equipment that did not meet the specifications he promised, and his customers' systems were completely non-functional as a result. On the other hand, I know of another OEM representative who has provided outstanding custom software to all his customers and will spend days at a customer's location trying to locate and repair flaws in programs he didn't even write. In the case of OEM's — above all, know your vendor.

Our recommendation to you is that you know your own needs, are realistic about the services you will require, and buy equipment from someone you trust. Check what is being offered by each type of vendor, if you want, so you really understand your choices. Ask probing questions and don't be in too much of a hurry. Thoroughly investigate all of the options.

Mitch's Market Sample of Word Processing Programs

INTRODUCTION

One way to get an idea of what a word-processing program can do is to take a look at some of the products on the market. Since there are literally hundreds of programs you can buy today for your microcomputer or personal computer, this is not such an easy task. To help, we have created this "sampler" to narrow the number of word-processing programs down to a manageable size. Its contents are representative of the different types of programs available. We don't believe you can teach people all they need to know about these products simply by presenting a huge matrix of names, addresses, and two-sentence descriptions of word-processing programs. We have left that kind of evaluation to a magazine reviewer. Suffice it to say that what follows is a very good representation of what is on the market today.

We have broken the market up by price, with the least expensive programs described first and the most expensive last. There are two parts to this chapter. The first presents each product and the author's general review of the product. The second part is a detailed chart of features. The first section covers company information, system requirements, and price. The author's subjective impressions of the programs are given in terms of recommended uses for the product, product overview, and summary comments. The second section has a chart which details important features on the product, including system requirements (intended computer and operating system), price, editing capabilities, editing ease, documentation quality, on-screen format capability, and printing ability. We also advise you of the presence of features such as mail merge, spelling correction, programmable function keys, help menus, horizontal scrolling, column handling, indexing, and table of contents generators. Be sure to check the current price with the distributor since prices are subject to change.

We have included the following 10 programs in this market sampler:

FRED
Applewriter
Text Wizard
Super-Text II

EasyWriter Professional
NorthWord
Magic Wand
Spellbinder
WordStar
Select

(Note: FRED is a free word-processing program written in BASIC and included in this book as an appendix.)

DEFINITIONS

To better appreciate the upcoming sample of word-processing programs, we will now give definitions for the features that are covered in the reviews.

Program Title:

This is the formal name given to the program by the manufacturer of the program. People tend to remember product names better than company names, so some clever companies name themselves after their product.

Company Information:

This is the name and address of the manufacturer of the program, or, if the manufacturer does not sell direct, the name and address of the primary distributor of the program.

Intended Computer:

Which computer(s) does the program run on? It could be several. We'll be mainly confined here to personal computers (IBM, Apple, Atari, Radio Shack, PET) and CP/M-equipped small business microcomputers.

Price:

This is the cost of the program in U.S.A. dollars, circa mid-1982. You should take this price with a medium-sized grain of salt, since prices in books do not remain current for long. Call the manufacturer of the program for the actual price, or check your local computer store or some recent issues of *BYTE* magazine.

Recommended Uses:

In this heading we summarize what we perceive as the intended best uses for this word-processing program. We tell whether the program is geared for sophisticated professional use, light semi-professional use, or the novice user. If the program is aligned towards a certain profession, we will say so here.

Product Overview:

Here we attempt to give some general statements that tell you basically what kind of program it is, what makes it special, what features stand out, and so on.

A Final Word:

Here we give summary remarks pertaining to the product. Note that in this and the two previous sections, the content is solely the author's subjective impressions which are not to be considered as recommendations for any particular product.

The following information is to be found in the chart:

Intended Computer: What computer is the word processor intended to work with?

Operating System: Which "mother" program or disk operating system (DOS) is required for the program to operate?

Editing Capabilities: How extensive are the editing features beyond the standard insert and delete commands?

Editing Ease: How easy are the commands to use? Does the key arrangement make the actual editing straight-forward, relatively simple, fairly complex, or non-sensical?

Documentation: How good is the documentation? How complete is it and how easy is it to use?

On-screen Formatting: This tells if the program is designed to represent text on the screen of the computer as it will appear on the paper, as covered in Chapter 4, or if you must insert print format commands in the text to tell the printer what to do with it.

Printing Capabilities: This tells how much you can do when printing out the final draft. This subject also is covered in Chapter 4.

Mail Merge: Mail merge is that feature that "merges" letters typed by the program with names from a data base. This is covered in Chapter 6.

Spelling Corrector: This tells if a spelling-corrector program is part of the package or if one is available from another source.

Programmable Function Keys: Can the keys that are used to denote the various functions of the program be altered by the user?

Help Menus: Are help menus provided with the program, and if they are how good are they? Is a computer-assisted tutorial provided?

Horizontal Scrolling: Horizontal scrolling is useful for wide documents, tables, sales figures, etc.

Column Handling: Can you mark and move columns of text?

Indexer: Is any kind of indexing function available?

TOC Generator: Can a table of contents be generated?

PRODUCT OVERVIEWS AND GENERAL INFORMATION

PROGRAM TITLE:	FRED
COMPANY:	Designed and programmed by the Waite Group, Inc., FRED is hereby contributed to the public domain.
INTENDED COMPUTER:	Any that uses a version of Microsoft BASIC, however a disk I/O specifically for Apple II.
PRICE:	Free with purchase of this book. Source code for FRED is included as an Appendix. You can just type it into the Apple and run it. Good weekend project.

RECOMMENDED USES: Personal word processing. This is for the home user who needs very basic editing or for someone who is interested in learning how a word-processing program is created. Flowcharts are included along with algorithm description.

PRODUCT OVERVIEW: FRED, which stands for FiRst EDitor, is a *line editor* which permits creation of up to 500 lines of a maximum of 254 characters each. Good to use for very simple applications. Lacks block move but has search-and-replace! Slow and clunky because it is written in BASIC.

A FINAL WORD: FRED is a teaching tool for those motivated to understand the internal programming of word processors, an esoteric subject but one with many fans. Although FRED is slow now, in a few years, when floating-point processors and 16-bit microprocessors are in full swing, we will see a version of BASIC you could run FRED under that will be as fast as today's assembler programs.

PROGRAM TITLE:	Applewriter
COMPANY:	Apple Computer, Inc. 10260 Bandley Drive Cupertino, CA 95014
INTENDED COMPUTER:	Apple II and Apple III
PRICE:	\$75.00

RECOMMENDED USES: Personal and light-business word processing. A program for light correspondence and document preparation where high flexibility in printing isn't critical.

PRODUCT OVERVIEW: A very good text-editing system for a home or light business user. Displays 40 upper-case characters per line unless a special \$300 board is purchased to convert to 80 columns (not required on Apple III). Characters to be printed in upper case are displayed in inverse video, but are stored in both lower and upper case form — it is just that the Apple II can't display lower case. A RAM-based word processor with good editing capability. This is one of Apple Computer Company's finest software products.

A FINAL WORD: Highly recommended for the home and light business user who doesn't require extensive formatting and editing capabilities. It is one of the best Apple programs, and has a nice built-in tutorial program. The limited Apple keyboard somewhat holds Applewriter back from fully being what it could be. The file holds up to 31,901 bytes in a 48K Apple II. That is about 11 pages at 3,000 characters per page. Applewriter is based on the underground TED editor that has been floating around for some time. The bottom line is that this is a remarkable system when you consider the price.

PROGRAM TITLE: Text Wizard

COMPANY: Datasoft, Inc.
19519 Business Center Drive
Northridge, CA 93245

INTENDED COMPUTER: Atari 400 or 800

PRICE: \$99.95

RECOMMENDED USES: Personal and light-to-moderate business word processing. This program is RAM-based and can hold up to 11 pages of text in memory. Like Applewriter, Text Wizard has separate Editor and Print Formatter sections. It also makes an excellent BASIC program editor as it's compatible with Atari BASIC programs.

PRODUCT OVERVIEW: An excellent word processor for the Atari 400 or 800 computers. Text Wizard received a perfect score of "all excellent" in InfoWorld Software Report Card (Nov. 30, 1981). It includes almost all required editing and printing functions — proportional spacing, condensed print, elongated text, subscripting, headers, footers, and so on. Requires an Epson MX80 or Atari 825 Printer. Users must be careful to avoid accidentally deleting text when performing some text operations. Note the unique dual-column automatic printing feature.

A FINAL WORD: The first word processor for the Atari computers. Text Wizard is similar to the inexpensive Applewriter, but has better print output format, displays in upper and lower case, and has a good keyboard. Highly recommended.

PROGRAM TITLE: Super-Text II

COMPANY: Muse Software
330 North Charles Street
Baltimore, MD 21201

INTENDED COMPUTER: Apple II

PRICE: \$150.00

RECOMMENDED USES: Excellent for light business and personal applications. Super-Text II is particularly good for mailing lists and form letters using mail merge. The Math module adds columns of figures in your document! As with all Apple II word processors, the storage capacity of the disk drive limits the size of documents. A Dan Paymar lower-case adapter permits display of lower-case letters, but for standard displays, upper-case letters are inverse video like Applewriter.

PRODUCT OVERVIEW: Super-Text II is a very nice text-management system that includes features usually found in more expensive word processors. It is very comprehensive, RAM-based, and has a separate print formatter. The Address Book module handles 700 names and addresses per disk. Note features such as calculation capability, column decimal alignment, and split-screen to review two sections of text simultaneously. It has drivers for Diablo 630, Centronics 737, and easy modification of print parameter table to work any printer.

A FINAL WORD: Super-Text II is a nice light business and personal text-management system, especially for users requiring column-math functions in documents. The optional form letter capability is very useful for limited direct mailing requirements. It has an Autolink feature that lets you link various files together so they can be affected globally by search-and-replace. Thus, you can search for a name in a list made up from many files. A Print Preview mode lets you see on the screen how the document will look when printed out. Only the first 40 columns of each line are shown. A diskette for text requires a special initialization process and won't handle non-Muse text files. A terrific word processor for the money.

PROGRAM TITLE: EasyWriter Professional System

COMPANY: Information Unlimited Software, Inc.
281 Arlington
Kensington, CA 94707

INTENDED COMPUTER: IBM Personal, Apple II

PRICE: \$250.00

RECOMMENDED USES: The first word-processing program for the Apple II, EasyWriter is excellent for light business, professional, and at-home personal text editing and printing. It's especially good for brief correspondence and documents. EasyWriter is much like AppleWriter and has several extra features. For example, a good user-definable character set allows you to substitute any keyboard character with another ASCII value when printing out. A very nice balance of features and clear menus make this one of the most popular word processors on the market and the one IBM chose to use in their personal computer.

PRODUCT OVERVIEW: When used with the Apple II computer and an 80 column video board, EasyWriter provides an 80 character by 24 line format with upper and lower case displayed. It has the advantage of displaying text on the screen as it will appear when printed. Because EasyWriter is a RAM-based word processor, you may only edit four or five pages at a time in a 48K Apple. In a 256K IBM, you can have up to 24 pages in one file. EasyWriter supports incremental printing, proportional spacing, and justification. Like AppleWriter, it has separate print-output and disk-file manipulation subsystems (program modules). Good balance of cursor control, deletion, scrolling, and print format commands.

A FINAL WORD: EasyWriter Professional System is the 80 column version of the original EasyWriter program for the Apple and has a huge following of users. Written in the esoteric language called FORTH by John Draper, alias Cap'n Software, EasyWriter was easily adapted by Information Unlimited Software for the IBM Personal Computer. Electronic mail and data base programs from Information Unlimited Software grow out of EasyWriter to make it the center of the office of the future that uses the Apple or IBM Personal Computer.

PROGRAM TITLE: NorthWord

COMPANY: NorthStar
1440 Fourth Street
Berkeley, CA 94710

INTENDED COMPUTER: NorthStar Horizon

PRICE: \$399.00

RECOMMENDED USES: NorthWord is a professional word-editing system that runs only on the NorthStar line of computers. It is designed much like WordStar and has a powerful complement of editing features including limited on-screen formatting. Key definition is on a stick-on label that goes above the keyboard, rather than on on-screen menus.

PRODUCT OVERVIEW: A sophisticated word processor with most of the capabilities required by a business user. It includes on-screen formatting of the text, excluding page breaks. Unlike the previous Apple and Atari products, NorthWord, like other programs that use CP/M, is a disk-based word-processing program that allows creation of documents as long as the storage capacity of the disks. Disks must be initialized with NorthWord before using for text.

A FINAL WORD: Used with NorthStar's other applications programs, NorthWord can form the basis of a powerful office text-processing and management system. Block move is only 20 lines, so authors may be limited. A nice cursor "game" program teaches cursor movement. Print Preview mode allows on-screen preview of printed text, plus the ability to edit while printing.

PROGRAM TITLE: Magic Wand

COMPANY: Peachtree Software
Suite 700, Corporate Square
Atlanta, GA 30329

INTENDED COMPUTER: Any that runs CP/M

PRICE: \$495.00

RECOMMENDED USES: Magic Wand is especially well geared for businesses that do extensive form-letter preparation (mail merge entering of real time variables from keyboard), and lawyers who need very flexible and easy to use boilerplate features. It may not be best for someone without solid preparation, but it can be programmed to do incredible things that few others can.

PRODUCT OVERVIEW: Magic Wand is an extremely powerful text editor with built-in mail-merge capability. In fact, Magic Wand is almost a data-base system in itself. It is best suited to the user who needs to include programming-like statements within documents to permit customization of text for unique applications. Magic Wand also has extensive boilerplating capabilities, including the ability to review and select sections of stored text for inclusion in files being edited. This disk-based word processor works with any of the many available serial terminals. (All CP/M-based word processors must work with a serial terminal to have mass appeal.)

A FINAL WORD: A truly incredible product for a certain type of word-processor user. We recommended it especially for businesses with form letter needs. It's not the greatest product for the author or writer. The manual is not adequate for easy reference, but it does teach the system. This is a programmer's word processor. Text formatting works like AppleWriter — special backslash characters are put into the text and it is then sent to the Print program. The nice part is that you can embed variables of all types into your text. These are entered at final printing, either from the keyboard, or from a previously set up disk file. It even has conditional print commands.

PROGRAM TITLE: Spellbinder

COMPANY: Lexisoft, Inc.
Box 267
Davis, CA 95616

INTENDED COMPUTER: Any that runs CP/M

PRICE: \$495.00

RECOMMENDED USES: Spellbinder is a very complete text editor especially geared for a business environment with frequent need for form letters culled from a simple data base. It is also good for the creation of boilerplate type letters and documents requiring line numbers (reference and legal works). Spellbinder has a unique *macro* capability that allows a programmer to set up powerful text-processing procedures. It's suitable for lawyers, politicians, direct mailers, pollsters, etc. Programming skills are required to use macros.

PRODUCT OVERVIEW: A disk-based text editor with a special macro capability that is much like a high-level programming language. Macros allow diverse and unique applications. Some built-in macros include two-column printing, several sorting macros, a form handler, mail merge, and boilerplate. Spellbinder has true proportional spacing with nice user-defined space tables for different print wheels. Editor and Command modes are separate, which means more keystrokes are needed to get something accomplished.

A FINAL WORD: A very powerful word processor, with great capacity for customization. A knowledgeable user can create great printing macros.

PROGRAM TITLE: WordStar

COMPANY: MicroPro International
1299 Fourth Street
San Rafael, CA 94901

INTENDED COMPUTER: Any that runs CP/M, IBM Personal

PRICE: \$495.00

RECOMMENDED USES: Offset printing document preparation, manuals, books, theses, company correspondence, light to moderate business use. WordStar has a good mail-merge program available as a separate module. Sorting and data-base programs are also available separately.

PRODUCT OVERVIEW: This is a very powerful on-screen editor, allowing extensive hard-copy formatting and on-screen shaping of text. WordStar is a disk-based word processor which permits a file size equal to the capacity of the disk. It has beautifully designed on-screen help menus for all commands and a key command structure that is engineered for easy memorization and finger control. One of the first on-screen formatting text editors, this program has become the standard of the microcomputer industry. It is possible to print, save, and read files while still in the Edit mode. WordStar is modeled after ED and Imsai's short-lived NED, by Rob Barnaby and Seymour Rubenstein. Micro Pro enjoys the status of being the largest CP/M word-processing company and is releasing several WordStar-compatible programs that go beyond text editing. These include CalcStar and DataStar.

A FINAL WORD: An outstanding word processor, WordStar provides the best word-processing capability for the heavy user. Highly recommended for professionals that need to see what a document will look like before it is printed. Perfect for authors and technical writers.

PROGRAM TITLE: Select

COMPANY: Select Information Systems
99 Sir Francis Drake Blvd.
Kentfield, CA 94904

INTENDED COMPUTER: Any that runs CP/M, IBM Personal

PRICE: \$595.00

RECOMMENDED USES: Select is a CP/M-based word processor which is especially designed for office use. Extensively tested in the office environment, Select contains a simple one-line Command menu which is very easy to remember. Select can be learned in 90 minutes using a special built-in teaching program. The commands in Select are very simple and no extra CONTROL key or ESCAPE key press is required. For example, you press 'C' to create a document, 'S' to check spelling, 'Q' to quit, etc. It's very good for non-computer oriented users, business people and semi-professionals who need large disk files and fine quality output.

PRODUCT OVERVIEW: Select is a disk-based, menu-driven word processor which contains most of the features of WordStar, as well as some extra ones. It comes equipped with the SuperSpell program (reviewed elsewhere in this book) and uses the concept of "tools" to perform editing functions. Has justification, proportional spacing, excellent cursor control, and a powerful print-format menu. The back slash character is used for print commands, the caret for superscripting. Due to its overlay modules, Select does less disk accessing than other CP/M-based word processors and therefore is faster at scrolling.

A FINAL WORD: Select is an excellent word processor for an office environment where many different people will be using the system. Select claims that over 100 people-years worth of testing and refining have gone into this program. The built-in teaching program makes learning Select very easy compared to other systems.

WORD-PROCESSING PROGRAM COMPARISON CHARTS

Program	FRED	Applewriter	Text Wizard
Company	The Waite Group, Inc.	Apple Computer, Inc.	Datasoft, Inc.
Intended Computer	Microsoft BASIC, Apple disk I/O	Apple II/Apple III	Atari 400/800
Operating System	DOS/SOS	DOS/SOS	Atari DOS
Price	Free. Code in Appendix 3	\$75.00	\$99.95
Editing Capabilities	Minimal	Good	Good
Editing Ease	Fair	Good	Good, uses Atari Special keys
Documentation	Good	Good	Good
On-Screen Formatting	No Way!	No	No
Printing Capabilities	Minimal	Fair, adequate for light user	Excellent ²
Mail Merge	No	No, but coming	No
Spelling Corrector	3rd Party	Yes, 3rd party	No
Programmable	Yes	No	No
Function Keys			
Help Menus	No (can be added by programmer/user)	No	No
Horizontal Scrolling	No	No	No
Column Handling	No	No	Not on screen, 2 col. printing
Indexer	No	No	No
TOC Generator	No	No	No

-
- (1) Stores text files for computer that can be accessed by other data-base programs.
 (2) Has underlining, page numbering, sub/superscript, even proportional spacing (on a dot-matrix printer).

Program	Super-Text II	EasyWriter Pro	NorthWord
Company	Muse Software, Inc.	Information Unlimited Software	NorthStar, Inc.
Intended Computer	Apple II	Apple II IBM Personal	NorthStar Horizon
Operating System	DOS/SOS	IBM DOS DOS/SOS	N*DOS
Price	\$150.00	\$250.00 ⁵	\$399.00
Editing Capabilities	Very Good	Fair	Very Good
Editing Ease	Fair ³	Good	Excellent
Documentation	Adequate	Fair	Fair
On-Screen Formatting	No	Yes	Yes
Printing Capabilities	Good	Very Good	Extensive
Mail Merge	Yes ⁴	Yes	Yes (\$299.00)
Spelling Corrector	No	No	No
Programmable Function Keys	No	No	No
Help Menus	No	Yes	No
Horizontal Scrolling	No	Yes, 132 cols.	Yes, 160 cols.
Column Handling	No, except dec. alignment	No	No
Indexer	No	No	No
TOC Generator	No	No	No

(3) A large array of commands to learn.

(4) Two optional modules: Address Book mail list (\$49.95); Form Letter, Mail Merge (\$100.00).

(5) Requires an 80-column display board for Apple.

Program	Magic Wand	Spellbinder	Word Star
Company	Peachtree Software, Inc.	Lexisoft, Inc.	MicroPro Intrl. Inc.
Intended Computer Operating System	Any that runs CP/M	Any that runs CP/M	Any that runs CP/M
Price	\$495.00	\$495.00	\$495.00
Editing Capabilities	Extensive	Extensive	Extensive
Editing Ease	Fair	Fair	Good
Documentation	Good	Good	Good
On-Screen Formatting	No, but Print-preview mode	No	Yes, excellent
Printing Capabilities	Extensive	Outstanding ⁷	Extensive
Mail Merge	Yes, very good	Yes, as Macro	Yes, Mail Merge
Spelling Corrector	3rd party	3rd party	Yes, SpellStar
Programmable Function Keys	Yes	Yes	Yes
Help Menus	No	Yes	Yes, extensive
Horizontal Scrolling	No	Yes, 160 cols.	Yes, 32, 767 cols.
Column Handling	No	No	Yes
Indexer	3rd party ⁶	No ⁸	3rd party
TOC Generator	3rd party	No	3rd party

(6) 3rd party means programs available from vendors other than the OEM. Add-on word-processing tools on the market can be located in advertisements in the pages of *BYTE* magazine, *Creative Computing*, *Kilobaud*, and so on.

(7) True proportional spacing!

(8) Macro could be designed to accomplish this.

Program	Select
Company	Select Information Systems Inc.
Intended Computer Operating System	Any that runs CP/M or MP/M
Price	CP/M, CDOS
Editing Capabilities	\$595.00
Editing Ease	Good
Documentation	Excellent
On-Screen Formatting	Fair
Printing Capabilities	Yes ⁹
Mail Merge	Excellent
Spelling Corrector	Yes
Programmable	Yes, SuperSpell
Function Keys	No
Help Menus	No
Horizontal Scrolling	No
Column Handling	No
Indexer	3rd party
TOC Generator	3rd party

(9) Automatic tracking of margin changes and justification modes.

The preceeding catalog should give you a good overview of what is on the word-processing program market. The next thing to do is to try a word processor out and then buy it. True, a word-processing program is cheap, but a computer is not. Perhaps you can find creative ways to finance your system (see Chapter 8). The author and his friends received loans from a bank because they could show that they would be able to offer superior manual-writing services as a result of owning a word processor. This author used the advance from a book to buy his system. The advance paid for the whole thing. With the prices of computers dropping 30 percent each year, we will probably soon see the day when a powerful word processor will cost the same as a standard electric typewriter, i.e., under \$500. Then, just like a chicken in every pot, there will be a text editor in every kitchen. HAPPY CURSORING!

Word Processing Buzz Word Dictionary

This glossary is not intended to make you a computer whiz. It's intended to keep you from going stark raving mad when you are discussing word processing with a potential dealer. Even the best word-processing salespeople have a tendency to forget that a word whose definition is obvious to them may be just one more brick in the wall of misunderstanding that they are building between seller and buyer. We hope you won't need to use this glossary to understand this book, as we have made every effort to explain words and concepts as they are introduced. One final note — many phrases, as well as words, need to be defined in this glossary. Rather than using a strictly dictionary-like approach to these phrases by presenting the noun, a comma, and then any modifier, we have chosen to list these phrases in alphabetical sequence, based on the first word in the phrase. For example, where a dictionary would list "proportional spacing" under "S" as "spacing, proportional," we have listed it under "P," as "proportional spacing." Sometimes it's very difficult to guess how people will look up a word, so we've taken the path of least resistance. We hope it works for you.

ABORT: To artificially stop a process before it has reached its natural conclusion. For example, "Why don't you abort the printing of the document so we can correct the error I just found on page 2?"

ALGORITHM: The sequence of steps to be performed to achieve a particular solution. For example, a *search algorithm* in a word processor would define for the computer the steps it should take to match the words of a document to a given *search word*, as follows:

- 1) Determine what the first letter of the search word is.
- 2) Search for a word which starts with the same letter.
- 3) Now determine what the second letter of the search word is.
- 4) Determine if the second letter of the found word is the same.
- 5) If it doesn't match, find another word that begins with the same letter as the search word.
- 6) If it does match, determine what the third letter of the search word is.
- 7) Determine if the third letter of the found word is the same.

This process continues until all letters of the found word are determined

to match those of the search word or until a mismatch occurs. In that case, the search algorithm would direct the computer to search for the next possible match, starting again at step one.

ALPHANUMERIC: A set of characters which includes all letters of the alphabet and all numbers. "Word-processing commands are created using alphanumeric characters, such as 'E' to indicate that you want to 'edit a document'."

ALTERNATE PITCH: *Pitch* refers to the distance between printed characters, expressed as the number of characters printed per inch. For example, a printer using 10-pitch would print 10 characters per inch on each line. An alternate pitch is a less commonly used pitch which can be accessed by a word-processing program and reproduced on a printer. "Most word processors use 10-pitch as the standard pitch with 12-pitch as the alternate pitch."

BI-DIRECTIONAL PRINTING: A printer's ability to print one line of text from left to right followed by the next line of text from right to left (the reverse of what you're used to). This feature saves unnecessary movement of the print mechanism. "Some printers have built-in bi-directional capability that can be accessed by sophisticated word-processing software."

BIT: A contraction of the phrase "Binary digIT." The smallest unit of information handled by a computer, it represents part of an alphanumeric character. "In most microcomputers, the microprocessor assembles 8 bits to represent an alphanumeric character." See also ALPHANUMERIC and BYTE.

BLOCK: (1) A continuous section of text which may be manipulated by the operator. "Why don't you move that block of text to the next page."
(2) Sometimes used as a verb interchangeable with the term "reformat" to define the restructuring of text to fit new margins and spacing parameters. "Now that you've changed the margins, you should block up that text." See also BLOCK MARKER.

BLOCK MARKER: A character entered by the operator at the beginning and end of a section of text to indicate a block that can then be manipulated (moved, deleted, or copied). "In WordStar, the beginning block marker looks like this: ."

BOILERPLATE: A section of text containing frequently used language that can be inserted in other documents or files. The use of boilerplates can permit an operator to produce a large volume of correspondence which is similar in content without repetitive retyping of text. "I think we should send this guy boilerplate letter number 8, which apologizes for the lousy service he got."

BOLDFACE: A printer feature which allows the printing of a character followed by a *partial* backspace and a re-strike of the same character. This makes the character appear darker and thicker on the printed page. "Many word processors use a printer's boldface capability to emphasize titles and section headings." Compare DOUBLESTRIKE.

BOOT: To "boot the system" is to turn on the computer and receive visual acknowledgement of the terminal screen that it is ready to begin operation. This term is derived from the phrase "bootstrap." Most computers have a built-in bootstrap program that does nothing more than tell the computer when it is first turned on to look at the disk in the drive for further instructions. So the computer pulls itself up into operation by its own bootstraps. "On some systems (like Apple or Atari), you just turn on the computer and insert a disk to boot the system. On other systems (like Altos), you must turn on the computer, insert a disk, and press RESET in order to boot the system."

BYTE: A unit of information perceived by a microprocessor as representing one character. In most microcomputers, an adjacent series of 8 bits of information represents a byte, or one character. "This byte represents an A, and the next byte represents a 2 ." Compare BIT.

CENTRAL PROCESSING UNIT (CPU): The "brain" of the computer, the CPU handles all calculation and logic functions for the computer system. It receives, processes, and distributes all information flowing to and from the computer system. "The CPU used in the Apple II is the 6502 microprocessor chip."

CHAIN: To indicate to a word processor that a series of documents should be treated as one document when a process is to be performed. This does away with the need for operator intervention to specify the next document in the series to be processed. "Many word processors will allow you to chain documents for printing. Then the word processor will automatically print a series of documents one after the other while you take a lunch break."

COMMAND: A direction given by the operator to the computer to perform a specified function. In most cases, commands are given by pressing a certain key (or sequence of keys) on the video terminal. "The command to edit a document is given by pressing the letter E ."

COMMENT LINE: A line of text which will appear on a video screen but will not be reproduced on a printer. "Comment lines can be inserted in a document to provide information to the operator which should not appear in the printed document." Comment lines are indicated by prefacing the information with a special code. If you wished to advise the operator to print 3 copies of a document for in-house distribution, you might

created a comment line that looked like this: "... 3 copies for in-house distribution." The two periods advise the computer that the following information is not to be printed.

CONFIGURATION: A complete computer system composed of individual pieces of equipment, for instance, a CPU, disk drives, video terminal, and printer all interconnected and operating as one system. "My hardware configuration is an Apple II, two disk drives, my color TV set, and a Epson printer."

CONTROL KEY: A special key on a video terminal that is pressed simultaneously with a normal keyboard character to indicate to the computer that the resulting character is a command. "If you press the letter 'D,' a 'D' will appear on your video screen; if you press the letter 'D' and the CONTROL key at the same time, one character will be deleted from the screen."

CPU: See CENTRAL PROCESSING UNIT.

CURSOR: A unique character displayed on a video monitor or terminal which indicates where the next character entered from the keyboard will appear. Derived from the Latin verb meaning "to run," the cursor "runs" all over the video screen. "Good word-processing packages permit cursor control, which allows the operator to move the cursor on the video screen in order to designate where typed information will appear."

DAISY WHEEL: A printer mechanism containing fully-formed alphanumeric characters, similar to those found on an electric typewriter. "Printers using daisy-wheel print heads can produce printed text almost indistinguishable from that produced by a human typist."

DATA DISK: A blank, *formatted* disk on which you can store all your text files. "Because space on a data disk is not used to store programs, you will be able to store a lot more of your text files on each one."

DESCENDER: That part of some letters of the alphabet (e.g., "p," "g," "q,") which descends below the normal line of text. Some video terminals and printers cannot effectively produce characters with descenders and will display those characters on the same level as the normal line of text. "It may be difficult to read the text displayed on those terminals that cannot effectively display characters with descenders."

DISK/DISKETTE: A circular piece of magnetic material enclosed in a protective jacket used to permanently store computer programs, data, and text information. A disk is very similar in appearance and function to a record album. "Most disks are 8 or 5¼ inches in diameter." See FLOPPY DISK and HARD DISK for descriptions of different types of disks.

- DISK DRIVE:** An electro-mechanical device, including a read/write head, into which a disk is inserted to be read from or written to by the computer system. It is similar in function to an audio tape player/recorder. "I have 8-inch disk drives, so I have to order an 8-inch floppy-disk version of my word-processing software."
- DOS:** An acronym for Disk Operating System, also called simply the Operating System. See OPERATING SYSTEM for definition.
- DOT MATRIX:** A process by which characters can be formed for display on a video screen or on paper. Each character is made up of a series of dots in a specific two-dimensional pattern. The number of dots available horizontally times the number of dots available vertically for each character describes the density of the matrix. "An 8 x 10 dot-matrix display on a video terminal normally provides adequate clarity of character formation for easy reading."
- DOUBLESTRIKE:** A printer feature which allows the printing of a character, followed by a *full* backspace and a re-strike of the same character, so that it appears darker on the printed page. "I think we should print the chapter headings in boldface, and the section headings in doublestrike." Compare BOLDFACE.
- EMBEDDED COMMAND:** A print formatting command inserted within the printing of a document and executed by the printer during the printing of the document. For example, some word processors allow you to include a command to print a heading at the top of each page. In WordStar, such a command would look like this: ".HE Section Two." The heading "Section Two" would be printed at the top of the page where this command was first encountered, and on all following pages. "Embedded commands are displayed on the screen and are acted upon at printing time."
- FLOPPY DISK:** A flat, circular piece of magnetic material which is used to store computer programs, data, and text information. These disks are somewhat flexible — hence the adjective "floppy." "Floppy disks are produced in 8-inch and 5¼-inch formats." See DISK; compare HARD DISK.
- FORMAT:** To prepare a blank disk for use by a particular computer by electrically laying out tracks on the disk (like tracks on a record) so that the computer will know where to store information. Many different kinds of computers can use the same type of disk; however, each type of computer will store information on the disk differently. Since a computer will expect to find information placed on the disk where it can locate it, you must format a blank disk before using it. Generally there

is a program provided with the operating system that will perform the formatting function. "If you don't format a disk before using it, the computer won't know where on the disk to write information."

FUNCTION KEYS: Specially marked keys on a computer keyboard that are used to indicate that a certain process should be performed by the computer. Many word-processing systems include function keys which, when pressed, direct the word processor to perform common functions such as inserting, deleting, or moving text. "A word processor with function keys can be very useful for the beginning operator who might have difficulty remembering all the word processing commands."

HARD DISK: A rigid platter covered with magnetic material and used to store computer programs, data, and text information permanently. "Hard disks permit the storage of enormous amounts of data and are generally required for a multi-user computer system."

HELP MENU: A list of available commands, displayed on the screen. Termed a "help" menu because it helps the operator remember commands while using a word processor. "The EasyWriter Professional Word processor includes a help menu which may be eliminated once the operator knows all the commands."

INSTALL: To customize a software program so that it will operate on a particular computer system. In most cases, customization requires that the program receive the appropriate communication between itself, the terminal, and printer. "The manufacturer will install Magic Wand so it will operate on your computer system."

JUSTIFICATION: The reproduction of a section of text with perfectly aligned margins. Justification is achieved by the distribution of spaces between words so that all lines of text are the same length. "This book is printed with right- and left-margin justification." See also MICROSPACE JUSTIFICATION.

K: See KILOBYTE.

KILOBYTE (K): 1024 characters (or bytes). Used to measure the storage capacity of a computer or disk drives. For example, the Altos 8000-2 computer system can store 600 kilobytes of information on each disk (or about 600,000 characters per disk). "The Apple II computer has 48K (kilobytes) of random-access memory."

MAIL MERGE: A software program that creates personalized form letters by inserting a new name and address each time the letter is printed. Generally, the letter and the list of names and addresses are stored in different locations on a disk, and the mail-merge program will automatically



search for and insert the next name and address into each letter. "The mail-merge capability of Spellbinder will permit you to insert additional personalized information into the body of each letter."

MB: See MEGABYTE.

MEGABYTE (MB): One million characters (bytes). Used to measure storage capacity of a computer or disk drives. "A Dynabyte computer can be attached to a 32 MB (megabyte) hard disk."

MENU: A list displayed on a terminal screen, providing a choice of functions or items which may be selected by number or letter. A sample menu is shown below:

WHAT WOULD YOU LIKE TO DO NEXT?

1. EDIT A DOCUMENT
2. PRINT A DOCUMENT
3. DELETE A DOCUMENT
4. EXIT THE PROGRAM

PLEASE SELECT CHOICE BY NUMBER:

MICROSPACE JUSTIFICATION: The production of a justified margin by the even distribution of spaces between *characters* on a line in increments as small as 1/120 inch. Microspace justification produces more even lines of text than does normal justification, which only distributes whole spaces between words. Normal justification may leave wide spaces between words if

there are several long words on one line. "Microspace justification produces a much more professional looking document." See JUSTIFICATION.

MONITOR: (1) A video screen similar to a television set that can display images or characters as transmitted from a computer system. (2) A small, built-in computer program which oversees the operation of the computer, performing housekeeping functions similar to those found in the operating system. "The monitor is part of the computer hardware itself and is available whenever the computer is turned on." Compare OPERATING SYSTEM.

MULTI-USER CAPABILITY: The ability of a computer system to simultaneously process information received from more than one operator. "A multi-user system requires a terminal for each user."

OPERATING SYSTEM: A software program that acts as a "housekeeper" for your computer. The operating system tells your computer how to store and where to find information on a disk. It also tells the computer the names of the files on the disk and how much space these files use. It handles miscellaneous housekeeping chores as well. The operating system is normally stored on the first few tracks of a disk so the computer can reference it for instructions as soon as power is turned on. "CP/M is one of the most common operating systems available in the microcomputer industry." Compare MONITOR, definition (2).

PITCH: A measure of the distance between printed characters, expressed as the number of characters printed per inch. For example, a printer using 10-pitch would print 10 character per inch on each line.

PROPORTIONAL SPACING: A printer function that assigns a specific, unique number of spaces to each character in order to achieve a uniform appearance on the printed page. For example, the space allotted to "l" will be less than that allotted to "m," so proportional spacing is necessary to provide consistent spacing between each character. "Proportional spacing permits the production of text that appears to be professionally typeset."

RANDOM-ACCESS MEMORY (RAM): This is the temporary, *working* memory of the computer where information currently being processed is stored until RAM is either cleared out or the information is transferred to permanent storage (on magnetic media). This type of memory is called "random access" because the computer can jump anywhere within the memory to locate and store information. "A good word processor uses at least 48K of RAM." Compare READ-ONLY MEMORY.

- READ-ONLY MEMORY:** A computer program stored on a microprocessor chip which may be accessed by the computer but may not be altered or erased. "The Apple II computer includes the BASIC language in ROM."
- SCROLLING:** The movement of displayed text on the screen, either vertically or horizontally, to permit the display of additional text. Scrolling is like moving a "window" that opens into a room full of furniture, so that you can view furniture in different parts of the room. "Horizontal scrolling allows you to create a wide page of columns of numbers on a screen that normally only permits you to display 80 characters on a line."
- TERMINAL:** An all-in-one unit comprised of a video monitor on which text is displayed as transmitted by the computer or the operator and a keyboard with which an operator may transmit information to the computer for processing. "Personally, I prefer terminals which use a typewriter-like keyboard and a screen which displays the characters in green." Compare MONITOR, definition (2).
- WORD WRAP:** A word-processing function in which the computer calculates how many words will fit within specified margins and then automatically moves words down to the next line when the right margin is reached. No operator intervention is required. "Word wrap permits an operator to type at full speed without having to watch for a carriage return at the end of a line."

Record Processing

WHAT IS IT?

Record processing is the term used to describe entering, retrieving, and updating information that is stored in records on the computer. A *record* is a particular arrangement of information of the kind found in business environments. For example a name, address, and phone number constitutes a single record in a mailing list. The mailing list itself, with all its associated information, is referred to as a *data base*. Other types of records are inventory records, invoice and order forms, accounting and employee information, shipping and receiving data, and so forth. You may think of record processing as similar to word processing, except in word processing we manipulate words and in record processing we manipulate records of information. Like text, records are usually stored on disks, which makes it easy for the record-processing program to manipulate the records — read, change or output them — as efficiently as words are manipulated with the word processor.

AN APPLICATION

A typical application of record processing would be the ordering procedure for a company selling several products. In most businesses today, orders are taken manually with pencil and paper. Usually, a special order form is filled out with the customer's name, address, phone number, purchase order number (PO #), the names and quantities of products ordered, and so on. Standard practice dictates that one copy of this form be sent to the shipping department to fill the order, another copy to accounts receivable for mailing to the customer, and a third is filed away for accounting purposes.

Suppose the company president wishes to see graphs or sales breakdowns for the recent year. Since the needed information is buried in piles of paper, retrieving and summarizing this information will be an extremely time-consuming task. A second problem with the paper-and-pencil process is that the actual order entry procedure is subject to human errors. There is no checking done to see if the order was taken properly or if the form was filled out completely. Furthermore, it is tedious work filling out the same information (such as the day's date) with each order.

By converting the manual order-entry system to a computer-based record-processing system, it is possible to reduce the labor involved in the order entry process, catch errors before they get into the system, and manipulate the stored data in ways that are impractical to do manually. In addition, since all data is in a file on the disk, it is easy to extract meaningful information and use it to create new files, reports, mailing lists, special packing forms, and so on. In a record-processing system, the paper form is reproduced on the CRT screen and the information is entered directly into that form with the keyboard. Information is scanned, verified, and validated before it is accepted into the data base. This checking process protects against entering a value that is too large or too small, a letter where a number is called for, and so on.

Any business that currently does order entry on a manual basis can profit from record processing on the computer in two ways. First, data entry is smoother and safer. Second, you can create snap shots of your business' finances — where they are and where they are going.

Word processing and record processing are important partners in the electronic office of today and tomorrow. With the word processor, we manipulate text and words with relative ease, greatly speeding up the creation and modification of letters, documents, memos, bulletins, and columns. You can have automatically personalized letters that utilize name-and-address data created with the word processor. Bringing in record processing, we now can have personalized letters with contents that are based on a complex data base. We can have letters with contents that are modified "on the fly" by an operator who is answering specific questions presented by a computer program. For example a form letter could say: "You have been awarded \$_____ for your _____ injury." The operator might enter "1,000" and "leg." With record processing we can look at the data and print it out clearly in revealing reports, graphs, and documents. This latter application of record processing is just getting off the ground in the microcomputer market.

THE MARKET

At this time there are several programs on the market that will completely perform or partially aid in record processing. Some of these are called data-base management systems, or DBMS. Some of these are complete data-base systems which have a word processor or editor built in. They provide calculating ability, row and column spread-sheet capability, report generation, and sorting. Some of these programs are designed for the beginner while others require that the user be more knowledgeable. Some of these record-processing programs need to be set up by a programmer. The resulting program is a simple one that is easy to use, even for the neophyte. Because it would take a separate

book to describe all the data-base management systems available for today's microcomputers, we urge you to stay tuned for *Record Processing Primer* and *Spread Sheet Primer*. Meanwhile, here's some fundamental information about how a typical record-processing program might work.

A TYPICAL DESIGN

Data entry occurs when the operator enters information into the data base. If this involves entering data through the word processor, the operator is in a difficult position, because putting information into a word processor is too *free form*, i.e., there are no boxes to fill or columns to check off. What is needed is a universal data-entry and retrieval program that creates a "form" on the computer screen to your exact specifications. The form would help the operator easily input the entry items and would contain the same editing features as the word processor. The record-processing program would allow the operator to enter, examine, and print records. In addition, there would be the ability to select certain records with certain key information in a certain order.

Form Generation

One approach would be to have two programs, one for making the form (used by the programmer) and one for using the form (used by the operator). The form-generation program should allow you to create any kind of input form for the CRT screen. You should be able to make a form just like the paper one you had been using. The width and length of the form should be adjustable. You should be able to build fields with the form generator — columns of information with a particular heading that indicate what will be entered in the field. A field usually has a type which describes what is in the field — numbers or letters, how many, where the decimal point is, and so forth.

Once this form has been created, you can run the actual record-processing program with which the operator interfaces. These programs can be designed to simulate intelligence. For example, you can set it up so that when the operator enters a unique, pre-assigned customer code number, the program will automatically fill in the "ship to" and "bill to" addresses from another file! Or, when you enter the product code number, the product description will immediately appear on the corresponding line on the form.

You could have a constantly updated file of product prices created with the word processor. Then the record-processing program could automatically fill in the current price when the product code was entered. There could be sets of files containing prices with certain discounts for certain customers. When the quantity was entered, the

program could calculate the total price and fill it in. If you had to copy several records with common information in them (a block of text or a date), you could set up the form so that the repeated information is copied onto the next form automatically.

The record-processing program should allow the highlighting of certain fields in reverse video, the alignment of the decimal point after the number is entered, the insertion of dollar signs, and the justification of columns after the entry. It should perform a variety of calculations with the data you enter — incrementing a certain field within a record, carrying a running total across records, addition, exponential functions, and so on.

Form Editing

What if the operator makes a mistake or wants to change something on the form? A record-processing entry system should have a simple set of editing commands, similar to a word processor used in an office, that allows you to move the cursor about the form.

The program should be able to set up must-enter fields. For example, you could have it programmed not to accept a form if the date is missing. You can have the program make sure that numbers have not been entered where letters are required, and vice versa. You can even make sure each number falls into an acceptable range, for instance only accepting months from 1 to 12, or days from 1 to 31.

Scanning Records

A most important part of any good record processor is called data retrieval. This gives you free access to your records. A good record processor should be able to view and change records, select from a set of records, and search for records that fall within a certain range.

You should be able to display records in the order in which they were entered or in any order desired. You should be able to refer to some field in the record, such as the date or a range of dates, as the *key*. Then you could tell the record processor to display the records using that key, or you could tell it to display the list of records alphabetically by name.

You would also want to be able to look at records that share any given characteristic. For example, you could view just those records that have the city of San Francisco in the address field. A good search feature allows the program to find a single record in a large data base. To utilize this feature you need only to remember one small piece of data pertaining to the record. If you knew, for example that you sold a red soap dish on May 30, 1927, and that the address was “San” something, you could find it in the data base.

Printing Records

A good record processor should allow you to print out specific records in special formats and integrate specific data from the records into a print-out form. You should be able to print out records in either entry or index order. You could, for example, print out mailing forms sorted by product, newsletters sorted by zip code, or parts inventories sorted by part numbers.

SO WHAT IS A RECORD PROCESSOR?

A record processor is a tool that solves two problems for the application programmer. It gets data into the system fast and allows more to be done with it once it's in there. Such a program is often designed to be used by a programmer as a *front end* program. This means that the intended application uses the record processor for data entry and data retrieval. The application program can then perform more complex operations on the record files and leave all the work of getting the data into the records to the record processor. Application programs that access and manipulate the data in the records can be written in Pascal, BASIC, FORTRAN, or COBOL.

WORD PROCESSOR + RECORD PROCESSOR + MAIL MERGE + SORTER

By combining several of the tools we have described, it is possible to create sophisticated effects with your data. For example, the record-processing program could be used to enter, retrieve, and update data; and a sorting program would merge, select, and rearrange the record information. The word processor with a mail-merge option of some kind would then produce printed results.

Consider this simple example. Assume you have a current list of customers, stored in records created with a record processor. Each record includes the account size in dollars, the contact name within the company, and the address and phone number. Suppose you want to test a certain product on specific selected customers, for instance, a group in the western states that buys more than \$50,000 but less than \$100,000 worth of merchandise per year. You use the sorting program to produce a data file of these records. Then with the word processor, you draft a letter, using the mail-merge commands for variables as described in the previous chapter. Because the mail-merge program can dip into the sorted data file for this data, your letter will then contain the volume in dollars and the contact person's first name. The letter will appear very personal. The recipient will think that someone studied his or her buying habits and knew them well. Record processing makes such sophisticated service a relatively easy task.

FRED: FiRst Editor BASIC Program

We figured that computer owners reading this book would be desperately craving to use a word processor but might be too financially conscious (i.e., cheap) to go out and buy one. Therefore, we have included here a listing of FRED, a simple word processor written in Applesoft BASIC which you can type in and begin using *right now*. It's hardly the world's most sophisticated word-processing program, but the price is right and it's small enough to type in without undue hardship. When you are finished, you will have a program featuring insertion, deletion, on-screen text review, paragraph format, disk save and load, printing, and string search. Since FRED has only the most fundamental word-processing features, it is very easy to use. But if you're ready for a challenge, FRED is expandable. It's a perfect jumping-off place for learning programming on a rainy day. Perhaps you could create an application which you could eventually market. We submit this core program to the public domain. Enjoy!

Technically speaking, FRED is a line editor with primitive but useful word-processing features. Since it's written in Applesoft BASIC (a Microsoft dialect), it is not very fast. Nevertheless, it is 100 times faster than your typewriter. We have provided the brave souls who wish to use FRED with a flow chart for converting it to another language like Pascal, FORTRAN or FORTH.

HARDWARE REQUIREMENTS

To use FRED you must have an Apple II or an Apple II Plus with a minimum of 32K of RAM, at least one disk drive, and a printer (optional). It is assumed that you are moderately familiar with Applesoft BASIC and the Apple's disk operating system.

Type in the program, check for any errors you might have made, and save it on disk.

PROGRAM DESIGN

FRED is a simple word processor which allows you to enter text into the computer from the keyboard, modify it if necessary, print it, and save it to disk. (FRED was written by Corey Kosak, a software engineer with The Waite Group, Inc.)

FRED is a *line editor*. This means that you type your text into a series of lines or *elements*. The FRED program can hold up to 500 such lines, and each line can contain up to 254 characters. Each line has a line number ranging from 1 to 500. When you first start entering text, you will be typing into element number 1. After you've filled it with 254 characters (or hit RETURN), the computer will go on to element number 2. Line numbers allow you to easily refer to blocks of text. For example, you can remember that Chapter 3 of your new book starts on line number 157. Later, if you want to edit parts of Chapter 3, all you have to do is tell the Edit function to start editing at line 157.

OPERATION OF FRED

The main menu has eight options. They are: Text Entry, Review/Edit, Read from Disk, Write to Disk, Print to Printer, Print to Screen, Search for a String, and Exit. Each of these is invoked by typing the first letter of the command.

When the program begins, the text buffer is empty. You may either start typing into it by choosing option T, Text Entry, or if you have a text file on disk, you may read the file into memory with R, Read from Disk. Let's start with option T.

Text Entry

This is the mode into which you will be typing your original text. The computer clears the screen and prints in inverse video the number of the first empty element it finds. This number will be 1 for an empty text buffer. Now you may type away to your heart's content. When you fill up an element (after typing 254 characters), the computer will print the next element number on the screen, and you may continue typing. You can type in this way until either you run out of memory or use up the 500th element.

If you make an error while typing, you can back up the cursor with the left-arrow key. Every time you press the left arrow, the cursor moves one space to the left and deletes a character.

To move from one array element to the next, press the RETURN key. If you press the RETURN key on an empty element (one that has nothing typed into it so far), you will return to the main menu.

FRED has three basic text-formatting functions. Formatting commands are invoked by typing a control character as the first character of an element. The control characters will appear on the screen in inverse video.

CONTROL-P marks the beginning of a new paragraph. When the file is printed out, the rest of the text will be indented five spaces.

CONTROL-N forces a carriage return and line feed and causes the text to be printed at the left margin of the next line.

CONTROL-S prints a blank line so you can space vertically.

Remember that these control characters only have significance when they are the first character of an element. Also note that each element is printed *immediately after* the last character of the previous element. We did this so paragraphs can span several elements without being broken up. Therefore, to start a new paragraph, you must use either a CONTROL-N or a CONTROL-S.

Review /Edit

The Review/Edit function helps you go over and perhaps modify your text. When you type an E from the main menu, FRED will ask what element number you wish to edit. Type the element number and press RETURN. The element will appear on the screen along with another menu, the Review menu.

The Review menu has six options. They are: Move Down, Move Up, Zap, Push Down, Edit This Line, and Exit.

Move Down moves to the next element. If you are at the 500th element, this command has no effect.

Move Up moves to the previous element. If you are at the first element, this command has no effect.

Zap deletes the current element, moving all the following elements up one spot to fill the void. For example, if you type Z (for Zap) on element 4, then element 4 will be deleted, element 5 will be moved down to element 4, element 6 will be moved to element 5, and so on.

Push Down moves all the elements down a spot, creating an empty element at the current position. For example, if you type P (for Push Down) on element 4, then element 4 will be moved to element 5, element 5 will be moved to element 6, and so on.

Exit, of course, exits back to the main menu.

Typing E for "Edit This Line" reprints the element at the top of the screen and enters Edit mode. In Edit mode you may move the cursor right or left, insert or delete a character, replace a character, or return to the Review menu.

To move the cursor right or left, press the right or left arrow, respectively.

To delete a character, move the cursor on top of the character you want deleted and type CONTROL-D. The character will be deleted and all the text that was to the right of the character will be moved to the left one character space.

To insert a character, move the cursor to the right of the character you want to insert after and type CONTROL-I. All of the characters to the right (including the character under the cursor) will be moved one space to the right. For example, let's assume the line you are editing is "My name is Zybar, omnipotent ruler of the planet Glyx." As any

good Glyxxian knows, the name “Glyx” is spelled with two x’s. (It would be blasphemous to think otherwise.) To rectify this somewhat touchy situation, move the cursor onto the x and type CONTROL-I. The x will move one space to the right and a space will be under the cursor. Now type an x (quickly).

To replace characters, just type over them. To return to the Review menu, press the RETURN key. Before proceeding to the next section, take the time to experiment with and get the feel of the Text Entry and Review/Edit functions.

Write to Disk

After typing in your epic novel, you will naturally want to save it on the disk. To do so, type option W, Write to Disk, and the name under which you want to save your file. The computer will save all the text you typed into the text file. (It will write the text buffer to the file.) If you neglect to type a file name and just hit RETURN, the computer will return to the Main menu without doing anything.

Read from Disk

To use this command, type option R and the file name you wish to read from the disk. The computer will then read the text from the file into memory. Again, if you do not type a file name and just hit RETURN, the computer will return to the Main menu without doing anything.

Search

This command is used when you do not know the line number of a certain section of your text. To use it, type S for Search followed by the string you wish to search for. When the computer finds your string, it will print the line number and the element in which your string was found. If you wish to have the computer continue looking (because the one it found was not the one you wanted), type a C. Otherwise, type an X to return to the Main menu.

List to Screen

This command will list your text buffer to the screen, formatted according to your format-control characters. To temporarily halt the listing, type a CONTROL-S. To resume printing, type another CONTROL-S. To abort the listing and return to the Main menu, type a CONTROL-Q.

Print to Printer

This command will print your text buffer to your printer (assuming you have a printer, of course). The text will be formatted according to your format-control characters.

Exit

This is the command to exit FRED. The program will ask “Are you sure?” If you type anything but a Y (Yes), the computer will return to the Main menu with the text buffer intact. If you type a Y, the computer will type the sign-off message and exit the FRED editor. Warning: If you type a Y, the text buffer information is not recoverable; make sure you have saved your file before you Exit FRED.

SUMMARY

FRED is a simple, easy-to-learn text editor. It has many of the features of the much larger word processors like WordStar. You can easily modify it to suit your personal needs. For example, you could add a Search-and-Replace routine which would replace every occurrence of one string with another. Or, you could enlarge the text buffer so it can hold more text. You could even put in *help messages* that would explain to the first-time user what different commands mean and how they work. We submit FRED to the public domain, so you may use this core program to create a full-fledged word processor which you can market.

How Will FRED Help You?

Almost anyone can use FRED. Even the Glyxxians (who don't have anything that even faintly resembles fingers) can use it (thanks to the new Moonbeam Computer Mind-Link board that slips into slot 3 of your Apple). You can use FRED to write letters, draw up reports, store recipes, print out chain letters, or maintain a list of everyone who ever forgot your birthday. Keep in mind, however, that FRED is a marginal text editor and for those really big jobs a larger text editor will be necessary.

Where Do I Go from Here?

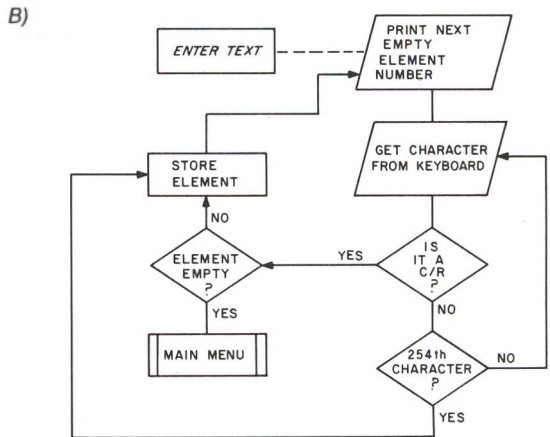
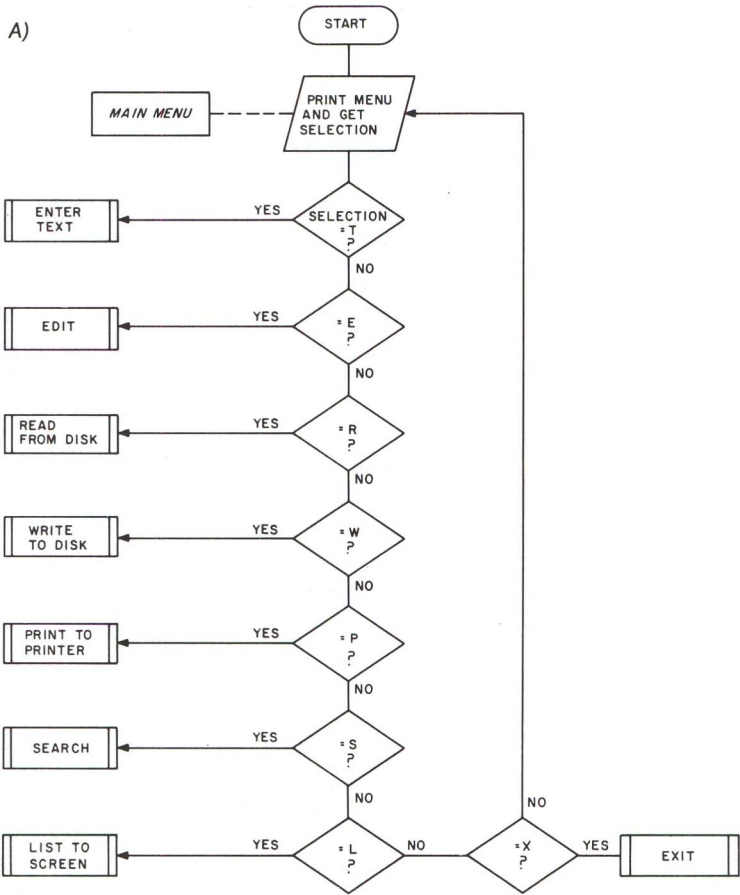
Get yourself well acquainted with FRED. Become friends. Use it and see for yourself how easy it is to type and correct errors on a word processor. After using it for a while, you can decide for yourself whether it is worth it for you to buy a full-fledged word processor.

Yes, Virginia, There Is a Flowchart!

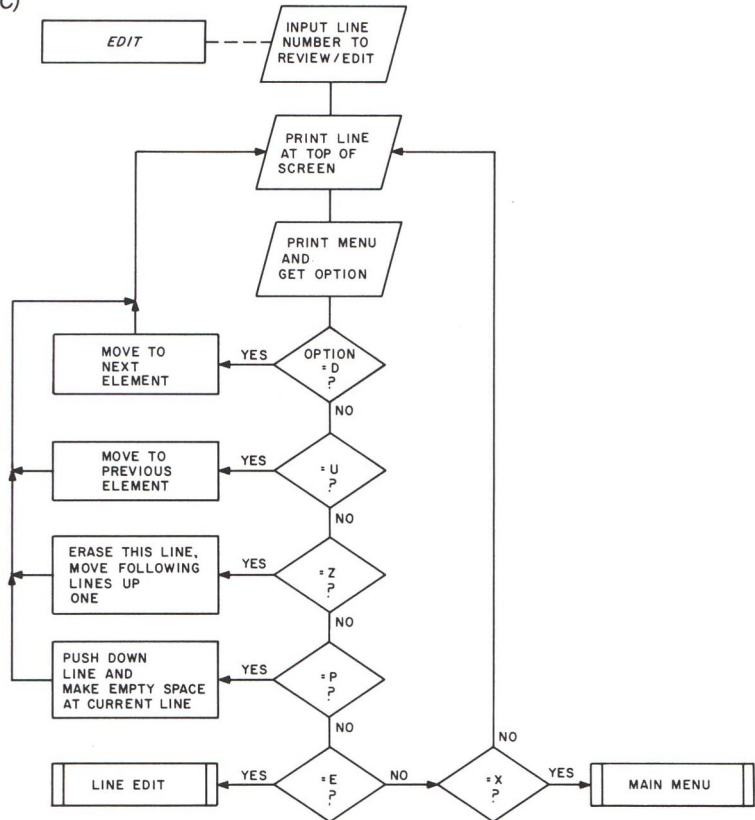
Figure C.1 is a flowchart for the FRED program. The flow begins at the little oval marked START and continues in the direction of the arrows. (In places where there are no arrows, assume that the program flows downward.)

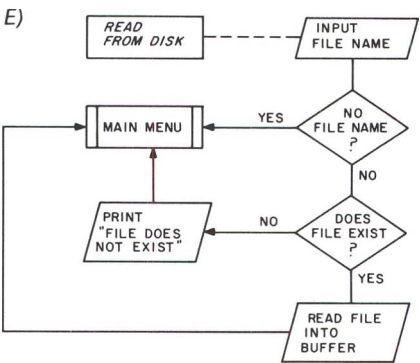
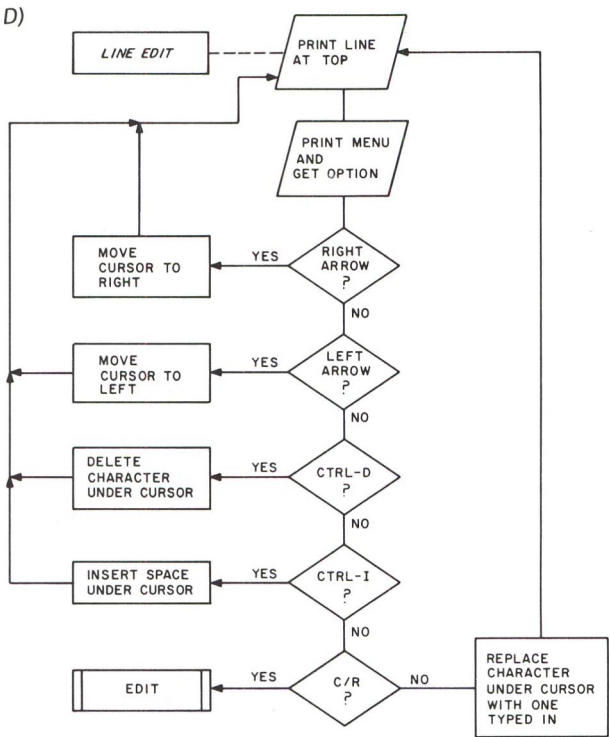
Part A of the flowchart illustrates the Main menu section of the program. It prints the menu, gets the user's selection, and branches to

the chosen subroutine. The parts of the figure following part A illustrate the various subroutines (described above) in the order that they appear on the Main menu.

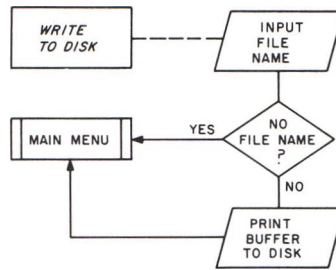


C)

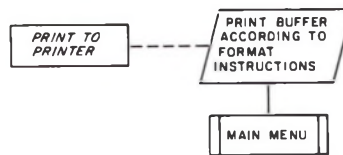




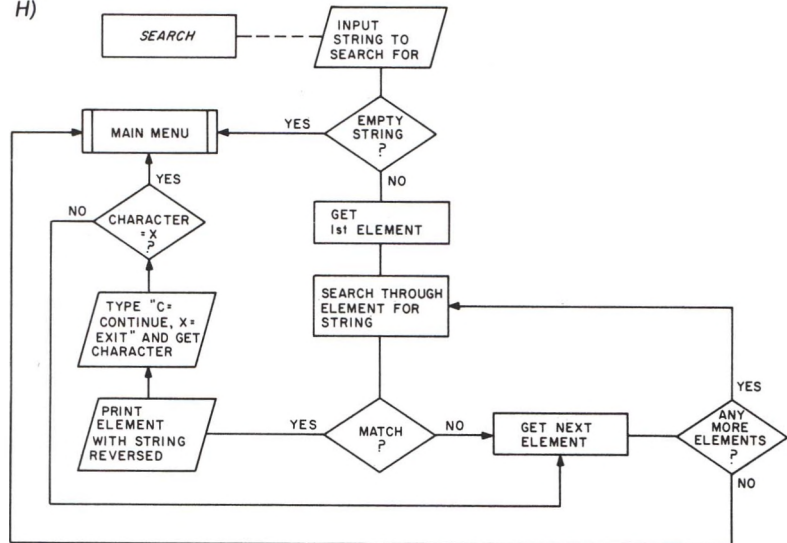
F)



G)



H)



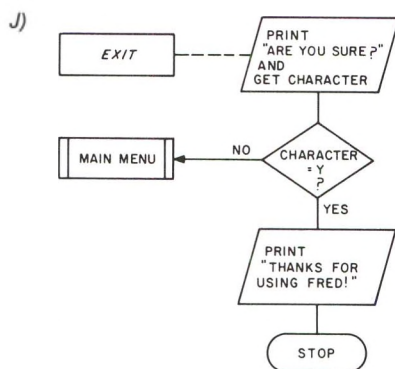
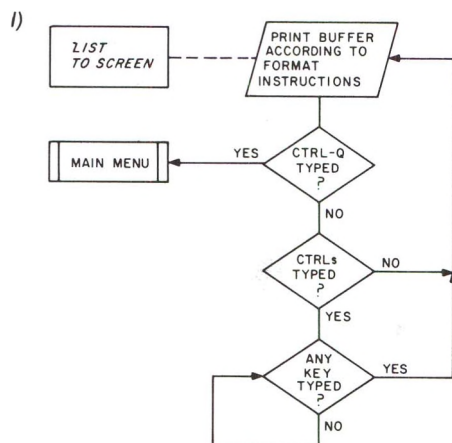


Figure C.1: Flowchart for FRED word-processing program.

FRED (FiRst EDitor)

```

100 DIM A$(500)
110 D$ = CHR$(13) + CHR$(4)
120 HOME
130 PRINT "          F R E D"
140 PRINT : PRINT "VERSION 1.0          MEMORY FREE: ";
150 X = FRE(0): IF X < 0 THEN X = X + 65536
160 IF X < 1024 THEN FLASH
170 PRINT X: NORMAL
180 PRINT : PRINT : PRINT "MENU
190 PRINT : PRINT "T. TEXT ENTRY          W. WRITE TO DISK"
200 PRINT "E. EDIT          P. PRINT TO PRINTER"
210 PRINT "R. READ FROM DISK    S. SEARCH"
220 PRINT "L. LIST TO SCREEN"
230 PRINT : PRINT "X. EXIT"
240 PRINT : PRINT "ENTER YOUR CHOICE: ";: GET A$
250 ON (A$ = "T") + (A$ = "E") * 2 + (A$ = "R") * 3 + (A$ = "W") * 4 + (A
$ = "P") * 5 + (A$ = "S") * 6 + (A$ = "L") * 7 + (A$ = "X") * 8 GOTO
4000,5000,1000,7000,8000,13000,14000,9000
260 GOTO 120
1000 HOME : VTAB 12
1010 INPUT "ENTER FILENAME TO READ: ";FL$
1020 IF FL$ = "" THEN 120
1030 ONERR GOTO 1130
1040 PRINT D$;"OPEN ";FL$
1050 PRINT D$;"READ ";FL$
1060 GOSUB 12000
1070 FOR X = 1 TO 500
1080 INPUT A$(X)
1090 NEXT
1100 PRINT D$;"CLOSE"
1110 POKE 216,0
1120 GOTO 120
1130 IF PEEK(222) < > 5 THEN STOP
1140 INVERSE : PRINT : PRINT "---- FILE DOES NOT EXIST ----": NORMAL
1150 PRINT D$;"DELETE ";FL$
1160 POKE 216,0
1170 GOTO 120
2000 QQ$ = LEFT$(A$(CL),1)
2010 IF QQ$ = CHR$(14) THEN PRINT : PRINT MID$(A$(CL),2);: RETURN
2020 IF QQ$ = CHR$(16) THEN PRINT : PRINT " "; MID$(A$(CL),2);: RETURN
2030 IF QQ$ = CHR$(19) THEN PRINT : PRINT : PRINT MID$(A$(CL),2);: RETURN
2040 PRINT A$(CL);
2050 RETURN
3000 LI$ = ""
3010 IF LEN(LI$) = 254 THEN RETURN
3020 GET A$
3030 IF A$ = CHR$(13) THEN RETURN
3040 IF A$ = CHR$(8) THEN 3080
3050 LI$ = LI$ + A$
3060 IF A$ < " " THEN INVERSE : PRINT CHR$(ASC(A$) + 64);: NORMAL
3070 GOTO 3120
3080 IF LI$ = "" THEN 3130
3090 IF LEN(LI$) = 1 THEN LI$ = "": GOTO 3110
3100 LI$ = LEFT$(LI$, LEN(LI$) - 1)
3110 PRINT A$;" ";A$;: GOTO 3010
3120 PRINT A$;

```

```

3130 GOTO 3010
4000 REM ENTER TEXT
4010 FOR X = 1 TO 500
4020 IF A$(X) = "" THEN 4040
4030 NEXT
4040 HOME
4050 CL = X
4060 PRINT : INVERSE : PRINT CL; ":"; : NORMAL : PRINT " ";
4070 IF A$(CL) < > "" THEN 120
4080 GOSUB 3000
4090 IF LI$ = "" THEN 120
4100 A$(CL) = LI$
4110 CL = CL + 1: IF CL < 501 THEN 4060
4120 PRINT "--- BUFFER FULL ---"
4130 GOTO 120
5000 HOME : VTAB 12: INPUT "LINE NUMBER TO REVIEW/EDIT: ";CL
5010 IF CL < 1 THEN CL = 1
5020 IF CL > 500 THEN CL = 500
5030 HOME
5040 VTAB 1: PRINT "LINE # ";CL: PRINT
5050 GOSUB 2000
5060 VTAB 12: HTAB 1: PRINT "OPTIONS:"
5070 PRINT
5080 PRINT "D- MOVE DOWN ONE LINE"
5090 PRINT "U- MOVE UP ONE LINE"
5100 PRINT "Z- ZAP THIS LINE"
5110 PRINT "P- PUSH THIS LINE DOWN."
5120 PRINT "E- EDIT THIS LINE"
5130 PRINT "X- EXIT"
5140 PRINT
5150 PRINT "ENTER OPTION: ";: GET A$
5160 IF A$ = "D" THEN CL = CL + 1: GOTO 5010
5170 IF A$ = "U" THEN CL = CL - 1: GOTO 5010
5180 IF A$ = "Z" THEN GOSUB 12000: FOR X = CL TO 499: A$(X) = A$(X + 1): NEXT
: A$(500) = "": GOSUB 12010
5190 IF A$ = "P" THEN GOSUB 12000: FOR X = 500 TO CL + 1 STEP - 1: A$(X)
= A$(X - 1): NEXT : A$(CL) = "": GOSUB 12010: GOTO 5010
5200 IF A$ = "E" THEN LI$ = A$(CL): GOSUB 6000: A$(CL) = LI$: GOTO 5010
5210 IF A$ = "X" THEN 120
5220 GOTO 5010
6000 HOME
6010 CP = 2
6020 LI$ = "*" + LI$
6030 HTAB 1: VTAB 1
6040 IF LI$ = "*" THEN 6100
6050 FOR X = 2 TO LEN (LI$)
6060 QQ$ = MID$ (LI$,X,1)
6070 IF QQ$ < " " THEN INVERSE : PRINT CHR$ ( ASC (QQ$) + 64); : NORMAL
: GOTO 6090
6080 PRINT QQ$;
6090 NEXT
6100 PRINT " "
6110 VTAB 12: PRINT "EDIT MODE:"
6120 PRINT
6130 PRINT "RIGHT-ARROW - MOVE CURSOR RIGHT"
6140 PRINT "LEFT-ARROW - MOVE CURSOR LEFT"
6150 PRINT "CTRL-D - DELETE CHAR UNDER CURSOR"
6160 PRINT "CTRL-I - INSERT SPACE UNDER CURSOR"
6170 PRINT "<RETURN> - EXIT EDIT MODE"
6180 PRINT : PRINT

```

```

6190 PRINT "ANY OTHER CHARACTER WILL REPLACE CHARACTER UNDER CURSOR";
6200 VTAB 1: HTAB CP - 1
6210 GET A$
6220 IF A$ = CHR$ (8) THEN CP = CP - 1: IF CP = 1 THEN CP = 2
6230 IF A$ = CHR$ (8) THEN GOTO 6200
6240 IF A$ = CHR$ (21) THEN CP = CP + 1: IF CP > LEN (LI$) THEN LI$ = LI$ + " "
6250 IF A$ = CHR$ (21) THEN GOTO 6200
6260 IF A$ = CHR$ (4) THEN LI$ = LEFT$ (LI$,CP - 1) + MID$ (LI$,CP + 1): GOTO 6030
6270 IF A$ = CHR$ (9) THEN LI$ = LEFT$ (LI$,CP - 1) + " " + MID$ (LI$,CP): GOTO 6030
6280 IF A$ = CHR$ (13) THEN LI$ = MID$ (LI$,2): RETURN
6290 LI$ = LEFT$ (LI$,CP - 1) + A$ + MID$ (LI$,CP + 1)
6300 IF A$ < " " THEN INVERSE : PRINT CHR$ (ASC (A$) + 64);: NORMAL : GOTO 6320
6310 PRINT A$;
6320 CP = CP + 1
6330 IF CP > LEN (LI$) THEN LI$ = LI$ + " "
6340 GOTO 6200
7000 HOME : VTAB 12
7010 INPUT "ENTER FILENAME TO WRITE: ";FL$
7020 IF FL$ = "" THEN 120
7030 GOSUB 12000
7040 FOR X = - 1 TO 0
7050 PRINT D$;"OPEN";FL$
7060 IF X THEN PRINT D$;"DELETE";FL$
7070 NEXT
7080 PRINT D$;"WRITE";FL$
7090 FOR X = 1 TO 500
7100 PRINT A$(X)
7110 NEXT
7120 PRINT D$;"CLOSE";FL$
7130 GOTO 120
8000 GOSUB 12000
8010 GOSUB 10000
8020 FOR CL = 1 TO 500: IF A$(CL) < > "" OR A$(CL + 1) < > "" THEN GOSUB 2000: NEXT
8030 GOSUB 11000
8040 GOTO 120
9000 HOME : VTAB 12: PRINT "ARE YOU SURE? ";: GET A$: IF A$ < > "Y" THEN 120
9010 PRINT : VTAB 12: PRINT "THANKS FOR USING FRED!"
9020 END
10000 PRINT D$;"PR#1"
10010 PRINT CHR$ (9);"8ON"
10020 RETURN
11000 PRINT D$;"PR#0"
11010 RETURN
11020 IF PEEK ( - 16384) = 145 THEN POKE - 16368,0: GOTO 120
12000 FLASH :A$ = "- WAIT, -": GOTO 12020
12010 NORMAL :A$ = " "
12020 HT = PEEK (36):VT = PEEK (37)
12030 HTAB 17: VTAB 1: PRINT A$;: NORMAL
12040 HTAB HT + 1: VTAB VT + 1
12050 RETURN
13000 HOME : VTAB 12
13010 INPUT "STRING TO SEARCH FOR? ";SS$
13020 IF SS$ = "" THEN 120

```

```

13030 LE = LEN (SS$)
13040 FOR X = 1 TO 500
13050 GOSUB 12000
13060 QQ$ = A$(X)
13070 IF QQ$ = "" THEN 13230
13080 LL = LEN (QQ$) + 1
13090 Y = 1:OY = 1:FF = 0
13100 IF MID$ (QQ$,Y,LE) < > SS$ THEN 13180
13110 IF NOT FF THEN PRINT : PRINT "LINE #";X
13120 FF = 1
13130 GOSUB 12010
13140 IF Y < > 1 THEN PRINT MID$ (QQ$,OY,Y - OY);
13150 INVERSE : PRINT SS$;: NORMAL
13160 Y = Y + LE:OY = Y
13170 GOTO 13190
13180 Y = Y + 1
13190 IF Y < LL THEN 13100
13200 IF NOT FF THEN 13230
13210 PRINT MID$ (QQ$,OY)
13220 PRINT "C=CONTINUE, X=EXIT: ";: GET A$: IF A$ = "X" THEN 120
13230 NEXT
13240 GOTO 120
14000 HOME : VTAB 2
14010 CL = 1
14020 GOSUB 2000
14030 CL = CL + 1
14040 IF PEEK ( - 16384) = 145 THEN POKE - 16368,0: GOTO 120
14050 IF PEEK ( - 16384) < > 147 THEN 14090
14060 POKE - 16368,0
14070 IF PEEK ( - 16384) < 128 THEN 14070
14080 POKE - 16368,0
14090 IF CL < 501 THEN 14020
14100 GOTO 120

```

Index

- American Standard Code for Information Interchange (ASCII), 25
- Apple II, 146–148
 - screen display, 72, 146
- Apple III, 146
- Applewriter, 146, 151
- Atari, 147
- Attorney, 117, 124–130
- Authors, 120, 124–128

- Bit, 24
- Block, 82–83
 - definition of, 156
 - deletion of, 55
 - glossary, 84
 - marking of, 82, 83, 120
 - movement, 83
 - read from file, 84
 - write to file, 84
- Boilerplate, 84
- Boldface, 75–77
- Boot, 157
- Buying (*See* Purchase)
- Byte, 24

- Carriage return, 64
- Chained printing, 98
- Character:
 - flag, 60
 - non-printing, 60, 64, 74–78, 82
 - width setting, 74
- Column move, 85
- Command reference guide, 130
- Computer store, 139–140
- Continuous scrolling, 53
- CONTROL key, 31
 - for cursor control, 48
- CP/M, 16, 33, 40
 - word processors compatible with, 149–150
- Cursor, 45
 - control, 45–49
 - control key, 48
 - reverse video, 47

- Database, 18, 165
 - law library, 117
 - mailing list, 119
- Database-management system (DBMS), 166-168
- Data entry, 167-168
- Datasoft, Inc., 147
- Dedicated word processors, 40-43
- Defaults for print formats, 72
- Deletion, 54-55
- Depreciation, 133-134
- Dictionary, spelling, 90-93
 - editing, 93
 - searching, 93
- Digital Equipment VT 100, 109
- Digital Research, 14
- Diskette, floppy, 23, 31
 - 3¼-inch, 17, 33
 - 5¼-inch, 23, 33
 - 8-inch, 23, 33
 - capacity of, 127
 - definition of, 158
 - care of, 32
 - formatting, 159-160
- Disk-based word processor, 39-40, 123
- Disk drive, 31
 - definition, 159
 - number of, 128
- Documentation, 130
- Dot matrix, 29
 - printer, 33-35
- Double strike, 77
- EasyWriter, 148, 152
- ED editor, 14
- Electric Pencil, 14
- Electronic mail, 118
- ENTER key, 30
- ESCAPE key, 31
 - printing with, 120
- Escape, Ltd, 111
- Find command, 79-82
- Financing, 133-136
- Flag characters, 60
- Floppy (*See* Diskette, floppy)
- Fonts, 35
- Footings, 73
- Form editing, 168
- Form generation, 167-168
- Formatting, 57
 - embedded commands, 68
 - off-screen, 57, 60, 68-78, 125-126
 - on-screen, 57, 62-67, 125-126
- Form letters, 97-101, 129
- FRED, 146, 151, 171
- Function key, 31, 160
- Global search, 81
- Glossary block, 84
- Grammar checking, 101-106
- Grammatik, 103-105
- Graphics-figure processing, 107-111
- Headings, 73
- Help menu, 129
- Home user, 121, 125-128, 130
- Horizontal scrolling, 52
- Hyphenation, 67
 - soft vs. hard, 67
- IBM Personal Computer, 36, 148
- Index for manual, 130
- Indexing, 95-96, 121
- Information Unlimited Software, 148
- Insertion, 53
 - in one keystroke, 82
- Interface, 37-38
- Investment tax credit, 134
- Justification, 65-66
 - embedded commands for, 69
 - microspace, 66, 74, 161
- K (kilobyte), 26
- Key, 168
- Keyboard, 30-31
 - touch, 31, 41
- Lear Siegler ADM-3A Dumb Terminal, 109
- Leasing, 135-136
- Legal word processing, 117
- Lexisoft, Inc., 149
- Lexitron, 11
- Line height, 72
- Line spacing, 71
- Lowercase, 41
- Magic Wand, 149, 153
- Mail merge, 97-101, 119, 169
 - definition of, 160
 - function of, 98-99
- Mail-order house, 137-138
- Mailing labels, 100
- Manual, 130-131

- Margin setting, 69, 73
- Mass mail business, 118-119, 124, 126, 130
- Memory, 24-27
 - read-only (ROM), 27
 - screen, 28
 - size, 26-27, 127
 - types of storage, 31-33
- Menu:
 - help, 129
 - on-screen, 12
- Magnetic Card Selectric Typewriter (MC/ST), 10
- Microjustification, 66, 74, 161
- MicroPro International, 14, 150
- Microcomputer, small business, 40
- Microprocessor, 24, 36
- Michael Shrayder Software, 14
- Microspacing, 66, 74, 161
- Mini-diskette (*See* Diskette, floppy, 5¼-inch)
- 3M Company, 11
- Mouse, 48
- Magnetic Tape Selectric Typewriter (MT/ST), 10
- Muse Software, 147

- Nested printing, 98
- New page, 73
- Non-printing character, 60-64
 - for blocks, 82
 - printer control commands, 74-78
- NorthStar, 148
- North Word, 148, 152

- On-screen formatting, 57, 125
- Operating system, 33
- Order processing, 165-166
- Original equipment manufacturer (OEM), 140-141
- Overprint, 78

- Page command, 74
- Page length, 72
- Page numbering, 73
- Page offset, 73
- Paper, 35
- Paragraph reformat, 56
- Peripherals, 23-24
- Pitch, alternate, 78, 156
- Plotter, 72, 111
- Printer, 23, 33-36
 - bidirectional, 156
 - control commands, 74-78
 - dot-matrix, 33-35
 - electrosensitive, 35
 - impact, 33
 - laser, 33
 - requirements for special formatting, 72
 - ribbons, 35
 - thermal, 35
 - use as plotter, 72, 111
 - wheels, 35
- Printing:
 - chained, 98
 - nested, 98
 - pause during, 78
 - records, 169
- Proportional spacing, 34, 66, 162
- Purchase:
 - criteria, 122-131
 - criteria for special users, 117-121, 124-130
 - vendor choice for, 137-141

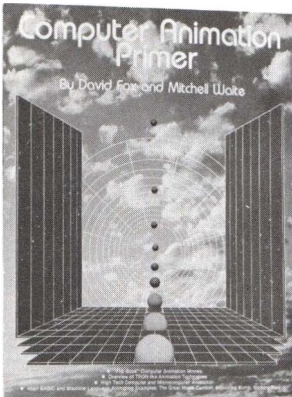
- Radio Shack Scripsit, 14
- Ragged right, 65
- RAM-based word processor, 39, 123
- Random-access memory (RAM), 162
- Read-only memory (ROM), 27, 163
- Reading level, 94, 101, 130
- Record processing, 165
- REFORMAT key, 62
- Reformatting, 56, 62-65
- Rental agreement, 136
- RESET key, 31
- Retail computer store, 139-140
- RETURN key, 30
- Ribbons, 35
 - colored, 78
- Ruler line, 59-60

- Screen:
 - memory, 28
 - organization of, 58-60
 - scrolling, 49-53
 - touch sensitive, 48
- Screen display, 27-30
 - in dedicated vs. micro system, 41
 - resolution, 122
 - size, 23, 122, 126
 - uppercase only, 41
- Scripsit, 14
- Scrolling, 49-53
 - continuous, 53
 - hardware controlled, 52
 - horizontal, 52
 - by screen or line, 51
 - vertical, 50

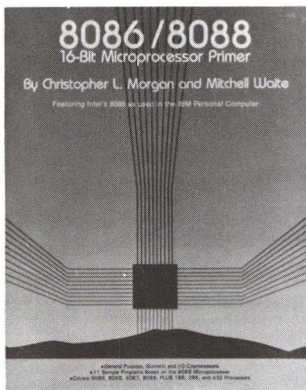
- Search, 79–80
- Search and replace, 80–81
 - global, 81
 - special characters for, 82
- Searching:
 - database, 168
 - in FRED, 174
- Secretary, 117–118
- Select (program), 129, 131, 150, 154
- Select Information Systems, 150
- Service contract, 43
- Single product vendor, 139
- Single-sheet feeder, 23
- Sorter, 169
- Spacing:
 - line, 71
 - proportional, 34, 66, 162
 - vertical, 72
- Spaces, soft, 64
- Spellbinder, 149, 153
- Spelling correctors, 88, 94
- Status line, 59
- Strikeout, 78
- String Search, 79–80
- Subscript, 77
- Super-Text II, 147, 152
- Superscript, 77
- Table of contents generator, 96–97
- Tax credit, 134
- Technical writers, 120–121
- Television terminal, 126
- Terminal, 23
 - definition of, 163
 - television, 126
- Text Wizard, 147, 152
- Threaded word processor (*See* Disk-based word processor)
- Track, 32
- Tractor printer, 23
- Training, 131
- Turnkey system, 42
- Typewriters:
 - early, 7–8
 - intelligent, 11
 - Magnetic Card Selectric, 10
 - Magnetic Tape Selectric, 10
 - semi-intelligent, 10
- Underlining, 75, 77
- Uppercase, 41
- Vendor:
 - choice of, 137–141
 - single product, 139
- Vertical scrolling, 50
- Vertical spacing, 72
- Voice recognition, 18
- Vydec, 12
- Word:
 - indexing, 95
 - count, 101
 - count analysis, 94
- Word processing:
 - definition, 5
 - program for, 38–40
- Word processor:
 - components of, 21–24
 - definition, 5
 - dedicated vs. micro-based, 40–43
 - disk-based, 39, 40, 123
 - functions, 6–7
 - future, 18
 - history, 7–18
 - integrated, 13
 - Japanese, 16–17
 - microcomputer-based, 14–15, 40
 - needs by occupation, 117–121
 - RAM-based, 123
 - video, 12
- Word profiler, 106–107
- Word Star, 14, 150, 153
- Word wrap, 60–61
 - definition, 163
 - with insertion, 53
 - turning off, 61
- Writers (*See* Authors, Technical writers)
- Xerox:
 - 820 SAM, 41, 42
 - Star, 41, 42

More BYTE Books Coauthored by Mitchell Waite

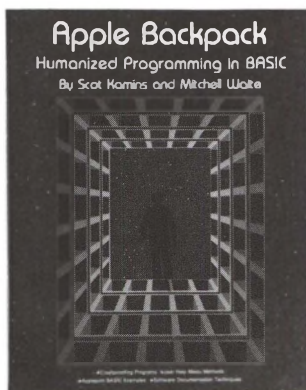
This is just one of four books coauthored by Mitchell Waite and published by BYTE/McGraw-Hill. You'll find the same friendly, easy-to-follow style and user-centered approach in each of the titles listed below. If you enjoyed and learned from this book, you'll certainly find the others in the Waite series equally helpful.



Computer Animation Primer, by Mitchell Waite and David Fox. Another in the ever-popular "Primer" computer book series, this engaging book introduces the exciting world of computer-animated graphics and presents the tools and techniques for creating original animated graphics on your personal computer. The book first describes the theory of animation and the potentials of various products, and then the authors go on to detail the actual programming techniques used in animation, focusing on character set, plotting, player-missile, and scrolling graphics. *Computer Animation Primer* features full-color illustrations and program listings in Atari BASIC and 6502 assembly language.



8086/8088 16-bit Microprocessor Primer, by Christopher Morgan and Mitchell Waite. The new, vastly more powerful 16-bit microprocessors are destined to become the basis for the next generation of personal computers, and this book provides the understanding you need to harness this remarkable advance in technology. Using a comfortable, down-to-earth approach, the authors detail the design and capabilities of the Intel 8086/8088 16-bit microprocessor. Also examined are two 16-bit "coprocessors," the 8087 Numeric Data Processor and the 8089 I/O Processor. In addition, the authors survey the current scene in 16-bit technology, reviewing software and products such as the new 8088-based IBM Personal Computer.



APPLE BACKPACK: Humanized Programming in BASIC, by Scot Kamins and Mitchell Waite. This book aids all computer users by establishing the "user-friendly" approach to programming in BASIC. The authors present concrete methods for developing programs that are not only easy to use, but also hard to misuse. Specific topics include clear screen formatting, crashproofing programs, developing built-in verifications and validations, presenting directions on the video display, and writing helpful, thorough documentation. Appendices feature an educational game program embodying the authors' user-centered approach and a humanized telephone-message-recording program with model documentation, both with complete Applesoft BASIC listings.

Word Processing Primer

The first book to focus primarily on the inexpensive microcomputer-based text-editing products, *Word Processing Primer* gives you a thorough rundown on this new and powerful way to electronically generate, correct, and manage all kinds of typewritten documents.

The authors begin with an overview of the word-processing field, detailing potential applications, suggesting ways to get started, and describing word-processing equipment and programs. A later section on text formatting shows you how to control the appearance of your final printed copy, while another describes programs that will check for grammar and spelling errors, create tables of contents and indexes, and generate personalized form letters.

Selecting a word-processing system is made easy as the authors describe the seven most important features to look for as well as the two most popular types of programs.

Also, "Mitch's Market Sample," a mini-catalog of available products, compares capabilities, features, limitations, and prices. Appendices include a glossary of word-processing terms, details on record processing, and a word-processing program called FRED (FiRst EDitor) that you can type into any personal computer that uses a version of Microsoft BASIC.

All of these features, enhanced by many illustrations, make this the most valuable, informative, and up-to-date word-processing guide available today.



Mitchell Waite is president of The Waite Group, Inc., a California company specializing in programming and book writing for the microcomputer and personal computer market. Mr. Waite has experience in electronics and communications and is fluent in the computer languages Pascal, BASIC, Fortran, Pilot, Forth, 6502, Z-80, 8086, and 8080 assemblers. He is the originator of the popular "Primer" computer book series. Mr. Waite is coauthor of the books *Microcomputer Primer*, *BASIC Programming Primer*, *Pascal Primer*, *CP/M Primer*, *Your Own Computer*, *Computer Animation Primer*, *8086/8088 16-bit Microprocessor Primer*, and *Apple Backpack* and sole

author of *Computer Graphics Primer*. His interests include swimming, hiking, and motorcycles.

Julie Arca is a consultant to the word-processing industry specializing in microcomputer-based systems. She has experience in preparing word-processing instructional programs and designing training manuals for personal-computer-based text editors. Prior to her consulting business Ms. Arca was manager of Computerland of Marin, California. Her interests include children's literature, science fiction, and English mysteries.

cover design by Timothy W. Taussig



McGraw-Hill Book Company
Serving the Need for Knowledge
1221 Avenue of the Americas
New York, NY 10020

ISBN 0-07-067761-1