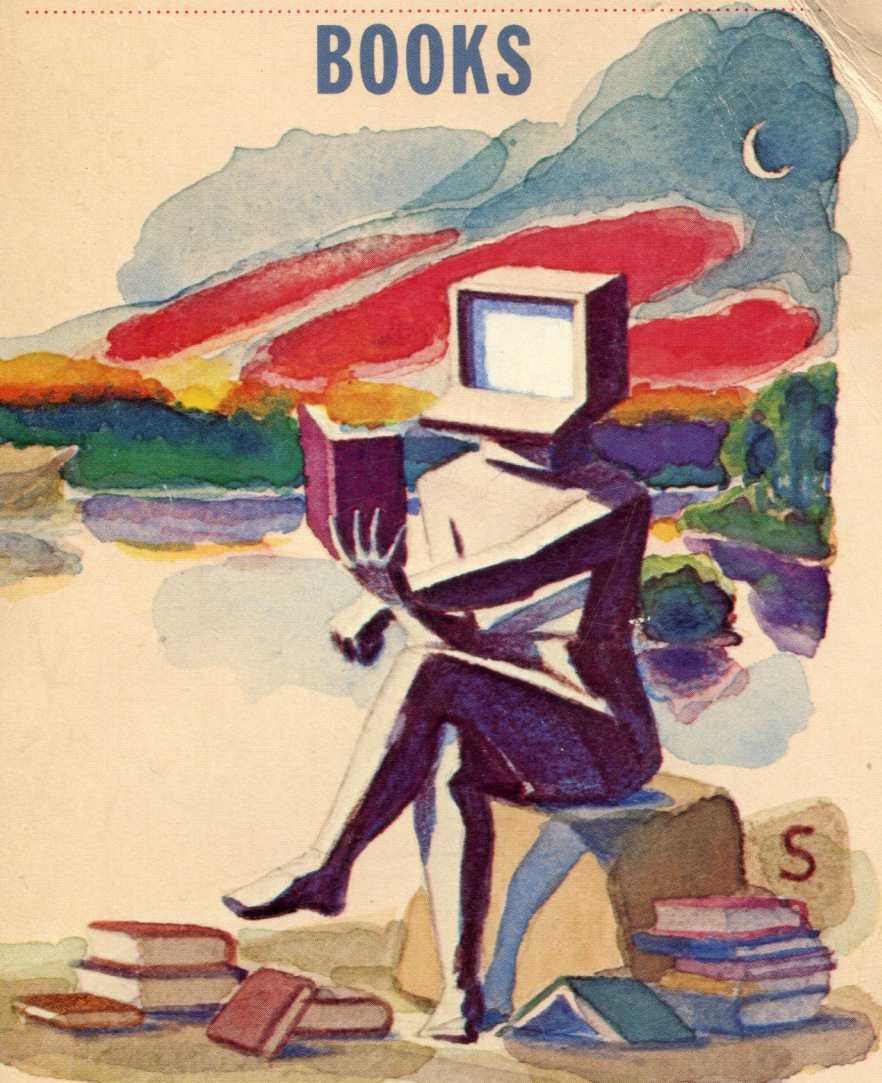


THE READER'S GUIDE TO
M I C R O
COMPUTER
BOOKS



BY MICHAEL NICITA & RONALD PETRUSHA

The Reader's Guide To Microcomputer Books

**By
MICHAEL NICITA and RONALD PETRUSHA**

Golden-Lee Book 1000 Dean Street Brooklyn, N.Y. 11238

Trademarks and Permissions

Apple Computer, Inc. - Apple®II, Apple DOS®,
Apple II Plus®, Applesoft®, Apple Pascal®,
Apple Logo®, AppleWriter®

Artsci, Inc. - Magic Window

Ashton-Tate - dBASE II

Atari, Inc. - Atari® 400/800

Bell Laboratories - UNIX

Commodore Business Machines, Inc. - PET®, Vic 20 CBM®,
PET BASIC®, CBM BASIC®

Compiler Systems, Inc. - CBASIC

Condor Computer - Condor Series 20

Cromemco, Inc. - CROMIX

Data General Corporation - ECLIPSE®

Digital Equipment Corporation - DEC®, PDP-11®

Digital Research, Inc. - CP/M®, MP/M®, ED®, PIP® CP/M-83, MP/M-86,
PL/I-86, Pascal/MT +

DJR Associates, Inc. - FMS-80

FORTH, Inc. - FORTH®

Hewlett Packard Co. - Pascal 1000®, HP1000®

Hunter & Ready, Inc. - VRTX

Information Unlimited Corp. - Easywriter

Information Unlimited Software, Inc. - Easywriter Prof.

Intel Corporation - PL/M-80

International Business Machines Corporation - IBM®

Kmega Microsoftware - VTS-80

Lazer Micro Systems - LISA Assembler

Lexisoft, Inc. - Spellbinder

Micro Data Base Systems - MDBS III

MicroPro International Corp. - WordStar®, Mail Merge®, SpellStar®

Microsoft Corp. - Microsoft®

MUSE Software - Super Text II

Ohio Scientific, Inc. - C4P DF

Osborne Computer Corporation - Osborne 1

Phase One Systems, Inc. - OASIS

Preachtree, Inc. - Magic Wand®

Programma International, Inc. - PIE/FORMAT

Que Corporation - Que

Regents of the University of California - used under a license from

SofTech Mircosystems, Inc. - UCSD Pascal, UCSD p-System

SELECT Information Systems - SELECT

Software Arts, Inc. - DIF, Keystroke Memory

Sorcim Corporation - SuperCalc, Pascal/M, Pascal/M-86
Tandy Corporation - TRS-80®Model I, TRS-80®Model II,
TRS-80®Model III, TRS-80®Model 16, SCRIPSIT, TRSDOS®
Timex Corporation - Timex®
Trustees of Dartmouth College - BASIC®
U.S. Dept. of Defence (ADA Joint Program Office) - ADA
Visicorp, Inc. - VisiCalc®
Zilog, Inc. - Z8®, Z80®, Z8000®

Any software product, component, or computer name omitted is assumed to be the trademark of the manufacturer.

“DID YOU EVER HAVE TO MAKE UP YOUR MIND?”

(John Sebastian) © 1965, 1966

The Hudson Bay Music Company. Used by permission.

All rights reserved.

Cover Art and Illustrations: Steve Salerno
Cover Design: Don Morris
Copy Editor: Laura Ross
Technical Editor: Dave Burstein

Library of Congress Cataloging in Publication Data

Nicita, Michael, 1953-
The Reader's Guide To Microcomputer Books
Includes Indexes.

1. Microcomputers—Book reviews. I. Petrusha, Ronald 1951- . II. Title.

QA76.5.N49 1983 016.00164 83-5510 ISBN 0-912331-00-3

Copyright 1983 by Michael Nicita and Ronald Petrusha.
World Rights Reserved. No part of this publication may be transmitted,
stored, or reproduced in any way, including, but not limited
to, electronic, mechanical, or photographic means, without the prior
agreement and written permission of the authors.

Published by Golden-Lee Book

Preface

Welcome to the first critical guide to microcomputer "bookware." This book evaluates a representative cross-section of the microcomputer titles which currently compete for your time and money. The individual reviews treat each book's content and quality of presentation, and in most cases conclude with our opinion as to its relative merits. We make no apologies for the sometimes acerbic opinions expressed in this first edition (we'll save the apologies for the second edition).

The need for such a guide is obvious. The publishing domain has certainly not been exempt from the flood of misleading, inferior products which have characterized other segments of the microcomputer market. Indeed, the scale weighing quality treatments against books of absolutely no redeeming value is just beginning to approach some measure of equilibrium. We hope our ratings allow you, whatever your specific microcomputer interests, to tip this balance in your favor.

We believe that this book is important for reasons which are even weightier than the appeal to a reader's pocketbook. Although computer technology has only developed within the last four decades, and microcomputers have appeared within the last ten years, their impact has been enormous. Their full potential as yet remains untapped; computer technology is capable of transforming the face of modern society in a way which will make the industrial revolution of the eighteenth and nineteenth centuries pale by comparison. The exact form of these changes, however, will be determined by the extent to which ordinary people are capable of understanding and shaping these developments. Viewed in this context, a bad book becomes more than simply a wasteful experience. The failure to teach results in a failure to learn; and the failure to teach a complex topic precludes later learning.

In order to guide the reader through current microcomputer literature, we have categorized more than four hundred books within six subject areas. Ratings ranging from ten to one hundred have been highlighted for quick reference. (Gradations can be regarded as follows: 90-100 excellent; 80-90 superior; 70-80 passable; 60-70 barely adequate; 50-60 inferior; 40-50 read at your own risk; 10-40 the best thing about these books may be the reviews.) Separate indices provide title, rating, author and rating within subject listings.

The level of our writing is tailored to our perception of each book's respective audience. We don't expect a reader of a purely introductory book to have any familiarity with personal computers; if the introduction happens to deal with Z80 assembly language programming,

however, we assume a minimal degree of programming and operational expertise.

Knowing which titles are worthy of purchase is not enough; they must be available. But booksellers frequently know as little about the quality of the books they are selling as do prospective readers. It is our hope that besides being a guide for the general public, this text will also serve as a sourcebook for computer outlets and bookstores; for that purpose, we have included an appendix containing three model inventories for stores of different sizes.

We extend our thanks to the talented production team responsible for this book: Leroy Spellman, Lee Heiman, Milt Heiman, Nelida Moya, Terry Toth, Laura Ross, Harriet Pinkston, Antoinette Hayes, Dave Burstein, Ron River, Gary Riecke, Steve Salerno, Don Morris, Kathleen Edwards, Samuel Conway, Kevin Jones, William Campbell, Brian Donovan, George Leibson, Noel Derecki, Software Emporium and the firm of Ackerman & Sobel.

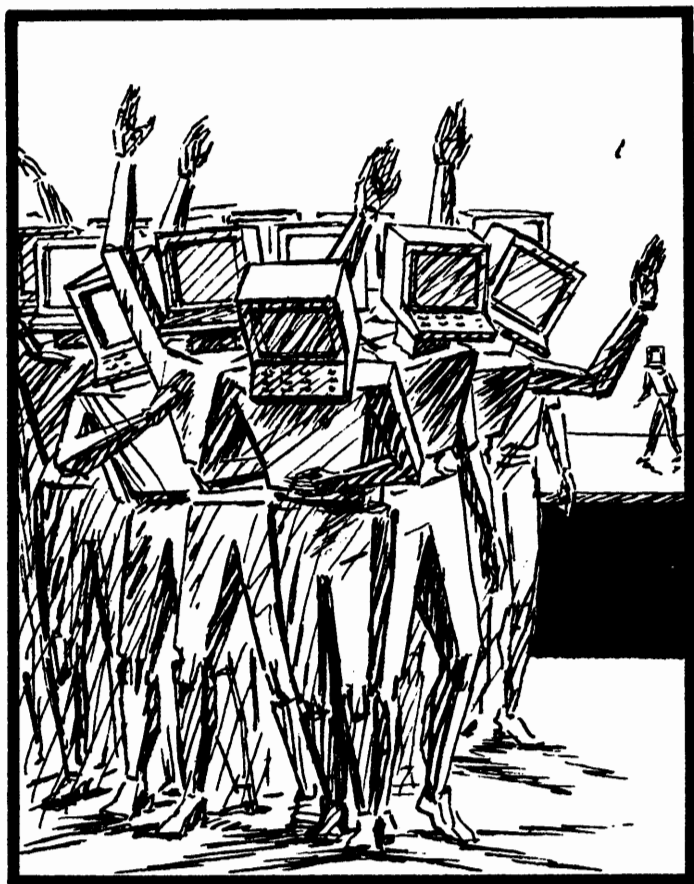
Without detailing their individual contributions, we give them our highest rating.

Michael Nicita
Ronald Petruska
New York, 1983

Table of Contents

1. Microcomputer Introductions	1
2. Microprocessors	63
3. Operating Systems and Hardware Design	91
4. Programming	
General	114
Basic	122
Fortran.....	147
Pascal.....	153
Others	167
5. Software and Applications	
General	182
VisiCalc.....	219
Word Processing	227
6. Specific Microcomputer Systems	
Apple.....	238
IBM	275
TRS-80.....	286
Atari.....	311
Commodore	323
Timex/Sinclair.....	332
7. Model Inventory/Appendix	345
8. Indices	
Alphabetic	349
Rating	361
Author.....	375
Rating within subject.....	397

To Wallis: long questions deserve long answers



1

Microcomputer Introductions

"In possibility, however, everything is possible. Hence in possibility one can go astray in possible ways, but essentially two. One form is the wishful, yearning form, the other is the melancholy fantastic—on the one hand hope; on the other, fear or anguished dread."

Søren Kierkegaard

*"Did you ever have to finally decide?
And say yes to one and let the other one ride?
There's so many changes and tears you must hide,
Did you ever have to finally decide?"*

John Sebastian

The first choices which confront the prospective reader of microcomputer books are the most bewildering.

Introductory titles abound. Recommendations conflict. For many, potential new dimensions of creativity and productivity fade with each turn of the page of the wrong first book. For others, such possibilities are completely extinguished by the purchase of the wrong machine.

The survey which follows is intended to brighten these initial encounters. Under the general umbrella of "Microcomputer Introductions," we have included books which provide a range of approaches to microcomputer topics: works on **COMPUTER LITERACY** attempt to increase general familiarity with computers, especially among non-users; those devoted to **MICROCOMPUTER SELECTION**, by focusing on computers in the abstract or on specific systems, aim at assisting the reader in choosing a system based on his or her needs; **REFERENCE WORKS** add to the user's knowledge and skill at

microcomputer use; finally, *COMPUTER-RELATED BOOKS* include sociological, journalistic or literary treatments of computers and computerization. Though the titles in this section are arranged in alphabetical order, we remind the more discriminating reader that referring to either of the ratings indices at the end of the book will quickly highlight superior treatments.

ARE YOU COMPUTER LITERATE?

Karen Billings and David Moursund

Level: Novice Rating: **65**

	Dilithium	1979	Paper	
148 Pages	ISBN: 0-918398-29-0		6" x 8"	\$9.95

This book was not written for computer users per se, but aims instead at teaching "everyone...something about computers."(v) This something is fairly minimal: a number of myths about computers are debunked, a few of the most essential concepts of computers are discussed, a wide range of applications are mentioned, and some of the social and political problems resulting from the computer revolution are presented.

Given the book's modest ambitions, its value to the reader very much depends on which particular "everyone" happens to be reading it. The book's language and focus make it most appropriate for general classes in computer literacy at the junior high school level. The appended bibliography listing introductory "secondary" works on computers for readers in the intermediate and high school grade levels and providing information on films about computers, indicates that the authors intended this to be their target audience. When used in the classroom, the book's general, non-technical treatment of computers and computerization provides a provocative medium for expanding the student's familiarity with computers and increasing his or her awareness of the possibilities and problems of their application.

The introduction to *ARE YOU COMPUTER LITERATE?*, however, also emphasizes its use for self-instruction, presumably by adult readers. For this purpose, the book is clearly inadequate. For actual microcomputer users, its treatment is most unsatisfactory, since the book is not designed to provide hands-on experience in using computers. For a more general adult audience, simple ideas are presented in simplistic ways, sentences rarely exceed seven or so words, and vain attempts to sustain the reader's interest are made through a series of very elementary quizzes which double as puzzles. The use of such a format to educate an adult

reader seems to assume that those who are "computer illiterate" are otherwise virtual illiterates as well.

THE BEGINNER'S GUIDE TO BUYING A PERSONAL COMPUTER

Richard Mansfield, Myron D. Miller, et.al.

Level: Novice Rating: **75**

	COMPUTE!	1982	Paper	
76 Pages	ISBN: 0-942386-03-5		6" x 9"	\$3.95

Considering both its size and price, THE BEGINNER'S GUIDE can do no more than introduce the reader to some of the factors involved in selecting a microcomputer, and steer him or her to more comprehensive sources of information. Although it makes rather exaggerated claims in early chapters, the book admirably succeeds in fulfilling this admittedly limited goal. In addition, ten pages of charts contain feature-by-feature comparisons of twenty-two popular microcomputers, including list prices of basic and expanded systems.

Provided that the reader looking for a simple system actually heeds the book's advice to "learn first, buy later,"(9) THE BEGINNER'S GUIDE provides a short, highly practical discussion of features which potential purchasers should consider. It therefore serves as a superior, inexpensive introduction to the process of selecting an appliance computer.

THE BEGINNER'S GUIDE TO COMPUTERS

Robin Bradbeer, Peter De Bono and Peter Laurie

Level: Novice Rating: **80**

	Addison-Wesley	1982	Paper	
208 Pages	ISBN: 0-201-11209-4		7" x 9"	\$10.35

THE BEGINNER'S GUIDE TO COMPUTERS was originally published by the British Broadcasting Corporation to accompany its series on computer literacy. The book includes a very accessible, non-intimidating introduction to computers for adult readers. Its most notable and valuable feature, however, is that it attempts to actively involve the reader in the computer revolution rather than to foster a purely passive understanding of computers.

The presumably uninformed reader is first introduced to the growing application of computers in all areas, and then led through a discus-

sion of the problem-solving abilities of computer hardware and software. The light, well-written treatment focuses on analogies and examples to make its points, and succeeds in bringing the computer revolution very close to home by including chapters on programming techniques and "You and Your Microcomputer."

The major shortcoming of *THE BEGINNER'S GUIDE TO COMPUTERS* is that it clearly shies away from treating the social and economic consequences of computerization. The discussion of computerized X-ray analysis, for instance, is disturbing, since it implicitly questions the "scientific" accuracy of computer-generated diagnoses—and then rapidly moves on to another topic without examining this problem more fully. The comparison of contemporary computer technology with printing in the fifteenth century raises some clear political questions which, again, the authors fail to develop further.

The book's assumption that the cause of computer literacy can best be served if computers are in the hands of everyone is, we believe, a correct one. It is for this reason that we recommend *A BEGINNER'S GUIDE TO COMPUTERS* as a sound introduction to computer literacy.

BUSINESS SYSTEM BUYER'S GUIDE

Adam Osborne

Level: Novice Rating: **80**

	Osborne	1982	Paper		
187 Pages	ISBN: 0-931988-47-0		5" x 8"		\$7.95

Adam Osborne, founder and president of Osborne Computer Corporation and former president and general manager of Osborne/McGraw-Hill, is one of the brightest stars in the growing constellation of individuals that populate the microcomputer industry. Dr. Osborne, a technical writer of considerable reputation (see author index), has, with *BUSINESS SYSTEM BUYER'S GUIDE*, published a non-technical survey of small computer systems.

This is a Peterson's guide for the small business manager journeying through the microcomputer countryside. Four case histories open the author's discussion of pitfalls to be avoided en route to successful computer selection. The primary thrust of the text which follows, aside from providing a rudimentary background in microcomputer fundamentals, is an elaboration of the themes first introduced in these case studies. Osborne recommends that system specifications be developed at the earliest stage of the planning process, and that computer consumers shop for software that adequately performs according to these specifications before hardware is considered. Certain paths are to be une-

quivocally avoided: among them are mail order companies, software that requires complex diskette handling, and vendors who "promise open-ended service at no additional cost."

Two other sections supplement the author's sound advice on microcomputer selection: an overview of computer system components and a cross-vendor product summary. The brief hardware survey displays examples of the most popular microcomputer systems and describes basic criteria for the selection of peripherals. The product summary spans packaged software, operating systems, programming languages and hardware offerings.

Our only qualification regarding this lucid, professional guide to computer selection is the author's reticence to critique any of the hundreds of products listed. Despite claims that "grading products is usually pointless," and "products change so quickly that any comparative analysis would include glaring omissions," this guide would be infinitely more valuable if Osborne's unequivocal advice extended to the otherwise bland recital of available microcomputer products.

BUSINESS SYSTEMS BUYER'S GUIDE presents a clear, adequate survey of the personal computer marketplace for the computer-confused small businessman.

THE COMPLETE BOOK OF HOME COMPUTERS

Van Waterford

Level: Novice Rating: **70**

Tab 1982 Paper
250 Pages ISBN: 0-8306-1423-0 6" x 9" \$10.95

The contents of Van Waterford's book actually illustrate one of the basic "don'ts" of computer selection—that the potential buyer should never select a system based on what he or she might do with it after a few years' time, using devices which have not yet been manufactured or perfected. Waterford clearly has his eye on the future, as is evidenced by his extensive discussion of networking (which occupies most of the first chapter), touch-input video display terminals, voice input terminals, voice recognition, and computerized homes. Not all of this, however, is of especially great value to the potential microcomputer purchaser interested in buying a machine in the present.

Perhaps the most useful portion of the book lists a range of the popular microcomputers now on the market, with special attention to Ohio Scientific computers, and particularly to the C8P DF, the system loaned to Waterford by Ohio Scientific so that he could familiarize himself with the operation of microcomputers. This section as a whole

(pp. 129-238) has been assembled with some care, and in many cases lists the unique or noteworthy features of each system, the peripherals which can be interfaced to it, and the range of software available for it.

But aside from the computer manufacturers which the book has overlooked (e.g., Osborne), *THE COMPLETE BOOK OF HOME COMPUTERS* remains decidedly incomplete. Waterford's discussion of microcomputer basics is schematic, and focuses more on available devices than on the reasons for their application. His guide to microcomputers is not critical in its focus, nor does it permit the reader easily to compare one machine with another. As a result, any discussion which might allow the potential purchaser to select a system based on his or her own needs is conspicuously absent from this book.

THE COMPLETE HANDBOOK OF PERSONAL COMPUTER COMMUNICATIONS

Everything You Need to Go Online With The World

Alfred Glossbrenner

Level: All Rating: **90**

St. Martin's Press 1983 Paper
295 Pages ISBN: 0-312-15718-5 6" x 9" \$14.95

The hottest item flying out of hardware distributors' inventories at the time of this writing are not joysticks, paddles or color monitors—or even system units—but modems. As the installed base of personal computer users matures, and owners become more technically sophisticated, many will opt, in Alfred Glossbrenner's words, to "go online with the world." *THE COMPLETE HANDBOOK OF PERSONAL COMPUTER COMMUNICATIONS* is a timely, informative guide through this "expanding universe of incredible size and power."

With the exception of one chapter which treats the technical details of telecommunications, most purely technical topics are avoided (packet-switching, for example, is never discussed). Rather, Glossbenner has compiled a consumer's directory to the communications super-market: hardware, software and service selection form the three primary topics of his book. "Online tips" are scattered throughout the text, giving the reader the benefit of the author's experience with modem adapters, particularly good reference manuals or operational short-cuts. Specific products are recommended in the course of these discussions; the book's real utility lies in using the author's advice on the best sources of

modems, services and software packages. In this respect, Glossbrenner has provided the literary equivalent of the valuable Bulletin Board service available to "telecommunicators."

One striking aspect of the text is its own structural similarity to many of the communications services it describes: an introductory section gives a quick, readable precis of telecommunication fundamentals and information service options, with subsequent chapters and appendices exploring these topics in greater detail. This "layered" approach simultaneously satisfies the needs of novices and experienced users. Combined with the author's vibrant, well-paced text, it makes for an ideal first book of microcomputer communications.

After versing the reader in the standards and naming conventions most often encountered in making initial hardware and software purchases, the author details three types of services currently available: encyclopedic databases, information utilities and news and business databases. Chapters on THE SOURCE, COMPUSERVE and THE DOW JONES NEWS RETRIEVAL SERVICE cover comparative costs, the type of information available, documentation and examples of system use (which rate as mini-operation manuals). Additional sections explain techniques for telecommunications troubleshooting, bulletin board systems and computerized type setting.

THE COMPLETE HANDBOOK OF PERSONAL COMPUTER COMMUNICATIONS meets a rising demand in the microcomputer marketplace with a quality, non-exploitative presentation. This is rare. Every microcomputer user interested in communications will benefit from Mr. Glossbrenner's useful, refreshing text.

COMPUTER CAPERS

Tales of Electronic Thievery, Embezzlement & Fraud

Thomas Whiteside

Level: ALL Rating: **85**

Mentor	1978	Paper	
166 Pages	ISBN: 0-451-62080-1	4" x 7"	\$2.50

Whiteside's book invokes a litany of ingenious computer professionals who, through anger or avarice, let themselves be seduced by data processing's "dark side." From the disgruntled employee who plants software timebombs in his company's computer to the classic case of the computer-assisted embezzler, each story entertains as it startles. Though few of the crimes are microcomputer-related (probably just a function of the 1978 copyright) the personal computer user will nevertheless find

himself fascinated by the elegance of some customized, rather perverse applications software (James Harlow's costing algorithms were our favorite).

The author's approach, to be sure, is a serious one: one appendix contains proposed legislation to control computer crime. Certainly, the individual cases he relates are government and management nightmares. Intentionally or not, however, one has difficulty suppressing amusement, even technical admiration, for *COMPUTER CAPERS'* "panoply of deviant geniuses."

COMPUTER CONSCIOUSNESS

Surviving the Automated 80's

H. Dominic Covvey and Neil Harding McAlister

Level: Novice Rating: **80**

Addison-Wesley 1980 Paper
211 Pages ISBN: 0-201-01939-6 6" x 9" \$6.95

The authors' common specialty in the field of medical computing evinces itself in the primarily mainframe flavor of this non-technical introduction to computing. This is especially true of those chapters which treat communications, operating systems and various financial approaches to computer ownership. The book possesses a much broader appeal, however, owing to its original discussions of the "geneology" of computer system components and a jargon-less, conversational presentation of data processing basics.

The typical hardware configuration is first dissected component-by-component: the chapter on "input devices," for example, explains the evolution of the current cathode ray tube from its earliest incarnation as "the primeval switch." The anthropological analogy even extends to phylogenic "trees," which show "the input kingdom's" various branchings over the course of evolutionary time to optical character readers and digitizers (paper tape becomes extinct). It is a graphically effective approach, which also works well for "the output kingdom."

Comparatively little attention is given to software, with major emphasis placed instead on the more operationally related considerations of hardware selection, buy/lease/rent options, and system installation. Bibliographies are included at the end of each chapter; unfortunately, they concentrate on dated *DATAMATION* and *SCIENTIFIC AMERICAN* articles and a few textbooks—not the most accessible stepping stones for the non-technical reader. In addition, the simplicity of some of the text's charts borders on pure inaccuracy; one graph, for example, shows ease-

of-use inversely proportional to hardware cost—a marginally true but also marginally dangerous assertion, as any “user-friendly” programming advocate will note.

COMPUTER CONSCIOUSNESS provides a very personal, unique slant on computing. Its shortcomings are generally compensated for by the fresh, comprehensive presentation of its topic.

COMPUTER DICTIONARY 3rd Edition

Charles J. Sippl and Roger J. Sippl

Level: All Rating: **95**

Sams 1980 Paper
624 Pages ISBN: 0-672-21652-3 6" x 9" \$15.95

COMPUTER DICTIONARY & HANDBOOK 3rd Edition

Charles J. Sippl and Roger J. Sippl

Level: All Rating: **95**

Sams 1980 Cloth
928 Pages ISBN: 0-672-21632-9 6" x 9" \$34.95

Despite their difference in titles, both the COMPUTER DICTIONARY and the MICROCOMPUTER DICTIONARY are primarily glossaries of terms relevant to microcomputer users, while COMPUTER DICTIONARY AND HANDBOOK is a hardcover equivalent of the COMPUTER DICTIONARY. The MICROCOMPUTER DICTIONARY contains some 5,000 entries; the number of words defined in the COMPUTER DICTIONARY is not mentioned anywhere in the introduction, although it certainly exceeds 5,000. The definitions provided in both books are usually quite comprehensible, and both works are interspersed with useful photographs and illustrations. The MICROCOMPUTER DICTIONARY also contains two appendices which explain the basics of microprocessors and microcomputer systems. Although it has no similar appendices, the production quality of the COMPUTER DICTIONARY is somewhat higher. On the other hand, aside from the same 624-page dictionary contained in the paperbound COMPUTER DICTIONARY, the cloth COMPUTER DICTIONARY AND HANDBOOK contains 14 lengthy and useful appendices which cover such varied topics as modern computer technology, mathematical and statistical definitions, computer languages, word processing systems, and the possible future impact of computerization on a variety of fields.

Since both dictionaries are intended as works of reference in the same field, share a common author, and have been issued by the same publisher, the question of the way in which they differ is sure to arise. An inspection of the contents of these two dictionaries, however, does not permit any hard and fast conclusions. The introduction to the COMPUTER DICTIONARY indicates that it is somewhat more oriented toward the business person, who can use it as a "browsing dictionary" in order to avoid embarrassment when trying to discuss the growing application of computers in business. Some of this difference, however, is merely window dressing; for example, this dictionary frequently contains abbreviated organizational names (ABA, ACM, ADAPSO, AFIPS, etc.) which are absent in the MICROCOMPUTER DICTIONARY. In other cases, terms or concepts which are of special importance in a business environment, such as phrases related to multi-user systems, are actually more fully covered in the MICROCOMPUTER DICTIONARY. Although a few conclusions can be drawn (the COMPUTER DICTIONARY, for example, tends to have more terms relating to gate logic), it appears that material has been included in either or both of these dictionaries on a random basis.

There is, obviously, some duplication of terms in these two volumes, and when this occurs, one definition tends only to slightly rephrase the other. The actual extent of this duplication, however, is fairly low, as the comparison of randomly selected pages from both dictionaries shows:

MICRO. D. PAGE	TERMS ON PAGE	% IN COMP. % IN DICT.	C.D. PAGE	TERMS	% IN M.D.
20	12	50.0%	50	19	21.0%
57	8	50.0%	123	20	25.0%
72	14	7.1%	179	31	0.0%
111	16	31.2%	226	21	19.0%
149	12	50.0%	274	16	31.2%
212	4	44.4%	341	14	42.8%
253	15	33.3%	387	15	26.7%
316	13	38.5%	462	25	24.0%
397	4	25.0%	526	21	28.6%
484	16	68.8%	601	25	25.0%

The extent of duplication is much lower in the COMPUTER DICTIONARY, then, largely because it contains more words in the first place, but also because many of its terms are very briefly defined and it is better cross-indexed.

Both of these dictionaries are outstanding works of reference, and both are highly recommended.

COMPUTER ESTABLISHMENT

The Inside Story of America's Most Dynamic Industry

Katharine Davis Fishman

Level: All Rating: **90**

McGraw-Hill 1982 Paper
 470 Pages ISBN: 0-07-021127-2 5" x 8" \$7.95

Katherine Fishman's narrative understandably revolves around that pre-eminent member of the computer establishment, IBM; the thousands of pages of testimony and internal documents released during the course of the U.S. Government's anti-trust case (dropped on January 7, 1982) provide the reader with "a slow motion scan of three decades of computer history."

These documents add a dimension of authenticity to the text, particularly to those chapters treating corporate jockeying at IBM, but are not essential to what is basically a history of the entire computer industry. While Fishman's preoccupation with IBM mirrors the industry's preoccupation with this corporate giant, it does not detract from the equally meticulous treatments of each of the "seven dwarfs"—those corporate players shadowed into relatively minor roles (and market shares) by IBM. Character studies of the Watsons at IBM, William Norris at CDC, and Kenneth Olsen at DEC, reveal the peculiar, lasting imprints that individuals have had upon the seemingly monolithic corporations they founded. Fishman's attention to detail, the result of years of extensive research, provides the reader with a journalistic tour-de-force; it is a study that every member of the computer community will come to value.

COMPUTER LANGUAGE REFERENCE GUIDE

Harry L. Helms, Jr.

Level: Novice Rating: **75**

Sams 1980 Paper
 109 Pages ISBN: 0-672-21786-4 6" x 9" \$7.95

The author of INTRODUCTION TO MICROCOMPUTERS FOR THE HAM SHACK (see our review) reappears with an equally brief text of much broader appeal. THE COMPUTER LANGUAGE REFERENCE GUIDE is a comparative study of the syntax and format differences of seven programming languages: Algol, BASIC, COBOL, FORTRAN, LISP,

Pascal and PL/I. Each language is given a short introduction and then analyzed according to a set of standardized criteria; these include program format, relational operators, procedures and functions, and conditional statements, as well as those "anomalies" unique to the syntax or structure of a given language.

The author assumes that the reader, being "fluent" in one of the languages his book treats, will have at least one reference point for a cursory examination of the companion languages analyzed. The most common versions of each language have been used for this comparison. Descriptions of high level languages are usually an obligatory topic for introductory microcomputer or programming books; none, however, provide the quick, succinct reference that Mr. Helms makes available.

THE COMPUTER LANGUAGE REFERENCE GUIDE accomplishes its task of providing a whirlwind tour of the major distinctions among the most popular programming languages; we recommend it to those programmers who have always been curious about how "the other guy" does it (and usually in fewer lines of code).

COMPUTER LITERACY

Problem-Solving With Computers

Carin E. Horn and James L. Poirot

Level: Novice Rating: **45**

	Sterling Swift	1981	Paper
304 Pages	ISBN: 0-88408-133-8	7" x 10"	\$13.95

COMPUTER LITERACY: PROBLEM-SOLVING WITH COMPUTERS is "...designed for use by the beginning student who has no previous related computer coursework." Judging from its content and presentation, we presume that the student is at the junior high school level. Other than a single chapter which discusses the sociological impact of computerization—including a summary of the Privacy Act of 1974—its utility would appear to be limited to this student audience.

The differences between a barely adequate textbook with little appeal for a wider audience (COMPUTER LITERACY), and a superior text which becomes a bestseller (BASIC BASIC) is one of presentation: in one case, the narrative thinly disguises the syllabus it is coerced into covering; in the other, topics are developed in the natural flow of a masterful delivery. One reads COMPUTER LITERACY and wonders which came first: the student exercises or the chapters whose material they test.

As if it were an extension of the initial chapter defining computer terms, a chapter which treats the history of computing accomplishes its task by "defining" the lives of prominent historical figures in concise,

easily-memorized sentences. Typical computer applications are then described. The one truly informative chapter in *COMPUTER LITERACY* follows; entitled "The Value of Information in Society," it treats the report of the Privacy Protection Study Commission, the Federal Computer Protection Act, and, as noted earlier, the Privacy Act of 1974.

Hardware and software overviews of microcomputers preface an introduction to BASIC using elementary flowcharting techniques. Here the text is clearly inferior to a number of popular treatments of the same subject (which have, we hope, already been read by a significant portion of *COMPUTER LITERACY*'s captive student audience). Minimal BASIC is covered in examples devoid of the slightest trace of originality. While the text may assist the junior high school student in identifying sections of a BASIC program, it's superficial treatment never imparts enough solid knowledge to enable him or her to write one.

COMPUTER LITERACY: PROBLEM-SOLVING WITH COMPUTERS is clearly a second-rate introduction to computing. We recommend that the general reading public avoid its purchase; we commiserate with those students who may be temporarily unable to exercise such sound judgement.

COMPUTERS AND EDUCATION

James L. Poirot

Level: Novice Rating: **50**

Sterling Swift	1980	Paper	
89 Pages	ISBN: 0-88408-137-0	6" x 9"	\$6.95

The introduction to *COMPUTERS AND EDUCATION* notes that most teachers received their training before the microcomputer revolution, and therefore lack a knowledge of computers; to rectify this problem, the book is devoted to spreading "teacher computer literacy."(iii) But what follows instead is a chaotic, rambling narrative about the application of computers in education which fails to address any of the major issues, and certainly fails to promulgate computer literacy among teachers.

Underlying Poirot's argument appears to be an assumption that, if a "better" technology is available, it should be applied, preferably without resistance from non-computer literate teachers. Poirot hopes to overcome their opposition by showing how computers might be used, but in the process he almost entirely overlooks the reasons for their application. As a result, Poirot has written a book which fails to teach the very audience to which it is addressed.

The educational institutions in our society are in a clear state of crisis, as evidenced by their growing failure to instill even basic literacy

among their students. Poirot's book parallels that crisis; it is symptomatic that, instead of attempting to educate his readers by raising the broader issues of computerization and its relationship to education, he chooses to close his brief book by presenting two sample designs for a microcomputer laboratory. By concentrating on form, **COMPUTERS AND EDUCATION** completely overlooks the content of education in general, and computerized education in particular.

COMPUTERS AND PROGRAMMING GUIDE

For Scientists and Engineers- 2nd Ed

Donald D. Spencer

Level: Novice Rating: 65

	Sams 1981 Paper		
463 Pages	ISBN: 0-672-21693-0	5" x 9"	\$15.95

As Spencer notes in the opening pages of his book, science and engineering have been closely associated with the development of computer technology in the last two decades, since they are the two prime areas in which computers have been most readily applied. If anything, Spencer understated his case—the use of computers has by now become virtually universal in the world of science and engineering.

In view of this, it is paradoxical that the contents of **COMPUTERS AND PROGRAMMING GUIDE** are extremely general and, aside from an introductory chapter devoted to applications and two later chapters devoted to problem-oriented programming languages and simulation languages, not specifically relevant to the needs of scientists and engineers. The book focuses instead on conveying to the reader the rudiments of "computer literacy," and is evidently intended as an introductory textbook to a course in computer science for undergraduates in the sciences and engineering.

Its clear, introductory textbook character as well as its concern with computers in general rather than microcomputers in particular, make **COMPUTERS AND PROGRAMMING GUIDE** of little interest to the microcomputer user, or to anyone who has already gotten his or her feet wet by gaining some familiarity with computers and programming.

COMPUTERS FOR EVERYBODY

Jerry Willis and Merl Miller

Level: Novice Rating: **90**

Dilithium 1983 Paper
262 Pages ISBN: 0-88056-094-0 6" x 8" \$5.95

This book was written for the reader with no technical background, who nevertheless suspects that he or she might find a microcomputer useful but lacks the knowledge to either select or use one. The book is therefore directed to the reader who either plans to use a computer exclusively as a household appliance (using only commercially available software) or at most intends to learn enough BASIC to write a few simple programs. Such users will become more and more common as microcomputers find growing acceptance in the home.

Given the book's intended audience, its treatment of microcomputer selection and application is excellent. COMPUTERS FOR EVERYBODY is well written, and avoids the obvious condescension of many similar works written for newcomers to microcomputers. Since the authors admit that their treatment is a necessarily simplified one, they provide suggestions for further reading for those who find that a particular topic engages their interest. The criteria for purchasing a particular microcomputer system are clearly spelled out at various points in the text; one of the final chapters surveys (and assesses) many of the more popular computers currently on the market. COMPUTERS FOR EVERYBODY also details a broad range of business, professional, educational and home applications for microcomputers, and even notes some of the software available in each of these four areas.

While this very elementary introduction to computer selection is inappropriate for any user who has at least a passing familiarity with computers, we do recommend it to those whose fear or lack of understanding of computers is mitigated by a desire to take advantage of the unique opportunities which microcomputers afford.

COMPUTERS FOR PEOPLE

Jerry Willis and Merl Miller

Level: Novice Rating: **60**

Dilithium 1982 Paper
200 Pages ISBN: 0-918398-64-9 5" x 8" \$7.95

COMPUTERS FOR PEOPLE is largely a rewritten version of the authors' COMPUTERS FOR EVERYBODY, with one major difference in emphasis. Evidently fearing that the plethora of appliance computers

now on the market will create confusion in the minds of unsophisticated readers and paralyze them in their selection of a real microcomputer system, Willis and Miller have decided to offer the reader a book whose major goal is product promotion for Atari 400 and Atari 800 computers. As the authors explain it, "throughout this book you will notice a general emphasis on Atari products. We wanted to write a good general introduction to small computers and feel that this can best be done by using one computer as an example all the way through the book. The Atari computer is an excellent choice for two reasons: friendliness and graphics. Between us we have had experience with most of the small computers available today. We can say without hesitation that, for most applications, the Atari computers are the best computers available for the price."(5) By the time the reader reaches the six-page table on "the current crop" (which lists, besides Atari computers, Texas Instruments 99/4A, TRS-80 Color and Model III, Commodore PET 4000 and VIC 20, and Apple II Plus computers), he has been so bombarded with lavish praise for Atari computers and profuse illustrations of Atari products that this section appears to be superfluous, as do the computer selection cards included in the appendices.

Any work on computer selection should at a minimum outline the criteria for selecting a microcomputer, and if possible relate these to real microcomputers. Willis and Miller have completely abrogated this responsibility in *COMPUTERS FOR PEOPLE* by focusing on the presumed virtues of a single manufacturer's microcomputers, to the exclusion of all others.

A CONSUMER'S GUIDE TO PERSONAL COMPUTING & MICROCOMPUTERS 2nd Edition

Stephen Freiberger and Paul Chew, Jr.

Level: All Rating: **65**

198 Pages	Hayden	1980	Paper	\$9.95
	ISBN: 0-81045-116-6		7" x 10"	

While the main thrust of this consumer's guide is on informing the reader about the various kinds of microcomputers currently (1980) on the market, a brief introduction is included to provide the inexperienced reader with a knowledge of the fundamental principles of microcomputers. This introduction, however, has been added largely as window dressing to supplement the main portion of the book, the listing of microcomputers; it is both too short (53 pages) and too vague to provide

the potential microcomputer buyer with very much information on microcomputers, and completely fails to specify criteria which the reader might use in selecting a system.

The CONSUMER'S GUIDE itself provides a select listing of microcomputers in five areas: full microcomputer systems (such as the Apple II); limited input-output systems and systems on a card (i.e., generally limited systems which serve as learning tools for the hobbyist, such as the Heathkit H8); microcomputers in a cabinet (such as Cromemco's Z-2 computer system, which is the central component of a system which the user is to assemble); and microcomputers on a card and CPU cards. Within any given category, all microcomputers are grouped together on the basis of the kind of microprocessor which they use. The actual discussion of microprocessors (pp. 180-185) and the "table of microprocessor types" (pp.186-190) do not provide enough information on the differences between microprocessors to make this classification useful or meaningful to the novice. Finally, the 1980 copyright date for the second edition indicates that this is hardly an overview of state-of-the-art microcomputer technology; most conspicuous in their absence are systems based on 16-bit microprocessors, as well as important additions to the microcomputer market which have appeared since this book went to press (the IBM Personal Computer, the Timex/Sinclair ZX-81).

If this work is compared to some other consumer guides to microcomputers, such as WEBSTER'S MICROCOMPUTER BUYER'S GUIDE, its shortcomings become even more apparent. Aside from the failure to survey independently developed software and available peripherals, the guide has evidently been compiled from information released by microcomputer manufacturers, with little or no attempt to be critical; to cite but one example, the fact that some microcomputer systems have membrane keyboards is never indicated in the text. More generally, because of the broad range of microcomputers which it surveys, this guide is of some value to the specialized hobbyist. The potential buyer who wishes to apply a microcomputer system to specific needs, however, would do well to rely on another, more informative reference work.

COURSEWARE IN THE CLASSROOM

Selecting, Organizing, and Using Educational Software

Ann Lathrop and Bobby Goodson

Level: Novice Rating: **65**

Addison-Wesley	1983	Paper	
187 Pages	ISBN: 0-201-20007-4	6" x 9"	\$10.95

On one hand, the use of computers has created the possibility of revolutionizing education and transforming the way people learn. On

the other, many educators, by divorcing the process of learning from the social context within which it occurs, adopt a purely technocratic approach to the application of computers, and simply opt for the replacement of traditional techniques by more up-to-date ones. This disparity between potential and reality is clearly apparent in *COURSEWARE IN THE CLASSROOM*.

The authors' prime motivation in the first half of the book—which provides a very cursory examination of computer applications in education and specifies criteria for selecting software—is in many ways a negative one: computers must not be allowed to join audio-visual aids (which were viewed as similarly revolutionary in the early 1960s) in the closet, where they will be unused and ignored. Their proposed method for avoiding this focuses in part on adopting the administrative measures necessary to integrate computerized education into the classroom: an entire section, for example, is devoted to organizing the courseware library, a topic which calls to mind James Poirot's designs for a model classroom. They advocate selecting the "right" software, and while the operational criteria they specify are unusually clear, these standards never rise above the mundane and obvious. In their system, computerized education seems the same as traditional education—except that it makes use of new equipment.

The second portion of the book is a "courseware directory," if such a glorified name can be applied to so uninformative a software listing. Having emphasized the need to evaluate carefully all courseware in the first half of the book, the authors provide a selective listing of 118 packages, which is completely uncritical in its focus and even fails to mention the minimum system requirements necessary for the proper running of these programs.

COURSEWARE IN THE CLASSROOM does specify some highly practical, frequently obvious criteria for selecting software. It also provides a minimal listing of courseware for those willing to do further research. It is not, however, a title we recommend.

CRASH COURSE IN MICROCOMPUTERS

Louis E. Frenzel, Jr.

Level: Novice Rating: **95**

Sams 1980 Paper
 254 Pages ISBN: 0-672-21634-5 9" x 11" \$19.95

The author revives programmed instruction techniques once popular in IBM/SRA introductory manuals for the purpose of imparting a broad spectrum of microcomputer knowledge (programmed instruction involves short, information-filled paragraphs interspersed by questions

which emphasize key word and concept retention). Though far from original, the presentation is effective and fast-paced, living up to the reader's "crash course" expectations. This book is spiral bound for easy opening, and contains colorful, accessible graphics: the microprocessor diagrams rate as didactic works of art.

Frenzel purposely expands his discussion past Central Processing Unit concepts and hardware considerations to encompass software, programming and peripherals. The detailed microprocessor chapters are consequently paralleled by equally detailed input/output interface chapters. Such topics are treated only after the patient, careful development of introductory material. The programmed instruction technique lends itself well to this "layered" approach to individual topics: this is one of the few introductory manuals with enough surface clarity for beginners and enough real depth (an 8080/8085 instruction set appendix) for intermediate microcomputer users.

CRASH COURSE IN MICROCOMPUTERS is a superior introduction to personal computer use. The novice will find himself retaining information almost against his will, and the advanced beginner will discover such hidden treasures as a classic description of Mosfet architecture and gate logic. Frenzel's book is a staple of any microcomputer library or specialized book section.

THE DEVIL'S DP DICTIONARY

Stan Kelly-Bootle

Level: All Rating: **90**

McGraw-Hill 1981 Paper
 141 Pages ISBN: 0-07-034022-6 6" x 8" \$8.50

Ambrose Bierce's DEVIL'S DICTIONARY was a compendium of biting, ironic definitions, generally political in nature, published at the turn of the century. DEVIL'S DP DICTIONARY is a funny, literate updating of the Bierce classic, filled with an insider's "embedded" stabs at data processing as a way of life.

Stan Kelly-Bootle's career in data processing spans the mercurial borderlines of mainframe, mini, and micro; so do his puns. A "random file" is a place where things can get lost in any sequence. The succinct definition for "endless loops" reads "see loop, endless." "SNA" becomes an acronym for "Scapegoat Network Architecture."

To repeat more of Stan Kelly-Bootle's definitions would be to rob the reader of the dozens of chuckles his book elicits (our favorite was "Extended BASIC"). Anyone who has ever "interfaced" (another good one)

with computers, or been subjected to the absurdities of data processing jargon will relish the rare wit of THE DEVIL'S DP DICTIONARY.

DON'T! Or How to Care For Your Computer

Rodnay Zaks

Level: Novice Rating: **85**

	Sybex	1981	Paper
218 Pages	ISBN: 0-89588-065-2	6" x 9"	\$11.95

Although microcomputers, because they contain few components and very few moving parts, are more durable and indestructible than either older generations of computers or larger mainframes, Zaks argues that their rate of failure is about the same. The explanation for this, he believes, is that the operators on more expensive systems were far more highly trained in the use of their computers, while the rapid dissemination of microcomputers has placed them in the hands of people with little or no training in their use. The problem is magnified, moreover, by the failure of the small user to keep a log of computer operations, so that the correction of problems resulting from the "time bomb" effect is magnified by the difficulty in locating the sources of the failure. Zaks attempts to rectify this problem by intensively surveying the do's and don't's of microcomputer care.

Most introductory works on microcomputers, of course, treat this topic to some degree, although their suggestions are generally obvious, to say the least; the reader is often warned, for example, that it is inadvisable to pour a cup of coffee into a diskette drive or to use an awl to engrave floppy disks. While Zaks' book includes these more elementary warnings, he also outlines the basic rules for the care of each major component of a microcomputer system (floppy disks and disk drives, hard disks, cassettes and cassette drives, printers, video monitors and keyboards, and the CPU itself). In addition, the book includes chapters on caring for software and, for multi-user systems or systems in a business or professional environment, on security measures aimed at protecting the integrity of the system and its data. Finally, Zaks presents an informative chapter on the layout and design of the computer room with an eye to making it both as comfortable as possible for humans and as safe as possible for the computer.

Advances in microcomputer product design and performance have made portions of Zaks' book obsolete. Most dated is the point of departure for his argument—the assertion that the rate of failure among microcomputers is comparable to that of larger systems. Despite this,

DON'T remains a valuable collection of sensible advice and practical suggestions for microcomputer care.

THE ELECTRONIC COTTAGE

Joseph Deken

Level: All Rating: **80**

Bantam 1981 Paper
386 Pages ISBN: 0-553-22863-3 5" x 7" \$3.95

Microcomputer technology and its subsequent sociological shock-waves form the broad theme of Joseph Deken's answer to Alvin Toffler's *THIRD WAVE* (the source of Deken's title). Unlike Toffler's book, however, the rambling text of *THE ELECTRONIC COTTAGE* does not focus on the development of one central thesis. Instead, it alights from topic to topic, frenetically pollinating the possibilities of one area of microcomputer research with possible future developments in another.

This diffuse discourse on pop-sociology and hypothetical technology is tempered by an original, informative presentation of microcomputer fundamentals. Deken is at his best when he uses his descriptive talents to unfold the "yes/no" world of personal computers. Specifically, his novel use of analogies in presenting complex technological subjects is one of the book's highlights. An egg carton filled with mobile eggs becomes a silicon lattice doped with impurities to insure electron movement among individual wells. A transistor is explained in terms of the mutually exclusive use of adjoining thin-walled classrooms by a classical orchestra and rock group. This level of imaginative, effective analogy extends from the construction of simple logic gates to complex microcomputer architecture.

Other topics covered by Deken include elementary statistical modeling, futuristic variations on personal computer games, biofeedback applications and a hardware overview. The introduction to the manipulation of microcomputer data through modeling, using Markov chains and recurrence relational models, is particularly well executed.

Though frequently preachy and self-absorbed in his treatment of specific topics, the author of *THE ELECTRONIC COTTAGE* has nevertheless fashioned an eclectic, diverting narrative on the pervasive effects, present and future, of microcomputers on "everyday" American life.

THE ESSENTIAL COMPUTER DICTIONARY AND SPELLER

For Secretaries, Managers, and Office Personnel

Charles J. Sippl

Level: All Rating: **90**

Prentice-Hall	1980	Paper	
258 Pages	ISBN: 0-13-284356-0	6" x 9"	\$6.95

Charles J. Sippl, author of THE MICROCOMPUTER DICTIONARY and co-author of the COMPUTER DICTIONARY (see our review), has produced another computer dictionary which aims at furthering "computer literacy" among persons in the business world. The 15,000 computer terms and acronyms which the dictionary contains, combined with its low price, make it an outstanding reference work for anyone interested in computer terminology.

This dictionary does not focus on definitions which are of special relevance to the microcomputer user, but rather provides definitions of computer terms in general, whether for larger mainframe systems or for microcomputers. The large number of entries included in the book, however, means that each definition must necessarily be concise and to the point, in contrast to the definitions contained in the two larger dictionaries assembled by Sippl. For example, a bus, among other things, is "a path over which information is transferred,"(30) an entry which provides the user with minimal information. For the computer user who needs a high-quality dictionary of technical computer terms, a more comprehensive dictionary is definitely necessary. But for those who do not require such a detailed and technically precise treatment of terms, THE ESSENTIAL COMPUTER DICTIONARY AND SPELLER admirably fulfills its intention of broadening the reader's familiarity with computer terminology and concepts.

THE FIRST BOOK OF MICROCOMPUTERS

The Home Computer Owner's Best Friend

Robert Moody

Level: Novice Rating: **90**

Hayden	1978	Paper	
139 Pages	ISBN: 0-8104-5121-2	6" x 9"	\$6.50

This is THE introductory book for the computer-traumatized, cyberphobic individual, who, for some fight-or-flight reason, must now learn about "them."

Moody's delivery is sugar-coated with cartoons (an elephant's head is main memory), with dialogues with an imaginary friend/teacher computer, and with a style so conversational that it prompts the reader to vocalize answers (know what I mean?).

In the course of winding through this Sesame Street menagerie of microcomputer topics, even the most unreceptive of readers will find Moody's original presentation effective. A "brief, brief history" of data processing is followed by a conceptual introduction to number systems and programming. A computer's components are then described, and some possible personal computer applications discussed. For those readers who manage to startle themselves by wanting additional information about personal computers, *THE FIRST BOOK OF MICROCOMPUTERS* provides ample appendices listing magazines, computer clubs and microcomputer retail stores.

We recommend Robert Moody's book as an entertaining, popularized treatment of microcomputer fundamentals.

FOUNDATIONS OF COMPUTER TECHNOLOGY

Joseph C. Giarratano

Level: Novice Rating: **70**

Sams	1982	Paper	
238 Pages	ISBN: 0-67221-814-3	9" x 11"	\$22.50

This is the first of a four-volume work by the author on the hardware and software concepts of microcomputers. The first two volumes examine the history of computers and the state of current microcomputer technology, while the final two volumes treat introductory and advanced programming in BASIC. Giarratano does not specify exactly for whom he is writing this first volume, or what prior knowledge of microcomputer technology he expects the reader to have. Nevertheless, this would appear to be a volume primarily for beginners.

The three major sections of this book can be regarded as virtually separate monographs. The first two chapters deal with the development of calculating equipment from the abacus through the present-day computer. The material in these chapters is presented in a straight-forward narrative style and provides a wealth of detail. The next three chapters examine the basic components of a microcomputer system: the central processing unit, memory, and input-output devices; in addition, Giarratano discusses programming and programming languages (machine code, assembly language, and higher-level languages). These chapters are well-organized, clearly and lucidly written, and informative, even if

they are not especially original in content. The chapter on computer terminals, moreover, is of practical value in outlining what kinds of CRT's or printers are currently available.

Giarratano's attempt to supply a beginning reader with the fundamentals of microcomputer history and technology, however, completely collapses in the final chapter, in which he discusses the manufacture of semiconductors and integrated circuits. This is a chapter clearly intended for the advanced reader, since the author makes absolutely no attempt to phrase his discussion of electronics in terms even vaguely comprehensible to the beginner.

The primarily textbook treatment which Giarratano offers, coupled with the book's high cost and its diffuse content, make this first volume of marginal interest to all but the most avid readers of microcomputer introductions.

GUIDE TO SYSTEMS APPLICATIONS

An Introduction to Microcomputers

John P. Grillo and J.D. Robertson

Level: Intermediate Rating: **95**

Wm. C. Brown 1981 Paper
 268 Pages ISBN: 0-697-09952-0 9" x 11" \$17.95

A quick glance at the first few chapters of GUIDE TO SYSTEMS APPLICATIONS reveals some incredibly amateurish photographs—programmable calculators, for example, are shown either with a flower-patterned bedspread as a backdrop, or “modeled” by a justifiably embarrassed 12-year old girl. No first impression could be more at odds with the true content of this book: it is a very professional introduction to systems application design for the intermediate BASIC programmer.

The introductory chapters on microcomputer components, central processing unit architecture and the nature of software, are uniformly excellent. One section includes an unusually coherent explanation of machine cycle times—a benchmark frequently used and frequently misunderstood when comparing the speeds of different microprocessors.

This is definitely not a BASIC primer, though features of the language are briefly discussed. The intermediate BASIC programmer for whom the book is primarily intended will find informative material in a chapter which treats the advanced functions of extended BASIC.

Only after this preliminary groundwork has been laid does the book begin to assume its uniquely system-oriented character. The requirements for a functional computer system are discussed. These include an analysis of possible options ranging from which dedicated printer to buy,

to whether to develop software in-house or to opt for a vendor's software package. This deliberate attention to planning and design carries over into the systems programming chapters which follow. The efficient use of advanced file manipulation techniques begins with flowcharting; as the coding techniques increase in complexity (linear, exchange and Shell-Metzer sorting), the reader/programmer begins to appreciate this design detail. If the book has a single programming theme, it is the utility of the Binary Sequence Search Tree (BSST) technique for structuring disk file data: three applications programs demonstrate its use in manipulating physical data. Indeed, one facet of the BSST technique is the distinction it makes between logical and physical structuring, a concept drawn from a systems approach hardly mentioned by the authors called data base management.

Bibliographies at the end of each chapter are among the best (and most eclectic) in the literature. Dozens of magazine articles and books are referenced as sources of additional information on each topic.

GUIDE TO SYSTEMS APPLICATIONS is a unique, professional introduction to microcomputer systems analysis and design for the experienced programmer. We highly recommend this book.

GUIDEBOOK TO SMALL COMPUTERS

William Barden, Jr.

Level: Novice Rating: **60**

Sams 1980 Paper
 127 Pages ISBN: 0-672-21698-1 5" x 8" \$6.95

The first two glaring omissions in this 1980 survey of microcomputer hardware are the IBM Personal Computer and the Timex/Sinclair 1000—two units which, by 1982, were selling faster than any of Mr. Barden's entries. The remaining treatment of the Apple II, Atari, TRS-80, et al., is reasonably good: standardized charts depict every feature from microprocessors to warranties, and the inclusion of schematics of each system's key pad is a nice comparative touch. Mr. Barden's guidebook is definitely not a critique of what the marketplace offers—his most scathing verbiage is reserved for the "scanty" Commodore PET documentation.

So, if you're confused about what personal computer to buy—but know it's definitely not an IBM Personal Computer or a ZX-81—and would like a listing of microcomputer systems and features with no ancillary analysis or comparisons, this latest William Barden effort is definitely the book for you.

HOME COMPUTERS

A Beginner's Glossary & Guide

Merl K. Miller and Charles J. Sippl

Level: Novice Rating: **55**

Dilithium 1978 Paper
 147 Pages ISBN: 0-918398-02-9 6" x 9" \$10.95

Charles J. Sippl, the perennial author or co-author of microcomputer dictionaries (see our reviews of THE COMPUTER DICTIONARY AND HANDBOOK and THE ESSENTIAL COMPUTER DICTIONARY AND SPELLER) has collaborated with Merl K. Miller (co-author of COMPUTERS FOR PEOPLE and COMPUTERS FOR EVERYONE) to produce yet another microcomputer dictionary, in this instance one which focuses on the needs of beginning microcomputer users. This particular work features a very brief, 81-page dictionary accompanied by an introduction to microcomputers and a short chapter on gate logic and electronic symbols.

The introduction attempts to survey the range of microcomputer-related topics (microprocessor technology, applications, computer memory, numbering systems) in a very unsatisfactory way. Clearly torn between the need to introduce a fairly complicated technical discussion of their topic and the need to provide an altogether unsophisticated beginner with the fundamental principles of microcomputers, the authors have failed to find a happy medium for organizing and presenting their discussion. The dictionary itself, "the most important part of this reference," (vii) provides highly condensed definitions of a relatively limited computer vocabulary.

Underlying the book's weaknesses is some confusion on the part of the authors as to how to deal with the "beginning" microcomputer user. This reference work attempts to address the interests of a broad spectrum of beginners, from those electronics hobbyists who are most interested in learning how the computer works to those who are most interested in finding out how to use it effectively as a tool for specific kinds of applications. Within the the confines of a very short book, the authors have attempted to include material which might appeal to these diverse groups; the result is, paradoxically, a reference work which is likely to be unsatisfactory to virtually all categories of beginners, and one which is far inferior to Sippl's other efforts.

HOME COMPUTERS VOL 1: HARDWARE

1024 Questions & Answers

Rich Didday

Level: Novice Rating: **95**

Dilithium 1977 Paper
225 Pages ISBN: 0-918398-00-2 5" x 8" \$11.95

HOME COMPUTERS VOL 2: SOFTWARE

1024 Questions & Answers

Rich Didday

Level: Novice Rating: **85**

Dilithium 1977 Paper
175 Pages ISBN: 0-918398-01-0 5" x 8" \$11.95

In both approach and quality of presentation, this is the MY DINNER WITH ANDRE of microcomputer introductions: both volumes are written in an interview format, with all the questions of a "bright interested newcomer" effortlessly answered in fascinating, conversational prose by a microcomputer expert (the copyright page contains a disclaimer that some sections of the book are based upon actual conversations). What results is one of the more original, comprehensive personal computer introductions currently in print.

While the first volume concentrates on hardware and the second on software, they contain common material on number systems and microprocessors. Appendices are duplicated, and volume two begins with a summary of relevant information from volume one. We would advise the novice not to treat the software text as a separate introduction, however, since its reliance on topics developed in the hardware volume far exceeds this brief summary.

VOLUME 1: HARDWARE introduces the reader to elementary gate logic by presenting a special "Prognosticator" circuit which predicts the viability of political candidates ("has a lot of money" is the most important single factor). Though far-fetched, the analogy works. The simple circuit discussion evolves into descriptions of each component of a microcomputer. The relation between the original circuit and the actual hardware is explained in a chapter on digital logic notation. A pleasant surprise in this first volume is a section on the types of diagrams the novice may confront in his exploration of microcomputer topics: functional block diagrams, circuit diagrams, timing diagrams, and even etching and drilling guides are covered in this one-of-a-kind section. Volume 1 concludes with inquiries into assembling microcomputer kits (specific kit references underscore the book's 1977 copyright) and microprocessor architecture.

VOLUME 2: SOFTWARE carries the microprocessor discussion to its natural sequel—machine language programming. The actual chapter titles of the machine, assembly, and BASIC programming sections begin with “what’s it like to program in ...” Although the implied goal of the chapters is not to impart programming expertise but to solidly ground the reader in programming fundamentals, discussions are quite comprehensive. If any criticism can be made of this software volume, it is that it rakes the leaves of introductory programming subjects with a bulldozer of overwhelming detail.

Both volumes effectively use analogies to convey their points (as is characteristic of normal conversation). The car/computer analogy is exploited in the hardware volume, while a cartographer’s map charts the relative “locations” of programming languages in the second volume. The questions themselves realistically digress from the normal development of a given topic (question number 422 is “I’m going to get some coffee—want some?”). Didday effectively uses the interview format to engage the reader’s attention in a manner that a more conventional treatment would preclude.

Rich Didday’s HOME COMPUTERS, despite some datedness and overzealous attention to detail, is a novel, lucid introduction to personal computing for the novice. We highly recommend the purchase of this two-volume set.

HOW TO BUY A PERSONAL COMPUTER

Carlton Shrum

Level: Novice Rating: **85**

	Alfred	1981	Paper	
64 Pages	ISBN: 0-88284-188-2		4" x 11"	\$2.95

This Alfred Handy Guide proved surprising in at least three respects: first, it is the cheapest microcomputer book reviewed in this survey; second, it is an updated fourth printing revised, so that it contains relatively current information about the changing personal computer marketplace; and third, it contains a 13-page comparative chart of the most popular systems and peripherals which alone is worth the price of the book. The introductory material is necessarily superficial, with only token treatments of the traditional “what is a computer” and “what features are important to you” topics. But given these limitations, HOW TO BUY A PERSONAL COMPUTER certainly presents one of the best buys in the introductory category.

HOW TO BUY A PERSONAL COMPUTER (WITHOUT ANXIETY)

Jonathan D. Lieff, M.D.

Level: Novice Rating: **50**

	Ballinger	1982	Paper	
113 Pages	ISBN: 0-88410-743-4		6" x 10"	\$9.95

Lieff is a psychiatrist who, because he could not find a work on computer selection which explained the topic in terms comprehensible to a layman, decided to write one himself; in addition, he also tries to confront "the emotional barriers many of us have about computers and suggest ways to overcome them." (3) The book is indeed written in terms comprehensible to a layman, and includes brief discussions of the history of computers, microcomputer basics, possible applications for computers in a wide range of fields (accounting, business, education, games, law, music, etc.), criteria for selecting a computer, and some of the major computers now on the market.

HOW TO BUY A PERSONAL COMPUTER (WITHOUT ANXIETY) is hardly the first book to appear which is written for the layman, nor is it the best. The major slant which justifies it—the fact that the author is a psychiatrist who presumably understands the anxiety which people have when dealing with computers—is disappointing, since the author's treatment of the problem does not rise above the level of simple platitudes. Many of the criteria involved in selecting a computer are, however, clearly specified, although very often briefly or only in passing; the book is simply too short to devote very much time to detailed explanations. While it may have been true that there were no intelligible works on selecting computers when the author was searching for one, this is no longer the case; even for readers who do not want to get overly involved with microprocessor basics—the audience to which this book is directed—far superior treatments are now available.

HOW TO BUY AND USE MINICOMPUTERS & MICROCOMPUTERS

William Barden, Jr.

Level: Advanced Rating: **45**

	Sams	1977	Paper	
240 Pages	ISBN: 0-672-21351-6		7" x 11"	\$10.95

Masquerading in the sheep's clothing of a general introduction to minicomputers and microcomputers is a wolf (or dog?) of an advanced

introduction to minicomputer architecture, software and programming. (The author states on page 18 that the term 'minicomputer' is used to apply to both micro's and mini's—a mercifully early tipoff to the game about to be played in the next 229 pages.) Not even a token effort was made to make the charts, benchmarks, and electrical diagrams remotely intelligible to the reader who really wants to know what the title promises. The 1977 copyright indicates either a presumption to timelessness in a chaotic, fast-moving industry, or a one-way ticket to the remainder bins of bookstores across the country.

HOW TO BUY AND USE MINICOMPUTERS AND MICROCOMPUTERS is, contrary to its title, a book only tangentially related to microcomputers; our rating distinguishes it as one of the unmistakable black sheep of current personal computer literature.

HOW TO BUY THE RIGHT SMALL BUSINESS COMPUTER SYSTEM

C. Roger Smolin

Level: Novice Rating: **50**

Wiley 1981 Paper
 156 Pages ISBN: 0-471-08494-8 7" x 10" \$8.95

The book is intended for the businessperson who has heard that there is a computer revolution going on, knows nothing about it, but hopes computerization may be just the thing to make his or her business more successful. The book is written in non-technical language which can indeed be understood by someone completely unfamiliar with computers and data processing. It attempts to "outline how a computer works and how to shop for the equipment and programs you'll need for small business systems," as well as to show how small business systems "should operate, what you should expect, and what you'll have to put into them to obtain something of value." (viii)

Smolin's treatment of all of these topics, however, is so superficial that it fails to provide satisfactory information about most of the questions which he addresses. This applies especially to the most basic question of all—whether or not to buy a small business computer in the first place; the criteria outlined are so vague, and Smolin's warnings of possible pitfalls so numerous, that the book is likely only to increase the potential buyer's sense of apprehension. The same applies to the selection of hardware; aside from providing criteria for choosing between dot matrix and impact printers, the book leaves the question of computer hardware at an abstract level, and provides very little useful information to help the small businessman determine what kinds of equipment are best suited to his or her needs.

The discussion of software is somewhat better, and does at least provide a list of minimum features which software packages in accounts receivable, order entry, accounts payable, etc., should possess. But even this high point does not permit the book to rise above an unmistakable level of mediocrity.

HOW TO CHOOSE YOUR SMALL BUSINESS COMPUTER

Mark Birnbaum and John Sickman

Level: Novice Rating: **30**

Addison-Wesley 1983 Paper
 150 Pages ISBN: 0-201-10187-4 7" x 9" \$9.95

We all know that peculiar species of world traveler whose ability to communicate in a foreign language is limited only by his ability to shout in English. Unfortunately, we do not all know his computer-book equivalent: the author whose ability to clearly relate complex ideas is limited only by his ability to truncate sentences and strip language of all words remotely polysyllabic. Sentences stay short. Words stay simple. See John's computer run.

What is even more ironic about the use of this method in this particular book is that HOW TO CHOOSE YOUR SMALL BUSINESS COMPUTER presumes to span both the micro- and the mini-markets. Certainly people in managerial positions who are making computer-related decisions of this magnitude deserve a non-intimidating but reasonably sophisticated presentation. "Computers are like small children," write Birnbaum and Sickman. And only a businessperson with a lemonade-stand mentality would seriously consider buying this book, we add.

HOW TO PROFIT FROM YOUR PERSONAL COMPUTER

Professional, Business, and Home Applications

T.G. Lewis

Level: Novice Rating: **30**

Hayden 1978 Paper
 192 Pages ISBN: 0-8104-5761-X 6" x 8" \$11.75

Given the widespread misunderstanding of computers and the love-hate relationship which many people have with them, HOW TO PROFIT

FROM YOUR PERSONAL COMPUTER offers the reader the hope of transforming a distant and vaguely menacing revolution into a means of making a profit.

So much for the imagery of titles. What the book is really about is how microcomputers might be applied and programmed to solve seven problems arising in a small (or even large) business environment. Whereas most authors try to provide a discussion of microcomputer basics to show the reader that, however complex computer technology may be, its essential features are at least comprehensible to a person of average intelligence, Lewis prefers to deal with this problem by debunking myths ("computers...are complicated machines built by geniuses, who are the only people able to understand them"). Whenever he does blunder into technical discussions, the results are clearly unsatisfactory; for example, his discussion of binary digits (the base two numbering system) could have confused a mathematician. But computer basics are not what Lewis' book is about. After introducing the reader to Tom Swift (THE Tom Swift), Dr. Goode and a host of familiar stereotypes, Lewis examines their business/personal problems and shows how to solve them using sound programming fundamentals and the BASIC language. This approach is incomprehensible to anyone without experience in programming and lacking a knowledge of BASIC, but it is superfluous as well to those who have been exposed to either or both; and Lewis makes no systematic attempt to supplement it by teaching the beginning reader the fundamentals of either programming or BASIC. Lewis would have been better off to eliminate his text and present the reader with a half-dozen or so references to other works.

In short, this book is badly mistitled; it really should be called HOW TO PROFIT FROM WRITING ABOUT PERSONAL COMPUTERS BY INCLUDING THE WORD PROFIT IN THE TITLE, since it is clear that the only person who might profit from reading Lewis' book is Lewis himself.

INTERNATIONAL MICROCOMPUTER DICTIONARY

Level: All Rating: **40**

Sybex 1981 Paper
118 Pages ISBN: 0-89588-067-9 4" x 6" \$3.95

Besides providing a "convenient reference" for those interested in microcomputers, this dictionary provides the Danish, Dutch, French, German, Magyar, Italian, Norwegian, Polish, Spanish, and Swedish equivalents of 171 microcomputer terms.

In theory, such a dictionary is a sound idea; for many computer users, the need to translate highly technical terms from English to an-

other language severely strains linguistic capabilities. But numerous features of this dictionary make it singularly unattractive for this purpose. It is not clear, first of all, why a small dictionary (which occupies the first 65 pages of the book) is more "convenient" than a larger one, which contains both more words and fuller definitions of terms. But the dictionary's main attraction and, presumably, its *raison d'être*—the fact that it provides translations of words in ten European languages—has also been poorly conceived and executed. There is no indication of the gender of foreign words, which means that this dictionary must be supplemented by at least a general foreign language dictionary, and in some cases a technical dictionary as well. Nor is it clear what criteria have been used in selecting these ten languages. Their unifying feature, however, is that they all use Roman script, which indicates that reduced printing expenses may have been instrumental in their selection.

In a nutshell, this book has been shoddily and cheaply assembled in order to capitalize on the rapid dissemination of computer technology. Even its low price does not compensate for the combination of a mediocre computer dictionary and an inferior lexicon of foreign computer terms.

INTRODUCTION TO MICROCOMPUTERS FOR THE HAM SHACK

Harry L. Helms, Jr.

Level: Novice Rating: **65**

	Sams	1979	Paper	
95 Pages	ISBN: 0-672-21681-7		5" x 9"	\$5.95

This eighty-four page monograph describes the communication capabilities of the microcomputer, specifically in the amateur radio environment (i.e., "shack").

Other than general introductory material and some individual topics which might prove of passing interest to the non-ham public—like the use of ROM to convert digital signals into Morse Code—its focus (wavelength?) progressively narrows to the world of the amateur radio enthusiast.

Microprocessor-based Morse Code receiving systems, slow-scan television reception and remote control systems are discussed. The author additionally argues the merits of the FCC's adoption of a packet radio network patterned after a recent Canadian experiment (packet radio offers the advantage of interference-free communications).

INTRODUCTION TO MICROCOMPUTERS FOR THE HAM SHACK offers little to the microcomputer user not already interested in amateur

radios; even for its admittedly limited ham audience, however, Helms' irritating brevity makes for little more than a marginal distraction.

AN INTRODUCTION TO MICROCOMPUTERS-VOL.0-BEGINNERS BOOK 3rd Edition

Adam Osborn and David Bunnell

Level: Novice Rating: **85**

233 Pages	Osborne 1982	Paper	\$14.95
	ISBN: 0-931988-64-0	7" x 9"	

Though not necessarily enticed by the prospect of beginning a four volume opus on microcomputers with "Volume 0," the reader will find this Osborne/Bunnell introduction highly informative. The book's focus belies the primarily microprocessor content of the volumes which follow; few "beginner's books" treat Boolean operations, functional logic, and data register manipulation in the detail apparent in AN INTRODUCTION TO MICROCOMPUTERS, VOLUME 0.

The book is divided into two sections of three chapters each. The first section covers microcomputer components, software and computer selection. The second discusses elementary programming and microprocessor architecture.

The non-technical section is reminiscent of Adam Osborn's approach in the BUSINESS SYSTEM BUYER'S GUIDE (see our review). After a brief description of computer components and peripherals, a case study highlights some of the pitfalls of precipitous computer selection (programmer Susan Kilobyte plays a key role). A product survey of available hardware and software concludes this first section.

While no specific programming language is introduced in any detail in the second half of the book, binary arithmetic and its relation to circuit logic is comprehensively described. The authors explore the "information paths" of data within the microprocessor; the unsuspecting reader is given an excellent grounding in data register operation and flag interpretation—preliminaries of assembly language programming.

Diagrams highlight the interrelation of Central Processing Unit components to one another and to the external world. A series of seven diagrams is especially effective in depicting the precise sequence of data transfer along a bus; sections of the CPU are graphically highlighted in order to emphasize the importance of instruction timing during such operations.

The distinctly microprocessor flavor of these last chapters, both conceptually and in detailed hardware considerations, gives a taste of

the material discussed in the next volume in the series, **VOLUME 1-BASIC CONCEPTS**. We recommend **VOLUME 0-THE BEGINNER'S BOOK** not for its market survey and component overview, but for its effective introduction to microprocessor concepts.

THE INVESTOR'S COMPUTER HANDBOOK

Rod E. Packer

Level: Novice Rating: **80**

Hayden	1982	Paper	
168 Pages	ISBN: 0-8104-5203-0	6" x 11"	\$10.95

The "handbook" of this title is actually a misnomer: Packer's work is really an "idea" book for the personal computer/market enthusiast. Four skeleton programs written in BASIC are included in appendices to be fleshed-out on the reader's microcomputer (the self-descriptive titles are Fotofolio, Stockstory, Chartart, and Stock Swap). The treatment's true value, however, lies in the author's original musings about possible hardware/software paths to be pursued by investment applications (or, at least those not currently being pursued by Mr. Packer's own microcomputer investment software group).

The first two sections of the book ground the reader in personal computer fundamentals, while the third section is devoted entirely to various software application strategies. Case histories of varying approaches to similar investment analysis are discussed. Appendices include most state-of-the-art software packages currently available to the micro user. This book is a valuable tool less for the actual information and programs it contains than for its general guidance in formulating a customized microcomputer-assisted approach to investing.

KIDS AND COMPUTERS

The Parents' Microcomputer Handbook

Eugene Galanter

Level: Novice Rating: **95**

GD/Perigee	1982	Paper	
195 Pages	ISBN: 0-399-50749-3	6" x 8"	\$7.95

The Children's Computer School, founded in January 1981 in New York, teaches microcomputer fundamentals to children ages five to four-

teen. Eugene Galanter, the school's director, has written a parent and teacher's guide to personal computing. His book informs on two equally important levels: first, it imparts a working knowledge of microcomputer operation and programming to parents and teachers as a prelude to their own "enlightened co-learning" with children and students; second, it relates the provocative opinions and real concerns of an experienced educator about the problems of computer instruction on the elementary school level.

Early chapters cover microcomputer fundamentals: memories, peripherals and initial operation. These discussions are tempered by considerations specific to the author's audience, such as cumulative radiation dosage from monitors and television screens, ambient lighting and optimum phosphorous/background contrasts. A chapter entitled "Programming by, for and with Children" begins an introduction to BASIC programming. The treatment is straightforward. The text used is a variation of the lesson plan from the "adult" program in the author's school; it encompasses rudimentary BASIC commands and functions, and concludes with the composition of a few BASIC programs.

In a final chapter entitled "Evaluating Computer Education," Galanter provides additional information about his rather unique school, as well as some of the fine points distilled from his own teaching experience (these include the best teacher:pupil:microcomputer ratios). Of special note is his emphasis on the need for both "higher quality instructional programs" and the continuous reinforcement of the curricular development process.

KIDS AND COMPUTERS is a significant addition to the growing body of literature which treats the computer education of children. Parents and teachers will find the value of the concrete microcomputer knowledge it conveys surpassed only by the originality of its author's insights.

MICROCOMPUTER BUYER'S GUIDE

Computer Reference Guide

Tony Webster

Level: All Rating: **90**

	Byte	1983	Paper	
351 Pages	ISBN: 0-9594624-4-9		7" x 11"	\$19.95

Authors of computer-selection books frequently spend so much time outlining the criteria involved in choosing a computer that they

rarely, if ever, relate their discussion to "real" computers. If these books are successful, the reader's personal computer needs begin to focus, but never crystallize in the form of a specific machine.

MICROCOMPUTER BUYER'S GUIDE is the answer to this epidemic of frustration spreading across the ranks of readers of introductory microcomputer books. It is a valuable reference manual crammed with precise, detailed comparisons of most microcomputers. Webster's guide lists large and small manufacturers as well as software vendors who provide packages for individual models. Suggested retail prices of CPU's and peripherals are included, a rare find in current microcomputer literature.

The chapters which precede the actual encyclopedic treatment of the micro market, however, fail to rise to the standards set by the rest of Webster's book; the text is poorly, sometimes ungrammatically written, and provides a generally superficial overview of personal computers.

Nevertheless, as a pure reference guide, the MICROCOMPUTER BUYER'S GUIDE stands alone as a tool for both the novice who would rather "let his fingers do the walking," and the experienced user searching for the Sears-Roebuck catalog of microcomputer options.

MICROCOMPUTER DICTIONARY 2nd Edition

Charles J. Sippl

Level: All Rating: **95**

	Sams	1981	Paper		
535 Pages	ISBN: 0-672-21696-5		5" x 9"		\$15.95

See our review of COMPUTER DICTIONARY AND HANDBOOK on page 9.

MICROCOMPUTER PRIMER 2nd Edition

Mitchell Waite & Michael Pardee

Level: Intermediate Rating: **75**

	Sams	1981	Paper		
384 Pages	ISBN: 0-672-21653-1		6" x 8"		\$14.50

MICROCOMPUTER PRIMER was revised and reissued in order "to provide the beginner with sufficient understanding to confidently tackle whichever particular microcomputer he or she selects."(5) The questions asked ("What is a microcomputer?" "What do I do with a computer?") as

well as the illustrations provided in the introduction are fully consistent with this focus on the needs of the microcomputer novice. So too is the book's second chapter, which provides, among other things, an excellent discussion of microprocessor architecture. The final two chapters of the book, on programming and operating systems, are likewise written with the newcomer to microcomputers in mind.

Sandwiched between these initial and final chapters, however, is an extremely sophisticated technical discussion which is sure to leave even a fairly knowledgeable microcomputer user far behind. Beginning with their discussion of power supplies, Waite and Pardee make no attempt to provide an increasingly confused reader with essential background material for subsequent discussions of microprocessors, memories or input/output interfacing. The brief glossary which they have included in the introduction is of no help here, since most of the technical terms which they have failed to define in the course of their discussion do not appear in it.

While the book does show some evidence of extensive revision, such as its discussion of 16-bit microprocessors for example, at other times it bears the clear imprint of a work written during an era in which most microcomputer owners were hobbyists with one-board or limited input/output systems; this is most evident in the chapter on microcomputer memories, which devotes comparatively scant attention to recent advances in computer technologies. The discussion of input/output interfacing also shares this focus on a previous generation of microcomputer technology when the authors examine LED's and binary switches.

Finally, portions of the book suffer from a certain vagueness and approaches to microcomputers only in the abstract. This criticism does not apply to the authors' detailed and informative comparison of seventeen individual microprocessors (pp. 148-179); but aside from this, their discussion of microcomputer principles is never related to any specific microcomputer system. This is most clearly reflected in their discussion of assembly language programming, which, because it does not closely focus on the instruction set of a particular microprocessor, permits the authors to note only the most general and universally applicable operations in assembly language programming.

Both Waite and Pardee have established themselves as highly capable authors of works on microcomputers. We hope that the third edition of MICROCOMPUTER PRIMER will rise to a level of quality consistent with their previous work.

MICROCOMPUTERS AND THE 3 R's

A Guide For Teachers

Christine Doerr

Level: Novice Rating: **85**

Hayden 1979 Paper
177 Pages ISBN: 0-8104-5113-1 6" x 9" \$9.75

Doerr has curtailed her discussion of microcomputer basics to an absolute minimum (pp 4-9), barely raises the question of computer selection (pp 17-18), and has wisely decided to forgo the obligatory attempt to teach the reader BASIC, in order to devote her attention more fully to educators' use of microcomputers in secondary schools. The result is a coherent, carefully reasoned and argued work on the revolutionary potential of the computer for education, and on the wide range of applications which microcomputers can find in the classroom. Although at times the tone of the book is both defensive and polemical (indeed, one can sometimes detect a certain element of resignation toward "converting" her more traditional colleagues to the application of computers), the overall quality of the book is high, and Doerr's suggestions for computer applications in secondary schools are both thoughtful and exciting.

In the course of her book, Doerr outlines the ways in which computers might be applied in problem-solving, instructional simulation, games and computer-assisted instruction; these applications are relevant, moreover, not only to science and mathematics, but also to the humanities and the social sciences. In each of these cases—but also in the more general theoretical discussion of the use of the computer as an aid to teaching—Doerr includes extensive bibliographic information which enhances the value of this book. She also notes the existence of high-quality software at various points in the text, although she does not hesitate to recommend that teachers write their own programs, provided that this is done with a full understanding of the difficulties involved and the vast amounts of time required for the development of effective software. Finally, included among the useful materials in the book is a syllabus for a course in computers and BASIC.

MICROCOMPUTERS AND THE 3 R'S offers teachers an excellent introduction to computer literacy at the same time as it provides all readers with a glimpse at the possibility of using computers to make education more stimulating.

MICROCOMPUTERS FOR BUSINESS APPLICATIONS

William Barden, Jr.

Level: Novice Rating: **55**

Sams 1982 Paper
 256 Pages ISBN: 0-672-21583-7 6" x 8" \$9.95

The good news is that, in comparison with his HOW TO BUY AND USE MINICOMPUTERS AND MICROCOMPUTERS, Barden has shown substantial improvement as an author of books on microcomputers; the bad news is that still further improvement is necessary if Barden wishes to write a marketable work in the area of microcomputer selection. Part of the problem is once again the title—a book entitled MICROCOMPUTERS FOR BUSINESS APPLICATIONS might reasonably be expected to discuss ways in which a microcomputer can be used to best advantage in a small business setting. But Barden completely overlooks this topic, and instead attempts “to ‘round out’ the reader’s education in small business systems and discuss how you might analyze a system for your special application.”(194) Barden uses this aim of “educating” the reader as an excuse to avoid focusing on any particular aspect of computer selection and application, and instead moves from topic to topic in a completely superficial, unhelpful and uninformative way. Having covered microcomputer basics (in a far oversimplified way) and types of microcomputer systems on the market, Barden actually devotes three chapters to teaching the reader BASIC.

The only sound chapters in this book are those on a potential user’s hardware and software requirements. Barden’s discussion of input/output operation time, main memory capacity, and system response time is quite well done, and goes a long way toward helping a potential microcomputer buyer determine what type of system will best meet his needs. Aside from these two chapters, this book never rises above the level of the author’s previous fiasco.

MODERN COMPUTER CONCEPTS

Joseph C. Giarratano

Level: Intermediate Rating: **90**

Sams 1982 Paper
 302 Pages ISBN: 0-672-21815-1 8" x 11" \$22.95

This is the second volume in the epic four-volume survey of computer technology by Dr. Giarratano, a Bell Laboratory employee specializing

in the area of software development for microprocessor-based products. While the series encompasses the entire spectrum of data processing hardware, the author's own background surfaces in its particularly detailed treatment of microcomputer topics.

MODERN COMPUTER CONCEPTS covers six areas of computer technology: memory devices, central processing units, mass storage, data communications, computer networks and architecture, and videotex. The author's textbook-type treatment is crammed with technical data relating to each of these subject areas, and is supplemented by some of the most fascinating micro photographs (especially of microprocessors) in the literature.

The mass storage chapter, for example, first categorizes magnetic memories according to whether the medium moves or is stationary, and then proceeds with a detailed description of everything from minifloppies to bubble memories. The theoretical groundwork is laid by a discussion of data encoding techniques (non-return-to and zero-inverted and phase encoding methods). The reader is then stepped through the computer version of a CHILTON AUTO REPAIR MANUAL. Disks are dissected. Micro photographs magnified 5000 times reveal iron oxide textures. Even factory techniques of disk substrate production and coating are discussed in detail.

The years of research required to produce this single volume is evident in the similarly detailed treatments of each subject area. MODERN COMPUTER CONCEPTS is both a technician's sourcebook which is a delight to read, and a reference work which belongs in every computer library. While the microcomputer user will find subjects discussed which do not directly relate to his particular system, the information contained in chapters which are microcomputer-specific more than justifies the purchase of this book. We highly recommend it to the intermediate to advanced user.

NAILING JELLY TO A TREE

Jerry Willis and William Danley, Jr.

Level: Novice Rating: **95**

	Dilithium	1981	Paper	
244 Pages	ISBN: 0-918398-42-8		5" x 8"	\$15.95

Willis and Danley's NAILING JELLY TO A TREE is an excellent work about the creation, use and application of software. It is written for the reader who "has at least a passing knowledge" of microcomputer hardware, software and programming techniques, and "who is interested in using and adapting the thousands of computer programs that are available today."(2) The book largely focuses on explaining programming

languages for microcomputers, and serves as a general introduction to the topic; it also aims at demystifying programming languages and software and admirably succeeds in whetting the appetite of the reader for other, more specialized literature.

The book concentrates on explaining the basic principles of machine language, assembly language and BASIC (the only one of the higher-level languages the authors examine in detail, since it is used in most microcomputer systems); because BASIC is not a standardized language, the authors provide a final chapter on conversion from one BASIC to another. Their treatment of these topics contrasts sharply with that of most general works on microcomputers, which do discuss low and high level languages but leave the reader with the distinct impression that assembly and machine language are too complex and cumbersome to merit much attention. As a result, many readers—including those with a good deal of knowledge about microcomputers—simply assume that lower-level languages are beyond their grasp, and hence are unable to adapt pre-existing programs to their own microcomputers. By debunking this myth, *NAILING JELLY TO A TREE* offers those readers who are not using their machines as simple household “appliances” the possibility of deriving greater enjoyment, knowledge and utility from their microcomputers.

NCR DATA COMMUNICATIONS CONCEPTS

Technical Publ. Dept. of NCR
Level: Intermediate Rating: **80**

Sams 1980 Paper
206 Pages ISBN: 0-672-24548-9 6" x 9" \$7.95

Written by the Technical Publication Department of NCR, this primer of communications concepts is intended for the computer user interested in refreshing or broadening basic communications concepts. It is not written specifically for the personal computer user; most of the actual applications discussed apply to more sophisticated data processing environments (e.g., elementary networking and facsimile transmission).

Besides the general theoretical chapters, which require a minimal electronics background, the microcomputer user would be most interested in the modem-related topics; these include modem testing and troubleshooting and the diagnosis of transmission line problems.

Though a rather dated treatment of communications concepts if judged solely by the original copyright date (1971), *NCR DATA COM-*

MUNICATIONS CONCEPTS is nevertheless a good overview for both the data processing professional and computer hobbyist.

PEANUT BUTTER & JELLY GUIDE TO COMPUTERS

Jerry Willis, Deborrah Smithy and Brian Hyndman

Level: Novice Rating: **85**

Dilithium 1978 Paper
215 Pages ISBN: 0-918398-13-4 6" x 9" \$9.95

A quick scan of PEANUT BUTTER's chapter titles reveals the immediately ingratiating "flavor" of its text: "So you want to be a computer freak," "Leonardo de Altair," and "You too can build an IBM 360 with only a screwdriver and a blowtorch."

Jerry Willis, that omnipresent Dilithium Press author, has crafted one of the more comprehensible guides to personal computers. It is also one of the few that, recognizing the inherent shortcoming of this genre of microcomputer overview, does not hesitate to recommend other books and magazine articles which provide better elaborations of individual topics. (Even if the reader discounts the suspiciously high number of Dilithium Press books among the recommendations, enough solid supplementary reading remains to warrant this accolade.)

Unfortunately, PEANUT BUTTER's text has gotten a bit stale since its original publication in 1978; the hardware references, which are laced throughout its chapters, could stand some major updating (especially the chip examples and microcomputer buying guide section). Nevertheless, the general introduction to personal computers has more than weathered the test of time, as have the consumer tips chapters, which read like a microcomputer version of COMPUTER CAPERS (see our review).

PEANUT BUTTER AND JELLY GUIDE succeeds in being "an easy to digest source of information on personal computing." We reserve a higher rating, however, for the second edition, which its popular but dated treatment appears to warrant.

THE PERSONAL COMPUTER BOOK

Peter A. McWilliams

Level: Novice Rating: **95**

Prelude Press 1982 Paper
274 Pages ISBN: 0-345-31106-X 6" x 11" \$9.95

It is a tribute to Peter McWilliams' fresh, original presentation that even those readers who dislike his eccentric style (there are more than a few) inevitably read, and begrudgingly enjoy, the entire book. If Rodney Zaks is the Isaac Asimov of microcomputer literature, Peter McWilliams is its Kurt Vonnegut. His treatment is irreverent, self-deprecating, sometimes humorous, coy, sometimes tedious—and immensely effective.

A rambling, conversational text combines with captioned illustrations (the hell leaf of the Bosch triptych is proposed as a new video game screen) to unfold the world of personal computers. Though technical accuracy is sometimes compromised in order to keep the content simple (silica is "a fancy word for sand"), this introduction is unquestionably informative. For example, McWilliams shuns the dry, encyclopedic litany of the most popular personal computers on the market to pick the few he considers the best, and explains why. His observations and opinions are cavalierly delivered, but decidedly accurate.

In one of those many opinions, McWilliams openly worries that first-time computer users who purchase the Timex/ Sinclair ZX-81 will be forever traumatized by some of its anti-ergonomic features. (It was a similar concern to assist the novice in avoiding microcomputer books that might prove dangerous to the health of his burgeoning interest that prompted us to prepare this guide.) THE PERSONAL COMPUTER BOOK is the type of introductory text toward which every computer-curious reader should be steered. This intellectual-joystick tour receives one of our highest ratings in this category.

PERSONAL COMPUTERS IN BUSINESS

An Introduction and Buyer's Guide

Barbara Gibson

Level: Novice Rating: **65**

Apple Computer 1982 Paper
50 Pages ISBN: 0-9609780-0-3 9" x 11" \$2.95

This is a non-technical introduction to personal computing published by Apple Computer, Inc. The authors, who have tried to remain "as objective as possible," nevertheless hope that their text will make "it

easier to see why... Apple personal computers are the best selling personal computers in the world." In spite of the undeniable fact that this magazine-like publication is an unabashed marketing vehicle, its style and price elevate it above comparable concise overviews of microcomputers.

Slick graphics and full-color photographs supplement a low-key, advertising-copy narrative. (Indeed, some of the material has been excerpted from Apple Magazine.) The discussion of how a computer works is limited to component descriptions, and the text focuses on the possible business applications of personal computers, ranging from word processing to accounting functions. As each application is treated, a "features to look for" section presents the potential buyer with a checklist of hardware and software options. Accounts Receivable packages, we are informed, should automatically post balances to both individual accounts and to a general ledger, as well as produce traditional agings. Whether in the context of a small business or professional office, these applications overviews are supplemented by evangelical vignettes in which a small businessman or professional lands his or her particular personal computer system (in the interest of "objectivity" no specific personal computer is mentioned, though the accompanying photographs all display what are unmistakably Apple components). Concluding sections cover additional sources of information, questions and answers, and a dictionary of "computerese."

PERSONAL COMPUTERS IN BUSINESS is the abbreviated, Apple-view of the benefits of personal computers. Though limited in scope and format, its magazine-treatment of microcomputer business applications is, if nothing else, readily digestible.

PERSONAL COMPUTING 2nd Edition

Daniel R. McGlynn

Level: All Rating: **90**

Wiley 1982 Paper
335 Pages ISBN: 0-471-86164-2 5" x 10" \$14.95

The original draft of our review of McGlynn's PERSONAL COMPUTING was rather unfavorable, and criticized the book for defining its topic too broadly at the same time that it failed to integrate much of the material it covered. Since having written the review, however, PERSONAL COMPUTING has invariably been the first work which we have consulted in checking the accuracy of a myriad of details, or to refresh our memory about some basic microcomputer principles. Although it remains true that PERSONAL COMPUTING makes no attempt to present

a smooth, highly-focused discussion of some aspect of microcomputer usage, criticizing it for this reason falls prey to the truism that "a foolish consistency is the hobgoblin of small minds."

McGlynn has written this book for an audience which, we would hope, forms the vast majority of microcomputer users; this is a book for those who both enjoy reading and wish to expand their knowledge of computers. For such readers, McGlynn's attempt to cover the waterfront of computer-related topics, far from being a disadvantage, offers a wealth of information which is readily available to the reader at the precise moment it is needed. Aside from presenting intrinsically interesting "trivia" about microcomputers (such as his analysis of the personal computer marketplace, based on a 1978 study conducted by the University of Southern California), McGlynn has effectively included material of interest to microcomputer users with varying degrees of sophistication; while the novice to microcomputers can benefit from the chapters on programming and programming languages and on personal, educational, professional or business applications, more advanced users will be interested in the sections on interfacing or on operating systems.

Also most impressive is the emphasis which PERSONAL COMPUTING places on finding alternative sources of knowledge or assistance. Although the bibliography which McGlynn has compiled is brief and highly selective, he also provides an extensive list of microcomputer periodicals, as well as appendices listing American, Japanese and West European microcomputer manufacturers, the addresses of computer stores in the United States, and computer clubs, all of which the reader can use as a springboard to still further sources of information.

We highly recommend McGlynn's renaissance tour of personal computing to microcomputer users of all levels and interests.

A PERSONAL GUIDE TO PERSONAL COMPUTERS

Peter Lundstrom

Level: Novice Rating: **65**

Apple Computer	1982	Paper	
50 Pages	ISBN: 0-9609780-1-1	9" x 9"	\$2.50

Published by APPLE COMPUTER, Inc., this magazine-like guide bears a striking resemblance to PERSONAL COMPUTERS IN BUSINESS; indeed, certain sections of text are identical. Where PERSONAL COMPUTERS IN BUSINESS highlighted the cost benefits of microcomputer business applications, PERSONAL GUIDE emphasizes operational and hardware considerations. The differences between RAM and

ROM are addressed, and a series of monitor screens demonstrate initial program loading. The titles of the pamphlet's four main chapters indicate the scope of its text: "What Will a Personal Computer Do For Me?" "How Do I Shop For a System?" "How Do I Run This Thing?" and "What Will It Cost?" Both publications (though sold in bookstores, they cannot be called books) contain similar full-color graphics. Both give the same simple overview of the hardware and software options available to the prospective microcomputer buyer. Both, since they are marketing brochures for APPLE COMPUTER, provide a glimpse of elementary microcomputer concepts.

A PERSONAL GUIDE TO PERSONAL COMPUTERS is a quick, pleasant, admittedly biased survey of microcomputers. And for the price of \$2.50, its significant shortcomings become increasingly palatable.

RUNNING WILD

The Next Industrial Revolution

Adam Osborne

Level: All Rating: **80**

Osborne 1979 Paper
178 Pages ISBN: 0-931988-28-4 4" x 7" \$3.95

It is the preeminent role which Adam Osborne still plays in the microelectronic industry, rather than his book's inherent qualities, which kindles interest in RUNNING WILD. Osborne, a technical writer, consultant and corporate president, has the dubious distinction of having the term "guru" used as a pronominal adjective in scattered references to him in articles and books; RUNNING WILD, his 1979 version of the shape of things to come, may well have started this trend.

The author speculates upon the future effects of microelectronic advances upon society. In addition to the insights it sheds upon the economic and sociological perceptions of its author, RUNNING WILD retains its fascination for a number of reasons. Adam Osborne, as the young aggressive entrepreneur, describes a history of Silicon Valley unrivalled by any of the more prosaic accounts in microcomputer literature. None, for example, treat the role of the space race of the sixties in creating a legion of technically expert, and ultimately unemployed, microelectronic engineers. Osborne's tracing of the evolution of the microelectronic watch and calculator industries in the seventies becomes a Darwinian adventure in survival-of-the-fittest marketing strategies.

Surveys of the effects of technological advances on the white and blue collar job markets cover robotics, electronic speech and telecom-

munications. Much of what Osborne predicted in 1979 has come true; much has not. As he envisioned, changes in the composition of the American work force have emphasized service industries; micro-electronics has dramatically accelerated the development of efficient prosthetic devices; and diskette copy-protect technologies still lag behind the abilities of unscrupulous users. However, many vendors may disagree with his prediction that "IBM will soon cease to be a significant force" in the computer-system marketplace.

Adam Osborne is at his best when explaining the corporate and technological histories of the microcomputer industry. While his speculations are of marginal interest (if for no other reason than their obvious datedness), *RUNNING WILD* remains a significant document in the evolution of the industrial revolution which it describes.

SMALL COMPUTERS FOR THE SMALL BUSINESSMAN

Nicholas Rosa and Sharon Rosa

Level: Novice Rating: **80**

	Dilithium	1980	Paper	
331 Pages	ISBN: 0-918398-31-2		6" x 9"	\$16.95

Recognizing that businessmen are as a rule more interested in what it does than how it works, the Rosas have written a work which is highly successful in introducing only those microcomputer basics which are absolutely essential for buying and using a computer in a business setting. Rather than surveying the esoteric features of microprocessor architecture, *SMALL COMPUTERS FOR THE SMALL BUSINESSMAN* focuses on whether a computer is useful, what kind of computer the businessperson should acquire and how it can be used.

By far the best discussion in the text centers around the advantages of computerizing business operations. Using Roger Williams' article "Consideration for Computer Implementation in a Small Business," which appeared in *INTERFACE AGE*, the Rosas emphasize the hidden costs of not computerizing; "you will never gain back the money you will have lost by delaying acquisition."⁽⁷⁾ Although the authors could have presented a more sophisticated and refined economic analysis of the cost and benefits of computerization, their arguments in favor of acquiring a computer remain convincing.

The book is not so successful in addressing the issues of computer selection and application, however. This is in part a function of its loose organization and repetitiveness, which tend to distract an increasingly weary reader. But it results especially from the book's dated

treatment. Even though the distinction between mini and microcomputers was becoming blurred by 1980, the capability of most microcomputers was still limited, and memory was comparatively expensive; the state of the 1980 microcomputer market led the Rosas to recommend a system with 32k of memory as the "minimum" necessary for business needs, although they made no attempt to enable the reader to determine what a theoretical maximum might be. Since 1980, the range of microcomputers on the market has expanded enormously, their power and capability have increased rapidly, and their price has declined; a business can now acquire an extremely powerful system at a relatively low cost. But the Rosa's book, although a competent guide to the 1980 microcomputer market, is far less helpful in enabling the reader to select the proper microcomputer system in 1983.

This criticism also applies to the discussion of computer applications and software. While the authors emphasize selecting a computer system based on the range and utility of software available for it, they note at the same time a relative dearth of software for business applications, a situation which they expected would change sharply in a few years. As a result, their approach to selecting computer hardware and software emphasizes the use of a systems consultant and an outside programmer for special applications. With the greater range of business software which has since appeared, however, a more concrete analysis of existing software is mandated.

For the businessperson who cannot decide whether or not to computerize, *SMALL COMPUTERS FOR THE SMALL BUSINESSMAN* retains its value. For those searching for a system, we would recommend a more current assessment of the microcomputer marketplace.

SO YOU ARE THINKING ABOUT A SMALL BUSINESS COMPUTER

1982/83 Edition

R.G. Canning and N.C. Leeper
Level: Novice Rating: 90

Spectrum 1982 Paper
203 Pages ISBN: 0-13-823617-8 9" x 11" \$10.95

Most books on computer selection seem to be cloned from the same standard format. In the course of a discussion about microcomputer basics, which is done in a way either too complex or too simplistic for the potential personal computer buyer, the reader is expected to pick up a few valuable "clues" relating to his individual selection criteria. Aside

from actually increasing the reader's latent apprehension about micros, these books conspicuously fail to address the major question which they purport to answer: how to select a microcomputer.

Such is not the case with **SO YOU ARE THINKING ABOUT A SMALL BUSINESS COMPUTER**, which rises far above this mediocre level. The authors attempt one of the most difficult feats in an introductory treatment of microcomputers—to coax the novice and simultaneously sate the experienced user—and succeed.

The discussion of microcomputer basics is detailed and lucid. The actual criteria for selecting a personal computer are discussed with great care and seriousness. Even the appendix reveals an expertise too rarely evident in similar offerings by Hayden, Sams, McGraw-Hill or even the publisher of this book, Prentice-Hall. This 1982/83 edition offers a state-of-the-art overview of microcomputers (and some low end minis); any enlightened selection would seem to mandate its perusal.

SOUL OF A NEW MACHINE

Tracy Kidder

Level: All Rating: **95**

Avon 1981 Paper
287 Pages ISBN: 0-380-59931-7 4" x 7" \$3.95

Data General's Eclipse MV/8000 minicomputer certainly does not seem the stuff of which Pulitzer Prize winners are made; yet Tracy Kidder's intriguing account of the lives that intensely focused on its birth catapulted machine and author into literary stardom.

This book provides a capsule history of the competitive, mercurial minicomputer market in the seventies. It relates the subtleties of computer architecture through a diary-like walkthrough of the entire design process. Ultimately, however, it is a book that is only parenthetically about computers; its true poignancy lies in its terrible insight into the psyches of men who are driven to succeed. One remembers the corporate infighting and technical breakthroughs of Kidder's narrative as mere backdrops for the brilliant characterizations of Tom West and his Eagle team.

The author quotes a fellow reporter as saying that the computer industry "has an insidious ability to reduce things to less than human dimensions." **SOUL OF A NEW MACHINE** more than adequately redresses that imbalance, in the process providing one of the more enriching experiences in computer-related literature.

UNDERSTANDING ARTIFICIAL INTELLIGENCE

Paul Y. Gloess

Level: Novice Rating: **65**

Alfred 1981 Paper
47 Pages ISBN: 0-88284-150-5 4" x 11" \$2.95

With this Alfred Handy Guide, "today's fast-moving adult on the run" can sample the varied disciplines which are normally grouped under the umbrella term "artificial intelligence." These areas of computer research include heuristic evaluation, automatic theorem proving, deductive reasoning, and natural language processing. Each, in turn, is briefly surveyed by the author. A special chapter introduces LISP, a language whose manipulation of symbolic expressions has made it uniquely suited for artificial intelligence research (and, conveniently, the subject of yet another Alfred Handy Guide).

The inherent limitations of such an admittedly superficial overview would seem to preclude criticisms regarding lack of depth. So rather than point out that the chapter on heuristic evaluation of game positions neglects to mention some of the better known searching algorithms, we will simply conclude this abbreviated review. "Today's fast moving adult on the run" wouldn't have it any other way.

UNDERSTANDING COMPUTER SYSTEMS

Harold W. Lawson, Jr.

Level: Novice Rating: **90**

Computer Science Press 1982 Paper
164 Pages ISBN: 0-914894-31-5 7" x 10" \$10.95

This introduction to computer systems by a Professor of Telecommunications and Computer Systems is noteworthy for a number of its novel approaches to computing. The book, like many others, intends to give a general conceptual explanation of how data is manipulated electronically to produce some desired result. It introduces its subject, however, by first exploring the abstract nature of processes; that is, the series of transitions which an element undergoes in order to attain a new state. The processes are then developed. The author posits that the computer is nothing more than "a system of cooperating processes." With the logical groundwork for his premise presented, each of the computer's component processes, whether logic gate architecture or auxillary storage, is described in detail.

"Dimensional" flowcharting, a diagrammatic pseudo-code, is effectively used in these early general process logic discussions, as well as in later chapters which "draw" the parallels to CPU process logic. Even the obligatory treatments of Boolean algebra and computer arithmetic are enlivened when viewed from a purely "process" perspective (different gate processes change input signals into different signals).

New data processing terms are highlighted throughout the text in boldface to assist the novice in learning about software, hardware and system architecture. A comprehensive glossary provides additional references.

While it addresses computers in general without regard to power or size, UNDERSTANDING COMPUTER SYSTEMS provides an engaging, original approach to the explanation of a computer's most basic functions.

UNDERSTANDING COMPUTERS

Paul M. Chirlian

Level: Novice Rating: **85**

Dilithium 1978 Paper
193 Pages ISBN: 0-918398-15-0 6" x 9" \$11.95

UNDERSTANDING COMPUTERS seeks to explain how a computer functions in terms of clever combinations of its simplest elementary structures—logic gates. This "atomic" approach to microcomputer operation forms a special sub-category of introductory literature; one of its earliest and best-known popular treatments is this Paul Chirlian book.

The focal subject of the introductory chapters is the conceptual link among Boolean algebra, gate logic and circuit design. A series circuit is used to represent an AND switch, and to introduce logic gate symbols; a parallel circuit represents an elementary OR switch. Variations of these two switches are then interconnected to produce a circuit that performs binary addition (the author's "half-adder"), again accompanied by a gate circuit diagram and Boolean truth table. The plot thickens as Chirlian deftly moves from this simple combinational circuit to circuits that "remember" electronic impulses—"flip-flops." Block diagrams and the now familiar gate circuit diagrams supplement his discussion of R-S and J-K flip-flops. These are in turn used as the building blocks for even more complex assemblages of shift registers and memories, until the basic workings of a computer have been described.

This orchestrated crystallization of a functioning computer from elementary switches is carefully and clearly managed by the author. Along the way, binary arithmetic and programming languages are dis-

cussed. It is extraordinary that the only section of this 1978 book which appears outdated is the final chapter which surveys available microcomputer systems.

UNDERSTANDING COMPUTERS remains a classic treatment of microcomputer architecture at the gate level.

UNDERSTANDING COMPUTERS

Thomas H. Crowley

Level: Novice Rating: **45**

McGraw-Hill 1967 Paper
 142 Pages ISBN: 0-07-014761-2 5" x 8" \$4.95

This dated McGraw-Hill paperback is of interest only in that it accentuates the reader's sense of the speed with which generations of computer technologies have become extinct since its original publication in 1967. The pictures of vacuum tube IBM 705's elicit an almost archeological curiosity. One chapter treats punched card technology. Tables summarizing memory capacities and access times read like ancient newspaper headlines.

For the most part, chapters which discuss the basic conceptual elements of computing have remained relevant. Number theory and gate logic, whether implemented in vacuum tubes, transistors, integrated circuits or bubble memories, have remained the same. Programming processes, like the translation of a high-level language into executable machine code, are also treated in chapters which continue to be informative despite the intervening one-and-a-half decades of technological change.

UNDERSTANDING COMPUTERS is also interesting for its sociological posturing. An underlying concern at the time of its writing was the role of computers in displacing part of the American work force. The author rather defensively argues the merits of a data processing-intensive economy (in a chapter entitled "1984") with amazingly familiar "trickle-down" hypotheses. Some guesses at probable paths of technological advance proved to be true (heuristic systems), others not (self-reproducing computer systems).

Though still actively marketed as a general introduction to computer technology, UNDERSTANDING COMPUTERS should, of course, be avoided for its technical obsolescence. It is this same datedness, however, which offers a rare window into an earlier generation of computer technology.

UNDERSTANDING DATA BASE MANAGEMENT

Michael J. Freiling

Level: Novice Rating: **80**

Alfred 1982 Paper
65 Pages ISBN: 0-88284-221-8 4" x 11" \$2.95

Alfred Handy Guides are the fast-food alternatives to the more weighty (and, to some, indigestible) offerings of microcomputer literature. Sized and priced for quick, no-nonsense reading, they do nevertheless offer snapshot overviews of different data processing topics. Though a Handy Guide on data base management may seem to a computer professional a bit tautological, Michael Freiling's treatment imparts more than a taste of the conceptual framework of modern data base design.

While it necessarily falls short of a definitive survey of this complex subject, the guide nevertheless manages to discuss relational modeling, schema design and basic network and hierarchical relationships. Bachman diagrams are used to show the inter-relatedness of different data base objects.

The reader will most probably not glean much true wisdom about data base architecture from this particular treatment, but as an inexpensive introduction to the topic it rates somewhere between a Monarch Note and a James Martin data base treatise; it is an accurate, concise, accessible survey.

UNDERSTANDING THE MICRO How It Works and What It Can Do

Judy Tatchell and Bill Bennett

Level: Novice Rating: **80**

Usborne 1982 Paper
48 Pages ISBN: 0-86020-637-8 6" x 9" \$4.95

Judging by the age of the few non-robots included in its comic-book illustrations, the audience of UNDERSTANDING THE MICRO is the pre-teenager who has expressed some interest in personal computers. The premise of this originally British publication, that the jump from the Hulk to hex is not a particularly difficult one, appears to be true.

Lilliputian robots literally guide the reader/viewer through micro-computer components, operations and programming. The motherboard of a Timex/Sinclair 1000 is dissected, with its logic chips, RAM and ROM

chips, voltage regulator and printed circuit board highlighted. A BASIC program is displayed (though this is not a BASIC primer) and a brief history of microcomputers is graphically illustrated.

There are problems that adults will experience with this kind of treatment, however. The explanation of elementary gate-logic is so muddled that it may inadvertently "turn-off" the uninitiated reader. The buyer's guide, which surveys some low-end microcomputer products (ZX-81, VIC-20, Atari 400), seems incongruous given the context and level of presentation of previous chapters. Some of the more obvious misprints are also rather irritating ("Micro Sound" is the heading of a section on microcomputer graphics).

But, given the limitations of its format and age level of its audience, UNDERSTANDING THE MICRO is an attractive, fun introduction to microcomputer basics for pre-teenagers.

USING MICROCOMPUTERS IN BUSINESS

A Guide for The Perplexed

Stanley S. Veit

Level: Novice Rating: **85**

	Hayden	1981	Paper	
142 Pages	ISBN: 0-8104-5152-2		6" x 9"	\$9.95

It is indicative of the pin-stripe pragmatism that Stanley Veit preaches that his chapters on microcomputer memory and general functioning are the final, and not the first, chapters in this book: the author correctly assumes that the average business "computerizer" is more concerned with the impact of the electronic box upon his daily operation than with the operation of the box's electronics. This is unquestionably a businessperson's book.

Emphasis is placed on topics such as the best methods of phasing-in a micro without disruption to the normal flow of activity and the actual selection of the correct hardware/software/consultant mix. The presentation is clear and reasoned.

Installation strategies and the anticipation of post-installation problems are discussed on a level suited to any data processing environment. (Mr. Veit even includes a few pages of a step-by-step recovery from a system crash.) While all of the necessary microcomputer specifics, and more, are included, the author never loses sight of his audience or its environment (the three-page how-to-program-in-BASIC treatment has thankfully been omitted). USING MICROCOMPUTERS IN BUSINESS is

a book to be recommended to microcomputer-oriented businesspeople of all perplexities.

WHY DO YOU NEED A PERSONAL COMPUTER?

Lance A. Leventhal and Irvin Stafford

Level: Novice Rating: **75**

Wiley 1981 Paper
278 Pages ISBN: 0-471-04784-8 7" x 10" \$8.95

WHY DO YOU NEED A PERSONAL COMPUTER?, like other books on microcomputer selection, attempts to outline some of the basic features of system design and usage while at the same time providing the reader with a sense of how to go about selecting a personal computer. The final chapter integrates much of the material from the preceding discussion as it explicitly focuses on the process of selecting a microcomputer.

While the book succeeds reasonably well in addressing this topic, its early chapters often create the impression that the newcomer to microcomputers is intellectually deficient, and that the questions posed ("how can computers seem to be so smart?") and explanations offered by the authors must be phrased with this in mind. But after the obligatory exercise in trying to teach the reader BASIC (which, it seems to us, rarely has any place in works on microcomputer selection), the quality of the book improves rapidly.

The authors distinguish three major kinds of microcomputers: one-board computers, appliance computers and modular computers. The first are typically computers with limited functions (such as dedicated devices or learning aids), while the distinction between appliance and modular computers is based in part on the ease with which a wide range of peripherals can be added; more importantly, however, appliance computers are ready-to-use systems, while modular computers must be assembled from individual components which are integrated with the central processing unit. The criteria for selecting a microcomputer are most clearly oriented toward choosing among these three categories of microcomputers, but do not clearly and explicitly focus on selecting a computer within any of these three categories. This shortcoming substantially reduces the value of the book to most buyers, who are concerned with making a choice within any one of these categories.

WITHOUT ME YOU'RE NOTHING

The Essential Guide to Home Computers

Frank Herbert

Level: Novice Rating: **50**

Pocket Books	1980	Paper	
304 Pages	ISBN: 0-671-43964-2	5" x 8"	\$5.95

In *WITHOUT ME YOU'RE NOTHING: THE ESSENTIAL GUIDE TO HOME COMPUTERS* Frank Herbert, the author of *DUNE*, has traded an ecology of sand for one of silicon; only the most obdurate of Herbert fans however, will be able to skirt its black holes of platitudes and literary digressions to salvage a few particles of microcomputer essentials.

After a promising start in which a computer is conceptually approached as "an assembly of switches," the reader is confronted with a marginally interesting discourse on computer vernacular as a "trade language" in process. This is followed by an entire chapter in which the author ruminates about "what we should call computers" (his punch-line conclusion is "Hey, you").

A token effort is made to explore the obligatory topics of RAM vs. ROM and the history of computing. The gravity of some unseen body (perhaps a more weighty treatment of the same topic) continues to draw the author off-track: the reader is given a soapbox monologue on the inevitable collapse of computer monopolies and the implicit deceptions they employ when marketing large, complex computer systems. A "buyers guide" section which follows never mentions a single specific microcomputer, but does laud the merits of the PDP-11, a minicomputer system (an implicit deceit never mentioned by the author).

Superficial discussions of programming in general and BASIC in particular mark the second half of *THE ESSENTIAL GUIDE TO HOME COMPUTERS*. The author, whose early chapters contain diatribes against computer jargon, proceeds to introduce the BASIC programming language by peppering the reader with fifteen BASIC statements in the course of three pages of text. This treatment is further confused by the additional introduction of Herbert's PROGRAMMAP System, a hybrid graphics-cum-flowchart approach to structured programming. The main narrative ends with the author's rather refreshing admission that he has indeed been redundant in the book's presentation, but that the use of repetitious examples is a "well-tested educational technique" employed only for the reader's benefit.

WITHOUT ME YOU'RE NOTHING: THE ESSENTIAL GUIDE TO HOME COMPUTERS is a perfect foil for Peter McWilliams' *THE PERSONAL COMPUTER BOOK* (see our review). Both take risks in an attempt to entertain while teaching, but only one succeeds. We can only hope that *WITHOUT ME YOU'RE NOTHING: THE ESSENTIAL GUIDE TO HOME COMPUTERS* is not the beginning of a new tetralogy.

YOUR FIRST COMPUTER

A Guide to Business & Personal Computing

Rodnay Zaks

Level: Novice Rating: **90**

Sybex 1980 Paper
 258 Pages ISBN: 0-89588-045-8 6" x 9" \$8.95

Although the title does not indicate it, this book has been written specifically for the small businessperson who is considering computerizing his accounting and bookkeeping operations; in addition to providing necessary technical information about how computers operate, the author attempts to present criteria which allow the reader to determine whether he or she should computerize at all, and, if so, what kind of microcomputer system should be purchased.

YOUR FIRST COMPUTER fulfills these goals extremely well, although in a few cases the criteria presented to enable the potential microcomputer buyer to choose an effective system are left at an abstract level, so that it is not immediately apparent to the reader how to translate these principles from theory into practice.

Despite this, the book has much to offer. First of all, the discussion of the advantages of computerization is one of the best we've read; generally, this question is unjustifiably ignored or dismissed after a few clichés. Zaks shows that he has taken it seriously and has devoted a good deal of thought and attention to the way in which a computer might affect a small business. The book is also very successful in explaining how microcomputers operate to the complete novice; Zaks has managed to present extremely technical issues in terms which are comprehensible without being overly simplistic. Finally, most of the criteria for computer selection which Zaks outlines are presented in a clear and practical way. At times, only a slight lack of specificity and concreteness prevents this from being the definitive work on microcomputer selection. For example, the discussion of the advantages of high-level compiled languages, such as FORTRAN or COBOL—as opposed to high-level interpreted languages, such as BASIC—is very well-handled, although the reader is not told how this affects actual computer operation time.

On the whole, this is one of the best "how to select your own computer" books. It ranks with Sams' CRASH COURSE IN MICROCOMPUTERS and Prentice-Hall's SO YOU ARE THINKING ABOUT A SMALL BUSINESS COMPUTER as superior general introductory microcomputer paperbacks.

YOUR OWN COMPUTER

Mitchell Waite and Michael Pardee

Level: Novice Rating: **90**

222 Pages Sams 1982 Paper
ISBN: 0-672-21860-7 6" x 9" \$7.95

The scenario of one old game show had contestants with a limited amount of time racing around department store aisles, frantically filling their shopping carts with what they perceived to be the most valuable merchandise. The results, graphically reviewed for the audience by an item-by-item inventory of their carts' contents, were always particularly revealing of each individual's personal value system.

Reviewing introductory computer books is like being asked to critique a television season's-worth of game-show carts; the authors of such books are faced with the impossible task of presenting their readers with a finite sampling of an almost infinite body of knowledge. And like those contestants, their choices are particularly revealing of their own hierarchies of importance.

YOUR OWN COMPUTER is one of the best examples of this very individual subject emphasis. Written by two veteran microcomputer authors, it is a thoroughly readable overview of the personal computer market, which distances itself from its counterparts by the obvious eclecticism of its topic selections.

It is part reference manual. A brief glossary of computer terms is included, accompanied by a separate explanation of microcomputer acronyms. Almost 40% of the book is a feature and price comparison of forty personal computers. (With the second edition published in 1982, the absence of the IBM Personal Computer is rather puzzling.)

It is part general introduction to the personal computer market. A brief history of microcomputers ends with a next-decade forecast of the penetration of micros into everyday life. Microcomputer architecture and functions are explained and illustrated for the novice.

It is an introduction to programming. Curiously, a cursory description is given to high-level languages, while inordinately strong emphasis is placed upon numbering systems and binary arithmetic, normally the purview of introductory assembly language texts.

YOUR OWN COMPUTER is a personal computer introduction that goes to great lengths to cover all of the relevant facets of its topic. While some of these emphases seem incongruous in an introductory text, its authors are to be congratulated for the variety and quality of the aisles sampled in this department-store tour of personal computing.

A 60-MINUTE GUIDE TO MICROCOMPUTERS

A Quick Course In Personal & Business Computing

Lew Hollerbach

Level: Novice Rating: **80**

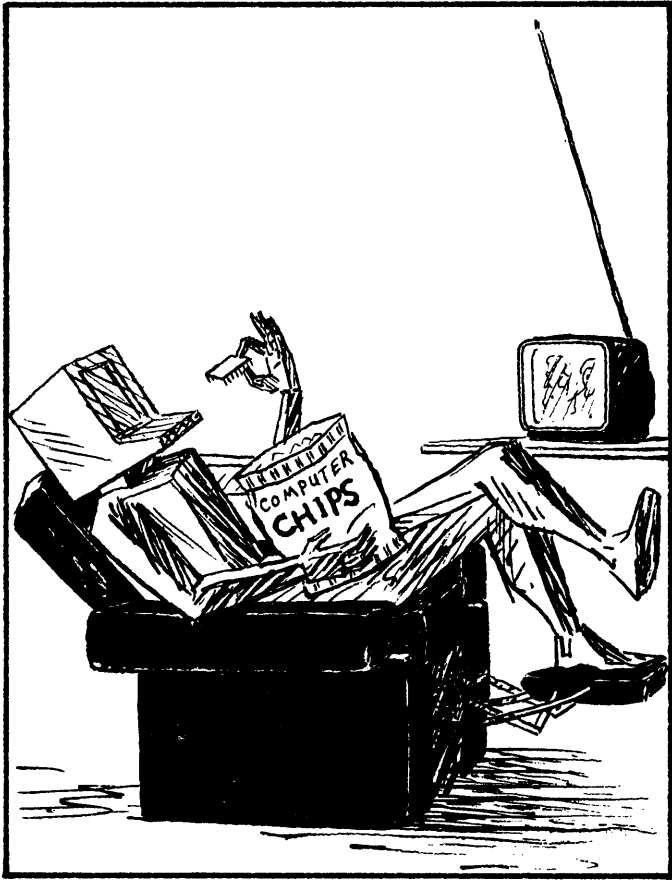
Spectrum 1981 Paper
137 Pages ISBN: 0-13-811448-X 6" x 8" \$6.95

Though Lew Hollerbach will never be accused of being either the Pierre Franey or Craig Claiborne of microcomputer literature, his "quick-course in personal and business computing" certainly achieves its admittedly limited objectives. Indeed, the refreshing appeal of this book, especially when compared to some of its ponderous, over-ambitious rivals, is that it is simply written, clearly presented and conveys most essential microcomputer basics in sixty minutes (or \$6.95, whichever comes first).

"Admittedly limited," however, is the operative definition: the few pictures in this 136-page tome are necessarily symbolic (actual photographs of microcomputers would waste precious minutes). Sentences:

- are frequently dismembered by a memo-like convention;
- use dots, and boldface to convey a topic's essential points;
- are much like the answers to a multiple-choice question;
- whose final, correct solution is 'all of the above'.

60 MINUTE GUIDE is a good, quick introduction for the reader who values his time more than both his money and a desire for real understanding. For a few dollars and a few hours more, however, we recommend the much more enriching experience provided by any of the highly rated titles in this category.



2

Microprocessors

Feeding the microcomputer revolution are the last decade's startling developments in computer chip fabrication technology. As the electronic pathways etched in silicon wafers become smaller and more complex and the chips themselves become cheaper, personal computing becomes an even more pervasive part of "everyday" American life.

Any appreciation of the power and elegance of microcomputer design requires an understanding of microprocessors. In this area, general introductory works like Rodnay Zaks' *FROM CHIPS TO SYSTEMS* are excellent. More specialized treatments of specific microprocessors refine this knowledge; one example in the 16-bit arena is Morgan & Waite's *8086/8088 16-BIT MICROPROCESSOR PRIMER*.

Aside from familiarizing themselves with developments in computer technology, many computer users will be attracted to microprocessors in order to learn assembly language programming. Included in this chapter are reviews of a number of quality works on programming the Z80, 8080, and 6502 microprocessors; others can be found in the section devoted to individual microcomputer systems.

FROM CHIPS TO SYSTEMS **An Introduction to Microprocessors**

Rodnay Zaks

Level: Intermediate Rating: 95

Sybox 1981 Paper
552 Pages ISBN: 0-89588-063-6 7" x 9" \$14.95

Besides being a photographic bestiary of the last decade of large scale integrated circuit technology, *FROM CHIPS TO SYSTEMS* is a hardware-oriented discourse on microprocessors. The "introduction" of

the subtitle should not be construed to mean that this is anything other than a microprocessor treatment aimed at a moderately sophisticated, programming-knowledgeable audience.

Historical and conceptual discussions revolve around an understanding of elementary microprocessor architecture. Readers familiar with Zaks' other microprocessor books (PROGRAMMING THE Z80, PROGRAMMING THE 6502) will immediately recognize the block diagrams of microprocessor components. The 8080 is initially used to follow the path of the main types of instructions through a microprocessor. These early chapters are marked by Zaks' habit of either assuming prior reader knowledge (binary arithmetic and shift and carry operations) or introducing topics which are only explained in detail much later in the text (MOS flip-flops). What remains, however, is an eminently readable revelation about the internal operation of microprocessors.

The author branches out to other microprocessor-based system components—specifically, memory and I/O interface chips. These chapters focus on specific chips, and display their components in order to illustrate different microprocessor topics (for example, the Intel 8251 USART for I/O, and the Motorola 6820 PIA for parallel I/O). A comparative survey of currently marketed microprocessors highlights the varieties of architectures and component interfaces that have arisen from the elementary building blocks of early integrated circuits. Bit width is used as Zaks' criterion for analysis, with groupings of the major 4, 8 and 16-bit microprocessors. This comparative evaluation of competing microprocessors (for example, the 8085 vs. Z80 vs. 6800), which necessarily includes some reference to their historical evolution, is one of the highlights of the book. On the 16-bit front, the architectures of the 8086/88, Z8000, and 68000 are analyzed.

After an additional synthesis of microprocessor components into a hypothetically "complete" system, microprocessor applications in four areas are presented: industrial systems, consumer devices, computer systems, and specialized applications (military, avionics, and aerospace). Assembly language programming and microprocessor system development tools conclude the technical chapters of Zaks' book.

The future, according to the author, will see advances in the three areas which have traditionally delimited microprocessor production and performance: production yield (the percentage of good chips per production batch), microprocessor internal speed and component densities.

FROM CHIPS TO SYSTEMS is an epic treatment of what Zaks describes as "the second industrial revolution"—the emergence of microprocessors. It is mandatory reading for the personal computer user who seeks to more fully understand the internal functioning of his or her machine. Paradoxically, however, one's appreciation of this book is, within limits, proportional to the amount of prior microcomputer

knowledge and experience that he or she brings to reading it. We therefore recommend FROM CHIPS TO SYSTEMS to the intermediate to advanced user.

FUNDAMENTALS OF DIGITAL COMPUTERS

Donald D. Spencer

Level: Novice Rating: **50**

Sams 1978 Paper
320 Pages ISBN: 0-672-21534-9 6" x 9" \$10.95

In order to capitalize upon the the burgeoning personal computer market, some editors search through their publishing companies' archives hoping to unearth old material that can be revived as new, saleable "product." Their combined efforts, which span publishing houses and continue undaunted even today, are manifested in that miscreant of microcomputer literature—the "retread."

FUNDAMENTALS OF DIGITAL COMPUTERS displays such attributes. This is the second edition of a work originally copyrighted in 1969; though the text and photographs have obviously been spruced up to reflect the technological changes of the subsequent decade, the reader is nevertheless mystified by such anachronisms as an emphasis on punch cards. The microcomputer user will also be disappointed by the primarily mainframe thrust of Mr. Spencer's book. While good overviews are given of the most popular high-level programming languages, this singular bright spot is not enough to rescue FUNDAMENTALS from its fundamentally dated treatment.

HOW TO PROGRAM MICROCOMPUTERS

William Barden, Jr.

Level: Intermediate Rating: **85**

Sams 1977 Paper
256 Pages ISBN: 0-672-21459-8 6" x 9" \$10.95

When William Barden, Jr., states in this book's preface that he intends to present programming "on the most basic, machine or assembly-language (level)," most computer literate individuals realize that this is the prelude to a rather demanding bit of reading. Not the Sams blurb-writer, however. Barden's use of the word basic is interpreted as "elemen-

tary" instead of "fundamental," and transmogrified into a back-cover description which lauds "a guidebook...written especially for beginning programmers." Few beginning programmers will read this book and graduate to intermediate.

This, of course, is not to detract from the inherent merits of HOW TO PROGRAM MICROCOMPUTERS, which is really a guide to microprocessor programming. William Barden, Jr. is a lively technical writer who finds his true metier when presenting microprocessor topics (his three non-technical general introductions, reviewed in this book, meet with far different evaluations). Specifically, the book is an introduction to the assembly-language manipulation of data using the 8080, 6800, and 6502 microprocessors. It proceeds by providing a brief conceptual background of arithmetic operations within the computer, and progresses to microprocessor architecture. Assembly language op codes are introduced in short programming examples, illustrating different but equivalent methods of moving data and bit processing among the three microprocessors. Barden's final chapter includes more than fifty microprocessor-specific subroutines and their functions, to be incorporated by the programmer into his own assembly language code.

HOW TO PROGRAM MICROCOMPUTERS is a deceptively engaging introduction to the world of microprocessors. Its in-depth treatment and comparison of the three popular microprocessors is recommended to broaden the reader's knowledge of machine-level data manipulation.

AN INTRODUCTION TO MICROCOMPUTERS-VOL. 1 Basic Concepts 2nd Edition

Adam Osborne

Level: Intermediate Rating: **90**

Osborne	1980	Paper	
430 Pages	ISBN: 0-931988-34-9	7" x 9"	\$15.95

This second volume of the four-volume INTRODUCTION TO MICROCOMPUTERS has become a classic in microprocessor literature. A wealth of material is concealed beneath its deceptively simple structure: five main sections treat Fundamental Concepts, Microprocessor Components, The Central Processing Unit, Logic Beyond the CPU, and Programming Topics. All sections are marked by a heavy reliance on diagrams to assist in explaining microprocessor data flow. All additionally adopt the convention of highlighting key concepts in boldface, with regular typeface used for explanatory text.

Osborne, in the introductory chapters on elementary microprocessor architecture, intentionally omits any discussion of chip fab-

rication technologies (unlike Zaks' FROM CHIPS TO SYSTEMS) to concentrate on how the existing hardware is used to manipulate data. After the obligatory review of binary arithmetic and Boolean logic, Central Processing Unit register use is displayed in a sequence of diagrams which track the flow of data during the execution of individual microinstructions. This treatment is graphically effective. Though "something of a tangent within the context of the products discussed in this book," a section on chip-slice control units is unique in the literature both for its content and clarity of presentation. Osborne's introductory material is filled with similar caches of enlightening text. Despite its 1980 copyright, the author remains technologically fashionable with discussions of bubble memories and 16-bit microprocessor architectures.

Interfacing is the primary consideration of the chapter entitled "Logic Beyond the CPU." External buses, transistor-transistor logic gates and I/O ports are covered. Circuit diagrams of simple external devices (for example, a temperature regulator) "ground" the conceptual aspects of the author's discussion of these topics. The concise, clear handling of communication protocol (BISYNC, SDLC, HDLC, ASYNC) which concludes this section is of particular utility for data processing professionals looking for background material for microcomputer network designs.

The final segment of VOLUME 1: BASIC CONCEPTS treats assembly language programming. Osborne's answer to minimal BASIC is a general 8-bit assembly language which he develops, using IEEE syntax and formatting standards, into a versatile, comprehensive instruction set. The CPU register technique of following an instruction sequence through a series of memory diagrams is again used to illustrate specific assembly language commands. Advanced topics include register-register instructions, parameter passing, and interrupt instructions.

An INTRODUCTION TO MICROCOMPUTERS VOLUME 1: BASIC CONCEPTS is the ideal introduction to microprocessor operation and non-specific assembly language programming for the intermediate programmer. Despite its title, this is not a book for the inexperienced personal computer user.

INTRODUCTION TO 8080/8085 ASSEMBLY LANGUAGE PROGRAMMING

Judi N. Fernandez and Ruth Ashley

Level: Intermediate Rating: 80

Wiley	1981	Paper	
303 Pages	ISBN: 0-471-08009-8	7" x 10"	\$10.95

Adopting a strict frame-by-frame programmed instruction technique, the authors of this Wiley Self-Teaching Guide present assembly

language programming to the experienced microcomputer user. Elementary programming concepts are not treated. Rather, the book focuses on building programming fluency using the 8080/8085 instruction set.

The development of this programming proficiency is approached quite logically. The reader first practices number systems conversions. The format of 8080/8085 assembly language statements (label-operation-operands-comments) is introduced. Subsequent chapters cover the simplest members of the instruction set, assembler directives and conditional instructions. As with many other programming language introductions, each new instruction is incorporated into a simple program in order to demonstrate its use alone and in conjunction with other instructions.

These introductory chapters treat assembly language code and register manipulation without the normal references to microprocessor architecture. Unlike different high-level language dialects, an assembly language program is inextricably bound to a particular microprocessor: the internal structure of the microprocessor determines the potential versatility of its corresponding assembly language instruction set. It is precisely for this reason that Adam Osborne (INTRODUCTION TO MICROCOMPUTERS, VOLUME 1) and Rodney Zaks (PROGRAMMING THE Z80) both rely heavily on architectural diagrams to supplement their chapters on introductory assembly language programming. This key element is missing from the text of INTRODUCTION TO 8080/8085 ASSEMBLY LANGUAGE PROGRAMMING.

After additional register instructions (LDA, STA, INR, DCR) and logical operations (ANA, ANI, ORA, ORI) are covered, the role of the stack for temporary data storage is treated. All elements are packaged into functional programs with the normal divisions of main line code, subroutines, and data area definitions strictly observed. In this context, emphasis is placed upon the use of PUSH and POP instructions in subroutines to accomplish a specific function without disrupting the integrity of data on the stack and in registers and memory. A final chapter describes advanced techniques for handling multibyte addition, multiplication and division.

INTRODUCTION TO 8080/8085 ASSEMBLY LANGUAGE PROGRAMMING is an adequate step-by-step approach to learning the elements of assembly language coding; for more comprehensive, equally accessible treatments, however, we recommend any of the highly rated texts in this section.

MICROPROCESSORS

An Introduction

Paul Kimberley

Level: Novice Rating: **70**

McGraw-Hill 1981 Paper
 280 Pages ISBN: 0-07-034548-1 5" x 8" \$6.95

This McGraw-Hill reprint of an originally English publication occupies a unique niche in microcomputer literature: half introduction to microprocessors, half computer dictionary, it does manage to give the reader a taste of the nature and importance of LSI technology in the microcomputer revolution. But the real question Kimberley's book raises is: "who would want to buy this book?" Any prospective reader sophisticated enough to pursue information about microprocessors would already have gleaned more knowledge from general introductory texts than Kimberley presents in his "specialized" treatment. In addition, any of the computer dictionaries reviewed in this book do a much better job than MICROPROCESSORS' 115-page treatment (yes, over 40% of this book is a dictionary).

The combination of a superficial microprocessor treatment and an inferior computer dictionary may well provide the 'professional layman' (Kimberley's purported reader) with a rudimentary knowledge of microprocessors. We doubt it. Less mediocre discussions of this inherently fascinating topic can be found in any one of the better introductory microcomputer books (rating of 85 or higher) or, if detailed knowledge is required, in any of the superior microprocessor titles.

OSBORNE 4 & 8-BIT MICROPROCESSOR HANDBOOK

Adam Osborne and Gerry Kane

Level: Advanced Rating: **95**

Osborne 1981 Paper
 1204 Pages ISBN: 0-931988-42-X 7" x 9" \$19.95

OSBORNE 16-BIT MICROPROCESSOR HANDBOOK

Adam Osborne and Gerry Kane

Level: Advanced Rating: **95**

	Osborne	1981	Paper	
777 Pages	ISBN: 0-931988-43-8		7" x 9"	\$19.95

These two volumes supersede volume II of AN INTRODUCTION TO MICROCOMPUTERS published by Osborne/McGraw-Hill. Taken together, they form a catalog of four, eight and sixteen-bit microprocessors on the market at the time the book went to press. In addition, the final chapter of 16-BIT HANDBOOK provides information on nine chip slice products in the 2900 series manufactured primarily by Advanced Micro Devices. (Chip or bit slice products are devices which implement the arithmetic and logical unit of traditional microprocessors, but do not possess their control unit.)

Instead of opting for a catalog of all microprocessors (which would have been of antiquarian interest only), the authors have wisely chosen to focus only on those which are not obsolete. Hence, the section on four-bit microprocessors treats only four-bit microcomputers (the TMS 1000 family, National Semiconductor's COP400 Series and the Rockwell PPS4/1 family) which integrate CPU, memory and I/O on a single chip, making them ideal as controller devices. Excluded are those chips which, even though they continue to be manufactured and will enjoy a high volume of sales in the future, have become obsolete and are now used only in products previously designed. The section on chip slice products is just as highly selective, and focuses on the 2900 series because it is "the clear leader in terms of sales and customer acceptance." (8-1)

In addition, these volumes examine only those support chips for a particular CPU which can be used with that CPU alone. Devices which support a particular microprocessor but can be used with others, such as Intel's 8255 and 8255A Parallel I/O interfaces, are treated in Volume III of AN INTRODUCTION TO MICROCOMPUTERS. Chips such as Intel's 8087 Numerical Data Processor are also not discussed, presumably because they were released after the book went to press.

This is in no way to downplay the scope of these two mammoth volumes, which provide a state-of-the-art survey of microprocessor technology. Nor is it to diminish the enormous achievement of the authors in assembling what is, on balance, an extremely comprehensive guide to existing microprocessors. A comparison of the contents of these two volumes with the standard for such microcomputer reference works serves to emphasize the unique value of these handbooks and the painstaking care and attention to detail shown by Osborne and Kane in compiling them.

The de facto standard for such technical works is a high-priced book which summarizes information provided by manufacturers. Sundry and diffuse data are generally tied together by an incomprehensible introduction which makes exaggerated claims for the importance of the volume but never quite focuses on what the book is intended to accomplish or how it is supposed to be used. In contrast, the almost two thousand pages of text in these microprocessor handbooks are, considering the sheer volume of information they contain, a bargain at the price. Also notable is their high production quality, reflected in a densely packed yet readable text, the effective use of screened boxes to enhance the readability of tables, and their durable binding. But what really distinguishes this work is the quality of its information. In the sections devoted to each chip, the authors have adopted a standard format to present material which includes: the ancestry of the chip (e.g., both the Zilog Z80 and the Intel 8085 having been based on the Intel 8080A); current manufacturers and variations among their products; microprocessor architecture, including registers, status flags, pins and signals; the instruction set and addressing modes; machine cycles and clock signals; and data sheets with electronic and timing data.

Although the text is a critical one which emphasizes the strengths and weaknesses of particular microprocessors, it is not the authors' intention to provide a comparative analysis. Similarly, although a benchmark program (which moves data from an input buffer to a permanent table) is provided for each microprocessor, the authors believe that this merely underscores "the capriciousness of benchmark programs."(xvi) The superiority of a given microprocessor depends on the application.

Given the high quality of these two volumes, it is unfortunate that they are of interest primarily to two highly specialized audiences which include designers of microprocessor-based products as well as hobbyists and those technically sophisticated readers who prefer to focus on real microprocessors instead of microprocessor technology in the abstract. But for those interested in a catalog of microprocessors, it would be virtually impossible to improve upon these two handbooks.

PROGRAMMING A MICROCOMPUTER: 6502

Caxton C. Foster

Level: Intermediate Rating: 65

Addison-Wesley	1978	Paper	
229 Pages	ISBN: 0-201-01995-7	7" x 10"	\$9.95

The contents of Foster's book reflect its 1978 publication date, a time when microcomputers were first beginning to appear, knowledge about

computers was readily accessible to comparatively few people, and assembly language programming appeared to be an enormously difficult and esoteric art. Given the obvious importance of a discussion of microprocessor architecture for an understanding of assembly or machine language programming, it is puzzling that Foster has largely evaded this prerequisite to programming by instead dragging out his "homuncular (little man) theory of computers,"(1) which likens computer operations to the duties performed by a file clerk. In addition, the book focuses on specific programming projects (such as measuring keybounce, working combination locks, playing popular tunes, operating a digital clock or keeping two trains from colliding at an intersecting track), but it does not provide the reader with the systematic knowledge of assembly language programming necessary for other, more general applications which may be of far greater interest to the contemporary microcomputer user.

Despite these limitations, the book does preserve some of its value as an instructional text for those wishing to learn how to program in machine language. A further peculiarity is that Foster developed his programs on MOS Technology's Kim-I, an early microcomputer incorporating a 6502 microprocessor. The book is written in a clear and comprehensible way which a beginner can easily understand. And, despite Foster's failure to present a systematic overview of 6502 architecture, individual portions of his discussion of microprocessor architecture (especially of pages and the stack) are extremely lucid.

But these strengths are not sufficient to compensate for the fact that the book is tied both to an early (and now obsolete) microcomputer system, and to an early breed of computer hobbyist. For novices wishing to learn 6502 programming, we recommend any of the fine works which appeared after the publication of Foster's book.

PROGRAMMING THE Z80

Rodnay Zaks

Level: Intermediate Rating: **95**

	Sybox	1982	Paper	
624 Pages	ISBN: 0-89588-047-4		6" x 9"	\$15.95

Mastery of assembly language programming using the Z80 instruction set is the goal of this Rodnay Zaks tour-de-force. In a sense, it is a typical Zaks treatment: it logically and meticulously dissects its subject, with the intention of laying the foundation for user mastery, not proficiency. It is a detailed, comprehensive, easy-to-follow Z80 text.

The development of the author's topic is quite simple: progressing from basic concepts, it introduces Z80 architecture, and then builds elementary assembly language programs which diagrammatically 'flow'

through the Z80 components. (Reader exercises, interspersed throughout the text, focus primarily on modifications of Zaks' programming examples.)

These elementary programs are then themselves the subject of a programming techniques discussion: simple arithmetic and logic operators are used to embellish the user's repertoire of Z80 commands. Special attention is given to the unique architectural features of the Z80 (for example, the use of the HL register as a 16-bit accumulator), and to the extended Z80 instruction set. Register management, loops and sub-routines are demonstrated using these "additional" Z80 commands, with references made to comparable 8080 and 6502 code.

The "heart" of the book is its alphabetic explanation of each Z80 instruction. A reference manual-within-a-manual, it includes the functions and format of each instruction, as well as timings, data flow diagrams and examples.

Addressing and input/output techniques form the book's "advanced" topics section, with programming examples again used to synthesize various op codes. In the addressing chapter, one of the Z80 addressing modes is demonstrated in a program which adds two blocks of data in memory; assembly code is used in the input/output section to mime the functions of a UART chip.

An "application example" chapter uses the instruction set and coding techniques to transform utilities familiar to the BASIC programmer (e.g. bubble sort) into their more efficient Z80 assembly language counterparts.

PROGRAMMING THE Z80 is the ultimate user's guide to practical manipulation of this powerful microprocessor. We recommend its purchase to anyone seriously considering Z80 assembly language programming.

PROGRAMMING THE 6502

Rodnay Zaks

Level: Intermediate Rating: **95**

Sybex 1980 Paper
386 Pages ISBN: 0-89588-046-6 6" x 9" \$13.95

Rodnay Zaks has written the gourmet guide to assembly language programming for the 6502 microprocessor, the microprocessor used in appliance computers manufactured by such companies as Apple and Commodore. As an introduction to assembly language programming for the newcomer, PROGRAMMING THE 6502 is outstanding; at the same time, it is equally valuable to microcomputer users with varying degrees of familiarity with 6502 assembly language programming. Finally, Zaks'

fourth chapter on the 6502 instruction set, which devotes at least one page to the comprehensive presentation of each instruction (including the addressing modes available with it and its effect on status flags), insures that his book will remain a valuable reference tool long after the user has become familiar with assembly language programming.

Zaks' approach to his subject is in many respects theoretical rather than practical. As late as the last chapter, Zaks notes that, "All the programs we have studied and developed so far have been developed by hand without the aid of any software or hardware resources." (343) This applies even to the "application examples" presented in chapter eight. While this may be viewed as a limitation, it is nevertheless Zaks' concern that the reader be able independently to combine the various instructions available for the 6502 microprocessor into programs capable of solving problems of his or her own choice. Throughout the text, the reader is encouraged to write his own assembly language programs (although, unfortunately, Zaks does not provide his own solutions to these). The level of discussion throughout the book is unusually thorough and comprehensive; this applies not only to Zaks' presentation of the 6502 instruction set itself, but also to the fundamentals of assembly language programming: his discussion of two's complement arithmetic, for example, is outstanding. As a result of his careful and thoughtful effort, the reader of Zaks' book is guaranteed to emerge with an outstanding foundation in assembly language programming for the 6502.

Z80 AND 8080 ASSEMBLY LANGUAGE PROGRAMMING

Kathe Spracklen

Level: Intermediate Rating: **85**

168 Pages	Hayden	1979	Paper	
	ISBN: 0-8104-5167-0		6" x 9"	\$9.70

For some personal computer fanatics the question "How does it work?" begins a lifelong quest: for a few the quest may end with an answer couched in the hieroglyphics of particle physics; for most, it will never end. All, however, seriously begin by learning to program in assembly language.

As Kathe Spracklen demonstrates, knowledge of the manner in which microprocessors manipulate binary expressions to multiply or divide is essential to successful assembly language coding. The book attempts to avoid hardware-specific discussions, concentrating instead on the mnemonics used by the Z80 and 8080. Though each microprocessor's instruction set possesses more similarities than differences

when compared to its counterparts (with the Z80 an extension of the 8080 set), the simultaneous presentation of both may prove confusing to the novice. In addition, better introductions to binary arithmetic are not difficult to find (the decimal analog is usually explored in greater depth). Finally, the approach of demonstrating the functions of different op codes within a chapter and having the reader synthesize these individual codes into a small program as a practice exercise is certainly useful; however, more sample routines with REM-type explanations included in the text might prove more beneficial. This is especially true of the abbreviated chapter on using assembly language to optimize execution time.

Despite these qualifications, Z80 AND 8080 ASSEMBLY LANGUAGE PROGRAMMING is one of the better introductions to this complex topic; its greatest utility will be found by the microcomputer user who has already had some exposure to assembly language and wishes to consolidate and expand his or her knowledge.

Z80 ASSEMBLY LANGUAGE PROGRAMMING

Lance A. Leventhal

Level: Novice Rating: **85**

Osborne 1979 Paper
629 Pages ISBN: 0-931988-21-7 7" x 9" \$16.99

The inclusion of both 6502 ASSEMBLY LANGUAGE PROGRAMMING and Z80 ASSEMBLY LANGUAGE PROGRAMMING in a single review stems from the identical format and content of these two volumes. Only the third chapters of each book, which present the instruction sets and addressing modes of the 6502 or the Z80 and compare them with similar microprocessors (in the case of the 6502, with the 6800; of the Z80, the 8080A and 8085), are different. Later chapters even treat the same programming problems, although the programs themselves use the respective instruction sets.

These books do "not discuss the general features of computers, microcomputers, addressing methods, or instruction sets."(1-1) In large measure, they do not discuss the specific features of these for the Z80 or 6502, either. Instead, these works presuppose that the reader, having read AN INTRODUCTION TO MICROCOMPUTERS, Volume 1, would find such discussions superfluous. Since he assumes that readers of varying levels of experience will learn assembly language programming from his text, Leventhal presents the most basic concepts in boldface and expands

upon these in lightface type. While this convention is applied in the first two chapters, it is used only sporadically thereafter; almost all lightface explanations from chapter three onward are inadequate.

The first two chapters, which are among the few comprehensible to a novice assembly language programmer, treat types of programming languages and the general operation of assemblers. They are, however, technically dated, and stress the unsuitability of high-level languages for microprocessor applications (1-10) and the slow execution time of resident assemblers.(2-16) The enormous improvement of microcomputer systems since 1979 makes these chapters far more intimidating than they are informative.

Leventhal's approach to teaching assembly language is similarly dated. His emphasis on microprocessor (as opposed to microcomputer) programming and the use of cross-assemblers indicates that his intended audience in 1979 consisted of engineers or programmers designing microprocessor-based products, supplemented by those few hobbyists who owned microcomputers (many of which were one-board systems). Assembly language still lay within the province of a technical elite.

The growing acceptance of microcomputers has made assembly language accessible to larger numbers of people. Typically, such users begin assembly language programming only after learning BASIC; in addition, they possess neither the technical knowledge of earlier assembly language programmers nor the same motivation for learning assembly language. For such novices, Leventhal's volumes on Z80 and 6502 assembly language programming are clearly inadequate.

For its intended audience, on the other hand, these volumes have weathered the test of time to become classics in their field. Having presented the 6502 and Z80 instruction set and addressing modes in a manner typical of a reference work rather than a tutorial, Leventhal presents the reader with programs illustrating assembly language topics. Areas covered include memory manipulation, arithmetic, loops, sub-routines and binary-hex-decimal-BCD-ASCII conversions. Leventhal saves his most detailed treatment, however for input/output programming and interrupt handling, topics which are almost exclusively within the purview of advanced programmers.

It is above all to the advanced programmer that these volumes are indispensable and to whom we recommend them highly.

Z80 INSTRUCTION HANDBOOK

Nat Wadsworth

Level: Advanced Rating: **80**

Hayden 1978 Paper
 108 Pages ISBN: 0-8104-6275-3 5" x 8" \$5.95

The author has grafted together the first chapter and 'appendix A' from his Z80 SOFTWARE GUIDE AND COOKBOOK (see our review) to create this 'spinoff' reference guide to the Z80 instruction set. A brief introduction to operation and architecture precedes a commentary on each of the Z80's basic instructions. Unlike similar treatments—notably Rodnay Zaks' PROGRAMMING THE Z80—this survey of individual instructions is not alphabetized; the fact that an alphabetic cross-reference is provided in an appendix does little to increase the book's utility as a quick source manual. But more could have been learned from Mr. Zaks' book. Instead of providing the three-digit octal and two-digit equivalent hexadecimal notation for each op code (which should have been relegated to an appendix), a more standardized treatment of status flags would have been in order.

Wadsworth justifiably assumes a prior knowledge of Z80 assembly language programming; this book is not intended as a general introduction to the topic, but rather as "a programming and program assembly aid." To a limited extent, it succeeds in providing a handy reference to Z80 mnemonics and usage. If a programmer is serious enough about his craft to be coding consistently in Z80 assembly language, however, we recommend that he consider any of the more comprehensive, highly rated treatments of this subject.

Z80 MICROPROCESSOR PROGRAMMING & INTERFACING

Joseph C. Nichols, Elizabeth A. Nichols, et.al.

Level: Intermediate Rating: **70**

Sams 1979 Paper
 302 Pages ISBN: 0-672-21609-4 5" x 9" \$12.95

Rather than teaching Z80 assembly language programming in general, the authors have written this volume for those interested in learning to program the single-board SGS-ATES Nanocomputer. The portions of the text which deal specifically with the Nanocomputer (and especially

the chapter on its operation on pages 47-77) make the book of correspondingly less interest to owners of other computers with Z80 microprocessors.

The book's presentation of many of the prerequisites of assembly or machine language programming—such as a moderately comprehensive discussion of binary arithmetic—is inadequate. In addition, its organization is superficial; while similar kinds of instructions are all grouped together in single chapters (such as those dealing with addressing modes, jumps, calls and returns, and logical instructions), the need to develop simple programs leads the authors to present numerous op codes before thoroughly explaining them.

Despite this, the book presents a complete if succinct overview of assembly and machine language programming for the Z80 microprocessor. Once the reader has learned how to operate the Nanocomputer, he or she is encouraged to attempt the series of "experiments" which conclude each chapter, thus consolidating familiarity with the material through actual practice. At the same time, however, the authors do not encourage the reader to develop his own assembly language programs.

At its best, Z80 MICROPROCESSOR PROGRAMMING provides an uninspiring introduction to Z80 assembly language. At its worst, it is a dated treatment of programming for an obsolete microcomputer system.

Z80 SOFTWARE GOURMET GUIDE AND COOKBOOK

Nat Wadsworth

Level: Intermediate Rating: **75**

	Hayden	1979	Paper	
322 Pages	ISBN: 0-8104-6276-1		5" x 7"	\$15.95

Each cookbook in this series lists assembly language recipes for one of the four most popular microprocessors: the Z80, 8080, 6502 and 6800. Though authors vary, the topic sequence and chapter structure of each text are identical: the "core" module of every section is a brief reference guide to a particular microprocessor instruction set, supplemented by subroutine listings that can be incorporated into user code. Nat Wadsworth's Z80 SOFTWARE GOURMET GUIDE AND COOKBOOK, the most comprehensive of these treatments, is the flagship of the Hayden Gourmet Guide Series.

Wadsworth first presents an overview of Z80 architecture which includes descriptions of register banks and CPU flags and then follows with an op code listing of each of the more than two hundred Z80 instructions. Because the explanation of the Z80 instruction set is in-

tended as part of a general introductory treatment, the instructions themselves are grouped by function rather than alphabetically. This severely restricts the utility of this chapter as a reference manual, despite its cross-indexing appendix. Zilog mnemonics are used exclusively, accompanied by three-digit octal and two-digit hexadecimal codes. Though some indication is made of the effect of most instructions on status flags, the lack of any standardized treatment makes spot-checking during coding rather tedious.

The Z80 SOFTWARE GOURMET GUIDE AND COOKBOOK is actually less an introduction to the Z80 architecture or instruction set than a source book for assembly language subroutines. Here it succeeds. Utility routines include general routines for clearing a section of memory, passing parameters or programming time delays.

Each program or routine contains self-explanatory internal comments as well as ancillary text provided by the author. Stack operations (the Z80 adds to its stacks in descending address order) and input/output processing are also covered. Other general subroutine topics include search and sort routines (fixed and free format tables), conversion routines (Baudot to ASCII to decimal to binary), floating point routines and decimal arithmetic routines. The final Wadsworth gem is the assembly code for a "game of skill and chance" called "Space Capture."

The Z80 SOFTWARE GOURMET GUIDE AND COOKBOOK, like the other offerings in the Hayden "Gourmet Guide" series, is best suited for the experienced assembly language programmer who needs quickly and efficiently to expand his or her library of subroutine code. It falls short of providing the beginner with an adequate introduction to Z80 assembly language programming, but certainly lives up to its recipe-laden "cook-book" title.

Z80 USERS MANUAL

Joseph J. Carr

Level: Advanced Rating: **45**

Reston 1980 Paper
326 Pages ISBN: 0-8359-9516-X 6" x 9" \$15.95

A reader of any microprocessor user's manual naturally expects it to treat two subjects above all—microprocessor architecture and assembly language programming. But Z80 USERS MANUAL obscures these topics by using the Z80 as the focal point of two case studies in applied mathematics.

The first portion of the book, which appears to be devoted to Z80 architecture, is simultaneously a study in integer division and truncation. More precisely, the author shows how a detailed and comprehen-

sive discussion of such topics as Z80 architecture, interfacing, interrupt handling and the use of support chips can be reduced in scale so that it becomes quite uninformative. This progressive truncation of otherwise useful information reaches its climax in the last chapter of the book's first section, which devotes five pages to the Z8 and Z8000.

The second portion of the book, on the other hand, uses the Z80 instruction set to illustrate multiplication or geometric progression. The built-in instructions available on the Z80 have been combined with its ten addressing modes to produce 405 individual instructions, each of which is listed in the last half of the book. Most works present this same information in a two or three page table which, while crammed with detail, is also easy to use. Z80 USER'S MANUAL, by treating "LD BC, nn" (instruction 166 on page 222) as distinct from "LD DE, nn" (instruction 167 on pages 222-223), which in its turn is distinct from "LD DE, (nn)" (instruction 171 on page 224), can claim no such distinction.

We highly recommend Z80 USERS MANUAL as a model for all those who are concerned with presenting information in as disorganized and incongruous a format as possible in order to render it meaningless. For those interested in either the Z80 microprocessor or in Z80 programming, however, we recommend some other work.

Z8000 ASSEMBLY LANGUAGE PROGRAMMING

Lance A. Leventhal, Adam Osborne and Chuck Collins

Level: Advanced Rating: **90**

	Osborne	1980	Paper
898 Pages	ISBN: 0-931988-36-5	7" x 9"	\$19.99

The emergence of a new state-of-the-art technology, in the shape of the 16-bit microprocessor, is reflected by the publication of volumes on the Z8000 and the 68000 in the Osborne assembly language programming series. But while the content of these books is new, they continue to follow closely the format adopted by Leventhal in his works on assembly language programming for 8-bit microprocessors; in addition, they are still ostensibly addressed both to beginning and experienced assembly language programmers. While both books are definitive reference works for advanced programmers, they are less responsive to the needs of novice programmers than were the volumes devoted to the 8-bit microprocessors.

This growing disparity results from the use of the same techniques to treat a more complex topic; the intrinsic difficulty of understanding assembly language programming has been magnified by the addition of

new features, such as the 68000's absolute short addressing mode or the Z8000's segmented and non-segmented memory addressing modes. The volume on the 68000 shows the greatest modification of these volumes' standard format in order to conform to the needs of the novice, although this is largely superficial. Rather than presenting the entire instruction set and all addressing modes at the outset, the instruction set has been moved to the book's last chapter; but even here instructions are not adequately explained as they are presented.

To argue that these volumes are unresponsive to the needs of the novice, however, is partly unfair, since the meaning of "novice" has changed since the books were published. These were among the first books on assembly language programming for the 16-bit microprocessors; in fact, very few others have appeared since. But the dilemma of being at the forefront of new technological developments is that new audiences which require a substantially different kind of treatment rapidly emerge. A "novice" in 1980 was a computer enthusiast who purchased a one-board 16-bit system from a semiconductor manufacturer; in 1983, he is more likely to be the owner of a full microcomputer system who simply wants to take a stab at assembly language programming.

But although these volumes are already becoming dated for one audience, they will continue to preserve their value for experienced programmers as long as the 68000 or the Z8000 themselves are not obsolete. They serve as a ready source of assembly language programs and subroutines for a wide range of applications. In addition, for those who possess the necessary background, they provide an extremely detailed and comprehensive treatment of 68000 and Z8000 programming. Typically, the most thorough treatment is given to advanced programming topics. These include passing parameters to and from subroutines, interfacing with peripheral devices operating at various speeds, programming external support chips and handling interrupts; in addition, the volume on the Z8000 treats memory management with the Z8010 Memory Management Unit. Finally, both volumes feature an excellent discussion of program development and design which illustrates sound programming techniques by presenting concrete examples of code to supplement the text.

Advanced programmers will find these two volumes indispensable goldmines both of programs and information. Until more accessible treatments become available, beginners too will have to rely on them, although we would recommend that they supplement volumes in the Osborne series with any of a number of highly rated works on programming 8-bit microprocessors, in order to gain an understanding of the underlying principles of assembly language programming.

6502 ASSEMBLY LANGUAGE PROGRAMMING

Lance A. Leventhal

Level: Advanced Rating: **85**

Osborne 1979 Paper
629 Pages ISBN: 0-931988-27-6 7" x 9" \$16.99

See our review of Z80 ASSEMBLY LANGUAGE PROGRAMMING on page 75.

6502 SOFTWARE GOURMET GUIDE & COOKBOOK

Robert Findley

Level: Intermediate Rating: **75**

Hayden 1979 Paper
204 Pages ISBN: 0-8104-6277-X 6" x 8" \$12.95

See our review of Z80 SOFTWARE GOURMET GUIDE AND COOKBOOK on page 78.

6800 SOFTWARE GOURMET GUIDE & COOKBOOK

Robert Findley

Level: Intermediate Rating: **75**

Hayden 1976 Paper
211 Pages ISBN: 0-8104-6281-8 6" x 8" \$12.95

See our review of Z80 SOFTWARE GOURMET GUIDE AND COOKBOOK on page 78.

THE 68000: PRINCIPLES AND PROGRAMMING

Leo J. Scanlon

Level: **Advanced** Rating: **55**

Sams 1981 Paper
238 Pages ISBN: 0-672-21853-4 5" x 8" \$14.95

The general consensus is that the new 16-bit microprocessors will enormously facilitate the speed and ease of assembly language programming once the programmer has mastered their more complex instruction sets and addressing modes. By some curious reversal of this conventional wisdom, Leo J. Scanlon, in his introduction to assembly language programming for Motorola's MC68000 microprocessor, appears to assume that if the process of programming is easier, then learning how to program the 68000 must be easier as well. Scanlon goes even further in assuming that the principles of 68000 assembly language programming are so accessible that he does not make them the sole focus of his short (204 page) book. Also included are a background chapter on the development and significance of the 68000, a discussion of cross macro assemblers and system development support products (for those readers using other computers as development systems for 68000-based software) and an examination of eleven 68000 support chips.

The heart of the book, however, does concern assembly language programming. Aside from a brief discussion of the format of assembly language coding and of pseudo-op codes, the whole of 68000 assembly language programming is treated in the third chapter (pp. 42-108), which presents the reader, in rapid fire succession, with all fourteen addressing modes and all 56 basic instruction types available on the 68000. Having completed this bombardment, Scanlon moves on to two chapters with applications examples (mathematical routines, 68000 MULS and MULU instructions, lists, and look-up tables) which are enormously helpful in consolidating the reader's grasp of material which he or she has failed to assimilate in the first place.

While Scanlon's introduction downplays the expertise necessary to derive some benefit from reading his book ("just a basic understanding of the rudiments of computer architecture...and familiarity with some type of assembly language"—p.5), his emphasis on development systems, as well as his hasty treatment of 68000 programming, indicate that this is a work intended for the professional programmer. Any reader who is not thoroughly conversant with some form of assembly language programming will certainly not learn how to program the 68000 by reading this work.

68000 ASSEMBLY LANGUAGE PROGRAMMING

Gerry Kane, Doug Hawkins and Lance A. Leventhal

Level: Advanced Rating: **90**

	Osborne	1981	Paper	
596 Pages	ISBN: 0-931988-62-4		7" x 9"	\$16.99

See our review of Z8000 ASSEMBLY LANGUAGE PROGRAMMING on page 80.

8080 SOFTWARE GOURMET GUIDE & COOKBOOK 2nd Edition

Robert Findley and Raymond Edwards

Level: Intermediate Rating: **75**

	Hayden	1976	Paper	
220 Pages	ISBN: 0-8104-5280-X		6" x 8"	\$12.95

See our review of Z80 SOFTWARE GOURMET GUIDE AND COOKBOOK on page 78.

8080/Z80 ASSEMBLY LANGUAGE Techniques For Improved Programming

Alan R. Miller

Level: Advanced Rating: **95**

	Wiley	1981	Paper	
319 Pages	ISBN: 0-471-08124-8		7" x 10"	\$10.95

Alan Miller, whose career deftly combines the roles of software editor for INTERFACE AGE and Professor of Metallurgy at the New Mexico Institute of Mining and Technology, has just as deftly combined two distinct topics in his 8080/Z80 ASSEMBLY LANGUAGE: a definitive explanation and reference source of 8080 and Z80 subjects, and an introduction to advanced assembly language programming techniques. This is one of the best treatments of both topics currently available.

Some familiarity with assembly language programming, though not essential, will definitely assist the programmer/reader. Miller's introduction to the 8080 and Z80 instruction set is more of a review than a plodding narrative designed for novices. Even given this condensed

treatment, the presentation of first 8080 and then Z80 instructions and flag interpretations is more comprehensible than the simultaneous approach employed by Kathe Spracklen in Z80 AND 8080 ASSEMBLY LANGUAGE PROGRAMMING (see our review). In the context of the more powerful Z80 set, special operations such as relative jumps are explained.

Base conversions, two's complement arithmetic and logical operations preface a chapter on "the stack:" the author emphasizes its role in saving and restoring register data and storing return addresses when subroutines are called. Preliminary assembly language discussions end with considerations of polling, interrupt handling and parity checking techniques. That this is not a typical text is evident in the originality of the coding examples used to demonstrate these techniques: one program creates a purely interrupt-driven keyboard (a system operating under CP/M interprets every key entry as a hardware interrupt and immediately loads keyboard data into a buffer). All of this general background material is logically developed and perfectly paced.

The advanced programming techniques section which follows includes the use of macros in assembly language programs (emulating Z80 instructions with an 8080 CPU is a one-of-a-kind treatment). Like many BASIC manuals which develop an application in order to reinforce retention of commands and introduce coding techniques, Miller develops a system monitor from scratch, explaining instructions and techniques at each phase of its construction.

Assembly language routines for ASCII/binary character conversions are covered, as are the more esoteric topics of program storage on paper and magnetic tape (Miller's work demands a paper feed of programs to a Z80 CPU which front-ends a larger mainframe). The author's versatility in interfacing assembly language programs with CP/M, mentioned earlier, is further demonstrated in a section which treats linking these programs to other CP/M subroutines.

An appendix lists the 8080 and Z80 instruction sets alphabetically and numerically (by hex), and additionally cross-references them. A detailed description of each 8080 and Z80 instruction is provided in a distilled version of Wadsworth's Z80 INSTRUCTION HANDBOOK (see our review).

For the advanced programmer, Alan Miller's 8080/Z80 ASSEMBLY LANGUAGE admirably achieves the author's goals of providing an 8080/Z80 sourcebook combined with a treatment of advanced coding techniques.

THE 8086/8088 PRIMER 2nd Edition

Intro. to Architecture, System Design & Programming

Stephen P. Morse

Level: Advanced Rating: **90**

	Hayden	1982	Paper	
276 Pages	ISBN: 0-8104-6255-9		6" x 9"	\$10.95

Novices will undoubtedly find the g-forces generated by Stephen Morse's instant acceleration into advanced 8086/8088 topics injurious to their intellectual health. Only programmers inured to the rigors of 8-bit microprocessor assembly language coding will truly appreciate the 16-bit discourse of THE 8086/8088 PRIMER.

Even the author's introduction is crammed with information about Intel microprocessors: the evolution of the 8008 into the 8086/8088 is treated in terms of the increased versatility of each new generation of microprocessors. Specifically, the additional addressing modes, multi-processor capability and faster data transfer rates of the 8086 are highlighted, as is the 8080-data transfer compatibility of the 8088.

Detailed discussions of 8086/8088 architecture and system design follow. Programmers new to Intel's 16-bit machine organization will notice the unique role of segment registers in building 20-bit words. Direct and indirect addressing modes are considered as a preface to the actual use of these modes by the 8086/8088 instruction set. An extensive analysis is conducted of each type of instruction, adhering to the traditional progression from arithmetic instruction to flag interpretation. The author does an exceptional job with the 8086 interrupt mechanism. Morse, one of the designers of the 8086, ironically mentions some design errors discovered in the original microprocessor (the chip would accept interrupts immediately after executing an instruction that moved a new value into the stack segment register).

A system design section next treats the 8086 family of components which together form a complete system. These include the 8259A, 8282, 8286 and 8288. A parallel discussion covers the circuit components used with the 8088.

The remaining half of Morse's text provides a survey of 8086 assembly language programming, PL/M and Pascal. Again, each of these chapters builds upon the author's assumption of prior reader experience in assembly and high-level language programming. ASM-86 is described, with emphasis on the new segmentation-definition statements (SEGMENT, END, ASSUME, ORG). The 8086 version of the Intel proprietary language designed primarily for microprocessor applications, PL/M-86, is surveyed. Its structure and syntax serve as interesting foils for the Pascal presentation which concludes the book.

THE 8086/8088 PRIMER is an expert analysis of this complex microprocessor by one of its designers. It provides first-hand insights into a microprocessor whose application versatility is still being explored.

8086/8088 16-BIT MICROPROCESSOR PRIMER

Christopher L. Morgan and Mitchell Waite

Level: Intermediate Rating: 100

	Byte	1982	Paper
355 Pages	ISBN: 0-07-043109-4	8" x 10"	\$16.95

The opening pages of this introduction to Intel's 8086/8088 16-BIT MICROPROCESSOR PRIMER fail to address the question of whom the book was written for or why anyone would want to read it; this is an unfortunate omission on the part of Morgan and Waite. Their book is a literate, enjoyable introduction to state-of-the-art microprocessor technology for anyone who both enjoys reading and is interested in microcomputers. The authors have done an admirable job of presenting extremely complex material in a comprehensible fashion without resorting to gross oversimplification. In addition, potential purchasers of 8086 or 8088-based microcomputer systems (such as the IBM Personal Computer) can benefit from reading the book for the light it sheds on the capabilities of Intel's 16-bit microprocessors. Owners of these machines can use this book as a first step toward fully exploiting their computers' capabilities.

Morgan and Waite's overview of the 8086 and 8088 microprocessors (the latter being identical to the 8086, except that it contains an 8-bit data bus instead of the 8086's 16-bit bus) is a thorough one which covers system architecture, methods of addressing memory (a particularly important topic, since the 8086 or 8088 in effect combines two distinct subprocessors, the Execution Unit and the Bus Interface Unit), assembly language programming, the capabilities of these chips, and a comparison of them to both the traditional 8-bit microprocessors and to other 16-bit microprocessors—notably Zilog's Z8000 and Motorola's MC68000.

Intel has designed the 8086 and 8088 as the central microprocessor for a series of chips which, when used together, can enormously increase the total computing power of a system. Morgan and Waite also survey these other microprocessors, including the 8087 Numeric Data Processor, the 8089 Input/Output Processor, and a variety of support chips (such as the 8284 Clock Generator, the 8288 Bus Controller, or the 8237

Programmable Direct Memory Access Controller). Their discussion of these chips largely follows the same format as the chapters which treat the 8086/8088 microprocessors.

Perhaps the greatest service which Morgan and Waite's book provides is the demystification of a complex technological development. Other authors, for example, feel that 16-bit microprocessor architecture is simply too complex to allow a relatively inexperienced user to attempt assembly language programming. Although this book is not a specialized assembly language manual, Morgan and Waite nevertheless show that while programming the 8086/8088 is more complex than programming an 8-bit microprocessor, the greater flexibility of its instruction set (which includes multiplication, division, and string handling capabilities) may even make 8086/8088 assembly language programming easier once its basic principles are understood. More broadly, in writing this book, Morgan and Waite have made an important contribution toward humanizing a technology which others prefer to treat as distant and inaccessible to the non-specialist.



3

Operating Systems & Hardware Design

Orchestrating the complex functions of individual microcomputer components is the one master program known as the "operating system." This chapter evaluates books on a number of such systems: the most popular, and most reviewed, is CP/M (Control Program for Microcomputers). Both introductory and advanced works on CP/M are examined. Other operating systems include the USCD p-System, which is usually used in conjunction with a Pascal compiler, and UNIX, an operating system developed by Bell Laboratories for use with the C programming language. Apple or TRS-80 users should consult the sections devoted to these machines for works on AppleDOS or TRS-80 DOS.

Hardware design normally falls within the province of the hobbyist interested in interfacing a microcomputer to a variety of peripheral devices. While microcomputer-specific interfacing is treated in Chapter Six, this section does include a number of fascinating books by Steve Ciarcia, the author of the monthly column "Ciarcia's Circuit Cellar," in BYTE magazine. Also included are books on "breadboarding," a technique for testing elementary circuit design.

BUILD YOUR OWN Z80 COMPUTER **Design Guidelines & Application Notes**

Steve Ciarcia

Level: Advanced Rating: 95

Byte 1981 Paper
332 Pages ISBN: 0-07-010962-1 8" x 11" \$15.95

Fascination with computer technology has led many people to fantasize about building their own computers virtually from scratch. For

some, the most acceptable starting point for this process is manufacturing their own chips. Those enthusiasts with at least one foot firmly grounded in reality, however, can realize their dream much more easily by following the detailed, step-by-step instructions offered by Steve Ciarcia in **BUILD YOUR OWN Z80 COMPUTER**.

Ciarcia's minimal Z80 microcomputer system design includes a power supply, the Z80 microprocessor itself, 1K of manually programmed EPROM (Electrically Programmable Read-only Memory), 1K of programmable memory (RAM), a 19-Key hexadecimal keyboard, a six-character LED display, a UART (Universal Asynchronous Receiver Transmitter) for serial input/output, and seven input/output ports. In addition to providing the reader with the hardware specifications, Ciarcia also includes an extensive discussion of the ZAP (Z80 Applications Processor) operating system supplemented by a machine and assembly language listing of the monitor program itself. Since Ciarcia's design emphasizes flexibility and expandability, the final two chapters of his book treat enhancements to the system (digital to analog converters, analog to pulse width converters and a CRT).

The architecture of his ZAP computer has been both thoughtfully and thoroughly designed by Ciarcia, who emphasizes both low cost and overall effectiveness in selecting the system's individual components. This fine attention to detail begins with the power supply (which is designed to protect the system against both low voltage and overvoltage), and extends to the inclusion of such features as a reset circuit and the use of buffering to insure that the system bus has adequate power to drive peripherals. Aside from suggesting the components which should go into making the system, Ciarcia includes detailed directives for testing them while the system is still being assembled.

While the book is detailed, it is also highly specialized; Ciarcia makes no attempt to offer explanations in terms which might be even vaguely comprehensible to those without substantial expertise in electronics and computer technology. It is above all for those specialized hobbyists who wish to take the next step beyond assembling a microcomputer from a kit that Ciarcia's book is extremely valuable.

CIARCIA'S CIRCUIT CELLAR, VOL.I

Steve Ciarcia

Level: Advanced Rating: 95

Byte 1979 Paper
127 Pages ISBN: 0-931718-07-4 8" x 10" \$8.00

CIARCIA'S CIRCUIT CELLAR, VOL.2

Steve Ciarcia

Level: Advanced Rating: **95**

	Byte	1981	Paper	
220 Pages	ISBN: 0-07-010963-X		8" x 10"	\$12.95

CIARCIA'S CIRCUIT CELLAR, VOL.3

Steve Ciarcia

Level: Advanced Rating: **95**

	Byte	1982	Paper	
228 Pages	ISBN: 0-07-010965-6		8" x 10"	\$12.95

Among those hobbyists who are interested in interfacing their microcomputers to a variety of exotic peripherals and gadgets, Steve Ciarcia has become a household name. These three volumes reprint Ciarcia's articles in *BYTE* magazine from September 1977 through June 1980 in an accessible and convenient form.

Although interfacing is generally a topic of concern primarily to the specialized, advanced and technically well-grounded hobbyist, these compendia also have much to offer to the less advanced user. Part of Ciarcia's purpose in writing these articles is to equip his readers with a more detailed and thorough understanding of computer technologies; Ciarcia feels that interfacing projects provide the most effective means of consolidating the learning process by giving knowledge a real and practical form. But this should not deter the computer enthusiast with absolutely no interest in performing the experiments from reading and enjoying Ciarcia's articles. His discussions of memory mapped input/output (vol. I, pp. 7-11), the Intel 8086 (vol. II, pp. 120-127), and EAROM or Electrically Alterable Read-Only Memory (vol. II, pp. 129-136)—to cite just a few outstanding examples—provide the reader with information in a succinct yet provocative form which transcends the treatment of similar topics in more standard and conventional works.

The non-hobbyist who nevertheless puts his or her microcomputer to a wide variety of uses will at some point confront a situation in which either a peripheral is not available in ready-assembled form for a particular application or it is far too expensive. For these cases, Ciarcia's articles provide a virtual catalog of inexpensive devices which the user can buy and connect to the microcomputer. Since Ciarcia knows that many people are intimidated by do-it-yourself electronics, he notes when the instructions accompanying these components will be comprehensible to the novice.

Finally, the interfacing projects themselves, which range from home control systems (for security, control of appliances, etc.) to methods for

enhancing a microcomputer's capabilities or efficiency (a dual voltage converter to provide adequate power for peripherals) to games (a microcomputer version of Simon), are far less formidable than they might appear on first sight. Although the projects do presuppose a certain amount of technical knowledge, Ciarcia's detailed diagrams, photographs and relatively full explanations permit less experienced users to perform and learn from these experiments. As a final touch, Ciarcia usually provides the software necessary to control the peripheral device in the form of BASIC programs, along with suggestions for refining them.

THE CP/M HANDBOOK

With MP/M

Rodnay Zaks

Level: Novice Rating: **95**

Sybex 1980 Paper
320 Pages ISBN: 0-89588-048-2 6" x 9" \$14.95

Rodnay Zaks' introductory guide to CP/M, like many of his other works, underscores his reputation as a leading author of books on microcomputers. Zaks offers a thorough and detailed overview of CP/M which is as accessible to the novice as it is useful to the more experienced microcomputer user. Once the reader has become familiar with many of the basic features of CP/M, the CP/M HANDBOOK retains its value as a reference manual. In Chapter Six, Zaks has included an alphabetical reference guide to CP/M commands which is both more detailed and more convenient than those provided in manufacturers' manuals. In addition, the commands used in two individual transient programs, PIP (which copies files from one device to another) and ED (the CP/M text editor), are summarized in five appendices.

The single drawback to Zaks' otherwise excellent book is the first chapter, in which Zaks presents a broad overview of CP/M. His explanations are so succinct, and the number of commands so voluminous, that even an experienced computer user can feel bombarded with abstract and apparently trivial details. If this is the case, however, the reader can and should simply skip the chapter, since the remainder of the the book presents essentially the same material in a gradual and thorough form which is much easier on his or her memory.

We recommend this Zaks title as the single best introduction to CP/M, and as the single most convenient reference work for operating system commands and syntax. THE CP/M HANDBOOK is an indispensable part of every CP/M user's library.

CP/M PRIMER

Stephen M. Murtha & Mitchell Waite

Level: Novice Rating: **90**

Sams 1980 Paper
 91 Pages ISBN: 0-672-21791-0 9" x 11" \$14.95

Probably the most popular of the Waite primers, CP/M PRIMER owes its fourth printing to the apparent ease with which it divulges the intricacies of this near-ubiquitous operating system. A conversational tone combines with humorous, memorable illustrations to guide the reader through the progressive unravelling of CP/M's use.

Gary Kidall's initial forays into PC/M at Intel and his eventual founding of Digital are treated in early historical notes. The book then structures its approach into four partitions: the installation of CP/M; its use through commands and utilities; the CP/M text editor, ED; and ASM, the CP/M assembler. One aspect of CP/M is explained, a brief example of its use given, followed by a further explanation of the actual example. This teaching technique pervades the text, building toward the assembly language examples used to demonstrate ASM. The book is most effective when used in combination with an operating version of CP/M; its true effectiveness, however, is revealed by its ability to impart a hands-on quality of understanding without necessitating the actual keying of commands (we read this book prior to booting CP/M, and believe our success in immediately using CP/M was primarily due to the clarity of its instructions).

An idea implemented in CP/M PRIMER, borrowed from the programming templates used in mainframe environments, is the CP/M reference card which detaches from the book's spiral binding to provide quick, easy access to naming conventions. Of equal value are the appendices which further explore CP/M architecture and list sources of CP/M-compatible software.

CP/M PRIMER is a well-written, enjoyable introduction to the de facto standard in microcomputer operating systems.

CP/M REVEALED

Jack D. Dennon

Level: Advanced Rating: **80**

Hayden 1982 Paper
 180 Pages ISBN: 0-8104-5204-9 7" x 10" \$13.95

This is not a treatment of CP/M for novices. The experienced 8080/Z80 assembly language programmer whose interests mandate a working

knowledge of this complex, powerful operating system will benefit most from Jack Dennon's text: a series of assembly language program exercises are designed to familiarize the user with CP/M architecture and functions.

As revealed in introductory chapters, the emphasis is on imparting an immediate operational facility with CP/M, rather than providing any historical or conceptual background material. (For the literary antithesis of this work, see *THE CP/M PRIMER*.) Within the first few chapters, the Text Editor and "Dynamic Debugging Tool" (DDT) are used in machine examples to write and debug assembly language programs. Though the only documentation provided with these programs are the author's internal comments, the experienced programmer will find Dennon's approach to exploring CP/M fascinating: the versatility of the 8080 instruction set is used to dissect the operating system. One program counts all the occupied spaces in the disk directory. Another displays the actual CP/M memory map. Not only does the user parenthetically learn about the nature of CP/M editing facilities and formatting conventions, but at the end of most chapters, he or she discovers a useful assembly language utility residing on the CP/M exercise diskette.

Specific advanced topics which are approached in this original fashion include the direct modification of machine code ("patching"), CP/M's special use of the record block table and its image in memory, and random data access methods. Two special utility programs are included on their own merits rather than for some didactic purpose: *COMMON* takes advantage of the control point facility of CP/M 2.2 to gain immediate read access to CP/M commands; *RESTORE* allows an extent-by-extent examination of a diskette file.

Two qualifications should be made of Dennon's advanced discourse on CP/M. The first is that the book, despite its fascinating detail, lacks a comprehensive explanation of the relation of CCP, BDOS, CBIOS and the TPA to one another in memory. The second is that redundancy in a book for advanced users tends to be especially irritating: the fact that CP/M files are not "threaded" and that disk space for a file is correctly reserved only when it is properly closed are topics which are repeated in the course of twenty pages of text.

CP/M REVEALED is nevertheless a valuable, novel introduction to this popular operating system for an elite audience of experienced assembly language programmers.

CP/M SIMPLIFIED

Jeffrey R. Weber

Level: Novice Rating: **80**

Weber Systems 1982 Paper
 316 Pages ISBN: 0-938862-04-9 5" x 10" \$13.95

This book presents an introduction to CP/M (up to version 2.2) and MP/M, and briefly discusses CBASIC (a compiler version of BASIC). The book assumes that the reader has no prior knowledge of computers, operating systems or programming. The explanations which Weber provides are usually well-written and clear, so that both the novice and experienced microcomputer user can benefit from them.

The book has only two drawbacks. The first is the inclusion of an entire chapter (pp. 17-29) in order to insert an advertisement for the software manufactured by Weber Systems Incorporated. The second is that CP/M SIMPLIFIED is at times CP/M disorganized; Weber frequently mentions a CP/M command in its proper place (at least from the viewpoint of a well-organized presentation), only to defer discussion of it until a later chapter. Although this does nothing to increase the clarity of the initial discussion, the fuller presentation of these commands in later chapters does succeed in clearing up any confusion the reader might have.

GETTING ACQUAINTED WITH MICROCOMPUTERS

Louis E. Frenzel, Jr.

Level: Advanced Rating: **80**

Sams 1978 Paper
 288 Pages ISBN: 0-672-21486-5 6" x 9" \$9.95

This book is written for the advanced beginner who, having learned about transistor radios by building one, now turns his attention to microcomputers. Mr. Frenzel's employer at the time of the book's publication was the Heath Company, one of the largest manufacturers of do-it-yourself microcomputer kits. GETTING ACQUAINTED WITH MICROCOMPUTERS is an effective presentation of the Frenzel/Heath philosophy of learning by building.

After the initial obligatory chapters treating microcomputer fundamentals (with an understandably atypical emphasis on microprocessors), the reader is instructed to purchase any one of a number of microcomputer "trainer" kits containing all the nuts and bolts required

to assemble an actual computer. Frenzel leaves the specifics of construction to the reader by providing general programming problems to be used as familiarization exercises on the functioning microcomputer.

While certainly still a valuable introduction for a special breed of computer hobbyist, GETTING ACQUAINTED suffers from that debilitating disease that afflicts much microcomputer literature—technical obsolescence. Kits mentioned by Frenzel as being available in 1978 (including Heath's own) have since been withdrawn from the retail market. Since Frenzel's book, however, remains one of the few breadboarding titles currently available, this limitation is easily overlooked by the microcomputer builder/enthusiast.

HOW TO GET STARTED WITH CP/M (Control Program for Microcomputers)

Carl Townsend

Level: Novice Rating: **45**

	Dilithium	1981	Paper	
158 Pages	ISBN: 0-918398-32-0		6" x 9"	\$13.95

The author's avowed intention of producing a CP/M manual for novice computer users is a noble one—but, unfortunately, it is not well served by the text of this book. Half of HOW TO GET STARTED is actually an appendix of CP/M hardware and software distributors; the remaining half is a hybrid introduction to personal computers and CP/M guide. Besides the incongruity of learning about software for the first time on page four and loading CP/M on page twenty-three, the book never adequately addresses CP/M architecture. Furthermore, the brief discussion of the STAT, PIP, and ED utilities is interrupted by a section outlining the thirteen rules for the care and handling of floppy disks. The material presented is a grossly simplified version of the CP/M documentation accompanying every purchase; it does accomplish the enormously difficult task of not improving upon VisiCorp's treatment of this complex operating system. CP/M is a valuable, necessary part of the microcomputer market; its effective use cannot possibly be gleaned from this atypically poor offering from Dilithium Press.

AN INTRODUCTION TO MICROPROCESSORS

Experiments in Digital Technology

Noel T. Smith

Level: Advanced Rating: **80**

	Hayden	1981	Paper	
176 Pages	ISBN: 0-8104-0867-8		9" x 11"	\$10.95

Subtitles play an inordinately important role in microcomputer literature: they frequently represent the first clue in deciphering the text's true audience. AN INTRODUCTION TO MICROPROCESSORS, unlike most other offerings on this same topic, is actually a lab manual for the microprocessor enthusiast. (A microprocessor enthusiast is that rare individual who, feeling guilty that he assembled his personal computer from a kit, now wants to make amends by really starting from scratch.)

Previous electronic and microcomputer experience is assumed by Mr. Smith, along with a willingness to use one of the text's recommended breadboards to complete the microprocessor experiments. The reader actually builds logic gates, testing advanced circuitry in the same manner as integrated circuit designers. This instructing technique is reserved for the electronics hobbyist, and should not be confused with any of the more accessible treatments of this same topic.

INTRODUCTION TO MICROPROCESSORS is a circuit cellar tour of integrated circuit functions; it is a specialized approach for a specialized audience of talented electronics tinkerers.

INTRODUCTION TO THE UCSD p-SYSTEM

Charles W. Grant and Jon Butah

Level: Novice Rating: **95**

	Sybex	1982	Paper	
300 Pages	ISBN: 0-89588-061-X		7" x 9"	\$14.95

This tutorial and reference guide provides the reader with an overview of the UCSD p-SYSTEM, the operating system which frequently implements versions of the Pascal programming language (and sometimes FORTRAN and BASIC as well). Like some of the other fine guides to software published by Sybex, INTRODUCTION TO THE UCSD p-SYSTEM offers a detailed, well-organized and comprehensive presentation of the features of this system which guarantees that the reader will rapidly become proficient in its use.

One novelty of the p-System is that it is a menu-oriented operating system. When first booted, the microcomputer will be in command mode, which allows the reader a choice of entering the other modes available to the operating system; these include the filer, edit, compile, link, run and execute, assemble, and debug modes. After presenting a two-chapter overview of the system which very effectively lays the groundwork for the book's later in-depth analysis of the p-System's features, Grant and Butah survey all but two of these command modes; they omit the p-System assembler—presumably because it is machine-specific—and the debug facility, which had not yet been implemented at the time the book was written. In addition, several secondary command modes exist in some versions of the p-System (User Restart, Initialize, Halt and Swap), although the reader is simply referred to the documentation provided with the operating system for an understanding of these.

After their initial introduction to the p-System, the authors devote lengthy chapters to the filer and edit modes, as well as to the compile and run or execute modes. Convenient reference guides to filer and edit commands are included at the end of their respective chapters. In the final chapter, the authors confront one of the chief limitations of the p-System—the limited memory which is available for storing programs and data—and provide the more-or-less advanced Pascal programmer with a detailed understanding of the use of the linker and the swap facility to overcome this shortcoming. Appendices deal with initial system configuration, compiler and run-time error messages, and the manipulation of the Pascal system library.

As a result of both the growing popularity of Pascal and the tendency toward larger microcomputer memories, the p-System will probably become more and more commonly used as an operating system. **INTRODUCTION TO THE UCSD p-SYSTEM** offers the most competent and detailed treatment of it to date.

MASTERING CP/M

Alan R. Miller

Level: Advanced Rating: **95**

	Sybex	1983	Paper	
398 Pages	ISBN: 0-89588-068-7		6" x 9"	\$15.95

Curiously, Miller chooses to begin his introduction to **MASTERING CP/M** with the almost obligatory complaint about the inadequacies of manufacturers' documentation. This justification, which is without doubt valid for introductory books on CP/M, does not apply to Miller's work, which is admittedly intended for the advanced CP/M user or the

8080 or Z80 assembly language programmer. Miller in fact makes no attempt to introduce the reader to CP/M, but rather assumes familiarity with all of its basic features.

The material presented in Miller's work is graduated both in terms of its difficulty and the level of expertise it requires from the reader. After examining CP/M's organization in memory, Miller focuses on the system tracks of a CP/M master diskette. As even most inexperienced CP/M users know, the contents of these tracks are not displayed in response to the DIR command and must even be copied onto new diskettes separately with the SYSGEN command; their very existence seemingly must be accepted as a matter of faith by the user, a very unattractive feature for those who would like to modify or enhance their operating systems. Miller shows not only that these programs exist, but that BIOS, the CP/M Basic Input/Output System, can be accessed, stored in data tracks, analyzed and even modified. The specific changes which he makes (the simultaneous transmission of output to console and printer, directing output to a special area of memory, or adding a "printer on" subroutine) hardly exhaust the possibilities for the user who is patient enough to disassemble BIOS and plan further alterations.

While these earlier chapters can appeal to all but the most inexperienced CP/M users, the remainder of the book increasingly focuses on topics of interest to the advanced programmer. After a brief introduction to macros, Miller guides the reader through the process of creating a fairly substantial macro library which can be used to bypass BIOS and instead use BDOS (the Basic Disk Operating System) to handle peripherals other than disks, to manipulate disk files with assembly language programs and to display a wider variety of information about disk files.

MASTERING CP/M offers the reader the results of a master craftsman's experience working with CP/M. It is a book which is indispensable to either the frequent or occasional assembly language programmer or the advanced CP/M user.

MICROCOMPUTER DISK TECHNIQUES

Paul Swanson

Level: Advanced Rating: **75**

	Byte	1982	Paper	
234 Pages	ISBN: 0-07-062582-4		6" x 9"	\$15.00

Most microcomputers, Swanson contends, are underutilized because of a lack of knowledge on the part of their owners. This is true above all of systems with one or more disk drives; "the average owner of a personal computer knows how to handle the disk in a relatively primi-

tive way.”(VII) The owner’s lack of knowledge, however, is itself the result of a lack of instruction—effective disk handling techniques begin at precisely that point at which the reference manuals leave off. Swanson has decided to remedy this sad state of affairs by writing a book “for the amateur programmer who is serious about learning how to effectively use a disk drive on a microcomputer system.”

But while Swanson’s observation about the underutilization of disks is accurate, his book does not at all promise to improve this situation by offering the reader effective instruction in disk file programming. Swanson’s instructional technique presupposes that the reader possesses that knowledge whose absence serves as the justification for writing the book. After introducing the reader to some of the most basic facts about disks in the first three chapters Swanson abruptly shifts from presenting facts in a comprehensible way to presenting his own programs for handling random access and key files, which the reader can use at his own risk but which Swanson never attempts to adequately explain. This problem is compounded by the diversity of disk handling commands under the various dialects of BASIC. As a result, those readers who are patient and adventurous enough to experiment with the numerous (and lengthy) programs offered by MICROCOMPUTER DISK TECHNIQUES may—with considerable pain and difficulty—learn something about the principles of effective disk programming, a topic which should have been the focus of Swanson’s book but which he in fact avoided.

MICROCOMPUTER OPERATING SYSTEMS

Mark Dahmke

Level: Advanced Rating: **85**

	McGraw-Hill	1982	Paper	
227 Pages	ISBN: 0-07-015071-0		5" x 8"	\$15.95

MICROCOMPUTER OPERATING SYSTEMS is not an abstract study of operating systems in general or an examination of one operating system in particular; instead, it is a highly practical work written for those who want to learn to design their own systems. This is a book for users who have accumulated a certain amount of expertise. At a minimum, a knowledge of assembly language programming is required; all of Dahmke’s routines are written using the Z80 instruction set, and must either be “cleaned up” by the user, adapted to his own microcomputer system, or translated into another microprocessor’s instruction set. The user should also be able to program a PROM or EPROM. Finally, some familiarity with the functions and general design of operating systems is indispensable, since Dahmke’s explanations tend to be cryptic.

Dahmke's focus is not so much on the design of an operating system as a whole as on individual features which it should possess. His presentation is graduated in its complexity: starting with small systems without mass storage devices, he in turn examines single-user systems equipped with disk drives and multiprocessing systems. The only complete collection of operating system routines which he provides are for the system monitor of a limited input/output system. Although in the other cases Dahmke outlines the features and considerations involved in system design and provides some flowcharts, the reader is left to his own devices in translating these into a working assembly language program.

If one of the major appeals of assembly language programming for many microcomputer users is the enhanced control it gives them over the basic operation of their system, then the prospect of designing an operating system should be especially appealing. For such users, Dahmke's book provides a treatment unique in microcomputer literature.

MICROPROCESSOR OPERATING SYSTEMS, VOL.II

John Zarrella, editor

Level: Advanced Rating: **85**

Microcomputer Application	1982	Paper
145 Pages	ISBN: 0-935230-04-1	6" x 9" \$12.95

MICROPROCESSOR OPERATING SYSTEMS provides a collection of nine essays on nine different operating systems available for microprocessors. Each of the contributors is actively engaged in developing or implementing one of these operating systems. An introduction, emphasizing the importance of rationally selecting an operating system for given applications or eventual system expansion, links the individual contributions together.

The nine operating systems surveyed fall quite naturally into three categories: real-time control systems (VRTX for 16-bit microprocessors); single-user systems (CP/M for 8-bit and CP/M-86 for 16-bit microprocessors); and multi-user, multi-tasking systems (multiuser OASIS for the Z80 or ZEUS, Zilog's extension of UNIX for the Z8000). While the articles do not aim at comparing operating systems (although CP/M is a frequent point of reference) and can at best present highly succinct overviews of each one, their distinctive features and strengths do emerge clearly.

The volume is a specialized one which, aside from those simply interested in operating systems per se, appeals primarily to persons engaged in selecting multi-user systems for business, scientific or profes-

sional applications. It is here that the availability of a particular operating system is of central concern in choosing a microcomputer system as a whole; although owners of home computers or single-user systems do at times choose microcomputers because they are CP/M compatible, selecting a computer independently of its operating systems is more common. But despite its limited audience, MICROCOMPUTER OPERATING SYSTEMS provides an excellent overview of nine very powerful operating systems.

OSBORNE CP/M USER GUIDE

For All CP/M Users

Thom Hogan

Level: Intermediate Rating: 90

	Osborne 1982	Paper	
286 Pages	ISBN: 0-931988-82-9	7" x 9"	\$15.95

Having found the reference manuals which accompany most versions of CP/M to be altogether incomprehensible to anyone other than an experienced programmer, Hogan decided to write a user's guide to CP/M which would be useful to a less sophisticated microcomputer user. He does not, however, precisely define his image of the non-professional programmer, which explains the highly diffuse range of material which he has chosen to include. Two of the longer chapters in his book discuss the assembly language utilities available under CP/M and the technical considerations involved in configuring CP/M to a newly expanded microcomputer system; at the same time, Hogan discusses the range of higher-level programming languages which can operate under CP/M, suggests criteria for microcomputer selection and urges the reader to "cut out the 'computerese'" by renaming CP/M transient command files so that their filenames resemble the basic operations which these programs perform (e.g., changing PIP to COPY, or CBAS2 to COMPILE).

Although the range of topics examined in the OSBORNE CP/M USER GUIDE is broader than in Rodney Zaks' THE CP/M HANDBOOK, the quality of Hogan's treatment is highly uneven. Portions of the book are written in flowing prose which is pleasurable to read; at other points, explanations are cryptic and at times even incomplete. Indeed, most of those portions of the book devoted to an examination of CP/M per se will probably awe and intimidate a completely inexperienced microcomputer user. Finally, the book is far less useful as a reference than as an instructional text. Despite these limitations, however, the Osborne CP/M USER'S GUIDE is an indispensable second guide to CP/M, and is cer-

tainly mandatory for anyone other than the completely inexperienced microcomputer user.

PERSONAL COMPUTERS HANDBOOK

Walter H. Buchsbaum

Level: Advanced Rating: **85**

Sams 1980 Paper
 286 Pages ISBN: 0-672-21724-4 6" x 10" \$12.95

Though touted by Sams as a microcomputer introduction for "everyone owning or interested in personal computers," Dr. Buchsbaum's book is actually geared for the electronic hobbyist or technically-advanced novice. This is not a book to be recommended to the neophyte, contrary to the publisher's jacket copy; excluding the first two chapters, it demands previous familiarity with electronic notation and/or computer architecture.

For the advanced beginner, PERSONAL COMPUTERS HANDBOOK provides an in-depth discussion of how a microcomputer functions at the gate-level—simple flip-flop gate logic is embellished until it becomes the notation by which the operation of everything from the Arithmetic Logic Unit to input/output ports are described. Part electronics handbook, part conventional personal computer introduction, Dr. Buchsbaum's treatment represents the most technical first exposure to microcomputers available in a trade-paper format. Unless your interests mandate additional information about the use of Schmitt-Trigger inverting amplifiers in microcomputer input lines, this is probably not suitable material for your personal computer library. It is, however, a superior technical introduction to microcomputers.

PROGRAMMER'S GUIDE TO CP/M

Sol Libes

Level: Advanced Rating: **90**

Microsystems Press 1982 Paper
 199 Pages ISBN: 0-916688-37-2 8" x 11" \$12.95

PROGRAMMER'S GUIDE TO CP/M is a collection of articles which were originally published in S-100 MICROSYSTEMS magazine from January 1980 to February 1982. The articles are primarily directed at the experienced BASIC and assembly language programmer whose code

directly interfaces with a system running under CP/M. Six subjects are covered: An Introduction to CP/M; The CP/M Connection (interfacing topics); CP/M on NorthStar Systems; Software Reviews; Utilities and Enhancements; and CP/M-86. A detachable CP/M Programmer's Reference Guide, listing built-in, transient and PIP commands, BIOS entry points and BDOS function calls, is included as an appendix.

The introductory material reviews CP/M's file and memory structures. A section on CCP functions includes a routine for converting BDOS hexadecimal addresses into their decimal memory block equivalents. A review of CP/M Utilities notes the differences the programmer can expect to encounter under versions 1.4 and 2.0 (for example, STAT.COM and SUBMIT.COM programs). One shortcoming of a consecutive article format is that there is some duplication of background material. CP/M disk formatting and BIOS sector skew are topics repeated in a number of early articles.

The CP/M interface section treats the modification of BIOS using assembly language calls and the use of CP/M utilities within BASIC and assembly language programs. The IOBYTE function, which allows the special designation of peripherals for specific output, is implemented in CP/M in another article, which additionally covers CP/M's logical I/O System.

The highlight of the Software Review section is a series of monographs on the C programming language; David Gerwitz briefly surveys the C language and provides a comparative analysis of popular C compilers.

PROGRAMMER'S GUIDE TO CP/M is one of the few advanced treatments of this complex operating system. It is a compendium of source and reference material unique in microcomputer literature.

THE S-100 & OTHER MICRO BUSES 2nd Edition

Elmer C. Poe and James C. Goodwin

Level: Intermediate Rating: 40

Sams	1981	Paper	
206 Pages	ISBN: 0-672-21810-0	5" x 9"	\$9.95

Poe and Goodwin's introduction emphasizes the tendency of a computer to "shrink" with age, as a given system becomes less and less able to cope with the growing demands and expectations which the user places upon it. Fortunately, the authors note, the computer's bus system provides a convenient means of system expansion. On the basis of this

introduction, the reader is led to believe that the authors' purpose is to examine the relationship between the type of bus a given microcomputer system has and the possibility of expanding the system.

Rather than pursuing this topic, however, the authors limit themselves to presenting some photographs of microcomputers and assorted control boards, along with the signal definitions and pinout designations of 21 different bus systems; the sections on each of the 21 buses are accompanied by an absolute minimum of discussion or text. Only in the final three chapters do the authors return to what should have been the major emphasis of the book—the conversion of a variety of bus signals (the Benton Harbor Bus, the TRS-80 Model I Bus, and bus systems for 6502/6800 microprocessors) to S-100 Bus signals (hence the book's title). Aside from readers interested in making these three kinds of conversions, this book is primarily of interest to those who enjoy looking at some neat illustrations of 21 bus systems.

THE S-100 BUS HANDBOOK

Dave Bursky

Level: Novice Rating: **65**

Hayden 1980 Paper
257 Pages ISBN: 0-8104-0897-X 9" x 11" \$15.75

The title of Bursky's book belies its contents, which aim at providing a brief (148 page) introduction to microcomputer systems for the hobbyist in particular. The S-100 Bus is discussed only in passing in the text (pp. 33-39), although two appendices do provide schematic diagrams of commonly used S-100 Bus boards and an examination of the proposed ANSI standard for the S-100 Bus. Aside from the scant attention which he devotes to the bus itself, Bursky provides a brief history of calculating and computing devices, a general introduction to microprocessors, a discussion of binary arithmetic and Boolean algebra, an examination of gate logic and flip-flops, and discussions of computer memory, input/output devices and interfacing, and assembly language programming for the 8080A microprocessor. Considering the limited space which is devoted to each topic, the treatment of the earlier topics (up to and including his discussion of gates and flip-flops) is actually quite good, although the quality of the book declines in its later chapters.

The contents of Bursky's book, as well as its copyright date, indicate that Bursky is writing about a previous era in the history of microcomputers. This is most obvious in his mention of the "control panel that contains switches and indicators that permit you to manually control all computer operations" (37) and in his discussion of methods for interfacing the computer to a paper-tape reader/punch (72-73). But it applies as

well to computer memory systems, which have become far more advanced than Bursky's discussion indicates, as well as to many specific devices discussed in the sections on interfacing. More generally, the book was written for an audience which, as microcomputers become more firmly integrated into everyday life, will continue to account for a dwindling share of the total market; Bursky focuses especially on the hobbyist interested in assembling a limited microcomputer system from individual components around the S-100 Bus.

The S-100 Bus is still one of the major bus systems in use today. Bursky's S-100 BUS HANDBOOK, however, does not provide an adequate guide to the bus, nor does it meet the needs of the contemporary microcomputer user whose system is based on the S-100 Bus.

A USER GUIDE TO THE UNIX SYSTEM

Rebecca Thomas and Jean Yates

Level: Intermediate Rating: **70**

	Osborne	1982	Paper	
508 Pages	ISBN: 0-931988-71-3		7" x 9"	\$15.99

Operating systems are rarely noteworthy for their user-friendliness, although their ease of use nevertheless does vary substantially. UNIX, an operating system developed by Bell Laboratories which supports the C programming language, occupies an almost unique position in this respect: command names often do not even vaguely resemble the functions they perform and are sometimes similar to very different commands used in other operating systems; where they exist, prompts and error messages are cryptic and obscure; the total number of commands is voluminous, and the possibility of misapplying them with disastrous consequences is very real; and documentation tends to be extremely technical and of little help to a befuddled user. The other side of this coin is that UNIX is extremely flexible and implements any number of commands and features unavailable in other operating systems.

Thomas and Yates have written this USER GUIDE in an effort to emphasize the strengths of UNIX while providing the user with sufficient knowledge of the system to manipulate it effectively. Their approach emphasizes hands-on practice: the first of two chapters on using UNIX consists of a collection of nine tutorial sessions which demonstrate some of the most basic UNIX commands; the second presents forty commands, and concludes the discussion of each with a tutorial on its use.

Unfortunately, in emphasizing the practical use of UNIX, the authors have de-emphasized the theoretical knowledge necessary to utilize the operating system successfully. Thomas and Yates' failure to discuss

the organization and functioning of the operating system adequately is at times most detrimental to the novice. For example, an experienced user of other operating systems can, with some reflection, figure out how and why LN, the link command, operates; to the novice, however, this command will seem as incomprehensible as it is silly. In other cases, the book fails to provide all users with the knowledge necessary to work with UNIX; this is true of the authors' treatment of UNIX'S most distinctive feature, its hierarchical file structure.

If the book fails to provide an adequate introduction to UNIX, so too does it fail to address those features which account for UNIX'S growing popularity. Although the authors note that UNIX "supports office automation" (375) and can be customized for diverse applications, their emphasis on UNIX'S flexibility resembles a rather crude sales pitch, since no supporting detail is provided. The authors' focus on mainframe UNIX magnifies this shortcoming for microcomputer users, and above all for those interested in adding UNIX as an operating system.

A USER GUIDE TO UNIX does provide a convenient reference work listing the syntax of the most common UNIX commands; we do not recommend this book, however, to anyone not already familiar with the UNIX operating system.

USING CP/M

A Self-Teaching Guide

Judi N. Fernandez and Ruth Ashley
Level: Intermediate Rating: **75**

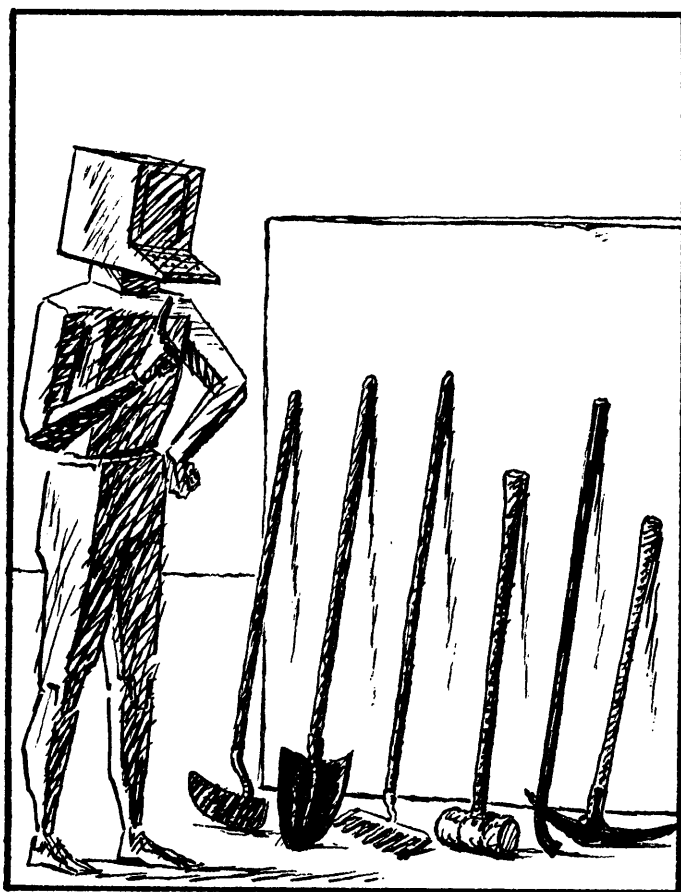
Wiley 1980 Paper
243 Pages ISBN: 0-471-08011-X 6" x 10" \$12.95

Without question, an assiduous reader of USING CP/M who answers every question in this Self-Teaching Guide and enters every sample command into his own operating system will triumph with a functional understanding of the book's topic. Since his time may be better spent with a more comprehensive treatment of CP/M, such a victory may well be regarded as pyrrhic.

Little attention is given to CP/M's architecture, the basis of both its power and flexibility. This architectural discussion normally evolves into a description of the nuances of the operating system—whose initial use by the reader is, presumably, the loading of CP/M into his microcomputer. USING CP/M barely treats this process. It is understandable that ASM, the CP/M assembler, would be given little attention in an introduc-

tory text, but even after the authors whet the appetite of the novice with a description of the general utility of some DDT functions, these functions are never explained.

In deference to the general quality of Wiley Self-Teaching Guides, the chapters which describe the CP/M text editor are accurately presented. Advanced ED functions are particularly useful. Neither treatment, however, sufficiently distinguishes USING CP/M from other less-than-exceptional works on this topic.



4

Programming

In time, the rudimentary tools of programming attract the attention of most microcomputer users. The works reviewed in this chapter reflect both the predominance of BASIC as a programming language for microcomputers and the emphasis on providing novices with introductory programming texts. The popularity of BASIC has, however, spawned the emergence of a variety of dialects which possess many subtle (and not so subtle) differences preventing a program written under one BASIC from running under another. As a result, a number of BASIC conversions dictionaries and encyclopedias have been published; the most outstanding of these is David Lien's *THE BASIC HANDBOOK*. Users with individual microcomputers can also refer to works on BASIC programming in the sections devoted to their own microcomputers; some of these, such as Lewis Rosenfelder's *BASIC FASTER AND BETTER*, are of interest to all programmers, and not merely to the owners of the particular machine to which the volume is devoted.

Present trends indicate the growing popularity of other programming languages. Some of these languages include: Pascal, a popular microcomputer language notable for its highly structured character; FORTRAN (FORmula TRANslation), an older language used extensively in engineering, statistics and the sciences; COBOL (COMmon Business Oriented Language), a traditional mainstay of programming for business applications; FORTH, a relatively new language which is unusual in that it is stack-oriented and uses Reverse Polish Notation; APL (A Programming Language), a scientific language permitting succinct and efficient code; PL/M (Programming Language for Microprocessors), a derivative of PL/I; C Programming Language, developed by Bell Laboratories; Ada, a comparatively new language developed by the Defense Department, and LISP (LISt Processor) an important language tool in artificial intelligence research.

We have included separate sections on FORTRAN, Pascal and these other languages. Pascal, the second most popular microcomputer language, is the focal point of a large number of texts and works of reference; among the latter titles, Jacques Tiberghien's *THE PASCAL HANDBOOK* is especially worthy of mention. Works devoted to other languages, and especially to COBOL and FORTRAN, however, are still relatively few in number, and often reflect a primarily main-frame emphasis; this is true, for example, of the Wiley titles on *STRUCTURED COBOL* and *FORTRAN IV*, even though the latter volume has supposedly been revised in an effort to include material of interest to FORTRAN microcomputer programmers.

GENERAL PROGRAMMING

BOOLEAN ALGEBRA FOR COMPUTER LOGIC

Harold E. Ennes

Level: Novice Rating: **95**

	Sams	1978	Paper		
128 Pages	ISBN: 0-672-21554-3	4" x 6"	\$6.95		

Why is it that two treatments of the same technical topic can differ so drastically in their ability to convey information? That is, why can one fail miserably while the other perfectly demonstrates an identical concept? The answer, at least in the case of microcomputer books, is a function of the author's effective use of examples—how universal they are, how real, how well they convey an analogous conceptual point. The answer for those microcomputer readers baffled by Boolean algebra is Mr. Ennes' book.

Books on microprocessor design and functions, regardless of how introductory, usually do poor jobs of interrelating binary arithmetic, symbolic logic, and circuit operation. *BOOLEAN ALGEBRA* admirably addresses this need. An elementary knowledge of algebra and electronics is required. The text, diagrams and schematics step the reader through truth tables, binary operations and simple gate logic. Venn diagrams and Karnaugh maps are used to simplify the manipulation of two and three variable problems. Thirty practice exercises are included in an appendix.

Any microcomputer user who has been grappling with the nuances of gate logic will find enlightenment (that transcendental 'connective or') within the covers of *BOOLEAN ALGEBRA FOR COMPUTER LOGIC*.

COMPUTER GRAPHICS PRIMER

Mitchell Waite

Level: Intermediate Rating: **90**

Sams 1981 Paper
184 Pages ISBN: 0-672-21650-7 5" x 7" \$14.95

COMPUTER GRAPHICS, like ancient Gaul, is divided into three main parts: a graphics overview, a concepts and hardware introduction and a programming guide. A working knowledge of BASIC is assumed.

The most immediately striking feature of the first section is its own use of graphics and illustrations: four-color plates of personal computer monitors eloquently support Mr. Waite's case for the special role of color graphics in education, industry, art, and, lest we forget, "fun and games." The conceptual discussion which follows traces the evolution of graphics technology from stroke graphics to dot matrix raster scan. Circuitry diagrams supplement this portion of the text. Sixteen "product profiles" of leading graphics-capable microcomputers are then presented. Each computer is evaluated on the basis of cost, total number of low and high resolution pixels, number of colors and a listing of graphic language statements. Digitizing plotters are also covered, but in a less standardized, less product-specific treatment.

The graphics programming section is, by the author's admission, the "meat of the book." Using AppleSoft BASIC, a series of program examples progresses from simple geometric and trigonometric displays to shape tables. Unlike other primers by Mr. Waite, the lack of a spiral binding complicates the entry of these programs into one's microcomputer. It is also unfortunate that not as much care was given to the accuracy of these programs as to the clarity of the book's graphics. (The program which demonstrates the plot of two cycles of two sine waves treats the number "1" and the variable "i" as interchangeable, resulting in output quite different from the author's intentions.) Mr. Waite's flair for readable text—as well as some of the truly stunning displays the demonstration programs produce—more than compensate for these shortcomings. High resolution graphics, shape tables and moving figure animation tables are all subjects of these elementary BASIC programs.

COMPUTER GRAPHICS PRIMER presents a comprehensive introduction to one of the more seductive uses of microcomputers for the intermediate BASIC programmer.

COMPUTERS IN NUMBER THEORY

Donald D. Spencer

Level: Novice Rating: **90**

Computer Science 1982 Paper
250 Pages ISBN: 0-914894-27-7 6" x 9" \$10.95

The author pursues the computer-as-mathematical-problem-solver topic familiar to readers of BASIC BASIC (see our review) with one notable exception—the computer discussion is paralleled by an equally fascinating discourse on number theory (large computers, we learn, are much easier to recognize than large primes).

The book begins with a brief overview of computers and BASIC. The BASIC language is explained using simple mathematical problem/examples. Spencer increases the complexity of these examples as the programming discussion evolves into the more advanced BASIC functions. The use of subroutines is demonstrated by a search for the roots of a quadratic equation, for example. A chapter on matrices not only mandates more complex examples, but is additionally prefaced by a purely mathematical introduction to the topic. The net result of this approach is that by the time the programming discussion gives way to Spencer's mathematical case studies, the reader has already been sufficiently "weaned" on BASIC problem-solving utilities.

The rest of the book is devoted to numbers: prime numbers, perfect numbers, Armstrong numbers, Fibonacci numbers, abundant numbers, deficient numbers, and square numbers, to mention just a few. Spencer's historical narratives, covering number theory from ancient Greece to Craig-1 research at the Lawrence Livermore National Laboratory, are fascinating. Scaled-down BASIC programs are used to demonstrate the use of the computer in number theory experiments; for example, a twenty-three line program computes all twin primes less than 1,000.

COMPUTERS IN NUMBER THEORY is filled with additional snippets of number theory history and computer examples. Donald Spencer has written an ingenious survey blending state-of-the-art mathematics with everyday BASIC programming. We highly recommend this book.

THE HEXADECIMAL CHRONICLES

Don Lancaster

Level: Intermediate Rating: **95**

Sams 1981 Paper
250 Pages ISBN: 0-672-21802-X 9" x 11" \$17.95

For readers with a vivid imagination who have not yet looked between the covers of Lancaster's book, the title may well conjure up the

fantastic vision of a lone hexadecimal as it valiantly wanders its way through... Closer inspection, however, indicates that a "chronicle" is in reality a conversion table, six of which Lancaster has included in THE HEXADECIMAL CHRONICLES as reference aids to assembly language programmers.

Although the price of the book is surprisingly high, its production quality is equally high. The book is spiral bound, an especially useful feature, considering that the reader must sometimes use two conversion tables simultaneously. Similarly, the alternating white and blue grids in the tables themselves minimize eyestrain and make the row or column which the user is seeking especially easy to find.

For the serious assembly language programmer, THE HEXADECIMAL CHRONICLES is an indispensable reference work. With the aid of this volume, the user can efficiently perform at least fifty-two different conversions between binary, octal, decimal, hexadecimal and ASCII-coded numbers; the conversions themselves can be performed for eight, sixteen, or twenty-four bit binary numbers. In addition, the sixth "chronicle" enables the user to perform simple hexadecimal arithmetic (including two's complement arithmetic) quickly and efficiently. It is unclear why Lancaster has included octal conversions, since he himself has declared this numbering system obsolete (p. 10). Nevertheless, without the tables, these numerical conversions are as tedious as they are time-consuming; with them, conversions from one numbering system to another can be both rapid and painless.

MICROCOMPUTER DATA-BASE MANAGEMENT

E.G. Brooner

Level: Intermediate Rating: 75

Sams	1982	Paper	
158 Pages	ISBN: 0-672-21875-5	5" x 8"	\$12.95

Brooner's book is both an introduction to general data base searching and sorting techniques, and a guide to a limited number of commercial data base packages.

Though programs are written in NORTHSTAR BASIC, this is a text valuable in the coding techniques it describes rather than as a source-book of ready-to-run programs. Searches (binary, sequential and random access) and sorts (bubble, merge and shell) are described in detail and demonstrated in program examples. Special consideration is given to the linked list and hash address methods of having data files internally describe data addresses through pointers and algorithms.

Brooner does not discuss the relational and hierarchical models of larger data bases; rather, the second half of his book focuses on the ways

in which three data base packages, WHATSIT, PROFILE, and SELECTOR, demonstrate the simple techniques previously described. Each package generates data base files based upon user-supplied answers to program prompts; each is given a business application problem to solve in order to display its unique query and relational features. The author concludes his analysis of each product with an evaluation which suggests additional applications suited to its particular strengths. A final section treats QSORT, a data base sort utility capable of sorting any CP/M compatible file.

MICROCOMPUTER DATA-BASE MANAGEMENT chooses to focus on a few data base searching and sorting methods, and even fewer data base software packages (dBASE II, for example, is never mentioned). While it does an adequate job describing those techniques and packages which the author has selected, we recommend a more comprehensive introductory treatment to programmers interested in further exploring this niche of software science.

POCKET GUIDE TO PROGRAMMING

John Shelley

Level: Novice Rating: **50**

	Addison-Wesley	1982	Paper	
62 Pages	ISBN: 0-201-07736-1		4" x 6"	\$6.95

The POCKET GUIDE TO PROGRAMMING is intended to provide the unifying link in the Addison-Wesley pocket guide series, the other four volumes of which concentrate primarily on the structure and syntax of an individual high-level language. In contrast, this pocket guide is supposed to transcend the peculiarities of individual languages in order to teach the beginner the general concepts and techniques needed to write successful computer programs.

Unfortunately, the book itself serves as a prime example of the dangers involved in not adopting a top-down, structured approach to a problem. Shelley presents programming as a miscellany of concepts, techniques and principles, none of which are carefully integrated into his discussion. The important points presented in the course of this brief text are obscured by mountains of trivia to which the author has paid an equal amount of attention. As a result, the reader is unable to see the forest for the trees.

Any short book which has not carefully delineated and focused on its subject matter must necessarily suffer from superficiality. This is certainly true of the POCKET GUIDE TO PROGRAMMING, whose discussion attempts to cover the waterfront of programming-related topics.

As a result, it is highly unlikely that this volume will actually enable the aspiring programmer to produce operable code.

PRACTICE PROBLEMS IN NUMBER SYSTEMS, LOGIC & BOOLEAN ALG.

Edward Bukstein

Level: Intermediate Rating: 90

Sams 1981 Paper
133 Pages ISBN: 0-672-21451-2 8" x 11" \$8.95

Whether the programmer's goal is to write more efficient assembly language programs or to better understand microprocessor gate logic, mandatory areas of study are number system manipulations and Boolean algebra. PRACTICE PROBLEMS is a workbook filled with sixty-four lessons (limited text and problems) that explore these related topics. A complete answer key is contained within each chapter.

This study guide actually provides a better introduction to number systems, logic and Boolean algebra than most programming texts. The number systems section, for example, not only treats the familiar decimal-to-binary-to-octal conversions, but delves into some interesting coding esoterica: two-out-of-five code and excess-three code (both convert decimal to binary on a digit-by-digit basis). Two's complement arithmetic is sensibly approached by first attempting decimal subtraction by the complement method, and then introducing binary. Even the logic gate/Boolean discussion rates as novel in its immediate use of simple circuit diagrams as illustrations for and/or relations.

The workbook format lends itself to effective treatment of advanced Boolean topics. Venn and Veitch diagrams, Karnaugh maps and truth tables are all covered in practice exercises. Though relatively little attention is given to NAND and NOR gates, PRACTICE PROBLEMS IN NUMBER SYSTEMS, LOGIC AND BOOLEAN ALGEBRA nevertheless warrants our recommendation as a superlative introduction to these microcomputer-related subjects.

PROBLEM SOLVING PRINCIPLES FOR PROGRAMMERS

Applied Logic, Psychology, and Grit

William E. Lewis

Level: Novice Rating: **90**

Hayden 1980 Paper
 163 Pages ISBN: 0-8104-5138-7 8" x 11" \$9.95

Lewis' book has become a mini-classic for programmers of all languages and levels. Problem-solving techniques are revealed through general non-computer problem solving, and then applied in programming examples. What makes PROBLEM SOLVING PRINCIPLES unique is the effective use of problem/examples (some original, but most rather well known) and the psychology of problem-solving in teaching programming.

Actual code is written in the author's own "interlingua," a pseudocode with cross-language applicability. Special treatment is given to top-down structured programming as well as to debugging techniques. The presentation is interwoven with Lewis' own brand of humorous-but-true programming axioms, and is as comprehensible as the techniques he teaches.

PROBLEM SOLVING PRINCIPLES FOR PROGRAMMING transcends the usual differences among individuals who program in various languages on different types of machines; it is must reading for anyone who intends to solve relatively complex real-world problems by using a computer.

SOFTWARE ENGINEERING FOR MICROS

The Electrifying Streamlined Blueprint Speedcode Method

T.G. Lewis

Level: Advanced Rating: **80**

Hayden 1979 Paper
 156 Pages ISBN: 0-8104-5166-2 6" x 9" \$6.95

SOFTWARE ENGINEERING FOR MICROS is a highly eccentric (and at times dogmatic) work which aims at imparting a "correct," disciplined and standardized approach to software development. Lewis' book is not

of interest to just any microcomputer programmer, but rather to a person engaged in program development for a large corporation or major software firm.

As he did in his *HOW TO PROFIT FROM YOUR PERSONAL COMPUTER*, Lewis spends a good deal of his time debunking "myths" and replacing them with "facts." For example, the myth that "software design must include consideration of a particular machine's architecture in order to take advantage of exceptional machine features" is replaced by the "fact" that "software design must be machine-independent, thus removing machine features from the design and precluding potentially unreliable programs." (7) In fact, however, the relationship between these two statements is not one of misconception to truth; either statement can be true, depending on the situation in which the dichotomy is applied. This is the case with most of the "myths" and "facts" presented by Lewis in the course of this book.

In large measure, Lewis has developed his "electrifying streamlined blueprint speedcode method" in response to the economic consequences of the computer revolution: while the cost of hardware has declined rather dramatically over time, the cost of software development has followed a sharp upward spiral. In order to restore a rough proportionality between the two, Lewis emphasizes maximizing hardware costs as a means of reducing more costly expenditures on software development. By using a streamlined (and top-down) approach to programming which involves the preparation of a zero-level blueprint (or very general program design), a speedcode, and kludgecode (or an assembly language program for a particular microprocessor), Lewis proposes a methodology to more fully utilize these expanded hardware capabilities, thus speeding program development while reducing total development costs. (Lewis does, however, later abandon assembly language in favor of a structured higher-level language, such as PL/M.)

Most of Lewis' argument is sound, although it ignores any number of very real considerations which limit its applicability. Since Lewis' overall solution assumes that it is feasible "to include several microprocessors in a system in order to increase the memory space available for resident programs," (3) it is not of great interest to the personal computer user who finds that a program and data cannot fit within available memory. Nor is it relevant to those who do their own programming, and particularly enjoy using machine-dependent features in order to produce efficient code—or even to learn about the capabilities of their machines more fully. But for large corporations or software development firms, a variety of economic and non-economic considerations—none of which have been recognized by Lewis—can intervene to undermine his assumption that the reduction of software costs and the production of easily generalized software should be a central goal of programming.

BASIC

ADVANCED BASIC Applications and Problems

James S. Coan

Level: Intermediate Rating: **90**

	Hayden	1977	Paper	
184 Pages	ISBN: 0-8104-5855-1		6" x 9"	\$11.50

In the tradition of his best-selling BASIC BASIC (see our review), author James Coan has treated additional BASIC programming topics and techniques in ADVANCED BASIC. The structure of individual lessons is identical in both books, with programs illustrating the solution to given mathematical problems.

After a brief review of elementary BASIC and the introduction of some extended BASIC features, Coan progresses to the "applications and problems" that are the focus of this current volume. These include file manipulation, coordinate geometry, matrices, statistics, polynomials, simulations and games. The mathematical discussions, along with their supplementary program/illustrations, develop from simple to more complex considerations. The chapter on statistics, for example, begins with a discussion of summation notation and ends with the computation of the coefficient of linear correlation of three sets of temperature data. Unbeknownst to the reader, Coan flirts dangerously close to elementary integral calculus with his chapter treating the calculation of the area under a curve. Of practical use to the non-mathematically inclined programmer is a chapter on BASIC files. Though Hewlett Packard and General Electric file systems are specifically discussed, the file handling techniques using serial, random access, ASCII and binary files are of general programming practicability.

Appendices include answers to selected problems, a short bibliography and a summary of statements in BASIC. ADVANCED BASIC is the logical extension of Coan's original mathematical presentation of BASIC programming to more complex topics. Readers of BASIC BASIC will find this companion volume equally informative.

BASIC 2nd Edition A Self Teaching Guide

Robert L. Albrecht, LeRoy Finkel, Jerald R. Brown
Level: Novice Rating: 80

Wiley 1978 Paper
325 Pages ISBN: 0-471-03500-9 7" x 10" \$12.95

The BASIC which the authors attempt to teach is not a microcomputer version, but instead closely resembles the original Dartmouth BASIC used on larger computers. Like BASIC FOR HOME COMPUTERS (see our review) and the other publications in Wiley's Self-Teaching Guide series, this volume aims at imparting the fundamental principles of programming in BASIC to the novice programmer. The remarks about the audience for which BASIC FOR HOME COMPUTERS was intended apply as well to BASIC, 2ND EDITION.

The range of topics covered in this volume is somewhat greater than that of BASIC FOR HOME COMPUTERS, and the level of condescension with which the reader is treated is somewhat reduced. In addition, the final chapter of the book deals with the use of external data files, a topic which is altogether overlooked in BASIC FOR HOME COMPUTERS. As a result, although BASIC, 2ND EDITION is not intended primarily for microcomputer users, and despite the differences between Dartmouth BASIC and microcomputer BASICs, the book is a more useful manual for teaching microcomputer users BASIC than is BASIC FOR HOME COMPUTERS.

BASIC: ADVANCED CONCEPTS

Joseph C. Giarratano
Level: Intermediate Rating: 80

Sams 1982 Paper
214 Pages ISBN: 0-672-21942-5 8" x 11" \$22.95

BASIC: FUNDAMENTAL CONCEPTS

Joseph C. Giarratano
Level: Novice Rating: 70

Sams 1982 Paper
198 Pages ISBN: 0-672-21941-7 8" x 11" \$22.95

Although these two volumes occupy the upper end of the price scale for BASIC language texts, they by no means occupy a similar position in

terms of quality. While the introductory volume provides a competent treatment of BASIC, it at the same time treats much of its material in a hasty and superficial way and omits a number of topics which are frequently included in introductory texts (e.g., BASIC string-manipulation commands and elementary data file handling). These topics have been included in the text for "advanced" programmers, which gives this second volume a highly uneven, motley character. Curiously, both books also include identical appendices.

Giarratano's goal in these two volumes is to teach the reader not one version of BASIC, but two. Throughout both volumes, he focuses equal attention on mainframe BASIC (for the DEC PDP-11) and on microcomputer BASIC (the Microsoft version for the OSI Challenger 2P and Challenger II). This dual treatment leads him to include two listings for each program presented in the text, even though the mainframe and microcomputer dialects are frequently similar. While the number of pages lost to this rather peculiar practice is minimal in *FUNDAMENTAL CONCEPTS*, it is quite substantial in *ADVANCED CONCEPTS*, and strongly contributes to the highly uneven quality of the latter volume. The OSI version of Microsoft BASIC is itself in many respects a peculiar one, although Giarratano treats these as part and parcel of Microsoft BASIC. More generally, while the emphasis on both BASIC dialects is a boon to users of both kinds of computer systems, it substantially reduces the value of these two volumes for persons without access to a larger computer, who are learning BASIC solely in order to program their microcomputers.

While *FUNDAMENTAL CONCEPTS* offers an introduction to BASIC with no frills, *ADVANCED CONCEPTS* does indeed treat advanced features and problems of BASIC programming which are simply noted—or even completely overlooked—in most other texts. The chapters on exact arithmetic calculations and accuracy and precision in BASIC are outstanding, since they emphasize programming aimed at overcoming the limited precision of BASIC. The chapter which uses PEEK and POKE commands for graphics applications, as well as to study and customize BASIC or to write self-modifying programs, is also highly distinctive, although Giarratano's discussion is heavily dependent on the memory map of OSI microcomputers. On the other hand, the chapter on string functions could easily have been presented in the introductory volume, while the discussion of file manipulation is altogether inadequate and simplistic. Even within his chapters on performing accurate and precise calculations with BASIC, the author interrupts his "advanced" discussion in order to present the elementary principles of binary arithmetic and binary-decimal conversion.

BASIC: *FUNDAMENTAL CONCEPTS* provides an altogether undistinguished (and overpriced) introduction to BASIC programming. *ADVANCED CONCEPTS*, despite its deficiency of advanced program-

ming techniques, is somewhat better, although it is not a volume we would recommend to any but the most dedicated BASIC programmers.

BASIC AND THE PERSONAL COMPUTER

Thomas Dwyer and Margot Critchfield

Level: Novice Rating: **95**

Addison-Wesley 1978 Paper
438 Pages ISBN: 0-201-01589-7 9" x 11" \$14.95

Fortunately for the BASIC reader, the "Personal Computer" of this book's title is a purely generic usage not to be confused with the microcomputer marketed by IBM. This is the reader's good fortune, since the broadest possible spectrum of personal computer users will benefit from the discourse on the BASIC language contained within its covers. A "vintage" treatment (note the 1978 copyright), its datedness may be a bit cloying at times (there are scattered references to the 'Apple I' and Cromemco 'Dazzler' boards). On the whole, however, BASIC AND THE PERSONAL COMPUTER has admirably withstood the onslaught of competing BASIC introductions; perhaps it is Critchfield's timelessly humorous illustrations, or the imaginative coding examples, or the clear, concise presentation which still rank it among the best in the literature.

Early chapters serve as a prelude to the now famous "'8 hour wonder' guide to BASIC programming," a brief overview of BASIC which provides the groundwork for later elaboration. All subjects, whether explanations of minimal BASIC or complex sorting algorithms, are taught by coding solutions to ingeniously-devised problems. (Yes, in this book it is the problems, and not the solutions, which are "ingenious.") Matrix manipulation, for example, is presented by providing a football coach with a Dallas Cowboy-quality analysis of scouting reports. Various programming techniques are "embedded" in a chapter on computer games; this same chapter also serves to consolidate the early lessons of the 8-hour wonder guide. The appendices and topics in advanced BASIC are cogent arguments for the book's additional value as a reference manual for microcomputer users well past those first eight hours of BASIC coding.

Learning Basic from Dwyer and Critchfield is a pleasurable, enriching experience. BASIC AND THE PERSONAL COMPUTER is a book we highly recommend.

BASIC BASIC 2nd Edition

Introduction to Computer Programming in BASIC Language

James S. Coan

Level: Novice Rating: **90**

	Hayden 1978	Paper	
269 Pages	ISBN: 0-8104-5106-9	6" x 9"	\$11.50

This second edition of James Coan's now-classic introduction to the BASIC programming language incorporates updates to the original text (primarily string functions), but leaves intact its unique approach: to wed the teaching of BASIC to the teaching of select topics in precalculus mathematics.

For example, the computer's utility as a mathematical tool is used as a springboard into the programming of solutions to quadratic equations. And, if we can solve for the axis of symmetry and turning point of a particular quadratic function—called a parabola—then we can also have the microcomputer graph the solution. Thus, the complexity of the BASIC code evolves in response to demands for more versatile programming through the use of more complex mathematical problems. This process of masterfully interweaving both topics is itself as fascinating as the lessons being taught in either. The mathematical topics themselves range from Euclidean geometry to elementary probability. Exercises are included at the end of each chapter, and an appendix on error diagnosis treats elementary debugging techniques.

BASIC BASIC is a very special introduction to this programming language. Combining the disciplines of mathematics and programming, it provides the reader not only with a working knowledge of BASIC, but also with a first glance at the scientific utility of microcomputers.

BASIC BASIC-ENGLISH DICTIONARY

For the Apple, Pet, and TRS-80

Larry Noonan

Level: All Rating: **80**

	Dilithium 1982	Paper	
150 Pages	ISBN: 0-918398-54-1	6" x 9"	\$10.95

As its full title indicates, this is both a BASIC dictionary and a conversions handbook. Each entry is followed by a brief definition, an example of the BASIC statement, and the status of the statement in Apple

Integer BASIC, AppleSoft, PET BASIC, and TRS-80 Level I and Level II BASIC. When a statement is not compatible with a particular version, Noonan also provides examples of substitute statements, if any are available.

The dictionary is concise and well-focused, with no entries for superfluous terms such as those relating to programming techniques, other programming languages, etc. Its format—all statements are arranged alphabetically, and all operators (and, colon, comma, >, <, etc.) appear alphabetically in a separate section—makes the dictionary both convenient and easy to use as a conversion handbook not only for Apple, PET, and TRS-80 users, but for users of other versions of BASIC as well.

THE BASIC BOOK

A Cross-Referenced Guide to the Basic Language

Harry L. Helms

Level: All Rating: **50**

	Byte	1983	Paper	
49 Pages	ISBN: 0-07-027959-4		6" x 8"	\$6.95

The definitive work against which all new BASIC cross-reference guides will be compared is David Lien's THE BASIC HANDBOOK (see our review). This latest 49-page reference work by that most concise of microcomputer authors, Harry Helms, compares unfavorably on all counts, even given its ostensibly cheaper price (\$6.95 for forty-nine pages vs. \$19.95 for 500 pages of David Lien's work).

THE BASIC BOOK is actually less a cross-reference guide to different BASIC dialects, as the cover copy proclaims, than a cross-system glossary of BASIC commands. This distinction is apparent in its treatment of the PLOT command: the entry reads "...turns on specified graphics block (Apple II and Atari only)." The reader is not told the syntax of the command or its relation to similar commands on other systems (for example, SET, RESET, or DOT). Since the utility of a cross-reference guide is presumably to assist the programmer in his translation of different BASIC dialects, THE BASIC BOOK fails in this singularly important respect.

Moreover, this "glossary" is divided into ten distinct command sections: system commands; variable and array commands; arithmetic, relational and logical operations; control and transfer statements; input and output statements; subroutines; string functions; numeric functions and statements; assembly language routines and statements; and graphics statements. The novice who is genuinely interested in a listing, with

definitions, of most BASIC commands, will have difficulty referencing individual commands without first searching a composite alphabetic index.

Of the definitions that are given, most are dangerously superficial. The RND function, we are told, "generates a pseudo-random number." No warning is given regarding the dialect-specific range of numbers that may be generated (Apple Integer BASIC, TRS-80 Level I BASIC and Sinclair BASIC all produce results which fall within their own particular numeric limits).

THE BASIC BOOK is a rather feeble entry into the microcomputer reference work genre. Its limited utility as a simple glossary of BASIC terms provides little reason for its addition to any personal computer library.

THE BASIC CONVERSION HANDBOOK

For Apple, TRS-80, & PET Users

Brain Bank

Level: All Rating: **40**

80 Pages	Hayden	1981	Paper	
	ISBN: 0-8104-5534-x		6" x 9"	\$7.95

Although their text is only 66 pages long, the authors have nevertheless managed to perform a seemingly impossible feat: they have put together a book which is cumbersome and difficult to use. Much of the problem lies in the book's organization. Rather than listing each BASIC statement alphabetically, the book's chapters are organized by the particular variant of BASIC into which the user wishes to convert a program. Chapter 2, for example, deals with conversions of AppleSoft or PET Basic into TRS-80 Basic, while chapters 3 and 4 treat conversions into AppleSoft or PET BASIC, respectively.

The chapters themselves contain conversion tables which indicate whether or not a particular statement is compatible in any two of these three dialects of BASIC. The information which these tables provide is altogether minimal; the BASIC user with a mental block for syntax will get no help here, but will instead have to rely on other reference works. Finally, anyone who wishes to convert a BASIC program into some version of BASIC other than these three will find THE BASIC CONVERSIONS HANDBOOK altogether worthless.

THE BASIC COOKBOOK

Ken Tracton

Level: All Rating: **75**

Tab 1978 Paper
 138 Pages ISBN: 0-8306-1055-3 5" x 9" \$5.95

THE BASIC COOKBOOK is a dictionary of the BASIC language and not, as the book's back cover ambiguously implies, an instructional text which will help a novice understand programming in BASIC. It therefore can serve a dual purpose: first, the BASIC COOKBOOK offers a comparatively inexpensive, but at the same time fairly comprehensive, summary of commands in BASIC; second, Tracton has paid some attention to variations among versions of BASIC, so that the book can double as a conversions handbook. Although Tracton does not specify which statements are permissible under particular dialects of BASIC, THE BASIC COOKBOOK can at least serve as an aid in debugging by directing the user's attention to variations in BASIC commands. For example, the user having problems with an IF-THEN statement will find that a number of variations of the statement exist, not all of which are recognized by the same version:

If (relation) Then (line)
 If (relation) GOTO (line)
 If (relation) Then (statement)

That this small and inexpensive dictionary contains a number of oversights, however, is evident in the one major variation of the IF statement which is not noted under this entry:

If (relation) THEN (statement):GOTO (line)

Although definitely a bargain at the price, the serious BASIC programmer will find BASIC COOKBOOK of increasingly limited value as his or her programming skills advance. We recommend it above all to those programmers who need at most a simple guide to BASIC statements and syntax.

BASIC FOR HOME COMPUTERS A Self-Teaching Guide

Bob Albrecht, LeRoy Finkel and Jerald R. Brown

Level: Novice Rating: **75**

Wiley 1978 Paper
 335 Pages ISBN: 0-471-03204-2 6" x 11" \$9.50

BASIC FOR HOME COMPUTERS is one volume in a series of Self-Teaching Guides published by Wiley which attempts to teach the com-

pletely uninitiated reader some of the basic principles of computer operation and programming. This particular volume aims at teaching the reader Microsoft BASIC, one of the most common versions of BASIC for home computers. It is written so that the reader can quickly grasp the fundamental principles or concepts of BASIC programming; the text is interspersed with questions or blanks which the reader must fill in, and each chapter concludes with a self-test, the answers for which are given at the back of the book. The discussion centers on the most essential features of BASIC, and completely ignores many of its more advanced features. Although it can be argued that some of these (such as CALL or USR and PEEK and POKE) may be beyond the comprehension of the novice, this cannot be said of the commands necessary to manipulate programs or data on mass storage devices, a topic which is completely ignored throughout the book.

This book is ideal for the person who has absolutely no understanding of programming, and who views computers and programming languages with varying combinations of awe, dread and mystification. It is extremely successful in dismantling an apparently formidable topic and treating each step in a demystified and comprehensible way. For anyone with even a minimal grasp of computer programming who does not share this computer phobia, however, the book is likely to seem both boring and condescending. A typical example of the kind of question posed throughout the text is the following: after explaining floating point notation and presenting the example of "4E+09," the authors ask that the reader fill in the blank of the sentence "The mantissa and the exponent are separated by the letter _____." This extremely elementary and simplistic approach encourages the reader who is not completely shell-shocked by the thought of studying BASIC to impatiently gloss over portions of the text, even if he or she is not thoroughly familiar with the material being discussed.

BASIC FROM THE GROUND UP

David E. Simon

Level: Novice Rating: **85**

	Hayden	1978	Paper	
222 Pages	ISBN: 0-8104-5760-1		6" x 9"	\$11.75

Though less mathematically intensive than BASIC BASIC, Simon's book continues its predecessor's computer-as-mathematical-tool approach to programming. All exercises tend to remain within the confines of elementary algebra. The author uses a "stereotypic" BASIC that he intends readers to follow more for conceptual understanding than precise syntax mastery. This technique works relatively well until such

system-specific topics as data file programming are encountered; an adequate treatment is still managed, although not without recourse to the author's suggestion to "consult your manuals" for additional details.

Readers of introductory BASIC texts (usually either reviewers or individuals still struggling to learn BASIC) will recognize the familiar topics of graph-plotting, random-number generators and number-guessing programs. The use of BASIC to solve algebraic expressions is the theme of most practice exercises. We found it rather paradoxical that the author had gone to the considerable trouble of including wrong examples of BASIC coding in his "Summary of BASIC" appendix, but had neglected to provide an answer key to any of the book's programming problems.

BASIC FROM THE GROUND UP nevertheless provides a comprehensive treatment of the capabilities of BASIC, couched in a less specialized mathematical context than its BASIC BASIC counterpart.

THE BASIC HANDBOOK

Encyclopedia of the BASIC Computer Language

David A. Lien

Level: All Rating: **100**

CompuSoft 1981 Paper
471 Pages ISBN: 0-932760-05-8 5" x 8" \$19.95

A variety of BASIC dictionaries and conversions handbooks—including some of quite high quality—are currently available for micro-computer users who have a tendency to forget the fine points of syntax, or who wish to translate a program written in one dialect of BASIC into another. Because of its encyclopedic treatment and its high production quality, however, THE BASIC HANDBOOK puts all of these lesser works to shame. An examination of the book's contents amply bears out the publisher's assertion on the back cover that this is "the most complete collection of BASIC words ever." Most BASIC commands have in fact been incorporated into the text. Although THE BASIC HANDBOOK is two to three times as expensive as other dictionaries or conversions handbooks, its price is more than justified by the meticulous and painstaking care with which it has been assembled.

And Lien has done far more than simply assemble words. At least a full page is devoted to each BASIC command or operator (terms appear in the front portion of the of the encyclopedia, operators in the back), so that it is both easy and convenient to use the book. Once the reader finds the term of interest to him, the handbook usually provides him with the following information:

- 1) The meaning of the command or word.
- 2) The proper syntax for its use.
- 3) If the term is an obscure one, the version of BASIC in which it is found.
- 4) One or more brief test program which enables the user to determine if a given BASIC word will operate under a particular dialect.
- 5) Programming alternatives if an individual version of BASIC will not accept the command.
- 6) Words or terms which perform a similar function or are in some way related to the referenced term.

Special appendices examine individualized features of Acorn ATOM, Atari, Tektronix, TRS-80 Color, and Disk BASIC. The index, aside from listing the pages on which terms occur, allows the reader to indicate whether that term will be accepted by his or her microcomputer, and to enter brief notes.

Whether the user simply desires a reference to BASIC statements or a guide to conversions among BASIC dialects, this is the finest work available. Because of its meticulous attention to the needs of the BASIC programmer, we have given it our highest rating.

BASIC PROGRAMMER'S NOTEBOOK

Earl R. Savage

Level: **Advanced** Rating: **90**

Sams	1981	Paper	
110 Pages	ISBN: 0-672-21841-0	9" x 11"	\$14.95

Written as a compendium of programming notes for the advanced BASIC programmer, this book is divided into four sections: an introduction which highlights documentation and memory-conservation techniques; a "notes" section, the main body of programming tips; a "programming aids" section which treats utility programs; and a "hardware-aids" section. Though written on a TRS-80 in Level II BASIC, the programs and techniques discussed by Savage are of interest to the users of other machines as well.

The introduction, after entreating programmers to follow the ten rules of "good programming," offers methods of coding which reduce the use of available memory and/or increase execution speed. The author leaves in the reader's hands the resolution of the maximum clarity vs. maximum efficiency dilemma.

Fifty program fragments and subroutines are listed in the "notes" section. A line-by-line analysis of each is included, with the intention of providing the BASIC programmer with techniques for everything from

error trapping to bubble sorts. Our favorite was a method of having a program disable the LIST command, so that the program itself could not be listed on either a monitor or printer. Each of the entries is a small gem of code which can easily be incorporated into larger programs; the author's use of minimal BASIC (actually a stripped-down version of TRS-80 Level II BASIC) was designed to facilitate this process. Flowcharts accompany the more complex coding examples.

Utility programs are briefly described in the "programming aids" section. A list of vendors for selected commercial programs directs the programmer to possible sources of utility code.

The "hardware aids" section is the only portion of the book of interest exclusively to TRS-80 owners. Hardware enhancements to the TRS-80 Model II which are discussed include memory expansion and the improvement of mass storage.

Earl Savage's spiral-bound BASIC PROGRAMMER'S NOTEBOOK is, by his own description, a "smorgasbord" of programming tips. We recommend it not only as a sourcebook of immediately useful code, but for its demonstration of original programming techniques.

BASIC PROGRAMMING PRIMER

Mitchell Waite and Michael Pardee

Level: Novice Rating: **90**

Sams 1978 Paper
236 Pages ISBN: 0-672-21586-1 6" x 9" \$11.95

The BASIC PROGRAMMING PRIMER is a textbook which aims to teach the reader the most important features of Microsoft BASIC, one of the most widely-used versions of BASIC for microcomputers. Although oriented largely toward the beginning BASIC programmer, the book can also be used to advantage by those who are familiar with BASIC programming on larger mainframe computers but do not know how these versions of BASIC differ from those used on microcomputers.

The book is clearly written and avoids treating the reader with the condescension that is so prevalent in many introductory texts. Although some of the more advanced features of Microsoft BASIC are not discussed in the BASIC PROGRAMMING PRIMER, its treatment of microcomputer BASIC is fairly comprehensive. In addition, two of the book's appendices are especially useful; Appendix C provides helpful hints on reducing the amount of memory consumed by a BASIC program, while Appendix D offers five suggestions to speed program execution time. The major shortcoming of the book—which becomes more glaring as more and more microcomputers are equipped with disk storage—is that it pays very little attention to file manipulation. This oversight makes the

book most useful for those who will include data in the program itself using READ and DATA statements, who will enter data interactively from the keyboard with the INPUT statement or who will depend on BASIC programs themselves to generate data.

But while BASIC PROGRAMMING PRIMER may not provide the most comprehensive introduction to BASIC programming, its detailed, even treatment of the material which it does cover makes it one of the best introductory texts available. We highly recommend this Waite and Pardee offering above all to newcomers to programming languages.

BASIC WITH STYLE: PROGRAMMING PROVERBS

Paul Nagin and Henry F. Ledgard

Level: Novice Rating: 55

Hayden	1978	Paper		
134 Pages	ISBN: 0-8104-5115-8	5" x 8"	\$8.75	

This entire series of volumes (PROGRAMMING PROVERBS FOR FORTRAN PROGRAMMERS, as well as BASIC, COBOL, FORTRAN, and PASCAL WITH STYLE) is concerned not so much with the details of the individual language to which each book is devoted as with developing a general set of sound programming practices while using each language. Effective programming is emphasized through a series of programming "proverbs" which emphasize the importance of multi-level planning and elaboration of a computer program in which the actual writing of code is only the final stage of a longer, well-conceived process. The series then focuses on the actual process of developing a program for a concrete application (generally Kriegspiel checkers, although COBOL WITH STYLE substitutes "the Atad Copper Mine problem," and the two FORTRAN volumes provide an additional payroll program) in order to show how these proverbs may be concretely applied.

What is most disturbing about this series is precisely its lack of attention to the features of any particular language, which suggests that these five volumes could easily have been consolidated into one. Most troublesome are the two books devoted to FORTRAN. PROGRAMMING PROVERBS is written with emphasis on FORTRAN IV and may be considered the first edition of this work; FORTRAN WITH STYLE treats FORTRAN 77 and is basically a second, revised edition. Not only is this slight difference of focus not indicated in their titles, but the minimal treatment of the features of each standard necessarily raises the question of why both books are in print simultaneously. More generally, given the peculiarities of individual languages which alternately encourage some

sound programming principles while discouraging others, it is surprising that the authors have—with only a few exceptions which are also largely unrelated to the languages themselves—included the same proverbs for the reader's edification. PASCAL WITH STYLE, for example, emphasizes the "Don't GOTO" proverb, although some texts no longer even mention that Pascal has a GOTO statement. But most disappointing in this regard is the volume devoted to BASIC. As numerous critics have pointed out, BASIC combines in a single language most of the features which discourage good programming techniques. The authors, however, do not even note this in the text, nor do they take any pains to show the reader how to impose order from without. In view of the pressing need for a longer, more detailed treatment, it is paradoxical that BASIC WITH STYLE is the shortest volume in the series (124 pages, excluding appendices).

Independent of the merits of the arguments about programming techniques presented in the course of this series—which are, admittedly, considerable—the basic similarity of content, the failure to focus extensively on the features of a particular language, and the at times uneven quality of the authors' treatment (reflected, for example, in the inclusion of such patently obvious proverbs as "Don't Panic" in all the ...WITH STYLE books) make it very difficult to recommend more than one volume of the entire PROGRAMMING WITH STYLE series, and leave the recommendation to purchase that one book a much less enthusiastic one than it otherwise would have been.

THE BASIC WORKBOOK

Creative Techniques for Beginning Programmers

Kenneth E. Schoman, Jr.

Level: Novice Rating: **70**

	Hayden 1977 Paper		
117 Pages	ISBN: 0-8104-5104-2	9" x 11"	\$7.80

BASIC as a mathematical tool is a topic which has now been treated in a number of introductory texts. Among them, BASIC BASIC and BASIC FROM THE GROUND UP provide treatments of varying complexity, with the latter requiring less grounding in precalculus-level mathematics. THE BASIC WORKBOOK continues this trend toward a less algebraically rigorous introduction to BASIC programming.

Only twenty key "words" are used in the author's presentation of minimal BASIC. Originally used as a lab manual for a high school computer course, the book contains an inordinately large portion of empty "homework" space for coding assignments. The "lecture" por-

tions of the book explain BASIC with great clarity and imagination. Yet, there is an intrinsic difficulty posed by the format: it never quite transcends the staccato classroom delivery of lecture-problems-coding required for consumption by the general public (and admirably achieved by its two predecessors, mentioned above). In addition to this format problem, more effort should have been spent elaborating upon the solutions to the many BASIC problems included in the workbook. (In some cases, the commands and functions required to efficiently solve these problems are formally introduced after the student assignment.)

THE BASIC WORKBOOK is a fine example of a high school level course outline treating the BASIC language; it fails, however, to make the transition to a BASIC text accessible to the non-student interested in learning BASIC.

A BIT OF BASIC

Thomas A. Dwyer and Margot Critchfield

Level: Novice Rating: 85

Addison-Wesley 1980 Paper
184 Pages ISBN: 0-201-03115-9 9" x 11" \$5.95

The authors who first collaborated on BASIC AND THE PERSONAL COMPUTER have distilled elements of that text into a less comprehensive but equally comprehensible introduction to BASIC. Actually, the only section not repeated from the earlier work (and this includes even the appendices), is the chapter which treats computer graphics. It is a tribute to the quality of BASIC AND THE PERSONAL COMPUTER, rather than any new effort by the authors, that A BIT OF BASIC is a superior BASIC introduction in its own right.

Illustrations and graphics have always been a Dwyer and Critchfield strong point, and this book is no exception; both new drawings and classics from PERSONAL COMPUTER artfully supplement the text. The book achieves its rather modest aim of imparting an understanding of the rudiments of BASIC PROGRAMMING, with an emphasis on graphics. The chapter which includes the authors' "8 hour wonder" course (the basis of the next spin-off?) is the starting point for a programming tour which winds through BASIC commands and ends with a brief discussion of the advanced features of extended BASIC. At each step along the way, care is taken to highlight differences in code which differentiate Microsoft, Applesoft, and Level II BASIC.

A BIT OF BASIC, though admittedly a derivative treatment of the topic, is nevertheless an effective introduction to BASIC for the beginning programmer, a fact which is more than adequately demonstrated by the continued success of its "parent" work.

CBASIC USER GUIDE

Adam Osborne, Gordon Eubanks Jr., and M. McNiff

Level: Intermediate Rating: **80**

	Osborne	1981	Paper	
215 Pages	ISBN: 0-931988-61-6		7" x 10"	\$15.00

CBASIC is a compiler version of BASIC which operates under CP/M. Compared with interpreted BASICs, it offers two major advantages. The first is speed of program execution, since a program is translated into object code or machine language only once rather than each time the program is run. The second is that it incorporates numerous extensions of BASIC which permit structured programming and modular program development. These make CBASIC much more suitable for a wide range of business and professional applications.

CBASIC USER GUIDE, one of whose authors is Gordon Eubanks, Jr.—the creator of CBASIC—provides a brief (196-page) introduction to this version of the language. Indeed, the volume's chief drawback is its brevity, considering the wide range of topics which it encompasses. Aside from examining those statements commonly discussed in introductory BASIC texts, the authors treat programming techniques and advanced programming topics such as effective formatting of input and output, CBASIC file handling capabilities and calls to assembly language subroutines. Although at times the authors merely provide programs to show how particular features of CBASIC (such as the ability to input several data fields simultaneously) can be handled most effectively, their discussion of advanced programming features is as informative as it is concise.

This text will not permit the novice to learn both BASIC and CBASIC simultaneously. We recommend CBASIC USER GUIDE to experienced BASIC programmers whose annoyance with the inconvenience of BASIC interpreters has led them to rely on compiled BASIC instead.

DATA FILE PROGRAMMING IN BASIC

A Self-Teaching Guide

LeRoy Finkel and Jerald R. Brown
Level: Intermediate Rating: **95**

	Wiley	1981	Paper	
342 Pages	ISBN: 0-471-08333-X		7" x 10"	\$12.95

This book departs from the typical Wiley Self-Teaching Guide's strict use of a frame-by-frame programmed instruction approach to mi-

crocomputer topics (see our review of BASIC: A SELF TEACHING GUIDE). Rather, it more closely resembles the conventional text's chapter-and-final-test format. Given the authors' assumption of a BASIC-knowledgeable reader and the relative complexity of their topic, this less segmented presentation is an appropriate modification of the Wiley technique.

Both technique and content combine to make DATA FILE PROGRAMMING IN BASIC an excellent study of this advanced BASIC topic. Examples are coded in a deliberately "wizedened" version of TRS-80 BASIC, with coding variations provided for Microsoft and NorthStar BASICs. (Surprisingly few are required until the random access file discussion, whose manipulation tends to be particularly sensitive to specific operating systems.)

A BASIC With Style-type precis of proper coding techniques introduces the reader to the world of file processing. 'Conservative' programming is emphasized; that is, programming which lends itself to easier debugging and ease of portability among systems. A brief review of BASIC follows, preparing the reader for the book's primary considerations—sequential and random access file processing. The methodology employed is first to explain how a file-type is coded in BASIC, and then to develop a 'utility' program which uses the new code. It is interesting to note that while some of these programs mime functions already available through DOS or CPM (like simple file copying), others are valuable as stand-alone operating aids (for example, changing header and detail data in a random access file).

The sequential file chapters highlight the addition of data to, and the merging of, these files. The random access file chapters treat the updating and search functions facilitated by their special indexing architecture. A concluding inventory application deftly combines both files in its use of a sequential file as a pointer for random access file data.

The appeal of this book is broadened by its special chapter on cassette type data files and NorthStar BASIC. Any BASIC programmer interested in the manipulation of microcomputer files should begin and end his search with the reading of this immensely comprehensible text.

DISCOVERING BASIC

A Problem Solving Approach

Robert E. Smith

Level: Novice Rating: **80**

	Hayden	1970	Paper	
164 Pages	ISBN: 0-8104-5783-0	6" x 9"	\$8.95	

Smith's approach to BASIC language instruction emphasizes the application of programming techniques to solve practical problems,

rather than the study and memorization of detailed explanations. Although most BASIC texts are now designed to be used while the reader is seated before a microcomputer, DISCOVERING BASIC is an early and extreme example of this technique: the reader begins to enter BASIC programs from page three onward, while discussion of BASIC syntax and statements is kept to a bare minimum, and practice is substituted for theory. The "lesson" section of the book occupies only the first ninety-five pages, while the remaining sixty-nine pages are devoted to review problems, in which the reader must write his or her own BASIC code based on Smith's flowcharts.

Although DISCOVERING BASIC was written before the appearance of microcomputers, it nevertheless has preserved its value as an instructional text. Much to his credit, Smith has not presupposed the stupidity of his audience. Beginning with very elementary exercises, the reader quickly progresses to more complicated programs which require some mastery of BASIC. This rapid advance in the programming abilities of the novice is made possible by using on-line programming to consolidate the knowledge gained from earlier material, and by supplying the reader with a wide range of challenging mathematical and statistical problems to solve.

Despite its dated treatment and its focus on mainframe BASIC, we recommend DISCOVERING BASIC to those beginners who feel that their programming ability will be enormously enhanced by using a sound problem-oriented text.

ELEMENTARY BASIC

Henry F. Ledgard and Andrew Singer

Level: Novice Rating: **65**

	Vintage	1982	Paper	
264 Pages	ISBN: 0-394-70789-3		6" x 9"	\$12.95

See our review of ELEMENTARY PASCAL on page 153.

FIFTY BASIC EXERCISES

Jean-Pierre Lamoitier

Level: Novice Rating: **85**

	Sybex	1981	Paper	
231 Pages	ISBN: 0-89588-056-3		9" x 11"	\$12.95

On the premise that "...the most effective way of learning a programming language is through actual practice," FIFTY BASIC EXERCISES

and its customized version, BASIC EXERCISES FOR THE IBM PERSONAL COMPUTER, purport to teach the reader with "a scientific or technical background" how to program in BASIC. In this context, the introductions which follow are most curious. Elementary flowcharting and a short example of a single BASIC program lead directly into algebraic exercises with thirty-statement program solutions. Syntax, individual comments and programming techniques are mentioned only in a nine-page appendix.

Once this "in medias rem" approach to BASIC is overlooked, however, a very eclectic series of BASIC exercises remain. Even the author's categorizations of problem types are novel: analytic geometry, data processing (that is, sorting and merging files), mathematical and financial computations, games, operations research (critical path analysis) and statistics. Problem complexity ranges from simplistic sales forecasting exercises to the coding of the min/max values in calculus equations. A common theme is the use of the microcomputer as a tool for solving mathematical problems.

FIFTY BASIC EXERCISES is a BASIC BASIC for the scientific, technical or mathematically adept programmer. Despite the author's claim to the contrary, it is not an introduction to the BASIC programming language; it will, however, provide the experienced programmer with a wide range of application exercises for his personal computer.

INSTANT FREEZE-DRIED COMPUTER PROGRAMMING IN BASIC 2nd Edition

Jerald R. Brown

Level: Novice Rating: **10**

	Dilithium 1982 Paper		
195 Pages	ISBN: 0-918398-57-6	9" x 11"	\$14.95

Jerald R. Brown, the co-author of a number of volumes in the Wiley Self-Teaching Guide series, has written what he modestly believes is "the greatest introductory tutorial on BASIC." This is a guide for the accidental microcomputer owner whose shock at discovering that the computer has a keyboard and therefore requires typing is surpassed only by the horror which the very thought of using the computer induces in the first place. We doubt that very many such readers exist, or that those who do will ever want to abandon their pre-written software packages in favor of BASIC programming.

Brown's instructional technique includes: an attempt to help the reader locate the proper key on the keyboard; an emphasis on numerous

pictures to stimulate the attention of the reader; the presentation of only very short programs in order to not overwhelm an undoubtedly confused reader; and the use of notations to remove any lingering doubts about whether a given portion of the text is to be read or entered into the microcomputer. BASIC instruction itself progresses slowly, although only the bare essentials of the language are offered. In short, Brown's technique is based on the assumption that learning is a painful process which progresses most smoothly if the mind of reader is properly anesthetized by humor, pictures and asides, and if all of the complexity is blanched from a topic. The overwhelming arrogance and condescension of this approach make Brown's book an unpleasant work which anyone who really wants to learn BASIC would do well to avoid.

INTIMATE INSTRUCTIONS IN INTEGER BASIC

Brian D. Blackwood and George H. Blackwood

Level: Novice Rating: 20

Sams	1981	Paper		
158 Pages	ISBN: 0-672-21812-7	5" x 9"	\$7.95	

This book is perhaps interesting as a sociological study or as a document testifying to a profound generation gap; as a text designed to teach the reader Integer BASIC, it is an unmitigated catastrophe. Despite the introduction, which assures the reader that "this book is written at the lowest possible level" and "it is assumed that you, the reader, have no knowledge of programming," (9) the authors' progression from topic to topic is so rapid and their explanations so cryptic that the book succeeds in disseminating confusion instead of knowledge. Computer programming from this approach is bound to appear more akin to magic than a rational method for solving problems.

The book is written by a father, George H. Blackwood—who has no real knowledge of computers—and his son, Brian D. Blackwood—who is a computer engineer. The preface was contributed by the elder Blackwood, who pathetically describes his attempts to learn to use his computer and expresses gratitude to his son for the consultations which prevented his "feeble efforts at programming" from being terminated;(10) even so, "the struggle goes on slowly, very slowly."(10) This preface, it seems to us, is a dead giveaway that Brian D. Blackwood's teaching techniques are in need of major revision.

But what is inconceivable is why anyone would want to use only Integer BASIC. As its name implies, it is a variant of BASIC that handles only integer numbers (i.e., whole numbers); decimals can be handled

only through a variety of programming machinations. In addition, any number in excess of $\pm 32,767$ cannot be handled by the computer. In practice, this means that if Integer BASIC is used in a checkbook balancing program, for example, deposits, checks or the account balance cannot exceed \$327.67 or the computer will produce an error message. As a result, "the best computer calculations are not as exact as the hand-held calculator calculations." (36) For those readers seriously considering a microcomputer with Integer BASIC, we would suggest that the \$7.95 which might be used for the purchase of this book be set aside toward the purchase of a hand-held calculator.

INTRODUCTION TO COMPUTER PROGRAMMING

Basic For The Beginners

Brian Reffin Smith

Level: Novice Rating: **80**

	Usborne	1982	Paper	
48 Pages	ISBN: 0-86020-674-2		6" x 9"	\$2.95

The difference between an "Usborne" book and an "Osborne" book is wider than the ocean which separates their respective publishers (Usborne is a British imprint). Unlike its Osborne counterpart, no Usborne title, with its comic-book illustrations and characters, will ever claim to be intellectually rigorous. It will claim, however, to "provide a fun introduction to computers and computing for absolute beginners." This is precisely where INTRODUCTION TO COMPUTER PROGRAMMING succeeds.

Most programs are written in a version of Sinclair BASIC for a predominately pre-teen audience. Robots and mechanical "bugs" inhabit a colorful landscape which serves as the backdrop for a series of frames treating programming concepts. Though the transition from simple BASIC commands to a full-fledged program is a bit precipitous, most of the elements of minimal BASIC, including graphics, nested loops and string variables, are eventually covered. The "no computer needed" emblem which appears on the front cover is highly suspect, however, since much of the book's learning is accomplished through short game programs which demonstrate programming principles.

INTRODUCTION TO COMPUTER PROGRAMMING is, in short, a short, light presentation of fundamental BASIC programming to a pre-teen "absolute beginner." Within its admittedly limited context, it receives our recommendation as a painless first book of programming.

THE LITTLE BOOK OF BASIC STYLE

How To Write A Program You Can Read

John M. Nevison

Level: Intermediate Rating: **90**

	Addison-Wesley	1978	Paper	
151 Pages	ISBN: 0-201-05247-7		6" x 9"	\$5.95

John Nevison's answer to Strunk & White's *ELEMENTS OF STYLE* is a book that treats the role of clear documentation in BASIC programming. The author contends that while the popularity of microcomputers has contributed to advancing computer literacy, computer "fluency" has not been equally well served.

The rationale for such coding in a business environment has always been the ease with which such readable code could be maintained; Nevison's approach, however, is an appeal to aesthetics. Sloppy code contains few comments, has no spaces between statements, is without any indentation, and is generally difficult to follow. Strong, "clean" code, on the other hand, is consonant with the *LITTLE BOOK*'s nineteen rules of stylistically "correct" coding, covering initializing, spacing and labeling conventions. "Before" and "after" examples are scattered throughout the book. Nowhere is their importance more evident than in the relatively long sample programs which Nevison includes in the final chapters.

Experienced programmers familiar with self-documenting programming techniques will find little use for the *LITTLE BOOK OF BASIC STYLE*. Intermediate programmers whose expertise in BASIC appears to be growing in direct proportion to the incomprehensibility of their code should confine their next "peek" to the text between its covers.

MICROSOFT BASIC 2nd Edition

Ken Knecht

Level: Intermediate Rating: **80**

	Dilithium	1983	Paper	
162 Pages	ISBN: 0-88056-056-8		6" x 9"	\$13.95

Two versions of MICROSOFT BASIC are the focus of this book: BASIC-80, release 5.0, and TRS-80 Model III BASIC. The author forsakes the traditional narrative approach characteristic of most BASIC surveys to compile an expanded glossary of the commands, functions and special features of this popular BASIC dialect.

An individual command is first categorized as belonging to one of several general function groups (elementary commands, strings, editing, disk commands, array and file commands, authentic functions, branching and loop commands). Within this group, the command is then "parsed" in each of its syntactical variations, with sample code providing actual BASIC statement examples. Most of these definitions are surprisingly comprehensive. For example, a chain to a disk file in MICROSOFT BASIC using the CHAIN command is explained with each of its possible parameters, from the most simple chain (chain "prog1") to its more esoteric variations (chain merge "prog1", 2000, delete 1000-5000). In the course of this description, Knecht refers to related BASIC statements which either expand or delimit the function of the command being discussed. In this example, the COMMON statement, we are told, will provide additional parameter-passing ability, while the existence of a DEFSTR statement in the original program will override any parameters passed by our CHAIN.

MICROSOFT BASIC is not the ideal introduction to the BASIC language for the non-programming user; it is, however, perfectly suited for the experienced programmer looking for a quick guide to mastering the nuances of a specific BASIC dialect.

POCKET COMPUTER PROGRAMMING MADE EASY

Jim Cole

Level: Novice Rating: **10**

	ARCsoft	1982	Paper	
128 Pages	ISBN: 0-86668-009-8		6" x 8"	\$8.95

The price is high, the pages are few, and the print is large, but the book's contents, as Cole repeatedly points out, are fun. The fun includes a "novelty 'alphabet counter'" (55-56) requiring the user to enter a 27-line program—plus an additional line if the user wants the pocket computer to beep at the program's completion—which successively displays each letter of the alphabet from A to Z. (Cole could have saved the reader a lot of boring typing by presenting a three-line variant of this program using the CHR\$ function, which is available on two of these pocket computers.) Loops in general seem to be fun. (61) But perhaps the most fun of all is the beep function, "a very special form of pocket computer output that very, very few other computers have." (56) Non-pocket computer owners should be thankful for small blessings.

But the astute reader will not be misled by the book's apparent infanthood. Cole's emphasis is both on structured programming (as his

eight-line program with only five GOTO statements on page 67 illustrates) and on a maximum economy of code in order not to deplete the pocket computer's limited memory (witness his novelty alphabet counter, discussed above). In his comprehensive overview of BASIC for pocket computers, Cole concentrates above all on PRINT statements (and in fact on very little else) and provides a detailed explanation of each of the lines of his short programs (as if they weren't already transparent enough).

The book's back cover presumptuously notes that "the BASIC instruction in this book is applicable to any computer, whether pocket-size or desk top. It's the perfect introduction to using BASIC in any computer." We suggest, on the other hand, that this is one book which even the owner of a pocket computer would do well to overlook.

PROFESSIONAL PROGRAMMING TECHNIQUES Starting with the BASICS

Richard Galbraith

Level: Novice Rating: **80**

Tab 1982 Paper
301 Pages ISBN: 0-8306-0128-7 5" x 9" \$10.95

Introductory BASIC texts have a very motley character. Many assume a shell-shocked reader who will find any higher-level language complex beyond words, so that they present only a subset of the language, focusing especially on PRINT and Assignment statements. Others share this assumption to a lesser degree, so that while their exposition of the elements of the BASIC language is more complete, they leave the reader with very little idea of how to combine BASIC statements into a meaningful program. A diametrically opposite assumption holds that programming is an elementary and transparent activity which requires little or no explanation; the reader can be safely left to glean whatever he or she needs to know largely from inspecting BASIC program lines. Fortunately, very little of any of these approaches is evident in Galbraith's PROFESSIONAL PROGRAMMING TECHNIQUES STARTING WITH THE BASICS. It attempts to steer a middle course by placing an equal emphasis on both BASIC language instruction and the development of effective programming techniques.

In comparison to other works, the book's treatment of BASIC is remarkably complete. Although Galbraith does ignore user-defined functions (the DEF FNX statement) and concentrates only on intrinsic

functions (such as RND, ABS, INT or LOG), this oversight is more than compensated for by his discussion of techniques and commands for handling disk files (he deals with sequential access files only).

But it is the emphasis on programming techniques which makes the book especially valuable for the beginning programmer. Whereas many introductory works highlight good programming practices in an abstract and general way, Galbraith's approach is extremely practical and detailed. For example, while most writers note in passing that good documentation is a necessity, the exact appearance that this documentation should take is seldom spelled out for the beginning programmer. Galbraith, on the contrary, not only discusses documentation in some detail, but even provides several examples of what effective documentation should look like. As a result, PROFESSIONAL PROGRAMMING TECHNIQUES STARTING WITH THE BASICS is one of the better introductions to BASIC available for the beginning programmer.

UNDERSTANDING BASIC

Richard G. Peddicord

Level: Novice Rating: **50**

	Alfred Publ.	1982	Paper	
48 Pages	ISBN: 0-88284-146-7		4" x 11"	\$2.95

Although BASIC is usually regarded as a programming language which is both straightforward and relatively easy to learn, a thirty-eight page text devoted to teaching the reader BASIC can convey only the bare backbones of the language in an extremely reduced format. This is the case with this volume in the Alfred Handy Guide series, which focuses on a selection of BASIC statements (assignment statements, REM, PRINT, READ, DATA, IF...THEN, FOR...NEXT, as well as those statements defining character strings and arrays) at the exclusion of all else. Large portions of the BASIC language (such as a functions and subroutines, for example) have been omitted, as has the attempt to teach the reader some fundamental programming techniques. The reader might legitimately expect the glossary which has been provided at the end of the book to convey some additional information about BASIC statements not discussed in the text, but in fact UNDERSTANDING BASIC's nine-page glossary defines a series of terms which are unrelated to the book's contents.

It is an especially inadequate text for the beginning BASIC user. In addition to the absence of any discussion of programming techniques, adequate explanations are rarely provided in this fast-paced book. Most explanations of BASIC statements are to be found in screened boxes surrounding a sample program or sample computer run. The net effect is

to overwhelm the reader with too large an amount of poorly organized information in too little space.

FORTRAN

BASIC FORTRAN

James S. Coan

Level: Novice Rating: **75**

Hayden 1980 Paper
235 Pages ISBN: 0-8104-5168-9 6" x 9" \$9.95

In this book, James S. Coan, the author of BASIC BASIC and ADVANCED BASIC, attempts to teach the beginning or intermediate programmer the basic features of FORTRAN. He largely focuses on FORTRAN IV along with a few features of FORTRAN 77, although he makes no attempt to specify his criteria for selecting one standard over the other, nor does he indicate which standard is being used and how they differ. Unlike some other books on FORTRAN programming, Coan's book is written for microcomputer users; all of the programs used in it were developed on a Cromemco microcomputer.

Unfortunately, the strong points of BASIC BASIC and ADVANCED BASIC are the weak points of BASIC FORTRAN, since Coan's approach to teaching the reader these two programming languages is identical: he attempts to combine instruction in a programming language with its use to solve mathematical problems. As a result, that portion of the text of BASIC FORTRAN which deals with the FORTRAN language per se is much shorter than the two-volume guide to BASIC, and much of what has been omitted are the detailed explanations which enable the completely inexperienced programmer to understand exactly what it is that FORTRAN statements are intended to accomplish. Because of this, although the book's treatment of FORTRAN is adequate for someone with prior programming experience, it is not an especially useful text for the novice. In other cases, the level of detail is insufficient for virtually any reader; this is the case, for example, in his discussion of the manipulation of external data files, to which Coan devotes scant attention because "there simply isn't room in this book to cover the subject thoroughly."

(93)

FORTRAN FUNDAMENTALS: A SHORT COURSE

Jack Steingraber

Level: Novice Rating: **45**

	Hayden	1975	Paper
90 Pages	ISBN: 0-8104-5860-8	5" x 10"	\$6.50

This is an introduction to FORTRAN which is not intended as an independent text or work of reference; the author has designed it to accompany the FORTRAN language reference manual for any particular computer. If this feature tends to reduce the book's attractiveness, so too does its 1975 copyright date; the book attempts to explain to the reader the rudiments of FORTRAN IV, rather than the newer and, for many microcomputer users, more common FORTRAN 77. Finally, the book is intended primarily for beginning programmers working on a mainframe computer; it is not at all oriented to microcomputer users.

The net effect of these three limitations is to severely reduce the value of this book, whose text (90 pages) is simply too short to be of much value to the novice programmer, and especially to someone attempting to learn FORTRAN without any formal instruction.

FORTRAN IV 2nd Edition A Self-Teaching Guide

Jehosua Friedmann, Philip Greenberg, et.al.

Level: Novice Rating: **85**

	Wiley	1981	Paper
499 Pages	ISBN: 0-471-07771-2	7" x 10"	\$12.95

The authors of this volume in the Wiley Self-Teaching Guide Series have devoted almost five hundred pages to a topic which most other authors manage to cover in about half the space. The result is a tedious, plodding work which nevertheless is quite effective in introducing the novice to FORTRAN.

Some of these extra pages are devoted to discussions of such topics as object-time FORMAT statements which are outside the scope of most introductory texts. In addition, the authors have done an excellent job of incorporating 1977 ANSI Standard FORTRAN into their discussion and contrasting it with the 1966 Standard. An appendix summarizing these differences in tabular form, however, would have been most useful. Finally, part of the rationale for the text's length is that it is "heavily oriented to microcomputers. Ninety percent of the material can be im-

plemented on microprocessors." In fact, the treatment of topics which are explicitly identified as microcomputer-related is largely confined to mentioning that one of the authors, Alan Hoffberg, has written a separate supplement on FORTRAN 77 for the Apple II. Almost no other attempt is made to distinguish between Standard FORTRAN and Basic FORTRAN for small computers; the latter is not even mentioned until page 334.

Despite the book's thorough treatment of the language, the techniques used to convey particular concepts are not always effective. The authors have attempted to select "fun," light examples which at times are inappropriate. For example, storing the phrase, "These statements are ridiculous" in a six-element vector (p. 273) completely obscures the way in which programmers use arrays. No complete program is presented until page 279, although the reader should have a sufficient command of FORTRAN to understand one far earlier. The questions which occur throughout the text are often trivial and non-challenging, while only after the seventh chapter do the exercises ask the reader to write complete FORTRAN programs. Programming itself is defined as something which causes nervousness.

While FORTRAN IV emphasizes structured, modular programming, the text itself makes for unstructured reading. The authors have included GOTO statements in numerous frames to indicate where the reader will find a more complete explanation of a topic. Worse, the authors have forgotten to include a GOTO when laying down the rules regarding variable types. The reader is told that names for integer variables must begin with the letters I through N (p.65). Almost three hundred pages later, he learns that these default definitions can be overridden by either the IMPLICIT statement or type declarations.

Pending the publication of a sound introduction to FORTRAN for microcomputer users, we recommend FORTRAN IV as a volume which manages to rise slightly above the level of mediocrity achieved by competing FORTRAN titles.

FORTRAN IV POCKET HANDBOOK

Daniel E. Alexander and Andrew C. Messer

Level: All Rating: 65

McGraw-Hill	1972	Paper	
91 Pages	ISBN: 0-07-001015-3	6" x 8"	\$3.95

The authors of the FORTRAN IV POCKET HANDBOOK have attempted to include in one small reference guide "a large amount of information normally found only in bulky FORTRAN text and reference books."(i) Any programmer who spends vast amounts of time thumbing through a variety of reference works to find the precise syntax of a given

FORTTRAN statement can readily appreciate both the need for, and the value of, such a compact guide. The appeal of this particular handbook, however, is limited by two considerations. First of all, it is a guide to statements in FORTRAN IV, the 1966 standard version of the language, and does not include statements from FORTRAN 77, the 1977 standard. In short, the book at this point is in need of revision and updating. Secondly, the utility of this book to microcomputer programmers is even more strictly limited, in part because some microcomputer versions of FORTRAN are subset FORTRAN 77, but also because the format of statements allowed in most microcomputer FORTRANs deviate somewhat from the FORTRAN standards.

The contents of this book lessen its quality still further. Although those sections of the text which deal with FORTRAN IV are still useful, a sizeable portion of the book is completely unrelated to FORTRAN programming. The first 23 pages include information on key punch machines, punched cards, teletypewriters and number systems; much of this information is primarily of historical interest, as disk storage and CRTs increasingly replace key punch devices and punched cards. In addition, each of the first thirteen sections conclude with one or two pages for the reader to enter his or her own notes. Each of these pages, instead of being left blank, however, contains a 15x26 box grid, which is guaranteed to render any notes that the reader might enter partly illegible. In short, the archaic character of some of the book's information, its low production quality and its limited audience all point to the need for a newer, better, and more user-friendly pocket handbook to FORTRAN.

FORTRAN WITH STYLE: PROGRAMMING PROVERBS

Henry F. Ledgard and Louis J. Chmura

Level: Novice Rating: **55**

Hayden	1978	Paper	
160 Pages	ISBN: 0-8104-5682-6	6" x 9"	\$8.95

See our review of BASIC WITH STYLE on page 134.

POCKET GUIDE TO FORTRAN

Philip Ridler

Level: Intermediate Rating: **70**

Addison-Wesley 1982 Paper
57 Pages ISBN: 0-201-07746-9 4" x 6" \$6.95

POCKET GUIDE TO FORTRAN is intended largely as a reference work for the FORTRAN programmer who either wishes to double check the precise form of FORTRAN statements or who has forgotten some of the precise details of FORTRAN programming. Very much like McGraw-Hill's FORTRAN IV POCKET HANDBOOK, then, this handy little guidebook attempts to dispense with the need to thumb through the pages of numerous bulky reference works; it is not primarily intended to teach the reader how to program in FORTRAN. Its utility to the FORTRAN programmer, however, is most apparently limited by two considerations: the first is that "the Guide is written for Standard FORTRAN," although the author does not specify which standard is used (in fact, it is the 1966 standard); the second is that the work does not attempt to examine the variations of FORTRAN available for microcomputer systems.

POCKET GUIDE TO FORTRAN is in many respects a vast improvement over the comparable McGraw-Hill reference booklet. Far more stringent criteria have been applied in determining what material should and should not be included, so that this guide is strictly confined to the details of FORTRAN programming. Its discussion of FORTRAN is generally comprehensive and of extremely high quality, with its only notable weakness being the treatment of FORTRAN functions. A number of valuable tables such as those which illustrate the results of mixed-mode arithmetic have been included, which enormously enhances the utility of the book.

At the same time, however, it is not altogether clear that the production format of this pocket guide is consistent with its stated objectives. Pages are numbered on one side in ascending order from 1 to 28, and in descending order on the other from 29 to 57. This makes the book somewhat bulky and cumbersome to use, however small its size. In addition, the overwhelming majority of the book consists of text and explanation, while the reader's expectation of a convenient guide to the format of various FORTRAN statements is poorly executed.

PROGRAMMING PROVERBS FOR FORTRAN PROGRAMMERS

Henry F. Ledgard

Level: Novice Rating: **55**

Hayden 1975 Paper
130 Pages ISBN: 0-8104-5820-9 6" x 9" \$8.95

See our review of BASIC WITH STYLE on page 134.

UNDERSTANDING FORTRAN

Herbert R. Ludwig

Level: Novice Rating: **50**

Alfred 1981 Paper
61 Pages ISBN: 0-88284-148-3 4" x 11" \$2.95

This volume in the Alfred Handy Guide series attempts "to introduce the reader to the FORTRAN language," although who this reader might be has not been specified. It should certainly not, however, be the beginning programmer, who needs to learn general programming techniques along with the way in which they are implemented by FORTRAN statements; fifty-two pages of text simply do not provide ample space to discuss the rudiments of FORTRAN fully and comprehensibly for the novice. This book is adequate, however, for those who wish to learn some of the basics of FORTRAN as a second language. Material is organized primarily by subject (handling of alphanumeric data, real/integer mode arithmetic) or by statement (input/output statements, control statements, do loops). The chief limitation of the volume is its lack of thorough, effective and detailed treatments of each topic, as might be expected from the size and price of the book.

PASCAL

A FIRST COURSE IN COMPUTER PROGRAMMING USING PASCAL

Arthur Keller

Level: Novice Rating: **85**

McGraw-Hill 1982 Paper
306 Pages ISBN: 0-07-033508-7 5" x 10" \$14.95

Keller's textbook of Standard Pascal aims at instilling sound, top-down programming techniques at the same time it teaches the reader Pascal. It succeeds extremely well in fulfilling both goals, and especially the former; the chapter on programming correctness and debugging, for example, is one of the best short introductions we've read. Until the last few chapters of his book (dealing with records, sets and files—a traditional weakness in Pascal texts) in which Keller largely relies on his programs to give the reader a sense of how to use these Pascal statements, Keller's approach to Pascal is detailed, comprehensive and informative.

One of the most notable features of Keller's treatment of Pascal is the attention he devotes to incorrect programming statements or procedures—i.e., to the use of syntactically correct statements to produce results which the programmer does not intend; after the reader is asked to identify the problem and propose a correct programming solution, Keller then proposes his own. This approach, which is rarely encountered in most books on programming, serves to highlight Keller's method, which focuses on the use of Pascal to solve practical programming problems.

Many introductions to a structured programming language seem to assume that sound programming techniques are an inevitable outgrowth of using the language. By rejecting this approach, and simultaneously emphasizing programming techniques in general and Pascal programming in particular, Keller has written a work which is highly effective in introducing the newcomer to Pascal programming.

ELEMENTARY PASCAL

Henry F. Ledgard and Andrew Singer

Level: Novice Rating: **65**

Vintage 1982 Paper
266 Pages ISBN: 0-394-70800-8 6" x 9" \$12.95

Hard on the heels of Nicholas Meyer's discovery of two previously unknown manuscripts by John H. Watson and of Michael Harrison's

discovery of Sherlock Holmes' alleged autobiography, Henry Ledgard and Andrew Singer claim to have unearthed a trunk full of unpublished works by Dr. Watson. The singularity of these manuscripts lies in the fact that they treat Holmes' attempts to apply a successor of Babbage's Analytical Engine in his efforts to establish detection as a science. In order to use Watson's writings to best effect, Ledgard and Singer have selected and edited them and translated their programs into two contemporary programming languages, BASIC and Pascal.

The fifty-three stories and four novels definitely written by "Dr. Watson" are all refreshing and original in both style and content. In contrast, the style of Watson's contributions to these two volumes is cramped and stilted, and the stories themselves frequently borrow key phrases or even whole sentences from his original work (such as, "To Sherlock Holmes it was always the engine," taken from "A Scandal in Bohemia").

But if someone has succeeded in dumping forged manuscripts on an unsuspecting Ledgard and Singer, they in turn are attempting to pawn off two barely passable programming texts on the unsuspecting reader. What is most striking about these volumes is their repetition, coupled, paradoxically, with their complete inadequacy of detail. Arrays, for example are used by the authors in the very first program; Holmes defines these as "table entries" (Pascal, p.27) or "each entry in the table" (BASIC, p.24), to which the authors add their own illuminating definition of "the rows of Watson's table" (BASIC, p.27). Their more detailed exposition of arrays later in the text treats arrays as "much like a chessboard," according to Holmes (BASIC, p.110, Pascal, p.134), or as "an ordinary table of entries," according to Ledgard and Singer (BASIC, p.115, Pascal, p.137). It is curious that the sole result of Holmes' and the authors' attempts to outdo each other in discussing the same programming statements is structured confusion.

More generally, both volumes attempt to place an equal emphasis on learning sound programming techniques and mastering the details of a particular programming language. Yet, the authors' approach to these two goals is reminiscent of that adopted by Ledgard in his Programming with Style series. While more attention has, of necessity, been devoted to the details of each language, minimal attention has been devoted to encouraging those features which make Pascal easier to structure, or BASIC more difficult to structure.

If one overlooks the appeal of stories involving Sherlock Holmes, these volumes offer an inferior programming text combined with an inferior introduction to BASIC or Pascal. We would have preferred to read about Holmes' original attempts to program Babbage's Analytical Engine.

FUNDAMENTALS OF MICROCOMPUTER PROGRAMMING INCLUDING PASCAL

Daniel R. McGlynn

Level: Novice Rating: **65**

Wiley 1982 Paper
335 Pages ISBN: 0-471-08769-6 6" x 9" \$14.95

Daniel McGlynn makes it quite clear that this book "is not intended as another book in the spirit of 'how to program in Pascal.'" The Wiley blurb-writer disagrees: "It features a step-by-step introduction to programming and programming languages—the increasingly popular Pascal in particular." The most complex, esoteric concept in microcomputer literature is not hierarchical data bases or microprocessor design but truth-in-advertising.

McGlynn has fashioned a book which examines the structural and linguistic roots of all programming languages. It is a treatment written more for designers of languages or language aesthetes than novice programmers. To use the author's own analogy, its intent is closer to a course in art appreciation than painting.

Introductory chapters on types of computer architecture, information theory and formal language theory are informative, but are marked by the author's irritating habit of alluding to topics without providing any further explanation. The reader never learns, for example, the differences among the four classes of computer architectures (Von Neuman, syntax oriented, direct and indirect execution). Other topics which do receive some attention might have benefited from additional elucidation: specifically, the Shannon-Hartley theorem and data flow diagrams. This important introductory section, which lays the groundwork for the programming examples which follow, represents a novel linguistic approach to program language design; however, the disparate elements of the author's discussion are not well integrated into the development of his stated theme.

Pascal as it reflects this modern view of programming is considered next. After a brief chapter reviewing Pascal's highly structured organization, the author launches into an equally cursory treatment of Pascal syntax. While this is an inferior introduction to the programmer's use of Pascal, with few examples, McGlynn's intention was not to provide another "how to" book. Unfortunately, his Pascal overview only minimally illustrates the general design features noted in the first section of the book; few references are made to any of the topics covered in the introductory chapters.

The book concludes with a reproduction of the ISO/DIS 7185 draft standard of Pascal and a survey of various mini and micro Pascal implementations, most notably UCSD Pascal. Token allusions are made to Modula-2 and ADA, two related high-level languages.

Despite an intent to discuss programming languages from a different, novel perspective, **FUNDAMENTALS OF MICROCOMPUTER PROGRAMMING INCLUDING PASCAL** rarely combines its disparate elements into the clear, cohesive discourse merited by its original approach.

INTRODUCTION TO PASCAL **Including UCSD Pascal**

Rodnay Zaks

Level: Novice Rating: 95

Sybox 1981 Paper
422 Pages ISBN: 0-89588-066-0 5" x 8" \$14.95

INTRODUCTION TO PASCAL is a work written for anyone, whether a beginner with no prior programming experience or an advanced programmer who is simply trying to learn a new language. Zaks' focus is on standard Pascal, although at the end of each chapter he notes divergent features of UCSD Pascal. At times, however, these differences are treated rather summarily; for example, only a page is devoted to UCSD's long integers, while UCSD string variables are discussed in the course of two pages. Aside from this rather slight limitation, Zaks has written a comprehensive, detailed and well-organized textbook which is one of the best introductory guides to Pascal currently available.

INTRODUCTION TO PASCAL stands out from other Pascal programming texts in two ways. The first is that Zaks has avoided laying down the "rules" of Pascal programming, opting instead to carefully examine the meaning and application of particular statements, their syntax, and the logic which dictates both statements and syntax. Because of this, the programmer can begin to view Pascal as a flexible tool to solve a problem rather than a confusing and seemingly contradictory set of rules combine to impede sound programming. Second, this book effectively applies Zaks' characteristic instructional approach to Pascal programming. Material is presented in a graduated fashion, with the more advanced topics (records, files, sets, pointers and lists) being treated in the book's later chapters. The broad scope and unusual comprehensiveness of Zaks' treatment make it valuable both as a work of reference and as a tutorial or introduction to Pascal.

All Pascal programmers, regardless of their level of proficiency or expertise, can benefit enormously from reading **INTRODUCTION TO PASCAL**.

PASCAL

George Ledin, Jr.

Level: Novice Rating: **90**

Alfred 1982 Paper
281 Pages ISBN: 0-88284-173-4 8" x 10" \$14.95

Had most programmers originally learned to code in Pascal, the later orthodontia of structured programming techniques might have been avoided. Ledin emphasizes the inherent clarity and structure of Pascal as he guides the novice with no previous programming experience through its elementary details.

After an introduction to computer concepts, Ledin's chapters logically follow the main divisions of a Pascal program. Simple, unstructured data definition statements and the classification of types of program data precede more detailed discussions of the procedural flow of the main program block. Pascal's ability to build complex definitions from simple structures is paralleled by the author's development of complicated code from simple statements. Syntax diagrams and Backus-Naur-Form notation (which are both clearly explained) are used to help the reader understand such language constructs.

Though variant records and advanced file handling techniques receive little attention, Ledin does manage to cover Pascal's use of linked list and binary tree access methods. The chapters which introduce Pascal vocabulary procedures and functions additionally highlight variations between the text's Jensen-Wirth standard and Pascal/VS, Pascal 8000, MP/Pascal, UCSD Pascal, HP 1000 Pascal, VAX-11 Pascal and Pascal 6000.

Individual chapters, whether on subprograms, procedures or sets and records, begin with a series of questions which the text proposes to answer; key concepts, words and symbols introduced in the chapter are then listed. The actual text is interspersed with summaries and reader exercises (one appendix is an answer key). Clearer, more professional presentations of programming topics are difficult to find.

If any qualification can be made of this work, it is that little consideration is given to the Pascal program's interface to the p-System operating environment. (While knowledge of p-code is not essential to an understanding of the Pascal language, both form the Pascal system needed to execute programs).

PASCAL, a programming language introduction by an experienced computer-science educator, is itself a good example of the benefits of a logical, "structured" presentation.

PASCAL: A PROBLEM SOLVING APPROACH

Elliot B. Koffman

Level: Novice Rating: **90**

Addison-Wesley 1982 Paper
426 Pages ISBN: 0-201-10341-9 6" x 9" \$14.95

Readers who find Koffman's PASCAL: A PROBLEM SOLVING APPROACH extremely informative—and we are certain most microcomputer users will—should not, however, be tempted to add his second work to their libraries. PASCAL and PROBLEM SOLVING AND STRUCTURED PROGRAMMING USING PASCAL are virtually identical; the former has been written for microcomputer users and includes UCSD Pascal, while the latter treats standard Pascal for mainframe computers.

Most authors, of course, now adopt "a problem-solving approach" to teaching programming, which features the use of programs to illustrate features of a language, and exercises to consolidate the reader's ability to use it to solve programming problems. While Koffman's treatment falls within this general rubric, his introduction to Pascal is at the same time highly distinctive. First, this approach is applied to all of Pascal, rather than to some subset of the language; the reader who completes his book will gain a good working knowledge of the entire range of Pascal statements. Second, the program examples and exercises which Koffman selects are a bit more challenging than those included in most texts, and insure that the user will actually gain mastery of Pascal programming. Third, given the diversity of Pascal statements or data types which might be used for a given application, Koffman examines their applicability for solving different kinds of problems; his discussion of user-defined variable types is an outstanding example of this. Finally, Koffman extends his problem solving approach not only to programming, but also to debugging; each chapter concludes with a discussion of the errors typically associated with individual Pascal statements.

PASCAL: A PROBLEM SOLVING APPROACH is notable for its effective combination of Pascal programming theory and practice. Because it offers a comprehensive and well-organized introduction which insures fluency in Pascal, it is a work which we highly recommend.

PASCAL FOR BASIC PROGRAMMERS

Charles Seiter and Robert Weiss

Level: Intermediate Rating: **95**

Addison-Wesley 1983 Paper
242 Pages ISBN: 0-201-06577-0 8" x 10" \$10.95

Since most microcomputers come equipped with BASIC, almost all owners have been forced to learn it at one point or another; consequently, it has become standard for authors of introductory works on Pascal for microcomputers to at least briefly compare Pascal with BASIC. Typically, such discussions focus on the unique features of Pascal, and especially on its idiosyncrasies (such as its user-defined variable types). These presentations do not, however, make effective use of the fact that the reader already knows BASIC in order to facilitate the teaching of Pascal.

In contrast, Seiter and Weiss' approach to teaching Pascal implies that if the reader already knows one programming language, the addition of a second one presents no great difficulty. The very first chapter of PASCAL FOR BASIC PROGRAMMERS contains a relatively in-depth comparison of the two languages which, instead of emphasizing the peculiarities of Pascal, focuses on its similarities to BASIC. Aside from enabling the reader to learn Pascal much more quickly and effortlessly, such an approach also leaves the reader with a real appreciation for the more highly organized and structured character of Pascal programming. By pursuing this technique of comparing and contrasting Pascal with BASIC, the authors have written one of the better introductions to Pascal currently available. Although Seiter and Weiss focus primarily on UCSD Pascal, the attention which they devote to other Pascal dialects makes this book valuable for the BASIC programmer wishing to learn any version of Pascal.

Since much of the book's strength lies in its use of BASIC to teach the reader Pascal, it is not surprising that the book is weakest when examining unique features of Pascal. Especially in the fourth chapter, which treats variable types not available in BASIC, the discussion of records, sets and pointers is simply too concise and too vague to be particularly enlightening. As we have noted in our other reviews, however, the inability to effectively present these concepts is a common failing of introductory works on programming in Pascal.

PASCAL FOR PROGRAMMERS

S. Eisenbach and C. Sadler

Level: Intermediate Rating: **90**

	Springer-Verlag	1981	Paper	
201 Pages	ISBN: 0-387-10473-9		7" x 10"	\$12.00

PASCAL FOR PROGRAMMERS is written for the reader who is familiar with any programming language and now wishes to learn Pascal. What this means, in practice, is that the authors can ignore both the presentation of basic programming concepts (although the injunction against the use of GOTO statements is not overlooked) and the explanation of common programming techniques (such as loops or arrays), while concentrating on Pascal statements themselves. The result is a short but information-packed, high quality introduction to Pascal.

PASCAL FOR PROGRAMMERS focuses on Niklaus Wirth's standard Pascal, with passing references to the differences between it and UCSD Pascal. The authors organize their introduction to the language around (usually brief) Pascal programs, which rapidly increase in complexity as the book progresses. The first five chapters, they believe, bring the programmer to a level of proficiency comparable to that of most introductory BASIC texts; the final five chapters treat "sophisticated" Pascal concepts, such as procedures (i.e., BASIC subroutines), functions, parameter passing, and recursion. These final chapters, which examine Pascal statements having no direct equivalent in BASIC, are above all noteworthy for their effective organization and clarity of presentation; it is these same concepts which are all too frequently mishandled in other introductory Pascal texts.

For the BASIC—and especially the non-BASIC—programmer who is growing increasingly weary of reading primitive definitions of loops or arrays each time he or she decides to learn a new language, PASCAL FOR PROGRAMMERS is made to order. We highly recommend it.

PASCAL FROM BASIC

Peter Brown

Level: Intermediate Rating: **80**

	Addison-Wesley	1982	Paper	
182 Pages	ISBN: 0-201-10158-0		6" x 9"	\$12.95

PASCAL FROM BASIC aims at teaching the BASIC programmer how to program in Pascal—and more importantly, how to think in Pascal;

"there is no such thing," Brown argues, "as a literal translation."⁽¹¹⁾ In an attempt to avoid "being solemn and stodgy,"(x) he has included many humorous asides in this extremely well-written text.

A short (162-page) text filled with a light, humorous tone, however, is not particularly conducive to mastering a new programming language. Brown's attempt to teach Pascal is most successful when the elements have direct equivalents in BASIC (although user-defined variables are a notable exception to this rule); similarly, the comparison of the strengths and weaknesses of Pascal and BASIC is generally well handled. But the distinctive commands in Pascal (sets, pointers, etc.) are not presented with such clarity, nor is the author's emphasis on structured programming in Pascal sufficiently detailed or thorough to restructure the BASIC programmer's approach to programming; paradoxically, Brown's approach is one from which literal translations follow as a matter of course.

Provided that the reader follow it with another, more detailed text, PASCAL FROM BASIC provides an excellent, refreshing introduction to Pascal programming for the more-or-less experienced BASIC programmer. As a single text designed to teach the user Pascal, however, it is inadequate.

THE PASCAL HANDBOOK

Jacques Tiberghien

Level: All Rating: **95**

	Sybx	1981	Paper	
485 Pages	ISBN: 0-89588-053-9		7" x 9"	\$18.95

Jacques Tiberghien's dictionary/encyclopedia of the Pascal language is as indispensable to the Pascal programmer as David A. Lien's THE BASIC HANDBOOK is to the BASIC user. Like THE BASIC HANDBOOK, this book is notable both for the meticulous care with which it has been assembled and for its high production quality, which makes using it pleasurable, convenient and rewarding.

Each entry in the handbook occupies at least a single page, which greatly contributes to this encyclopedia's ease of use. The term is briefly defined and the reader is informed of its relationship to the Pascal language (i.e., whether it functions as a symbol, an identifier or a concept) and of the Pascal dialects which support it. In all, Tiberghien focuses on six major variations of Pascal: Jenson and Wirth's original standard Pascal; Jenson and Wirth's CDC implementation; UCSD Pascal; Pascal/Z; HP Pascal 1000; and OMSI Pascal-1. Tiberghien supplies the syntax of each term, offers a detailed description of its functions and usage, examines its implementation-dependent features (although he

does not generally provide alternatives to these), and provides an example of the term's usage in a longer program so that the reader's understanding of a particular entry will be practical rather than abstract.

Perhaps the most striking feature of *THE PASCAL HANDBOOK* is the clarity and ease with which it allows the user to distinguish among these six versions of Pascal. Aside from simply noting the presence or absence of a particular statement or feature, Tiberghien also lists syntactical differences and potential errors in the implementation of given features. For this reason alone, *THE PASCAL HANDBOOK* can stand as a model for all programming language dictionaries and conversions handbooks.

PASCAL PRIMER

David Fox and Mitchell Waite

Level: Novice Rating: **85**

Sams 1981 Paper
206 Pages ISBN: 0-672-21793-7 8" x 10" \$16.95

PASCAL PRIMER is a guide to UCSD Pascal for the beginning programmer, and especially for the microcomputer user who has just learned the rudiments of BASIC. Throughout the text, the authors compare Pascal statements to BASIC equivalents. Fox and Waite's approach to Pascal is that it should be fun; the text is interspersed with asides (the life of Blaise Pascal, scientific notation explained, etc.), semi-humorous illustrations, quotations from Uncle Pascal and true/false self-tests. At times, however, this emphasis on fun obscures the learning process, since these frequent interruptions divert the attention of the reader and disturb his or her flow of thought. The true/false tests are frequently as silly as they are condescending.

The comprehensiveness with which Fox and Waite treat Pascal also varies throughout the book. Paradoxically, when the language's statements or concepts become complex, the authors give them comparatively short shrift; because of this, the later sections on user-defined data types and sets are unsatisfactory. A certain range of Pascal statements are excluded altogether; there is no discussion, for example, of the handling of files or records in Pascal. In contrast, however, the book's treatment of many of the most basic Pascal statements and concepts (standard variable types, loops, decision-making statements, subroutines and functions) is both detailed and informative. On balance, the *PASCAL PRIMER* is a very good introductory text for the beginning programmer, but one which the beginner will very quickly outgrow as he or she gains more experience in using Pascal.

PASCAL USER MANUAL AND REPORT

2nd Edition

Kathleen Jensen and Niklaus Wirth
Level: Intermediate Rating: 85

Springer-Verlag 1974 Paper
167 Pages ISBN: 0-387-90144-2 6" x 10" \$9.50

It has been argued that innovators come to be so dominated by their creations that they eventually lose sight of their own achievements, and become unable to describe or explain them satisfactorily. Fortunately, in the case of Niklaus Wirth, the creator of Pascal, nothing could be further from the truth. The PASCAL USER MANUAL AND REPORT, which he has co-authored with Kathleen Jensen, is a model of both clarity and brevity. It is, in short, a work of far more than passing historical interest.

The volume is not, however, intended to be an introduction to Pascal for the beginning programmer; instead, it provides a highly organized overview of the structure of the Pascal language. This approach is calculated to appeal to the more-or-less experienced programmer who uses some other (preferably structured) language, or to the Pascal user who wants a concise reference work. The overall value of the volume is reduced somewhat by its rather curious contents, which consist of two distinct sections: a brief (129 page) USER MANUAL, which the authors intend as a tutorial to introduce programmers to Pascal; and an even more concise (33 page) REPORT, which is largely a rehash of the USER MANUAL with all of its detail removed. But despite the fact that its treatment of the language is collapsed into slightly more than one hundred pages, the book is an important contribution to the literature on Pascal.

As a summary and surprisingly detailed description of the elements of Pascal, the USER MANUAL is excellent. Its quality is further enhanced by its treatment of the evolution of Pascal and by Jensen and Wirth's occasional comments on the applicability of some of Pascal's data structures. For those who have always wanted to learn a programming language "straight from the horse's mouth," PASCAL USER MANUAL AND REPORT is highly recommended.

PASCAL WITH STYLE: PROGRAMMING PROVERBS

Henry F. Ledgard, Paul A. Nagin, John F. Hueras
 Level: Novice Rating: **55**

Hayden 1979 Paper
 210 Pages ISBN: 0-8104-5124-7 6" x 9" \$8.50

See our review of BASIC WITH STYLE on page 134.

POCKET GUIDE TO PASCAL

David Watt

Level: Intermediate Rating: **65**

Addison-Wesley 1982 Paper
 58 Pages ISBN: 0-201-07748-5 4" x 6" \$6.95

Like the other three volumes in the Addison-Wesley Programming Pocket Guide Series which examine individual languages, this small volume attempts to serve as a "memory-jogger" or reference tool for the Pascal programmer. By no stretch of the imagination is it suitable as an introduction to Pascal programming for the novice, if only because of its organization (which focuses on broad topics such as variable types instead of individual statements). Moreover, its emphasis on syntax and the rules of Pascal make it unsuitable to the needs of the beginner.

Even as a reference tool, the book has some clear limitations. Its treatment of Pascal is accurate and detailed, and pages are packed full of information; there is, in fact, too much information presented in too little space. The result is a text which is not only hard to read, but also one whose potentially valuable tables and charts are all but incomprehensible. They are particularly gratuitous given the widespread acceptance of Niklaus Wirth's clear and informative syntactical diagrams. For those who find that they do need a memory-jogger, we recommend any text containing the Wirth diagrams, rather than POCKET GUIDE TO PASCAL.

PROBLEM SOLVING AND STRUCTURED PROGRAMMING IN PASCAL

Elliot B. Koffman

Level: Novice Rating: **90**

Addison-Wesley 1981 Paper
 430 Pages ISBN: 0-201-03893-5 6" x 9" \$17.95

See our review of PASCAL: A PROBLEM SOLVING APPROACH on page 158.

SPEAKING PASCAL A Computer Language Primer

Kenneth A. Bowen

Level: Novice Rating: **65**

Hayden 1981 Paper
 236 Pages ISBN: 0-8104-5164-6 5" x 8" \$11.95

Bowen's SPEAKING PASCAL is designed to make "learning to program in the language Pascal a pleasant experience" for "beginners with no previous programming experience." The "fun," as Bowen calls it, however, is entirely confined to the chapter headings, all of which continue Bowen's imagery of speech; hence, we have "Talking to Computers," "Money Talk," "Repetitive Conversations," etc. But once the reader moves beyond the chapter headings to the text, all of this pleasure comes to an abrupt end.

Part of the problem is a certain lack of clarity in presentation. This first becomes notable, for example, when Bowen discusses the use of the ORD and PRED functions, leaving the reader in some doubt about whether these functions rank characters according to an order defined by the computer, by the user or by both. These sources of confusion multiply throughout later chapters, as the level of detailed explanation provided by Bowen becomes increasingly inadequate for teaching the beginning programmer Pascal.

In fact very little of the text has been written with the needs of the beginning programmer in view. Bowen's approach to Pascal is a highly compartmentalized one in which each statement is examined and its rules are laid down. He rarely examines why these rules exist or how individual statements relate to the language as a whole. The result is a work which may succeed in informing the reader what individual Pascal statements should look like, but which does not aid him or her in

integrating these into a knowledge of Pascal sufficient to solve actual programming problems.

UNDERSTANDING PASCAL

George Ledin, Jr.

Level: Novice Rating: **75**

Alfred 1981 Paper
63 Pages ISBN: 0-88284-149-1 5" x 11" \$2.95

At \$2.95, this volume in the Alfred Handy Guide Series proves an outstanding buy for the programmer interested in learning Pascal. In view of the book's need to present a bare-bones, highly compact introduction to Pascal, Ledin has provided a surprisingly detailed and informative discussion of the fundamentals of Pascal. Like the the volume on BASIC included in this series, Ledin organizes his discussion of individual Pascal statements around brief programs, which serve to illustrate how the statement functions within a larger program; unlike the Alfred guide to BASIC which uses screened boxes to indicate important points or concepts, Ledin focuses on these more carefully in the book's text.

At the same time, however, the reader will only succeed in learning the bare rudiments of Pascal from reading this book. Works for the beginning programmer generally progress from extremely simple material on to the more difficult; UNDERSTANDING PASCAL does not adopt this rather straightforward approach. Instead, the concept of parameters in subroutines and functions, for example, is introduced before the discussion of data types (integer, real, character or Boolean) or of the rules for evaluating mathematical expressions.

Finally, any work which devotes only sixty or so pages to teaching a programming language must necessarily do so superficially. In this case, all of the complexities of Pascal have either been vastly simplified (e.g., the discussion of passing parameters for subroutines or functions) or virtually overlooked (e.g. the passing mention given to arrays, and to Pascal's structured data types generally). From reading UNDERSTANDING PASCAL, the beginning programmer can perhaps gain some idea of how to use the most basic statements of the language; whether that knowledge will also permit the reader to solve some practical programming problems, however, is highly debatable.

OTHER PROGRAMMING LANGUAGES

ADA: AN INTRODUCTION and ADA REFERENCE MANUAL

Henry F. Ledgard

Level: Intermediate Rating: **80**

Springer-Verlag 1981 Paper
317 Pages ISBN: 0-387-90568-5 8" x 11" \$16.80

Developed by the U.S. Department of Defense ("the largest software consumer on earth") and named for Lord Byron's daughter, Countess Augusta Ada Lovelace, the programming language Ada has remained within the confines of its original, military environment. We have included a review of the Ledgard text in this section, however, because Ada's striking similarity to other "civilian" microcomputer languages provides a unique insight into the general structure and syntax of high level programming. In addition to those programmers whose current projects mandate a knowledge of Ada, this book will be of interest to software grammarians tantalized by Ada's answers to the problems confronted, with varying degrees of success, by other programming languages.

The book's two titles, ADA: AN INTRODUCTION and ADA REFERENCE MANUAL, reflect the fact that it is comprised of two discrete texts. The first is Henry Ledgard's twelve-chapter introduction to the programming language; the second is the draft version of the proposed standard for Ada composed by Honeywell's French affiliate, Cii Honeywell Bull.

Ledgard's introduction begins with five short program examples demonstrating typical Ada features. The Pascal programmer will immediately recognize some striking structural similarities. Both languages begin programs with the strictly defined "typing" of variables; both continue with explicit descriptions of procedures, subroutines and sub-programs; and both conclude with a main executor module which sequences code execution. Even Ledgard's treatment resembles comparable Pascal texts in its description of each one of these program sections.

A discussion of general, primitive and array types precedes chapters on Ada's versions of Computational Statements. Language components are assembled into an annotated, functional program which presents the Ada solution to the classic "eight queens" problem. Final chapters treat some of Ada's specialized features, including Input/Output and parallel processing, and the handling of exception conditions.

The actual ADA REFERENCE MANUAL is an eminently less readable version of Ledgard's introduction. For programmers learning to use Ada, it serves as a convenient, definitive supplement to the author's language overview.

ADA: AN INTRODUCTION/ADA REFERENCE MANUAL provides a composite survey/reference manual to Ada. While a unified treatment may have made an initial reading more comprehensible, the double work nevertheless succeeds in imparting a novice-level fluency in the Ada programming language.

ANS COBOL 2nd Edition A Self-Teaching Guide

Ruth Ashley

Level: Novice Rating: **65**

Wiley 1979 Paper
265 Pages ISBN: 0-471-05136-5 7" x 10" \$9.50

Using a frame-by-frame approach to introductory programming, this Wiley Self-Teaching Guide covers much of the same material as STRUCTURED COBOL (see our review). According to the author, while STRUCTURED COBOL also teaches American National Standard COBOL, ANS COBOL does not necessarily teach structured COBOL. That ANS COBOL and structured COBOL represent mutually exclusive approaches to this programming topic should be cause for some concern in the COBOL programming community.

An equally insightful description of COBOL follows. Each of the four main divisions of a COBOL program are explored, with coding exercises reinforcing the reader's facility with statement syntax and reserved word vocabulary. (We assume that the presence of multiple GOTO's in a few of the program examples is to be construed as support for the introductory disclaimer.) File types are considered next. Experienced COBOL programmers (one of the many groups who should definitely not read this book) will enjoy the opportunity to reminisce about the good-old-days in a chapter which treats card files. The shortcomings of the book's simplistic approach become apparent as the discussion evolves to include more technically complex topics: the chapter on disk files, for example, inadequately explains different file access techniques.

While ANS COBOL will impart an adequate rudimentary knowledge of COBOL programming, it is by no means a superior introduction.

THE C PRIMER

Les Hancock and Morris Krieger
Level: Intermediate Rating: **90**

McGraw-Hill 1982 Paper
235 Pages ISBN: 0-07-025981-X 6" x 9" \$14.95

C, a programming language developed by Dennis Ritchie to operate with the Bell Telephone Laboratories' UNIX operating system, is of interest especially to systems programmers involved in designing operating systems, compilers or interpreters. Hancock and Krieger have written THE C PRIMER to serve as an introduction to Ritchie and Brian W. Kernighan's THE C PROGRAMMING LANGUAGE, which, they believe, makes for relatively difficult reading even for programmers experienced in other languages.

But instead of being simply an introduction to another more difficult introduction, this short book is able to stand on its own merits. Hancock and Krieger have written a light, sometimes humorous text which is enormously effective in familiarizing the reader with C programming. The book is successful largely because its treatment is tailored to those with experience in other languages. This permits the authors to begin with a sample program ("How C Looks"), and gradually to introduce more complex ones as illustrations of the features of C. They have also avoided line-by-line analysis of their program examples and concentrated instead on using them to depict the general features of C. Individual elements of their discussion—such as the treatment of recursion—are among the best and clearest we've read. An appendix provides a highly organized overview of C commands and functions not discussed in the text.

Without question, the novice will find THE C PRIMER difficult; the book's organization presupposes some exposure to high-level languages. For example, input/output, which usually serves as the starting point for an introductory work, is treated only in the last chapter. But the elementary explanations of such concepts as global and local variables or structures (i.e., records) sometimes makes the text tedious as well for those with previous exposure to higher-level languages. This shortcoming is almost inevitable, however, given the authors' audience: the diversity of other high-level languages, and the importance of some features of low-level languages in C (such as its extensive use of pointers) requires such a treatment.

With the growing popularity of UNIX as well as the implementation of C in microcomputer systems independently of UNIX, C can be expected to increase in popularity. THE C PRIMER provides an excellent, informative introduction to the language.

COBOL WITH STYLE: PROGRAMMING PROVERBS

Louis J. Chmura and Henry F. Ledgard

Level: Novice Rating: **55**

	Hayden	1976	Paper	
148 Pages	ISBN: 0-8104-5781-4		5" x 8"	\$9.95

See our review of BASIC WITH STYLE on page 134.

DISCOVER FORTH

Learning and Programming the FORTH Language

Thom Hogan

Level: Novice Rating: **35**

	Osborne	1982	Paper	
145 Pages	ISBN: 0-931988-79-9		7" x 10"	\$14.95

The inspiration for DISCOVER FORTH came from the author's frustration at "wandering through the aisles of bookstores looking for any book that would explain FORTH to me."(1) Unfortunately, the publication of DISCOVER FORTH has done nothing to change this. Two-thirds of the way through his text, Hogan notes, "So far, what you've learned about FORTH makes it no more sophisticated than a memory calculator..." (88) The second-to-last chapter begins with, "If you were wondering if all the information in this book would ever coagulate into a coherent whole..."(108) At the very end of the book, Hogan abruptly redefines the object of DISCOVER FORTH; it aims merely at helping the reader decide whether or not "to proceed with using FORTH."(119) Hogan recommends that those who wish to do so begin to gather more information about FORTH programming.

While the functional building blocks of FORTH programming are "words," few FORTH programmers would equate words with programs. Hogan, however, fails to present a single example of even a simple FORTH program in the course of his text, and instead concentrates on presenting individual FORTH words. He is more successful in making a case for FORTH'S efficiency and speed of execution vis à vis other high-level languages, but this argument badly misses the point: for the user to reap the benefits of this greater efficiency, he or she must be able to write something other than the most mundane of programs. DISCOVER FORTH does not even attempt to teach the reader to do this.

The most interesting part of the book is its very first page, where Hogan discusses FORTH'S appeal. It is a language, he implies, which

tends to attract eccentric programmers while it underscores their idiosyncrasies. To its adherents, FORTH "is not a programming language, but a religion."(4) We are confident that very few readers of DISCOVER FORTH will wish to join the sect.

FORTH PROGRAMMING

Leo J. Scanlon

Level: Novice Rating: 45

Sams 1982 Paper
246 Pages ISBN: 0-672-22007-5 5" x 8" \$16.95

The way in which an author chooses to teach a programming language bears a direct relationship to the reader's ability to learn to wield it effectively. In his excellent work on BASIC programming, for example, Lewis Rosenfelder argues that the user-defined function statement is one of the most under-utilized features of the BASIC language; he finds the explanation for this in the tendency of authors to select a mundane and practically useless example to illustrate the use of the concept.

Similarly, this use of inappropriate techniques and examples substantially detracts from the value of this Blacksburg Continuing Education Series introduction to FORTH-79 and fig-FORTH. As was the case in the Blacksburg group's introduction to MMSFORTH, Scanlon has chosen the stack-oriented character of FORTH PROGRAMMING as his point of departure. The opening chapters of the book detail the manipulation of the stack with a series of commands and arithmetic operations using Reverse Polish Notation. From these tedious chapters, the reader rapidly forms the opinion that FORTH provides a lengthy, expensive and complex method of performing the same operations which are ideally suited to a four-function calculator. Since a general intuitive principle of computer programming is that computers should not be used when other traditional methods are more effective, we suspect that most readers' attempts to learn FORTH by reading this book will be short-lived.

What is more, once the reader goes beyond these initial chapters, the author's instructional technique does not improve. Scanlon's treatment of FORTH is organized around the examination of FORTH words (a "word" is a series of characters which together define a single operation) in individual categories; chapters treat such topics as memory operations, loops, string processing and disk operations. In the course of the text, the author focuses primarily on explaining individual commands, and rarely provides examples of any but the most elementary FORTH programs. As a result, the reader's knowledge of FORTH is likely to

consist of a series of commands, which cannot be integrated into a program for any but the most simple applications.

A GUIDE TO PL/M PROGRAMMING FOR MICROCOMPUTER APPLICATION

Daniel D. McCracken

Level: Novice Rating: **70**

Addison-Wesley 1978 Paper
 262 Pages ISBN: 0-201-04575-3 8" x 9" \$16.95

PL/M-80, developed by Intel in 1973, represents an early example of a high-level structured programming language for microprocessors. McCracken's introduction to Intel's 1976 revision of PL/M serves as a reminder that structure alone is not an adequate substitute for flexibility.

McCracken's GUIDE, which was first published in 1978, is dated in a number of respects. Most obvious is the fact that it does not treat PL/M-86, the more flexible version developed by Intel for use with its 8086 microprocessor. But it is dated, too, insofar as it teaches a programming language which, not surprisingly, was firmly shaped by conditions specific to the early history of microcomputers. The small size and high cost of memory account for the language's awkwardness—reflected in its ability to handle only integers—and its limited precision. (A single item of data can occupy either one or two bytes in memory.) The use of microprocessors as dedicated devices or controllers, and the lack of development of microcomputer systems are reflected in McCracken's inclusion of case studies on such esoteric topics as furnace control or the numerical control of a machine tool.

GUIDE TO PL/M PROGRAMMING, while it retains some relevance as an introduction to high-level programming for industrial control applications, is now of almost exclusively historical interest to microcomputer users. We recommend that those who wish to learn a structured programming language look elsewhere.

INTRODUCTION TO FORTH

Ken Knecht

Level: Novice Rating: **45**

Sams 1982 Paper
 142 Pages ISBN: 0-672-21842-9 5" x 9" \$9.95

This volume in the Blacksburg Continuing Education Series provides a brief overview to MMSFORTH, a version of FORTH developed

by Miller Microcomputer Services for the TRS-80 Model I and Model III microcomputers. Although Knecht does not presuppose that his readers have prior programming expertise, he does compare FORTH statements with their BASIC counterparts throughout the text. Neither these comparisons, nor the author's more general approach, however, help to advance the reader's ability to program in FORTH.

FORTH differs from the more "conventional" programming languages in several ways. First of all, it uses Reverse Polish Notation, a system of data entry in which mathematical operands are specified only after the data on which these operations are performed have been entered. For example, the Arithmetic Operating System notation of $10 + 5$ becomes $10\ 5\ +$ when using Reverse Polish Notation. Although Knecht assures the reader who is unfamiliar with Reverse Polish Notation that it will be adequately explained in the course of the text, this is not the case; and while the use of RPN for very simple, one-step operations is straightforward enough, its extension to complex mathematical formulae is not so apparent.

The second peculiarity of FORTH is that it is a stack-oriented language. Aside from the usual return stack, FORTH employs a parameter stack to store data values. While Knecht seems to feel very comfortable with employing the term "stack," it is not so apparent that his readers will feel as comfortable reading it, since he almost completely fails to explain just what a stack is or what the implications of using a stack-oriented language are.

More generally, the fact that FORTH differs in many respects from other programming languages would seem to require a greater emphasis on its divergent features if a book is to succeed in teaching the reader the rudiments of that language. Knecht's brief, 133-page text, however, rules out such a focus. The author's highly condensed treatment of the elements of FORTH, and the very brief and elementary programs which he provides throughout the text to illustrate the language, can at best provide the reader with a minimal grasp of the elements of the language, and leave the him to integrate these as best he can in order to use FORTH as a tool for problem-solving.

LISP

Patrick H. Winston and Berthold K.P. Horn

Level: Intermediate Rating: **90**

Addison-Wesley 1981 Paper
430 Pages ISBN: 0-201-08329-9 6" x 9" \$17.95

To reduce the beauty of a snowfall to the symmetrical repetition of one simple crystalline form is to describe LISP as a hierarchical "list

processor." It is the most inherently aesthetic microcomputer programming language, with an almost organic ability to construct elegant, complex structures from paradoxically simple elements.

The authors of LISP, both affiliated with one of the leading LISP research centers at the Massachusetts Institute of Technology, have crafted a lucid two-part introduction to the language: part one acquaints the reader with the nuances of LISP naming conventions and functions; part two demonstrates the expressive power of complete LISP systems. While the text uses the MACLISP dialect exclusively in these discussions, an appendix notes INTERLISP variations.

Winston and Horn have made LISP's symbol manipulations easy to learn. Conceptual approaches alien to widespread Von Neumann methodologies are carefully introduced; rather than attempting to alter the value of a variable as it is routed through a specific sequence of instructions, LISP treats all input as the first layer in a series of functional transformations whose goal is to produce a desired result. Using LISP functions, elementary data types called "s-expressions" are combined into hierarchical groups. These in turn are functionally combined to produce the "elegant, complex structures" of programs. This first section highlights LISP's unique recursive properties, its flexibility when defining functions and its repertoire of search strategies. The authors' deliberate pace and numerous coding examples emphasize clarity; to this end, the dotted-pair notation used in other LISP texts has been purposely avoided.

Even the non-programmer will recognize a few of the famous examples of LISP systems analyzed in the second part of the book. Weizenbaum's DOCTOR is reduced to a transparent example of LISP's matching function. The blocks-world system, a complete computer environment where geometric objects are "acted-upon," demonstrates the construction of a goal-tree using functions to define other functions.

Few microcomputer users will purchase LISP compilers to explore the subtleties of LISP programming; all, however, would gain valuable conceptual insights into the nature of language systems by reading this Winston and Horn text.

PROBLEM SOLVING PRINCIPLES FOR ADA PROGRAMMERS

Applied Logic, Psychology, and Grit

William E. Lewis

Level: Novice Rating: **45**

	Hayden	1982	Paper	
185 Pages	ISBN: 0-8104-5211-1		6" x 9"	\$9.95

In a rather clumsy bit of publishing legerdemain, Lewis has translated the solutions to his PROBLEM-SOLVING PRICIPLES FOR PROGRAMMERS from 'pseudo-code' into Ada, FORTRAN and COBOL. Little else has changed. Since the message of the first popular presentation had more to do with process than praxis, there is little else to be gained by reading the same solutions to the same problems in a different programming language. Avoid these books. We continue to recommend Lewis' original classic, PROBLEM-SOLVING PRINCIPLES FOR PROGRAMMERS (see our review).

STARTING FORTH

An Introduction to the Forth Lang. & Oper. System

Leo Brodie

Level: Novice Rating: **60**

	Forth	1981	Paper	
348 Pages	ISBN: 0-13-842922-7		7" x 9"	\$15.95

STARTING FORTH is the first full-length introductory treatment of FORTH to appear. It is also far superior to all of the brief introductions reviewed in this volume. Yet, this merely serves to accentuate the disparity between it and an adequate introduction to a programming language.

Paradoxically, the difficulty involved in teaching FORTH is in part inherent in its flexibility. Unlike most traditional programming languages which rely on fixed statements, a "word" is the functional unit of operation in FORTH. The two are not, however, equivalent, since a FORTH word can be defined by the user or created from combinations of other FORTH words. The way in which words are created is therefore a central focus of FORTH instruction; in none of the standard high-level languages such as BASIC, Pascal or FORTRAN is this true to the same degree. The result, as reflected in all of the FORTH texts reviewed here, is

that "the word" becomes all-encompassing; rather than focusing on the use of words as a means of solving problems, these works focuses on words as ends in themselves.

In Brodie's work, this displacement of ends by means is apparent in his failure to present a complete program until the tenth chapter; those few examples of FORTH programming which he does provide come too little and too late. The book's exercises similarly ask the reader to define words, but not to write programs. As a result, despite its relatively complete presentation of the language, STARTING FORTH does not enable the reader to transcend the level of the individual word.

For those users who are obstinate enough to persist in trying to learn FORTH from published books, Brodie's is the only serious work available. Yet, its severe limitations compel us to recommend waiting for the publication of a text which will adopt a more effective instructional technique.

STRUCTURED COBOL A Self-Teaching Guide

Ruth Ashley

Level: Intermediate Rating: **80**

Wiley	1980	Paper	
295 Pages	ISBN: 0-471-05362-7	7" x 10"	\$10.95

Insofar as any high-level language introduction can teach its subject without recourse to an actual computer, this Wiley Self-Teaching Guide succeeds. Since COBOL compilers presumably have not enjoyed widespread microcomputer use, this book avoids the lab exercise approach of its BASIC counterpart, substituting coding examples for suggestions to "try it on your machine." Learning COBOL in this rather ethereal environment is like trying to teach sailing without ship, wind or water: the context places powerful constraints upon the utility of what is being taught.

The book requires an elementary data processing background. A division-by-division approach is taken, with hierarchies within divisions introduced as the coding discussion evolves. Sequential and random file processing are explained, concluding with a brief treatment of ISAM and VSAM processing techniques. Debugging using TRACE and EXHIBIT statements, is, ironically, the author's bon voyage into the "real" world of compiling COBOL programs. Like other Wiley Self-Teaching Guides, topics are presented in a frame-by-frame question-and-answer format.

STRUCTURED COBOL is an adequate elementary introduction to this language, using structured coding techniques; compared to similar

BASIC introductions, however, the microcomputer reader will find the discussion much less machine-oriented in either its teaching method or coding examples, and, consequently, much less effective.

UNDERSTANDING APL

Susan M. Bryson

Level: Novice Rating: **50**

Alfred 1982 Paper
45 Pages ISBN: 0-88284-220-X 4" x 11" \$2.95

In a curious (and, we think, self-serving) reversal of a conventional publishing practice, Bryson has defined the focus of this short guide, as well as the audience for which it is intended, in the book's final sentence rather than in the introduction. "If this brief introduction has interested you," she writes, "you will enjoy looking at the texts available on APL listed in the bibliography." (37) Especially when considered in conjunction with the preceding pages, this statement indicates that UNDERSTANDING APL was written with no intention of actually teaching APL; instead, it is aimed at those indecisive readers who think they might want to learn APL, but want to know a little bit about it first.

It must be conceded that the book is almost ideal for this purpose. An impressively wide range of APL primitives, functions and statements is discussed in the course of very few pages, so that the reader can get some idea of the style and ease of programming in in APL. Even so, thirty-seven pages are sufficient for treating only a subset of the language, and that in a superficial way. The reader who finishes APL will, at best, be able to use APL in its immediate execution mode as an outrageously expensive and tedious kind of calculator, but will not have the least idea how to write longer and more complex general-purpose programs.

So, if your decisiveness is at a low ebb, this is a good book to read in order to strengthen your resolve to learn APL. For those whose desire to learn APL is less ambiguous, however, we recommend that the \$2.95 which might be spent on this Alfred Handy Guide be used instead for the purchase of a more comprehensive and more serious treatment.

UNDERSTANDING COBOL

Richard G. Peddicord

Level: Novice Rating: **50**

Alfred 1981 Paper
 47 Pages ISBN: 0-88284-147-5 4" x 11" \$2.95

This is the only Alfred guide whose text is written vertically across the page; that is, one must rotate the book 90 degrees and read across the spine. The six programming examples contained in the guide provide the most rudimentary overview of COBOL. Four paragraphs do explain the four main COBOL divisions, but this explanation unfortunately falls after the first programming example which uses them. Another example, which demonstrates the COBOL convention that each record must end with a period, itself omits a period after a DISPLAY statement.(11)

The author unabashedly states on page 37, "If you plan to do further work in COBOL, you will need a good textbook." We suggest that the Handy Guide be revised, so that a preface present the beginning programmer with a "GOTO PAGE 37" statement. Inability to understand what this instruction means would then be the only condition under which further reading would be warranted.

UNDERSTANDING LISP

Paul Y. Gloess

Level: Intermediate Rating: **80**

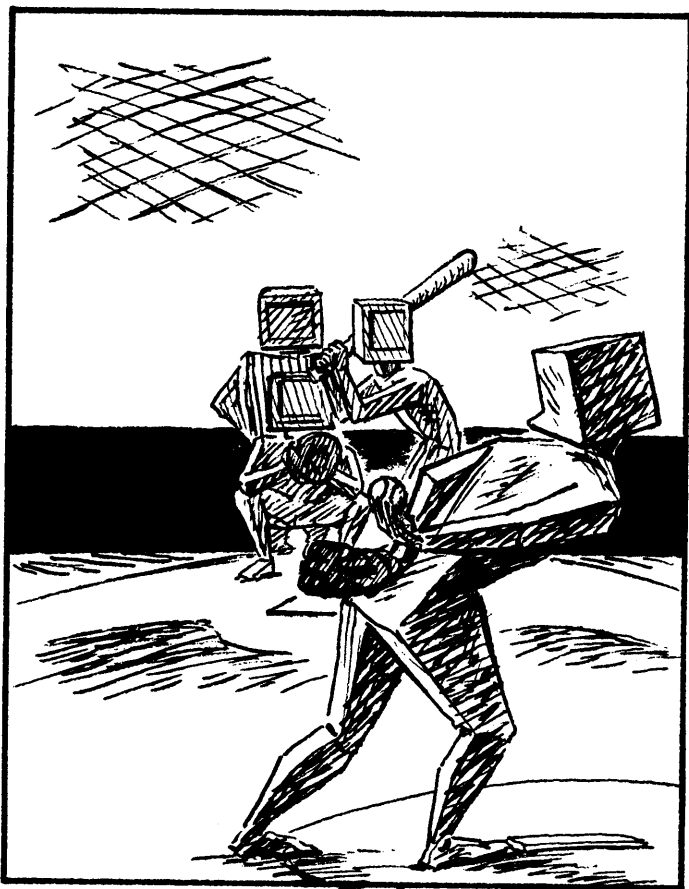
Alfred 1982 Paper
 64 Pages ISBN: 0-88284-219-6 4" x 11" \$2.95

Few readers will gain even a rudimentary understanding of LISP from Gloess' condensed 64-page treatment; as a quick, economical (\$2.95) reference guide to a complex programming language, however, it represents one of the better offerings in the Alfred Handy Guide Series.

A brief history and a language overview preface a description of LISP's single data type, the "s-expression" (footnotes in this chapter testify to the author's seriousness, regardless of format). LISP builds upon these elementary forms, using language and user-defined functions, to create versatile programs. Two varieties of s-expressions, "atoms" and "dotted-pairs," are then considered. List notation (LISP stands for "list processor") and special functions are explained and then demonstrated in a sample program. To Gloess' credit, two exceptional application programs are included: the first translates algebraic expres-

sions into their equivalent in Polish notation form; the second computes the differential equations for trigonometric expressions. Advanced LISP functions are covered in a concluding chapter.

The author freely admits that a number of important LISP features have not been explored in his abbreviated presentation; these include macros and the selective evaluation of arguments. We have nevertheless given UNDERSTANDING LISP a superior rating for its utility to the LISP-knowledgeable programmer as an easy-to-use reference guide.



5

Software & Applications

Performance determines successful microcomputer use. The titles included in this general software and applications chapter, despite their diversity of topics and approaches, all key on maximizing machine performance.

The largest number of books aim at providing the reader with a handy collection of software for specific applications; the user merely enters lines of BASIC (or, in a few cases, Pascal) code into the computer and, if he or she successfully avoids the effects of Murphy's Law, runs the program. In many instances, authors have written these collections in an attempt to improve the the reader's ability to program in BASIC; besides providing explanations of a program's operation, they include suggestions for further enhancements and improvements.

A second variant of applications literature aims at enhancing the user's knowledge of commercially available software packages. In this category, we have included works on VisiCalc—the powerful electronic spreadsheet manufactured by VisiCorp—as well as on word processing, which uses a microcomputer as a typewriter and storage device for text. The word processing section includes both general titles—which acquaint the reader with the functions of word processing software and suggest criteria for selecting a specific package—and introductions to WordStar, one of the more popular and flexible word processing packages for microcomputers.

Those users who want to develop their own software will be most interested in works which deal with programming for specific applications. Topics covered range from games (such as Richard Mateosian's *INSIDE BASIC GAMES*), and mathematical puzzles (Snover and Spikell's *BRAIN TICKLERS*) to artificial intelligence research (John Krutch's *EXPERIMENTS IN ARTIFICIAL INTELLIGENCE FOR SMALL COMPUTERS*). More traditional applications include programming in business (John Nevison's *EXECUTIVE COMPUTING*) or the the sciences (Eric Burgess' *CELESTIAL BASIC*).

BASIC BUSINESS SOFTWARE

E.G. Brooner

Level: Intermediate Rating: **85**

	Sams	1981	Paper	
141 Pages	ISBN: 0-672-21751-1		5" x 8"	\$11.95

Two audiences whose microcomputer interests frequently overlap will be interested in this Blacksburg Group offering: the small businessperson using a personal computer, and the BASIC programmer.

Brooner's attention is primarily directed toward the businessperson interested in the specific applications which can be used on his or her computer. By stepping through each of the development phases of nine different business programs, the author divulges the intricacies of running a payroll, analyzing an inventory or simply recording checking transactions. Brooner's introduction of different programming techniques is accomplished by highlighting the code used in successive application programs.

Brooner is also interested in showing the businessperson how to modify the "canned" software packages and/or operating systems to better suit his particular business needs. This discussion, along with some hints relating to the translation of Microsoft BASIC to CBASIC to NorthStar BASIC, provide valuable tools for the programmer.

BASIC BUSINESS SOFTWARE assumes a prior exposure to both BASIC programming and personal computer operating systems. Though high-priced when judged in relation to its total number of pages, the quality of this text more than justifies its recommendation to the microcomputer-minded businessperson.

BASIC COMPUTER GAMES

Microcomputer Edition

David H. Ahl

Level: Novice Rating: **85**

	Workman	1978	Paper	
183 Pages	ISBN: 0-89480-052-3		9" x 11"	\$7.95

The customary fare in books containing microcomputer program listings is a melange of public domain software, with programs that are mundane and uninteresting. For the dubious privilege of entering these lines of BASIC code into the microcomputer, the user is generally expected to pay a sum in excess of \$10.00. In contrast to this, David Ahl's BASIC COMPUTER GAMES and MORE BASIC COMPUTER GAMES

offer the computer game enthusiast a large selection of programs (101 in GAMES, 84 in MORE GAMES) at a very reasonable cost. While their quality varies greatly, very few of the programs are of the twenty-line variety which prove to be more trouble to key than they are worth.

The low cost of these two volumes has, however, necessitated some sacrifices. The most notable is in the legibility of the program listings. While Ahl defensively protests in advance that the dot matrix printer used to print the programs "gives them exceptionally high legibility, even when reduced to 64% as it has been done in here,"(xi) we in fact found that the attempt to type from these listings over an extended period of time is virtually guaranteed to cause blindness. More immediately, typing from the pages of these books is certain to prove extremely frustrating, since some letters (for example N and M, or B and D) are frequently indistinguishable from one another. In addition, for the user who, for whatever reason, is having difficulty in getting the programs to run properly, the omission of flowcharts, REM statements or any sort of documentation which indicates how the program is supposed to work, makes debugging an onerous process.

The quality of the games themselves varies greatly. At the low end of the spectrum, for example, the computer will generate "a different maze every time it is run," although the user can do nothing further with it (see Amazing in GAMES, p.3); MORE GAMES (pp.101-103) offers an "improved" version, in which the user can watch a near-sighted mouse (represented by two asterisks) blunder its way to the exit. But besides these duds, the two books do contain a number of interesting and enjoyable games, such as the advanced version of Football in GAMES (pp. 16-17).

A final comment concerns the character of the programming. Most of these programs were written initially for mainframe BASIC and then translated into microcomputer BASIC. At times, however, this conversion is incomplete; Artillery 3 (MORE GAMES, pp.2-3), for example, uses the MAT ZER statement (which is supposed to initialize the elements of an array to zero), although it will produce an error message under most versions of BASIC for microcomputers. Nor are the games especially user-friendly; in almost all cases in which a program calls for input from the terminal, the user is not supplied with any precise indication of the character or form of the response expected by the computer. Finally, some of the games can be won by "cheating" the computer; in Bombardment (GAMES, pp. 22-23), the goal of which is to destroy 4 enemy platoons hidden in a 5x5 grid, the player need only enter the same location four times to win, since no precautions have been taken to insure that all four platoons must be destroyed.

Both GAMES and MORE GAMES offer the reader countless hours of enjoyment at an extremely reasonable price. At the same time, however, the very real limitations of these volumes should be noted; and we would hope that some attempt might be made to make future editions of these

volumes, as well as of published software generally, both more user-friendly and more reader-friendly.

BASIC COMPUTER PROGRAMS FOR BUSINESS, VOL.1

Charles D. Sternberg

Level: Novice Rating: **85**

Hayden 1980 Paper
264 Pages ISBN: 0-8104-5162-X 7" x 9" \$10.95

BASIC COMPUTER PROGRAMS FOR BUSINESS, VOL.2

Charles D. Sternberg

Level: Novice Rating: **85**

Hayden 1982 Paper
376 Pages ISBN: 0-8104-5178-6 7" x 9" \$13.95

This two-volume opus of application programs is unique in that it is more than theoretically possible to computerize a fairly complex business given the contents of its texts. Every important business function, from the printing of mailing lists to the analysis of income statements, falls under the author's programming scrutiny.

Each program is meticulously documented by a preface which includes flowcharts, error recovery procedures, operational considerations and a complete listing of file and variable names. Structured programming techniques have been used so that it is relatively easy to discern one functional module from another. All programs are written in the Disk Extended BASIC, Revision 4.1, of the Altair 8800b Microcomputer System. While simple programs using common BASIC commands can be entered into most personal computers with little translation difficulty, certain Altair BASIC instructions, particularly file-related commands, are specific to the 8800b. For example, commands like "MID\$" and "MKI\$," reviewed by the author in an appendix designed to help with "compatibility problems," require rewriting before they can be entered into systems using more popular BASIC dialects. There is no question, however, that the quality and scope of the author's work more than justifies such user modification.

The thirty-five programs contained in Volume I are divided into three general areas of business applications: Financial Control and Analysis, Inventory Control and Analysis, and Production Planning and

Control. A bookkeeping system spreads journal entries into a general ledger, produces a trial balance, and even prints comparative profit and loss statements. A series of forecasting programs extrapolates business data using either the least squares regression, moving average or exponential-smoothing methods. Automatic reorders are generated by an inventory program's analysis of buyer criteria and stock levels.

Volume II contains programs categorized as belonging to one of four subject areas: Marketing/Sales Planning and Analysis, Personnel Recording and Analysis, Administrative Assistants, and Miscellaneous Programs and Routines. "Administrative Assistants" includes mailing lists, price lists and word processing programs. The miscellaneous section contains a collection of utility programs: a base chart formatter, file copy and "dump" programs, and scheduling system are typical entries.

If any critique can be made of Sternberg's programs, it is that input prompts are uniformly inadequate. While the programmer obviously knows the format of the numeric data (decimal, etc.) to be entered into input fields, the average end-user receives little help from program prompts.

BASIC COMPUTER PROGRAMS FOR BUSINESS is a rough gem of applications programs: each program requires some user modification to be of real utility in a business environment, in terms of translating from the Altair BASIC dialect, and in increasing their overall "user-friendliness." This is not to detract from the amazing breadth of applications that Sternberg's detailed, crafted code encompasses. With a little polishing, the businessman/microcomputer user will find these texts invaluable sourcebooks of new business applications.

BASIC COMPUTER PROGRAMS FOR THE HOME

Charles D. Sternberg

Level: Novice Rating: **70**

	Hayden	1980	Paper	
330 Pages	ISBN: 0-8104-5154-9	6" x 10"	\$10.95	

BASIC COMPUTER PROGRAMS FOR THE HOME contains eighty computer programs grouped into ten areas: home financial programs, automobile-related programs, kitchen helpmates, scheduling programs, list programs, miscellaneous programs, tutorial programs, conversion programs, recreational programs and hobbyist programs. Sternberg has taken great care to insure that these are compatible with most versions of BASIC, thus enabling the user with minimal programming experience to easily modify or improve on them. Sternberg avoids complex program-

ming routines, provides a brief description of each program and includes REM statements and variable definitions throughout so that the reader can easily understand what each portion of a program is expected to accomplish. Should the reader encounter compatibility problems between these programs and the version of BASIC installed in his or her microcomputer, Sternberg has included an appendix on some of the language features used in his program which might cause problems, and in many cases provides suggestions for overcoming them. In short, this book has been well-conceived and planned, well-produced, and well documented.

The major drawback of this volume is the contents of the programs themselves, many of which are not designed to produce information which is either sufficiently important or sufficiently inaccessible to justify the expenditure of time necessary to enter the program and data and do the computer run. This may be an inevitable consequence of a volume which offers so many programs at such a low price. The income tax recording program, for example, does no more than sum income and a variety of tax-deductible expenses which the user is to input with DATA statements; the simple process of preparing a DATA statement will probably take more time than using a hand-held calculator to produce the same totals. Similarly, "TV Scheduling—Extended Version" explicitly points out conflicts in viewing preferences which should be more than obvious before the data is input into the computer.

BASIC COMPUTER PROGRAMS FOR THE HOME adopts a format which is designed to promote the user's programming skills. The content of its programs, however, make it likely that this volume will merely remain unused on the bookshelf.

BASIC COMPUTER PROGRAMS IN SCIENCE AND ENGINEERING

Jules H. Gilder

Level: Novice Rating: **80**

Hayden	1980	Paper	
247 Pages	ISBN: 0-8104-0761-2	5" x 8"	\$10.95

BASIC COMPUTER PROGRAMS IN SCIENCE AND ENGINEERING contains a total of 114 programs, most of them in electronics or engineering. There are eight programs in basic electricity, seven in basic electronics and communications, and eleven in attenuator pads. The remaining programs focus on mathematics and data analysis.

Most of the programs are quite short, and have been written so that they will run with minimal modifications under most versions of BASIC;

our test of several of the mathematical and data analysis programs in fact showed this to be the case. Gilder has adequately fulfilled his intention of providing the reader with "a sourcebook of basic programs that can be referenced from time to time when the need arises." Gilder is far less successful in meeting his secondary goal of showing "the computer newcomer...how a particular problem has been solved," since the explanatory text accompanying each of the programs presupposes that the user has a certain technical knowledge of the problem, even if he or she does not know the algorithm by which the computer program can solve it. This is, in short, a handy and convenient software library for a highly specialized audience.

BASIC PROGRAMS FOR HOME FINANCIAL MANAGEMENT

W.B. Goldsmith, Jr.

Level: Intermediate Rating: **95**

314 Pages Spectrum 1981 Paper ISBN: 0-13-066514-2 8" x 11" \$12.95

The manager of a private income tax practice has compiled thirty-three programs of immediate utility to the personal computer user interested in computerized financial management. The primary subject areas are money management, credit control, major asset management and investment factors.

All programs are written in a "minimal" BASIC and adopt the general form of input prompts-calculation-printout; indeed, many of the programs appear longer than they actually are because the author has thoughtfully coded extra print statements to make input prompts more "user friendly." Conversion to other BASIC dialects is made easier by this simple program structure, and actually involves only the translation of certain system-specific string variable and TAB statements.

The documentation for each program is excellent. The book's more complex topics, like return-on-investment and bond investment analysis, are preceded by pertinent background information. "Programming notes" explain each of the functional segments of the program, and suggest the areas which may prove most troublesome if debugging is needed. "Operating notes" include sample mentions of prompts which require particular user responses. (For example, in one yes/no prompt, anything not beginning with "n" is treated as a "yes".)

The practicality of Goldsmith's topics to the home computer user makes this book among the best available. These "straightforward dollar-and-cents programs" include an "Electronic Checkbook," "Credit Card

Organizer," "Consumer Loan Analyzizer," "Net Worth Statement," and "Stock Analyzer." Each combines accounting savvy with the substantial programming expertise needed to produce simple, functional code and to generate efficient programs that are easy to use, and easier to understand.

BASIC PROGRAMS FOR HOME FINANCIAL MANAGEMENT is a valuable source of "load-and-go" programs which will instantly increase the return on the reader's personal computer investment. We highly recommend its addition to any program library.

BASIC PROGRAMS FOR SCIENTISTS AND ENGINEERS

Alan R. Miller

Level: Novice Rating: **90**

Sybex 1981 Paper
 318 Pages ISBN: 0-89588-073-3 7" x 9" \$14.95

The identical sets of programs in BASIC PROGRAMS, FORTRAN PROGRAMS and PASCAL PROGRAMS were written for practicing scientists and engineers as well as for juniors or seniors in engineering school; their successful application, Miller believes, also demands that the user have some knowledge of one of these programming languages. The programs themselves are grouped into ten areas: mean and standard deviation, vector and matrix operations, simultaneous solution of linear equations, curve-fitting (including least-squares and non-linear curve fitting), solutions of equations by Newton's method, sorting, numerical integration and a variety of advanced applications. The BASIC programs were written in Microsoft BASIC-80, version 5, and then "simplified to the lowest common denominator" in order to run under most BASICs. Miller recommends that these programs be upgraded by the user, especially by expanding variable names into meaningful words (he includes REM statements with suggested expanded variable names for this purpose). The greater uniformity of Pascal poses a smaller problem of syntax errors. Miller developed his Pascal programs with a Pascal/M compiler, but also ran them on Pascal/MT+, Pascal/Z, and JRT Pascal compilers, so that only the UCSD Pascal compiler may pose problems of compatibility to the user. The FORTRAN programs were written on Z80-based microcomputer and follow the 1966 standard.

The quality of these books is uniformly high. They can, first of all, be used to advantage by virtually anyone interested in engineering, the sciences, statistics or advanced mathematics. Aside from simply presenting the user with a powerful collection of programs, Miller has also

provided explanations of the scientific or mathematical principles with which these programs deal, thus allowing the book to be used as a learning aid for those who wish to expand their knowledge about any of these subjects. Program documentation is ample, and the way in which the programs operate is adequately explained. As a result, users should have little difficulty in adapting programs to their needs or to their own microcomputers.

But perhaps the most interesting feature of Miller's books is the emphasis placed on determining the limitations of a particular programming language used on a particular microcomputer. In the case of the BASIC interpreter, Miller presents simple programs to assess the degree of rounding error, the "randomness" of a random number generator and the likelihood of return addresses expanding in the stack until they fill all available Random Access Memory. For FORTRAN and Pascal compilers, Miller focuses on the precision and range of floating-point operations and the accuracy of some "intrinsic" functions.

BEAT THE ODDS

Microcomputer Simulations of Casino Games

Hans Sagan

Level: Intermediate Rating: **85**

	Hayden	1980	Paper	
210 Pages	ISBN: 0-8104-5181-6		6" x 9"	\$9.50

The allure of this book for the microcomputer user is most apparent in the seemingly incongruous disciplines represented in each of its appendices: Appendix A is a monograph on mathematical probability; Appendix B is a summary of BASIC commands peculiar to the ACCESS HP2000 BASIC dialect; Appendix C is a "Classification of Bets in Ascending Order of Expected Losses;" and Appendix D is a French-English mini-dictionary. All are required by Hans Sagan's eclectic text.

The author, a professor of mathematics, has used his knowledge of BASIC programming and probabilities to simulate five casino games: trente-et-quarante (thirty-and-forty, a European casino game); roulette; chemin-de-fer (literally, "railroad," a card game); craps and blackjack. Sagan's approach is as fascinating as the games he describes. A computer run first shows what the computer's version of the game looks like: how the betting proceeds, what input is required, and how "chance" is simulated. The author then offers some background material on the origin and rules of the actual casino game. He additionally proposes strategies which optimize the player's chances ("Beat the Odds") based

upon mathematical analyses of the games' specific probabilities. An annotated program listing follows, with suggestions for user-modifications to further enhance the accuracy of the computer's simulation.

Since the "international language of gambling" is French, Sagan has retained the appropriate French expressions in his programs (hence, the French-English appendix). All of his descriptions, whether of a particular game, program or law of probability, are detailed and entertaining. (Sagan never forgets the inherent fun of game-playing.)

The programs are written in HP2000 ACCESS BASIC, and, as mentioned earlier, an appendix notes potential dialect differences in the handling of FOR-NEXT loops, GOSUB and GOTO statements, delimiters, character strings and random numbers. Owing to complexities of function, each of the five simulator programs contains more than two-hundred lines of code.

BEAT THE ODDS is an engrossing combination of microcomputers, mathematics and casino games. Even if individual programs are never actually keyed, the author's lively, unusual narrative is alone sufficient reason for its addition to a personal computer library.

BRAIN TICKLERS: PUZZLES & PASTIMES For Programmable Calculators & Personal Computers

Stephen L. Snover and Mark A. Spikell

Level: Novice Rating: **80**

	Spectrum	1981	Paper	
162 Pages	ISBN: 0-13-081000-2		7" x 9"	\$5.95

BRAIN TICKLERS is less a collection of programs than a collection of problems, for whose solution the reader is expected to mobilize his more or less substantial programming skills. The book itself poses twenty-six mathematics-related problems of varying degrees of difficulty, none of which can be solved with programs capable of a general application; for example, a more or less typical problem requires that the reader identify the four narcissistic cubes (i.e., positive integers whose cube equals the sum of the cube of their digits). This is perhaps the greatest weakness of the book—that it emphasizes programming for highly specific and individual applications, rather than for more generalized ones.

Despite this, however, the low cost of the book and the challenge posed by the problems themselves make BRAIN TICKLERS an almost ideal work for further refining the reader's programming skills. After presenting the problem to be solved, Snover and Spikell provide the reader with programming hints and comments on the algorithm which

might be used to find a solution; the reader with a good deal of self-confidence can overlook these simply by not turning the page. The authors then present their BASIC-language solution to the puzzle, and provide some suggestions for further expansions or modifications of the problem.

The authors' programming solutions themselves, however, are not necessarily optimal ones. Curiously, in a collection of programs which seem to require FOR/NEXT loops, they carefully avoid these and instead rely on IF and GOTO statements. Their solution to a sample problem for the TI-59 Programmable Calculator similarly does not make optimal use of its instruction set. (This is the only instance in which they present a program for programmable calculators, although our own attempt to program a number of these problems on the TI-59 showed that their solution was fairly straightforward.)

BRAIN TICKLERS is an excellent addition to the beginning programmer's library, not because of the high quality of the author's programming examples, but because it provides an inexpensive means of permitting the reader to focus on concrete and interesting problems to develop programming skills.

CELESTIAL BASIC

Astronomy on Your Computer

Eric Burgess

Level: Novice Rating: **90**

Sybex 1982 Paper
300 Pages ISBN: 0-89588-087-3 7" x 9" \$13.95

CELESTIAL BASIC is a collection of twenty-three astronomy programs in BASIC, which are grouped into four areas: time (which includes programs to calculate the day of the week for any date or the position of stars in relation to the Earth in any year); the moon (including phases of the moon on any date or dates of lunar eclipses in any year); the planets (their position relative to Earth on any date, the horizon plots of visible planets, the times of rising and setting of Mercury or Mars); and general and tutorial subjects (astronomical conversions, dates of meteor showers, constellation recognition and the time of exposure necessary for optimum photographs of planets).

In addition to the intrinsic value of this wide range of programs for those interested in astronomy, this book shows every evidence of having been assembled with care. Although the programs were written for the Apple II, Burgess has made every effort to avoid complex or Apple-specific programming statements, thus enabling their use on a wide

variety of microcomputers. Where Burgess has used AppleSoft high resolution graphics, he provides alternative programs in an appendix to insure their maximum applicability. Burgess does not, however, specify the minimum RAM necessary to run a single program, although he does note that 48K is ample to combine several programs.

These programs should, in short, require only minor modifications to run under almost any version of BASIC. The user's success in adapting the programs to a particular microcomputer will then be a function of his or her understanding of what the programs are trying to accomplish, and more generally of his or her knowledge of astronomy. Although Burgess feels that he has adequately interspersed REM statements throughout the programs, this is not always the case. But more importantly, despite Burgess' cursory introductions to each program, this remains a specialized work for those who already possess at least some expertise in astronomy.

CHESS AND COMPUTERS

David Levy

Level: Intermediate Rating: **95**

Computer Science 1976 Paper
145 Pages ISBN: 0-914894-02-1 5" x 8" \$11.95

In a book which is in many respects oriented more toward the chess player than the computer user, Levy offers an enjoyable and thorough history of "chess machines" from 1770 (when Baron Von Kempelen first exhibited his infamous Automaton Chessplayer) to early 1976. Aside from the first chapter, which is devoted to pioneering attempts (including several fraudulent ones) at automating chess, Levy's focus is on computerized chess programs (although he does not present actual programs). The period covered by CHESS AND COMPUTERS is one in which these programs were the exclusive preserve of a few specialized research institutions such as Northwestern University (CHESS), Carnegie-Mellon (TECH) or the Institute of Control Science in Moscow (KAISSA).

Chess offers a seemingly perfect field for computerization, since the astronomical number of possible variations from a single position can be handled with relative ease by the sheer number-crunching ability of computers. Indeed most chess programs at least partly incorporate this "brute force" approach. But the major problem in chess programming is that this number-crunching utility does not provide the criteria for assessing which move, out of a universe of possible ones, is optimal. It is not even necessarily true that the ability to envision innumerable variations is especially advantageous; research has shown that a chessmas-

ter's "look-ahead" averages only 1.6 to 1.9 variations from a given position. In addition, successful play at chess is not based on the application of rigid mathematical formulae, but rather requires the employment of complex evaluation functions.

A major focus of Levy's discussion are the varying attempts made to incorporate chess heuristics in a program's evaluation function, and the limitations from which these attempts have suffered. The specific criteria and methods used in programs to evaluate chess positions are explored in some detail in the text, and their strengths and weaknesses are examined by Levy in the annotated listings of games played by some of the programs. Our favorite example was a 1971 game in which COKO, a Bell Telephone Laboratories program, had no fewer than four possible mates in one position against GENIE, from the Fleet Computer Center in San Diego, but lost the game because the program suffered from a classic case of indecision: paralyzed by the multiplicity of strong moves available to it, the program instead chose to move its king between two innocuous squares while one of its opponent's pawns queened.

Despite its (understandable) omission of chess program listings, we highly recommend CHESS AND COMPUTERS to those interested in writing their own chess programs, as well as to the chess/microcomputer enthusiast who is curious about the principles underlying computerized chess.

CODES, CIPHERS AND COMPUTERS

An Introduction to Information Security

Bruce Bosworth

Level: Novice Rating: **80**

Hayden 1982 Paper
259 Pages ISBN: 0-8104-5149-0 6" x 9" \$13.95

Part of this book's focus on "cryptography as a means of concealing and protecting confidential data"(i) makes it of little interest to most microcomputer users, who have no particular reason to fear eavesdropping or the theft of their data files. Because of this, Bosworth's discussion of the Data Encryption Standard and the Public-Key Cryptosystem, two major information security systems, is somewhat tedious and out of place.

But ever since Sherlock Holmes succeeded in unraveling the meaning of the dancing men, cryptography—the assorted techniques of concealing the meaning of information from unintended recipients—and cryptanalysis—the technique of deciphering or decoding such information by unauthorized users—have attracted a sizable number of devotees.

Bosworth's book, which attempts to advance cryptography and cryptanalysis into the computer age, was written in part with this group in mind. More than this, Bosworth hopes to improve the microcomputer user's command of BASIC and of programming techniques by stimulating his or her interest in codes and ciphers.

These various foci of attention permit Bosworth only briefly to survey methods of encrypting information, and to devote far too little attention to cryptanalysis (pp.130-141, where only frequency analysis and computerized trial and error are discussed). Despite this, the quality of both the discussion and the computer programs, as well as of the questions at the end of each chapter which encourage the reader to try out or modify the programs, make this book both interesting and rewarding.

CREATE WORD PUZZLES WITH YOUR MICROCOMPUTER

Ernest E. Mau

Level: Novice Rating: **95**

Hayden	1982	Paper	
304 Pages	ISBN: 0-8104-6251-6	7" x 10"	\$14.95

Ernest E. Mau is a professional freelance creator of word puzzles and games who, in the course of the last three-and-a-half years, has used his MITS 300/25 microcomputer system to prepare the word puzzles which he submitted to unsuspecting editors. The publication of his book is probably as much a shock to those purists who insist that word puzzles cannot be effectively computerized as it is a bonanza for those microcomputer users who enjoy doing word puzzles. The book's seventeen programs (which can produce twenty-three different kinds of word puzzles, although the traditional crossword puzzle is not among them) are all written in Microsoft BASIC so that they can run under CP/M without modification; even where this is not the case, however, Mau feels that alterations to the program should be minor and well within the reach and understanding of the word puzzle devotee. While the programs themselves contain few or no REM statements, Mau does provide detailed explanations of what the programs are intended to accomplish and how they should work. The proper operation of the puzzle programs requires a microcomputer system with at least 40K of user-available RAM, at least one disk drive and a printer.

Mau's book is extremely impressive for two reasons. The first is that the programs themselves are exceedingly complex and intricate, so that through applying them the microcomputer user can expand his or her

knowledge of programming in BASIC. The second is, of course, the enjoyment derived from the puzzles themselves. We should add, however, one warning note: the quality of the output depends very heavily on the amount of work the user is willing to input into this project. Computers cannot themselves generate words; instead, these must be provided to the computer in advance. Obviously, the greater the number of words and the more diverse the contents of data files, the higher the quality of the word puzzles which the computer can produce. Mau's own data base consists of some four hundred topical files, seven special list files, twenty six "alpha-list" files, fifty-three files for the acrostic puzzle program, and four quotation files. The user willing to expend this effort to create word files will find that his or her enjoyment of Mau's book far outweighs any labor involved.

DATA BASE MANAGEMENT SYSTEMS A Guide to Microcomputer Software

David Kruglinski

Level: Novice Rating: **85**

	Osborne	1983	Paper
260 Pages	ISBN: 0-931988-84-5	7" x 9"	\$16.95

Kruglinski's book assails one of the last bastions of mainframer snobbery: data base management systems (DBMS). While certainly not the James Martin of the microcomputer set, Kruglinski does impart a decent conceptual overview of the nature of a data base and more pragmatically, provides the novice with some operational understanding of a few popular data base packages.

The reader is introduced to the characteristics of a data base management system and its ability to maximize the number of ways any user can quickly access data. Its flexibility in presenting these multiple "views," as well as its limitations in a microcomputer environment, preface a more specific discussion of the three types of data base management systems. File Management Systems (FMS), Relational Data Base Management Systems (RDBMS) and Network/ Hierarchical Data Base Management Systems (NDBMS) are treated in Kruglinski's alphabet-soup survey. The conceptual introduction concludes with a chapter on physical data access techniques: these include binary search trees, hashing, and direct, sequential and random access methods.

Within the context of this conceptual groundwork, the author evaluates four data base packages: Condor Series 20, dBASE II, FMS-80, and MDBS III. These critiques are detailed, thoughtful and should be required reading for the prospective DBMS-package buyer. A standard

checklist reveals each product's versatility, price and system requirements. The author steps through a few sample applications, replete with photographs of screen prompts, to give a realistic sense of each package's actual operation. Kruglinski is never ambiguous regarding relative strengths and weaknesses. dBASE II, for example, is lauded as a "true relational data base management system," but chastised for its lack of security and its difficulty with indexed sorts of large files with long record keys. The treatment of those packages which include their own programming language (dBASE II and FMS-80) are supplemented by appendices providing sample program listings. Capsule reviews cover other less popular DBMS packages on a machine-specific basis.

DATA BASE MANAGEMENT SYSTEMS is a shopper's guide to DBMS packages. Kruglinski provides a comprehensible conceptual overview combined with well-documented evaluations of typical data base management systems. This is one of the better treatments of DBMS from a functional and operational perspective.

dBASE II USER'S GUIDE

Adam B. Green

Level: Novice Rating: **85**

151 Pages	Spectrum ISBN: 0-13-196519-0	1983 Paper	9" x 11"	\$29.00
-----------	---------------------------------	---------------	----------	---------

Material on dBASE II has been limited to references in a few popular microcomputer titles, among them Kruglinski's DATA BASE MANAGEMENT (see our review). Adam Green's dBASE II USER'S GUIDE is a welcome addition to this void in the literature; Green provides a lucid presentation suitable for both novices and experienced dBASE II users.

The author's treatment of Version 2.3 is divided into two main sections. The first is a sample dBASE II session designed as a tutorial for first time users; the second is a tour of advanced dBASE II functions through the development of a complete dBASE II mailing list system.

The introduction to the function and syntax of dBASE II commands is excellent. The reader is masterfully stepped-through a simple file application as elementary dBASE II commands—CREATE, DISPLAY, DELETE, PACK, LIST—are demonstrated in frames depicting actual input and output. This first session ends with discussions of primary and secondary memory, macros and the modification of data file structure.

As a quick preliminary to the advanced programming section, Green reviews the conventions of dBASE II's powerful data description language. Nested loops, user prompting and the program generator ZIP are all considered as Green launches into the initial design of a complete

dBASE II system. Programming techniques are demonstrated and sound documentation practices are reinforced as the author explains each element of a mailing list application. An annotated listing of these modules include menus, files, programs and techniques for linking multiple files to the original application. A final chapter treats interfacing dBASE II with SuperCalc and WordStar.

In a literature where knowledge carries an inordinately high price tag, Green's book nonetheless manages to be conspicuous. Though a superior treatment, there would appear to be little relation between its content or size or production quality and its retail price. We therefore recommend the dBASE II USER'S GUIDE exclusively to those prosperous personal computer owners who are searching for a comprehensible supplement to their dBASE II manual.

ELECTRONICALLY SPEAKING

Computer Speech Generation

John P. Cater

Level: Novice Rating: **95**

	Sams	1983	Paper
230 Pages	ISBN: 0-672-21947-6	6" x 9"	\$14.95

Computer speech generation, one of the newest innovations in microcomputer technology, is the subject of this highly readable text by an expert in "digital synthetic speech applications."

The complexity of the problems involved in speech synthesis become immediately apparent in the author's discussion of the physiology of human speech (forty-nine separate muscles modify "glottal pulses" to produce intelligible words). A chapter on linguistics with an emphasis on phonetic speech analysis concludes the "human model;" it is this model which, since the early twentieth century, man has attempted mechanically and electronically to simulate. A historical survey of these early attempts follows, ranging from Von Kempelen's "speaking bellows" to the star of the 1939 World's Fair, the "Voder Synthesizer."

Cater develops these topics with the masterful ease of a polished Nigel Calder or David Attenborough. Every inquiry into the nature of speech, whether organic or mechanical, provokes a child-like wonder and an equally awe-inspiring, reasoned explanation. In these early chapters, which lay the foundation for subsequent discussions of current speech synthesis techniques, Cater shows both his knowledge and continued fascination in a style seductive for its clarity and grace.

Those techniques currently employed in the microcomputer synthesis of speech form the technical "heart" of the book. In order of

increasing complexity, these are the waveform encoder, the analog format synthesizer and the linear predictive coded (LPC) synthesizer currently available to the general public.

ELECTRONICALLY SPEAKING: COMPUTER SPEECH GENERATION is an engaging overview of a number of sciences which have collaborated to produce "talking computers." Its style and substance make it one of the most highly recommended books in this guide.

EXECUTIVE COMPUTING

How To Get It Done On Your Own

John M. Nevison

Level: Intermediate Rating: 90

Addison-Wesley 1981 Paper
319 Pages ISBN: 0-201-05248-2 6" x 9" \$9.95

John Nevison is the Socrates of personal computer-toting corporate executives, rambling through boardrooms while conducting dialogues on the nature of bottom-line beauty. His text is written in the form of question-and-answer case studies designed for the executive who seeks to hone his analysis-generating ability with the aid of a microcomputer. It is thoughtfully presented, informative, and immediately useable.

Microcomputers first began to mushroom in the dark executive-office recesses of the corporate jungle when centralized data processing facilities proved unresponsive to the daily operational needs of end users. The availability of inexpensive hardware in the form of microcomputers, and of user-friendly software, notably VisiCalc, solved the immediate computing problem; Nevison's book expands the micro's capabilities past that initial beachhead into the single most valuable tool at the businessperson's disposal. Inventory modeling, net-present value analysis, and project scheduling are explained and coded in application examples relevant to most corporate environments. Each technique is explored in-depth, with exercises and additional references (many on pure business topics) at the end of each chapter. Nevison, the author of **THE LITTLE BOOK OF BASIC STYLE**, has also included appendices on structured programming and utility code. (All of the BASIC programs included in the book are available on cassette or disk.)

EXECUTIVE COMPUTING was one of the early treatments of microcomputer applications in the executive suite; even with the additional entries in this category which followed the growing popularity of personal computers as a business tool, Nevison's book still remains one of the best.

EXECUTIVE PLANNING WITH BASIC

X.T. Bui

Level: Intermediate

Rating: **90**

Sybex 1982 Paper
197 Pages ISBN: 0-89588-083-0 7" x 9" \$12.95

Modern managerial theory stresses the use of quantitative methods in reaching sound business decisions. Modeling, decision trees and critical path analysis are examples of such methods; their transformation into interactive BASIC programs is the subject of X.T. Bui's book.

EXECUTIVE PLANNING WITH BASIC assumes a prior knowledge of BASIC programming, though a token introduction to BASIC is included as an appendix. It covers three main areas of quantitative business analysis: decision models under certainty and uncertainty, forecasting and investment modeling and multi-criteria decision aids. Each area is further sub-divided into application chapters which each address a business problem using the appropriate method. These sub-topic chapters include exponential smoothing (under forecasting), critical path analysis (under decision models), and a specific program, "Electre," for the structured analysis of decisions involving preferences (under multi-criteria decision aids).

A brief explanation of the method to be demonstrated precedes each business problem, with sample data and output provided as supplements to the authors coded "solution." An economic-order-quantity chapter, for example, shows how optimal inventory and order quantities can be ascertained by supplying data to different program variables. Though the reader may have difficulty finding an answer to his particular set of business decisions, the modular program structure allows for easy modification. (Manufacturer-specific prompts in the economic order quantity program can be deleted in favor of a distribution-type model.) An appendix treats the elementary statistical conventions of mean, variance and standard deviation which surface in some of the text solutions.

EXECUTIVE PLANNING WITH BASIC is a fascinating combination of the disciplines of quantitative business analysis and programming; it will prove of interest not only to the businessperson practiced in the use of microcomputers who has pushed VisiCalc to its limits, but also to the new corporate user eager to supplement his decision-making ability with a powerful analytic tool.

EXPERIMENTS IN ARTIFICIAL INTELLIGENCE FOR SMALL COMPUTERS

John Krutch

Level: Novice Rating: **90**

Sams 1981 Paper
 110 Pages ISBN: 0-672-21785-6 6" x 9" \$8.95

Joseph Weizenbaum's 1966 Eliza program, which simulated the responses of a Rogerian psychotherapist to the prompts of a computer user/patient, became one of the most famous (or infamous) early experiments in artificial intelligence. John Krutch briefly discusses Eliza as an example of natural-language processing; but the scope of his book goes beyond this historical review of the field to expand the reader's awareness of recent developments in artificial intelligence research that range from pattern recognition algorithms in chess programs to computer-generated novellas.

The pattern of Mr. Krutch's book is similarly elegant: the reader is first introduced to the conceptual homework supporting each of the main branches of artificial intelligence research. These are automatic programming, pattern recognition, problem solving and natural-language processing. Simple BASIC programs are then presented to illustrate each approach. Since none of the programs exceeds a few hundred lines of code, the prospects of having your microcomputer generate some rather original haiku, for example, are very real.

EXPERIMENTS IN ARTIFICIAL INTELLIGENCE FOR SMALL COMPUTERS uses the reader's personal computer as a laboratory to demonstrate some of the more arcane avenues of artificial intelligence research. It is an original, fascinating treatment of this topic.

FORTRAN PROGRAMS FOR SCIENTISTS AND ENGINEERS

Alan R. Miller

Level: Novice Rating: **90**

Sybex 1982 Paper
 280 Pages ISBN: 0-89588-082-2 7" x 9" \$15.95

See our review of BASIC PROGRAMS FOR SCIENTISTS AND ENGINEERS on page 188.

FUN WITH MICROCOMPUTERS AND BASIC

Donald D. Spencer

Level: Novice Rating: **60**

Reston 1981 Paper
 128 Pages ISBN: 0-8359-2214-6 9" x 11" \$9.95

Despite the author's contention that this is a BASIC introduction "ideally suited" to secondary school students or beginning adults, FUN WITH MICROCOMPUTERS AND BASIC is, at most, a supplementary BASIC text for elementary school programmers (their code is legion). Half of the book is a computing-with-cartoons treatment of elementary BASIC programming; the remaining half is a collection of simple game programs and suggested exercises. The cartoons which mathematically join both sections are uniformly less-than-memorable.

A brief general introduction to computers precedes an equally cursory introduction to BASIC. Only the most common commands and programming techniques are mentioned. The "Problems and Programs" section provides approximately fifty programs which reinforce the dozen BASIC commands presented in the book's early chapters. Typical examples are "Russian Roulette," "Wheel of Fortune" and "Buried Treasure." Possible modifications are suggested in the explanatory text which accompanies each program. An appendix provides a glossary of computer terms.

FUN WITH MICROCOMPUTERS AND BASIC may provide a ten to fourteen-year-old with a few hours of programming diversions. Even this audience, however, will quickly tire of its unoriginal, superficial treatment of introductory BASIC programming.

GAME PLAYING WITH BASIC

Donald D. Spencer

Level: Novice Rating: **60**

Hayden 1977 Paper
 166 Pages ISBN: 0-8104-5109-3 6" x 9" \$10.50

Exactly what Spencer was trying to accomplish when he wrote GAME PLAYING WITH BASIC does not emerge clearly in the course of examining the book. In part, he seems to have some desire to teach the uninitiated reader BASIC, although the chapter which has been included for that purpose (pp.12-30) is far too short and vague to be of any value. More generally, his introduction emphasizes the value of pro-

programming games for their "ideal supplemental learning experiences when learning about problem solving, probability, computer programming, statistics, logic and decision making," although the book does not focus on helping this learning experience along either.

Spencer at times attempts to provide some ready-to-use software which might enable the reader to play games with the microcomputer; but the number of programs included for this purpose is surprisingly low. In the case of most of his fourth chapter on "number recreations," moreover, it is not even apparent how the programs and text (which deal with prime numbers, perfect numbers, Fibonacci numbers and amicable numbers) are related to game-playing. Finally, in numerous instances, especially toward the end of the text, Spencer vaguely sketches the rules of a particular game, leaving it to the reader to do the programming; the reader, however, would have a far broader range of games to choose from by simply buying a copy of Hoyle's and attempting to prepare computerized versions of those games which strike his or her fancy.

THE GIANT BOOK OF COMPUTER SOFTWARE

73 Magazine

Level: Advanced Rating: **75**

503 Pages	Tab 1981	Paper	ISBN: 0-8306-1369-2	5" x 8"	\$13.95
-----------	---------------	-------	---------------------	---------	---------

73 MAGAZINE is an amateur radio periodical; this book, contrary to its title, is a specialized collection of magazine articles on topics of interest to the electronic and radio hobbyist.

An introductory section provides an overview of high-level languages, as a prelude to a guide to elementary BASIC. The BASIC introduction is poorly developed, rambling through a series of programming non sequiturs: it is the only BASIC introduction in the literature where subscripted variables are discussed one page after the first mention of the PRINT statement. Just when the confounded reader begins to absorb, by osmosis, some BASIC fundamentals, the nameless authors (we begin to suspect why) introduce an assembly language program. Without a single explanation of any assembly language op codes, a "driver routine for the Heath H-14 line printer" magically appears.

Having dismissed THE GIANT BOOK OF COMPUTER SOFTWARE as the BASIC primer its jacket copy proclaims, we find that it may prove to be of some value as a hobbyist sourcebook of technical essays. Electronics, ham radio operating, slow scan televisions, and antenna manipulation are among the book's program-subject areas. Of special note are

the digitizing techniques used in the spherical trig routines used by radio operators to "eavesdrop" on weather satellites, and some computer-assisted integrated circuit design programs.

This is not a book for the novice programmer; it additionally demands a moderate level of electronic and/or radio expertise. *THE GIANT BOOK OF COMPUTER SOFTWARE* fails miserably as "a step-by-step guide to creating your own programs." It succeeds, however, as a highly specialized collection of scientific and technical programs of value only to the hobbyist, scientist or engineer.

INSIDE BASIC GAMES

Richard Mateosian

Level: Intermediate Rating: **80**

Sybex 1981 Paper
345 Pages ISBN: 0-89588-055-5 7" x 9" \$14.95

Although *INSIDE BASIC GAMES* does contain nine game programs written in BASIC, it is not intended to serve as a convenient collection of software enabling the reader to enter lines of code into a microcomputer and then begin playing games. Instead, Mateosian attempts to use the reader's enthusiasm for games as a means of refining his programming skills and providing him with a deeper proficiency in the use of BASIC; this is a book for those who ultimately aspire to writing their own BASIC programs. Aside from one chapter on time games which requires that a microcomputer be equipped with a system clock (as is the Commodore PET and the IBM Personal Computer), Mateosian's approach is remarkably dialect-free; he has clearly devoted some attention to variations in BASIC syntax and notes these in the footnotes accompanying his lines of code. *INSIDE BASIC GAMES* is a book which is at once challenging and rewarding.

At the same time, however, it is unclear to what degree Mateosian has succeeded in meeting his stated objective. At its most abstract and generalized level, Mateosian's emphasis on top-down structured programming, the use of Free BASIC (a pseudocode incorporating some actual BASIC statements), and the development of generalized programs which can easily be cannibalized to produce still other high-quality programs is highly effective. The more complex task of bringing these generalizations down to earth so that the reader can utilize sound programming principles to produce effective programs, is inadequately covered in the course of the book.

Mateosian's concrete instructional technique focuses on the detailed examination of each of the programs which he has developed for the book; in effect, he presents the reader with a fait accompli—here is

how I did it, see how well it is done? A more effective (although admittedly more difficult and time-consuming) approach would have been to analyze programming alternatives and discuss with the reader the superiority of one algorithm among the many available. In his discussion of guessing games, Mateosian begins to do just this, but in a highly truncated and unsatisfactory form. Nowhere else, however, is this attempted.

But even assuming that Mateosian has chosen the most effective instructional technique available, the ability of the reader to follow and learn from his programs could have been substantially enhanced. In part, Mateosian has violated one of the principles of sound programming practice while he has over-used still another. Aside from the Free BASIC code on which he increasingly relies as his book progresses, Mateosian has provided insufficient documentation with his programs; in particular, the reader could have benefitted enormously from a list of variables used in each program. In part, Mateosian has made far too liberal a use of subroutines, so that the actual program flow becomes somewhat difficult to follow. Mateosian does in fact note this criticism at one point in the text, but chooses to disregard it. In itself, this use of subroutines (and subroutines which call other subroutines) is not necessarily a fault; when combined with the actual format in which the book's programs are presented (with portions of a program scattered throughout an entire chapter), it becomes very difficult for the reader to gain a sense of how the program as a whole operates.

Despite these limitations and their resultant frustrations, *INSIDE BASIC GAMES* remains the single best work to date which attempts to satisfy the desire of the microcomputer user to simultaneously play some well-programmed games and improve his or her BASIC computer and programming skills.

INTRODUCTION TO COMPUTER ANIMATION

Nat Wadsworth

Level: Intermediate Rating: 80

	Hayden 1979	Paper	
77 Pages	ISBN: 0-8104-6279-6	8" x 11"	\$9.95

Low resolution graphics on an Apple II, TRS-80 Model II and Commodore PET are introduced by the author, using relatively simple coding techniques for display and then "animating" shapes on the monitor. The reader is guided from elementary uses of the PLOT function

through manipulation of graphics based upon derivation of the equation for the slope of a line. All coding avoids the use of either shape tables or assembly language subroutines.

The expostulation of Mr. Wadsworth's techniques is accomplished by 'walk throughs' of three programs of increasing complexity: a card-drawing program, an animated clown and a microcomputer variation of football. Though all programs utilize the color capabilities of each of the three systems listed above, users with monochrome monitors can achieve some interesting black-and-white effects. The low-resolution coding techniques focus on eradicating existing grid patterns and substituting others. The author's presentation is "reader-friendly," with ample explanation of each section of code and suggestions for user modifications.

The programmer without the sophisticated skills required by more complex displays will find in INTRODUCTION TO COMPUTER ANIMATION an easy, enjoyable way to explore the graphics capabilities of his personal computer.

LEGAL CARE FOR YOUR SOFTWARE

A Step by Step Guide for Computer Software Writers

Daniel Remer

Level: All Rating: **90**

	Addison-Wesley	1982	Paper	
247 Pages	ISBN: 0-201-06272-0		7" x 11"	\$19.95

Home computing may prove to be the ultimate "cottage" industry: not only are microcomputer users buying software applications for everything from game-playing to gourmet-recipe filing, they are writing their own very saleable programs and attracting the interest of software publishers in their national distribution. LEGAL CARE FOR YOUR SOFTWARE is the BASIC BASIC of legal protection for the software writer.

Each of the three main forms of protection (trade secret, copyright and patent) are explored through hypothetical case studies. The 'step-by-step' of the subtitle even extends to a line-by-line, H & R Block-type explanation of Copyright Office form TX, the copyright registration certificate required by the federal government. The fundamentals of contract law are discussed as a prelude to various software contractual agreements: between software writer and publisher, between company and programmer, and between software developer and test-site proprie-

tor. Perhaps the best feature is the hundred-page appendix filled with standard legal forms (non-copyrighted, we assume) that can be used in each one of the situations analyzed.

LEGAL CARE FOR SOFTWARE is a do-it-yourself guide, written in easily understandable non-legalese. Whether one writes programs or hires program-writers, its purchase and use is incontrovertibly sound advice.

MORE BASIC COMPUTER GAMES

David H. Ahl

Level: Novice Rating: **90**

	Workman	1979	Paper	
185 Pages	ISBN: 0-89480-137-6		9" x 11"	\$7.95

See our review of BASIC COMPUTER GAMES on page 182.

MORE CHESS AND COMPUTERS

David Levy and Monroe Newborn

Level: Intermediate Rating: **70**

	Computer Science	1980	Paper	
117 Pages	ISBN: 0-914894-07-2		5" x 9"	\$11.95

MORE CHESS AND COMPUTERS is the sequel to David Levy's CHESS AND COMPUTERS, and attempts to bring the reader up to date on developments in computerized chess from 1976 to 1979. This was a period which saw enormous advances in computerized chess programs: not only did computers defeat grand masters for the first time, especially in blitz play, but the advent of microcomputers also permitted the development and rapid dissemination of chess programs to chess enthusiasts generally.

MORE CHESS AND COMPUTERS, however, does not catalog these developments nearly as successfully as did Levy's first volume. In part, Levy is concerned with relating the story of his bet that no computer program could defeat him in a match by August 1978; while this provides sufficient melodrama for the book's opening chapter, it is largely used as a substitute for a more thorough discussion of how chess programs changed and improved during this period. Indeed, relatively detailed analysis of chess heuristics and the richly annotated records of chess games in the first volume have largely been replaced by listings of games,

only sometimes supplemented by a chess program's own commentaries made while play was in progress, but often without any annotation at all. The reader is left with a sense that this period does indeed mark the birth of a new era in chess programming, but without any understanding of how changing approaches to chess programming have underlaid this development.

The book also attempts to focus more closely on chess programs for microcomputers by including a chapter devoted to this topic. This chapter, written exclusively by Newborn, notes and briefly evaluates both dedicated chess games (CHESS CHALLENGER and BORIS) and chess software for microcomputers (MIKE and SARGON II), and provides a slightly annotated listing of two microcomputer chess games.

THE MOST POPULAR SUBROUTINES IN BASIC

Ken Tracton

Level: Intermediate Rating: **85**

	Tab	1980	Paper	
182 Pages	ISBN: 0-8306-1050-2	5" x 8"	\$6.95	

Ken Tracton has collected short BASIC subroutines in a variety of application areas and created a valuable reference guide of quick programming "fixes." Most subroutines are less than ten lines long and were originally written in minimal BASIC for a CDC Cyber (not your everyday micro). Their incorporation into a user program requires some dialect conversion. In addition, since all code is reduced to its barest essentials, variables are described in a commentary which precedes the actual code, but never within the subroutine itself. To write a program around the subroutine which computes the future value of a present day investment, for example, the reader must provide his own input prompts (and validations) for the subroutine's variables.

The real utility of this book lies in the sheer number of application problems its subroutines tackle. A sampling of chapter topics includes Annuities, Conversions, Electronics, Graphs, Hyperbolics, Inertia, Mathematics, Physics and Vectors. Test data is provided for each subroutine. In most cases, the code is simply a translation of a formula into a single BASIC statement, solving for a specific variable. This seems rather elementary—unless the given formula is for the annual section polar moment of inertia (whose equation is beyond the scope of this book's computer-generated typeface).

A concluding chapter demonstrates the author's method of incorporating subroutines into application programs. "Conservation of Mo-

mentum" and "Active Filter Design" are programs characteristic of the scientific/technical nature of this particular section, and of the book's general focus.

THE MOST POPULAR SUBROUTINES IN BASIC is a sourcebook whose code can be used to supplement large, existing BASIC programs, or as the kernels of smaller stand-alone applications. We recommend it for its general programming utility.

MURDER IN THE MANSION 2nd Edition And Other Computer Adventures in Pocket Basic

Jim Cole

Level: Novice Rating: **30**

Arcsoft 1981 Paper
96 Pages ISBN: 0-86668-501-4 6" x 9" \$6.95

MURDER IN THE MANSION is a collection of nineteen computer games, the title of which was inspired by an Agatha Christie novel. Cole is clearly excited about these programs, which he feels reflect his editing skills as a programmer:

Working with the Pocket Computer's version of the Basic language requires a sharp editing pencil. Remarks, software explanations, lengthy strings are out. Honing, fine tuning and waste trimming are in...I believe such required editing sharpens the programming skills. It requires maximum efficiency in program writing. The Pocket Computer is a wonderful trainer.(5)

Despite whatever proficiency as an editor he may have, Cole has fine-tuned all meaning out of his programs. Even editors, evidently, are subject to the truism "garbage in, garbage out." All but seven of the programs contained in this little book are of the Russian Roulette variety: the computer randomly selects a number, person, place or thing, and the user must guess it, usually without any prompting from the computer, making the entire proceeding an elaborate exercise in luck. Among the seven exceptions are two simple memory tests, one scrambled word game in which the user must unscramble any of twelve words available to the computer, and three games in which the user must guess a number with some prompting from the computer (too high or too low). Clearly Cole, with his penchant for whodunits (which are the subject of two programs), should have based his title on Christie's THE MURDER OF

ROGER ACKROYD, which, as its critics contend, was an exercise in guesswork rather than in deductive reasoning.

PASCAL PROGRAMS FOR SCIENTISTS AND ENGINEERS

Alan R. Miller

Level: Novice Rating: **90**

Sybx 1981 Paper
374 Pages ISBN: 0-89588-058-X 7" x 9" \$16.95

See our review of BASIC PROGRAMS FOR SCIENTISTS AND ENGINEERS on page 188.

PLAYING THE STOCK & BOND MARKETS With Your Personal Computer

L.R. Schmeltz

Level: Novice Rating: **70**

Tab 1981 Paper
308 Pages ISBN: 0-8306-1251-3 5" x 8" \$9.95

In his search for information on investment analysis, the author, already a personal computer user, uncovered "a large void" in microcomputer literature. His subsequent attempt to publish a combined stock market/personal computer introduction to fill that void, while certainly an admirable effort, falls far short of being a definitive treatment.

Both introductions confuse simplicity with superficiality. Other than a brief survey of financial ratios, the nature of the stock market is never discussed in any detail. Bonds, in any of their various forms, are never even mentioned. Rather, the reader receives general advice on broker and stock selection ("the ideal time to buy your stock is at market bottom") and is tutored on a few formulaic approaches to investing. One particular approach outlined in THE ART OF LOW RISK INVESTING by Michael Zahorchak involves the relation of volatile stocks and market cycles and is the subject of one of PLAYING THE STOCK AND BOND MARKET's BASIC programs. (Mr. Zahorchak's book is mentioned so many times throughout the text that the reader is surprised to find a bibliography which lists other investment-related books.)

The personal computer introduction provides a cursory tour of the hardware market. No conceptual treatment of computing or programming is presented; only two chapters in the book, one a system overview and the other a thirteen-page homily on "computer operations," focus solely on microcomputers.

If this book succeeds at all, it is not in either introductory area, but rather in its synthesis of computing and investment analysis. Sample programs demonstrate comparative stock analysis, calculation of financial ratios and transaction record-keeping. All programs are written in Applesoft BASIC. In addition, appendices provide listings of commercially available financial packages, computer and investment periodicals and telecomputing services.

PLAYING THE STOCK AND BOND MARKET WITH YOUR PERSONAL COMPUTER is a book of interest more for its novel idea than its sparkling execution. We recommend as superior the combination of a good personal computer introduction and a sound stock market guide.

PRACTICAL PASCAL PROGRAMS

Based on the Book PRACTICAL BASIC PROGRAMS

Greg Davidson

Level: Novice Rating: **90**

	Osborne	1982	Paper
207 Pages	ISBN: 0-931988-74-8	9" x 11"	\$15.99

If the reader notes some similarity in style and content between PRACTICAL BASIC PROGRAMS and COMMON BASIC PROGRAMS, it is due not only to their common publisher and author, but to the fact that many of the programs in the former are refined versions of their COMMON BASIC counterparts. (Poole freely admits that programs incorporate elements of similar ones in the earlier text.)

Forty programs treat a spectrum of financial and mathematical applications, but in a considerably more interesting, detailed format. Program introductions double as narratives which provide background material on application topics. These capsule summaries of intrinsically interesting subjects—whether queuing theory (as part of a Swedish Machine Program) or government regulations on profit sharing contributions (as part of a Profit Sharing Contribution Calculator)—elevate PRACTICAL BASIC PROGRAMS far above Poole's previous effort and most of the program-anthology crowd. In many ways, this text is closer in content and spirit to the approaches of BASIC BASIC and EXECUTIVE COMPUTING (see our reviews).

Other topics covered include Accrued Interest on Bonds, Net Present Value and Internal Rate of Return in the financial area, and Markov

Analysis, Lagrangian Interpolation, and Statistical Estimation Theory in the mathematical category. Documentation includes not only the background introductions mentioned earlier, but program examples, practice problems, author modifications of the original program listings, and bibliographies of additional reference material on a given topic.

PRACTICAL BASIC PROGRAMS not only provides a compilation of ready-to-run BASIC programs which address mathematical and financial problems, but—as an extra bonus—opens windows into subject areas rarely encountered by the inquisitive personal computer user.

SIXTY CHALLENGING PROBLEMS WITH BASIC SOLUTIONS 2nd Edition

Donald D. Spencer

Level: Novice Rating: **10**

Hayden 1979 Paper
128 Pages ISBN: 0-8104-5180-0 5" x 9" \$8.50

The only difference between SIXTY CHALLENGING PROBLEMS WITH BASIC SOLUTIONS and SIXTY NOT-SO-CHALLENGING BASIC PROGRAMS is the editor who decided to place the program descriptions in the first half of the book and the program listings in the second half of the book. The programs, rather, "problems," have titles like "I am thinking of a number" and "Sharky-the Card Player." They are trite variations on the same number guessing games, dice and compound interest programs found somewhere between the covers of most hastily assembled BASIC program anthologies.

The highlight of the book is actually the illustrations prepared by John R. Beaty, a high school student in Daytona Beach, Florida.

Our rating would not be so high were it not for John's laudable effort.

SOME COMMON PASCAL PROGRAMS

Based on the Book SOME COMMON BASIC PROGRAMS

Vicki Marney-Petix, editor

Level: Novice Rating: **85**

	Osborne	1982	Paper	
235 Pages	ISBN: 0-931988-73-X		9" x 11"	\$14.99

It is refreshing to find a book whose content accurately reflects the description offered by its title. This is especially true of program anthologies, where the general rule is that one's expectations of the book's text are inversely proportional to the claims made by its title. (Hence, the generally inferior quality of titles like 49 EXPLOSIVE GAMES FOR THE ZX-81 and 33 CHALLENGING COMPUTER GAMES FOR THE TRS-80, APPLE, PET.)

SOME COMMON BASIC PROGRAMS contains seventy-six programs treating very specific, simple applications. The programs, while not categorized by subject area, address either financial or mathematical problems. Most require only minimal configurations of the systems for which they were written; those that generate printer output can be easily modified to display screen output instead. All programs, which average approximately fifty lines of code, are also available on diskette.

Each entry explains the problem which the program sets out to solve, provides an example with typical program output, and then lists each line of program code. A supplementary "option" section proposes some minor modification of the original program and demonstrates this modified version in its own sample program run. The documentation, though less than extensive, adequately supports the jacket copy's ready-to-run claims.

Typical financial programs are a Federal Withholding Tax Calculator, a Check-Writer and a Return-on-Investment Analyzer. The more interesting mathematical applications treated include a number of algebraic, geometric and statistical programs; representative topics are linear interpolation, the roots of a quadratic equation and chi-square distributions.

SOME COMMON BASIC PROGRAMS is a sourcebook of immediately useable BASIC code for financial and mathematical applications. True to the book's title, these programs "perform a variety of common, practical tasks."

STIMULATING SIMULATIONS 2nd Edition

C.W. Engel

Level: Novice Rating: **85**

	Hayden	1977	Paper	
100 Pages	ISBN: 0-8104-5170-0		6" x 9"	\$5.95

STIMULATING SIMULATIONS (ATARI VERSION) 2nd Edition

C.W. Engel

Level: Novice Rating: **85**

	Hayden	1979	Paper	
118 Pages	ISBN: 0-8104-5197-2		6" x 9"	\$5.95

STIMULATING SIMULATIONS provides the microcomputer user with BASIC programs for fifteen simulations which can be run on most microcomputers with relatively minor adaptations. In addition, Engel has provided a brief introduction which discusses computer simulations and the process of creating them, and suggests enhancements which the user might make to each of the programs.

The most striking feature of these two books is that they are largely identical—the whole of STIMULATING SIMULATIONS is reprinted, line-for-line and page-for-page, in STIMULATING SIMULATIONS (Atari Version), which provides information on the necessary program conversions for the Atari microcomputer in a 16-page appendix not contained in STIMULATING SIMULATIONS (only thirteen of the original fifteen programs will run on the Atari, however). The back cover of the Atari version obscures the similarity of these two titles by noting that “original program listings are provided in Atari BASIC.” We will leave the question of why a publisher should issue two almost identical volumes under two slightly different titles to the imagination of the reader.

The “simulations” themselves are frequently not so much genuine computer simulations as traditional computer games; this is true of all three Soccer games, Monster Chase, Lost Treasure, Gone Fishing, Nautical Navigation, Rare Birds and the Devil’s Dungeon. Genuine simulations include Art Auction, Space Flight, Starship Alpha, Forest Fire, Business Management and Diamond Thief (a whodunit). In any case, whether these are “simulations” or “games”, most or all of these programs promise to be extremely enjoyable. Aside from the program listings themselves, Engel has provided suggestions for expanding and

modifying them which the reader can use as a basis for improving their quality. Especially in view of the intensive effort needed to develop good simulations, the programs in *STIMULATING SIMULATIONS* are, at \$5.95 for fifteen programs, a definite bargain.

SUCCESSFUL SOFTWARE FOR SMALL COMPUTERS

Structured Programming in BASIC

Graham Beech

Level: Intermediate Rating: **85**

	Wiley	1982	Paper	
182 Pages	ISBN: 0-471-87458-2		7" x 10"	\$14.95

Structured programming is a time-tested method of producing readable, easily-maintainable program solutions to business and scientific problems. The general technique begins with a formal "structuring" of a problem and ends with the coding of its "substructures" into program subroutines. *SUCCESSFUL SOFTWARE FOR SMALL COMPUTERS* treats the entire process of "top-down," structured BASIC programming, with an emphasis on that interim step from formal structure to efficient code.

The author assumes both a working knowledge of BASIC and a moderate mathematical background (problems in simultaneous linear equations and differential calculus are indicative of the general level required). His approach is novel in that flowcharts are first coded into "Program Description Language (PDL)," a pseudo-code which bridges the gap between diagram and program. PDL is then "translated" into a complete BASIC program (all BASIC programs are written in TRS-80 Model I Microsoft BASIC). Three main application areas are explored using these structured techniques: data processing (storage, sorting, merging and retrieval); simulation (continuous and discrete); and data structure (dynamic and static). Each chapter contains a separate bibliography and set of programming projects, with answers listed in an appendix.

SUCCESSFUL SOFTWARE FOR SMALL COMPUTERS is an introduction to structured programming techniques for the technical or scientific microcomputer user. Given the relative complexity of the programming problems it discusses, we recommend that its consideration be limited to this mathematically sophisticated audience.

TEN EASY PIECES

Creative Programming for Fun and Profit

Carl D. Meyer, Jr. and Hans Sagan
Level: Intermediate Rating: **75**

Hayden 1980 Paper
180 Pages ISBN: 0-8104-5160-3 7" x 10" \$9.50

Deriving its structure from a musical composition, TEN EASY PIECES is divided into a prelude, ten "movements" and a postlude. Each movement is actually a BASIC program with annotations describing not only the program's operation but the key BASIC commands and functions it employs.

The programs themselves range from the unoriginal (Hangman and Lunar Lander) to the vulgar (in "graffiti" PRINT statements define a nude female dancer) to the marginally distracting (a wargame entitled "Fox-hole"). Parts of the text borrow heavily from Sagan's infinitely more interesting BEAT THE ODDS (see our review).

The intermediate BASIC programmer will benefit most from the text which accompanies each program. Variations in BASIC dialects are highlighted (though usually mainframe versions), and general programming techniques are explained. The authors' mathematical backgrounds (both are professors of mathematics at North Carolina State University) are evidenced by the special treatment given the translation of formulae into BASIC code. The Lunar Lander program, for example, uses a form of the quadratic equation in conjunction with Optimal Control Theory to calculate a "soft-landing" in the shortest possible time. A strictly mathematical program is the "tenth movement," which offers a computerized calculation of the value of pi. The book's "postlude" involves the explanation of the "Linear Congruential Method" of generating random numbers. The authors are most informative when, in the BASIC BASIC tradition, they use programming as a pretext for clearly explaining complex mathematical concepts.

TEN EASY PIECES succeeds in conveying to the intermediate BASIC programmer advanced coding techniques which use the micro-computer as a mathematical tool. However, the programs' "movements," around which these discussions revolve, are trite variations of the same old theme.

33 CHALLENGING COMPUTER GAMES FOR TRS-80/APPLE/PET

David Chance

Level: Novice Rating: **75**

	Tab	1981	Paper		
256 Pages	ISBN: 0-8306-1275-0		5" x 8"		\$7.95

The 33 BASIC programs presented in this volume allow the user to play a variety of war games, games of adventure and disaster, of calculation, of speed and order, or to play games which focus on learning or self-improvement. In addition, Chance includes three games using TRS-80 graphics which cannot be run on either the Apple or the PET. The games themselves are interesting, and promise the microcomputer user countless hours of enjoyment.

But if the quality of the games is high, the quality of the accompanying programming and the documentation leaves something to be desired. Although a flowchart is provided for each program, these are not sufficiently detailed to assist the reader in debugging a program should it not run properly the first time. The sample run which accompanies each program and contains output from a correctly-operating program only serves to accentuate the disparity between the reader's expectations and the actual performance of the program. The attempt to fix relatively minor bugs is complicated by the absence of ample REM statements within the programs, as well as by the author's liberal use of GOTO statements; this unstructured, poorly documented technique makes the process of following the program flow difficult and frustrating.

Many of the thirty-three programs in Chance's volume are indeed challenging. The greatest challenge of all, however, comes not from playing the games, but from trying to get them to run in the first place. This is a volume we recommend above all to those who enjoy debugging.

50 PROGRAMS IN BASIC-2nd Edition For the Home, School and Office

Jim Cole

Level: Novice Rating: **45**

	ArcSoft	1981	Paper		
96 Pages	ISBN: 0-86668-502-2		6" x 9"		\$9.95

Based upon the prescient thesis that "the 'small' desktop computers of...the early 80s will be the museum pieces of the late 80s," this book presents a poor man's potpourri of BASIC programs for the pocket

computer. These very elementary programs, designed "specifically" for the business, educational and hobbyist markets, will consequently prove to be of general disinterest to just about everyone who uses BASIC on any size computer. The programs are rehashed versions of such well-worn public domain classics as metric conversion (part of the 'educational' section), compound interest and, in an audacious attempt to make the GUINNESS BOOK OF WORLD RECORDS, five (5) random number generators! 50 PROGRAMS IN BASIC FOR THE HOME, SCHOOL, AND OFFICE is the sequel to MURDER IN THE MANSION (see, or better don't see, our review). While it is not a book to be avoided at ALL costs, we suggest that \$9.95 should suffice (at least until the publisher decides to change its retail price).

55 ADVANCED COMPUTER PROGRAMS IN BASIC

Wm. Scott Watson

Level: Novice Rating: **55**

	Tab	1981	Paper		
252 Pages	ISBN: 0-8306-1295-5		5" x 9"		\$9.95

Almost one half of Watson's collection of "advanced" programs consist of the usual fare of public domain software offered in published works; the user can, for example, enter a short program to calculate means and standard deviations, or derive a range of information about repaying a loan after entering its principal, rate of interest and term of repayment. A notable exception to this is a comparatively long (twelve-page) "calculating dairy cattle feed rations" program, which most users are sure to find extremely useful.

But while the "miscellaneous and specialized" programs, as well as the "business and personal management" programs which Watson provides are of general disinterest to virtually everyone, many of the twenty-six games which he includes are distinguished by their novelty, creativity and imagination. These include a maze game in which the user must find his or her way to the exit (in contrast to other programs which simply display a maze), Concentration (based on the 1960s television quiz show series) and a game of computerized pool.

Unfortunately, however, execution has lagged somewhat behind conception. Watson wrote his programs for a TRS-80 Level II microcomputer, although he does not feel that this should lessen the appeal of his book. In order to atone for any inconvenience suffered by the user of a machine with another dialect of BASIC, Watson includes a command guide of BASIC statements which emphatically does not replace a BASIC

conversions handbook or dictionary. Watson's programming, moreover, relies heavily on the peculiarities of TRS-80 BASIC, some of which may not be familiar to the non-TRS 80 user. All statements which use the BASIC random number generator (the RND function), for example, will have to be rewritten, since the TRS-80 version is virtually unique in that it returns a number which is between one and the function's argument. Similarly, the TRS-80 PRINT @ statement does not have a ready counterpart in many other BASIC versions. In addition, we discovered at least one instance of careless programming: a FOR/NEXT loop with two possible exit points from the loop in the decimal to base conversion program (pp.75-76) produced a syntax error which terminated program execution. In short, the non TRS-80 user who wishes to try out these programs must have both a good working knowledge of BASIC and a great deal of patience.

57 PRACTICAL PROGRAMS & GAMES IN BASIC

Ken Tracton

Level: Novice Rating: **50**

	Tab	1978	Paper	
204 Pages	ISBN: 0-8306-1000-6		5" x 7"	\$7.95

Aside from seven games (Blackjack, Craps, I Ching, Number Guess, One-Armed Bandit and Star Wars I-II), the fifty-seven programs contained in this volume focus primarily on applications in mathematics and statistics. Many of these programs are standard ones which are frequently provided in introductory programming texts (Arithmetic Mean, Bubble Sort, etc.). Others hardly seem worth the effort required to enter them; Chi-Square for example, requires that the user enter both observed and expected frequencies—but if the user must calculate the expected frequencies, it is very little extra trouble to calculate the Chi-square statistic itself.

In theory, all programs have been “written in such a manner that they will operate even with a simplified subset of full Basic.”(5) However, our own attempts to run selected programs under Microsoft BASIC required extensive debugging, which makes this volume a dubious value in view of the simplicity of its programs and the comparatively small number of statements they contain.

67 READY TO RUN PROGRAMS IN BASIC

Graphics, Home & Business, Education, Games

Wm. Scott Watson

Level: Novice Rating: **85**

	Tab	1981	Paper		
182 Pages	ISBN: 0-8306-1195-9		5" x 8"		\$7.95

Written in TRS-80 Level I BASIC, this TAB anthology of "Ready-to-Run Programs" is a notch above other purely public domain compendiums of BASIC code. Of the categories of programs treated—graphics, home and business, education and games—it is undoubtedly "games" which receives the most attention in terms of code: almost 50% of this book is comprised of game programs. Some of the more original programs, which will all run in approximately 4K of memory, are: "Putting Practice" (graphic golf), "Russian Roulette for Two" (a Deer Hunter derivative), and "Salary Evaluator" (whether it's "packed with provocative information" really depends upon your tax bracket).

Each program uses the standard TAB format of program prologue, variables list, suggested program variations and actual code listing. The author has, unfortunately, complicated the process of translating TRS-80 BASIC into other dialects by the use of an abbreviated command notation ("P.A." is "print-at," "N." is "next," and "P." can either be "print" or "point").

Nevertheless, 67 READY-TO-RUN-PROGRAMS IN BASIC is one of the better offerings of its type; a handful of its funnier, simple games programs more than justifies its purchase price.

VisiCalc

DOING BUSINESS WITH VISICALC

Stanley R. Trost

Level: Intermediate Rating: **85**

	Sybex	1982	Paper		
259 Pages	ISBN: 0-89588-086-5		7" x 9"		\$11.95

The hobbyist eager to build his own Z80 microcomputer will probably purchase a Ciarcia text; the game-player anxious to graduate to even more arcane levels of keyboard dexterity will begin to modify code with the assistance of the latest compendium of high-res BASIC subroutines;

and the businessperson/VisiCalc user, having had a taste of the real power of this software tool, will inevitably cross paths with Stanley Trost's **DOING BUSINESS WITH VISICALC**.

Though introductory material is presented, a working knowledge of VisiCalc is assumed. The author's special VisiCalc notation is easy to understand; diagrams assist the reader when the spread sheet under consideration is deemed too complex for notation alone.

VisiCalc's inherent applicability to business forecasting, a consequence of its ability to recalculate entire assemblages of data based upon their mathematical interrelationships, is put to good use in each of the author's forty-seven applications. These applications range from the plain vanilla of standard P & L's to the fudge ripple of Form 1040 and schedule A calculations. All spread sheets are designed solely within the realm of VisiCalc's command set, without recourse to external BASIC subroutines.

Though minor misprints appear in some of the grid-locations of Trost's VisiCalc notation (the discount bond application is a prime example), **DOING BUSINESS WITH VISICALC** is a valuable source of pure code for the personal computer user interested in transforming his machine into a potent financial tool.

EXECUTIVE VISICALC FOR THE APPLE COMPUTER

Roger E. Clark

Level: Intermediate Rating: 70

Addison-Wesley	1983	Paper	
130 Pages	ISBN: 0-201-10242-0	10" x 8"	\$14.95

EXECUTIVE VISICALC FOR THE APPLE COMPUTER is unique in current VisiCalc literature in that it does not contain a single complete program/model. Rather, the author has employed a "tips and techniques" treatment which uses accounting examples to highlight special formulae or functions. It provides the experienced VisiCalc user with the insights required for an expert manipulation of this powerful software package.

The author describes his audience as DOS Version 3.3 (16 sector) VisiCalc users who have completed the first four lessons of the VisiCalc tutorial manual (this represents roughly half of Fylistra and Kling's text). After a brief discussion of the construction of VisiCalc formulae, a "Weekly/Monthly reporter" is used to demonstrate Boolean logic functions and their role in a scheduling model. Overlays, a topic which receives a scant one-page treatment in the VisiCalc manual, is the topic of a chapter featuring a "Sales Analysis System." This VisiCalc ability—to

blend spread sheets and thereby reduce repetitive calculations—is a good example of the type of supplementary information contained in Clark's book. This feature, like others, is described but never explicitly demonstrated in the context of a VisiCalc "program." It is as if a text on advanced BASIC programming discussed the code necessary to randomly access data in a disk file, but never related the discussion to the actual BASIC statements embodied in a functional program. Ironically, VisiCalc's own DIF files receive precisely this type of treatment in a chapter on the DATA SAVE function.

Of special interest are two chapters which contain a survey of software and hardware products available to the VisiCalc user: these include memory expansion boards, formatting aids, utilities which consolidate VisiCalc individual models into a single master model, and others which facilitate the transmission of VisiCalc files via communication between processors.

EXECUTIVE VISICALC FOR THE APPLE COMPUTER is a VisiCalc user's notebook of the advanced features of this software product. It will be most useful to those satisfied by a cursory, practically code-less description of these functions.

HOW TO USE VISICALC SUPERCALC

Carlton Shrum

Level: Novice Rating: **65**

Alfred 1982 Paper
48 Pages ISBN: 0-88284-223-4 4" x 11" \$2.95

The real value of this Alfred Handy Guide is not in its use as an operational guide to interactive spread sheet programs, as the title purports. Rather, it best serves as a comparative evaluation of the VisiCalc and SuperCalc software packages.

The discussion of the features available to the spread sheet user does cover the formatting, copying and replicating functions through a profit-and-loss statement demonstration. It somehow neglects to mention the equally important window manipulation, bar graphs and repeating label commands. More esoteric functions like DIF format file saves are never given the slightest allusion. The 'how to use' treatment is noticeably inferior to the publisher's documentation which usually accompanies these packages.

However, in the process of explaining each of his eclectic choices of important features, the author does parenthetically note the differences between VisiCalc and SuperCalc. SuperCalc, for example, has a protect

command which presents inadvertent user alteration of the contents of a cell. An appendix compares forty different spread sheet features, noting each of these VisiCalc/SuperCalc differences.

The only utility of this 48-page guide, admittedly a dubious one, is in this two-product comparison. See the VisiCalc reviews in this book for sources of better introductory guides to spread sheet programs.

MASTERING VISICALC

Douglas Hergert

Level: Intermediate Rating: 90

Sybex 1983 Paper
 217 Pages ISBN: 0-89588-090-3 7" x 8" \$12.95

Not only is MASTERING VISICALC's explanation of VisiCalc's functions more lucid than any of the chapters in the Visicorp reference manual, it expands this software product's utility far beyond its original design.

The first half of the book discusses the inherent capabilities of VisiCalc by elaborating upon simple examples; a spread sheet is constructed using most BASIC commands until the user is confronted with a structured analysis of his raw data that would have taken weeks to program. (One of the authors, an historical statistician by profession, entered data from an 1890 Russian census to "test" this early discussion in MASTERING VISICALC; by the end of his first session he had constructed an analysis that two research assistants had spent days performing.)

This first section, with its clear, easy-to-read descriptions of all of VisiCalc's formatting and naming conventions, more than cost-justifies the purchase of this book. The second half, the real 'extra,' provides a technique for the unlimited manipulation of data used by VisiCalc. By using BASIC subroutines included in an appendix to read VisiCalc's specially formatted save files (DIF files), the user can perform any calculation not part of the VisiCalc command set (e.g., variance and standard deviation). Data can then be restored for additional display by VisiCalc. MASTERING VISICALC is a relatively new addition to this particular niche in microcomputer literature; its style and scope destine it to become one of the most popular treatments of the subject.

VISICALC, ADVANCED VERSION

Worksheets for Business

Van Wolverton

Level: Advanced Rating: **85**

229 Pages VisiPress 1983 Paper ISBN: 0-912213-00-0 8" x 10" \$18.95

This VisiCorp publication contains twelve model worksheets designed for use with the advanced version of VisiCalc. All focus on financial applications: budgeting, forecasting and fundamental business reporting form the primary subject areas.

Each of these worksheets highlights features not available in the original version of VisiCalc (Keystroke Memory, modulo arithmetic, extended financial calculations and a variable column width). VisiCalc users interested in a sourcebook of worksheet applications may still benefit from VISICALC, ADVANCED VERSION FOR BUSINESS, however; Van Wolverton has provided guidelines for approximating the functions of most of these advanced features with combinations of VisiCalc commands.

Though a standardized convention for listing VisiCalc instructions does not exist, Wolverton's notation is immediately comprehensible. The cursor position for each entry is included for quick reference, and an appendix lists the values of every cell of every worksheet. Since the main difficulty with keyboard entry of a VisiCalc listing involves the accurate duplication of formulae, each application contains a special section describing calculations used and providing explanations of the role of each variable.

Advanced techniques employed by Wolverton include the use of consolidated DIF files and Keystroke Memory functions to handle the setup characters for condensed report printing. Modifications of the completed worksheet are frequently suggested: in a cash flow application, for example, the effects of a separate line entry for capital purchases upon other sections of the worksheet are discussed.

Users of the advanced version of VisiCalc will find Van Wolverton's book an immediately useable introduction to its extended functions through a series of practical business applications. VisiCalc owners, with some modification of the author's code, will discover an additional source of clearly documented financial worksheets.

THE VISICALC BOOK-APPLE EDITION

Donald H. Beil

Level: Novice Rating: **75**

Reston 1982 Paper
 301 Pages ISBN: 0-8359-8397-8 7" x 9" \$14.95

THE VISICALC BOOK-ATARI EDITION

Donald H. Beil

Level: Novice Rating: **75**

Reston 1982 Paper
 298 Pages ISBN: 0-8359-8393-5 7" x 9" \$14.95

Both editions of this book place most of their emphasis on mastery of VisiCalc's naming conventions and built-in functions—unfortunately at the expense of integrating these names and commands into a coherent use of the VisiCalc utility. Though diagrams and brief examples accompany each description of a command, few full demonstrations of VisiCalc incorporate a series of such commands into a working spread sheet. The forty practice exercises included in the final chapter might well have served this purpose; however, the absence of any answer key or even token explanation of the correct solutions merely highlights this serious deficiency in the book's presentation.

Original points do include a trouble shooting guide to the most common VisiCalc errors, and suggestions for user-friendly template creation. The chapter on external spread sheet documentation represents one of the most professional approaches to VisiCalc's business use.

At its worst, THE VISICALC BOOK is a glorified glossary of VisiCalc terms; at its best, it is an encyclopedic reference work of value to VisiCalc users. In either case, it is not the ideal vehicle for learning to use this popular software product.

VISICALC FOR APPLE II PLUS

Microcomputer Power Series

Edouard J. Desautels

Level: Novice Rating: **90**

Wm. C. Brown 1982 Paper
 155 Pages ISBN: 0-697-09967-9 9" x 11" \$16.95

VISICALC FOR THE IBM PERSONAL COMPUTER

Edouard J. Desautels

Level: Novice Rating: **90**

Wm. C. Brown 1982 Paper
155 Pages ISBN: 0-697-09967-9 9" x 11" \$16.95

VISICALC FOR THE TRS-80 MODEL I & III COMPUTERS

Edouard J. Desautels

Level: Novice Rating: **90**

Wm. C. Brown 1982 Paper
152 Pages ISBN: 0-697-09956-3 9" x 11" \$16.95

VISICALC FOR THE TRS-80 MODEL II & 16 COMPUTERS

Edouard J. Desautels

Level: Novice Rating: **90**

Wm. C. Brown 1982 Paper
152 Pages ISBN: 0-697-09955-5 8" x 9" \$16.95

This spiral-bound series customizes its treatment of VisiCalc to individual microcomputer systems: introductory chapters summarize IBM Personal Computer and TRS-80 operating instructions, and explain each keyboard with respect to its VisiCalc functions. (In fact, there is very little difference between IBM Personal Computer and TRS-80 VisiCalc operation: the inclusion of machine-specific load instructions is presumably designed for first-time users who have purchased micros in order to use VisiCalc.)

VisiCalc "letter" codes and formatting features are explained in machine exercises which grow in complexity. The author's pseudo-code for VisiCalc commands is simple and clear, incorporating prompts indicating proper command key use directly into the code. For example, ">D1 ENTER TOTAL" indicates where to initially position the cursor ("D1"), what key to press (ENTER), and what is to be entered at the assigned position ("TOTAL").

The full VisiCalc instruction set of commands, functions and formatting features is covered using this notation in application examples. These include an IRA investment analysis and a class test average/summary. Throughout the book, reference is made to parallel chapters in

the VisiCalc reference manual. The author merely elaborates upon topics which he feels receive inadequate manual treatment (e.g., DIF Files), rather than the fundamentals discussed by the manufacturer. (This is rather paradoxical for a text which does re-explain very fundamental operating instructions.)

A special "Case Studies" section reviews and reinforces preceding lessons. Among the applications developed are a "Portfolio Evaluator" (which keeps track of stock gains and losses based on a fluctuating cost/share) and a "Sales Forecaster" which averages yearly compounded growth in order to project future sales.

"When to Avoid VisiCalc" is a one-of-a-kind chapter which reminds users that situations do exist which require more programming power than even VisiCalc can muster. (For the unbelievers, these include the use of multiple data files and complex IF/THEN calculations.)

VISICALC FOR THE IBM PERSONAL COMPUTER is a thoughtfully designed, simple introduction which requires the VisiCalc reference manual but no microcomputer programming or operating experience. We highly recommend its purchase to novices whose primary personal computer application involves the use of VisiCalc.

VISICALC HOME AND OFFICE COMPANION

David M. Castlewitz and Lawrence J. Chisausky

Level: Novice Rating: **70**

	Osborne	1982	Paper	
181 Pages	ISBN: 0-931988-50-0		9" x 11"	\$15.99

The VISICALC HOME AND OFFICE COMPANION is just that—a companion volume to the manual which accompanies the VisiCalc program. The authors do not intend to improve their readers' ability to use VisiCalc by explaining how the program actually operates. Instead, they offer the user fifty models, complete with printouts, listings demonstrating how to key in each entry and suggestions for enhancing their quality or level of detail. The applications are primarily business and financial, although six deal with personal finance, two with social planning and one with household management.

The models were developed using the Apple version of VisiCalc, which is also what we used to test several of them. The reader who actually uses this book will find that it becomes a rather time-consuming companion—the models as the authors entered them were long and cumbersome, in large measure because they made minimal use of one of the most powerful features of the VisiCalc package—its ability to repli-

cate lines. While this volume may be ideal for the VisiCalc user who has absolutely no idea how to use the nice computerized spread sheet, it is of minimal value to the VisiCalc enthusiast who hopes to refine his or her skills at using VisiCalc.

WORD PROCESSING

COMPUTER WORD PROCESSING

Do You Want It?

R. Dean Boyer

Level: Novice Rating: **80**

Que 1981 Paper
149 Pages ISBN: 0-88022-000-7 5" x 7" \$14.95

Que is a midwestern management consulting group that publishes a wide range of microcomputer texts aimed at providing the personal computer user with tools for evaluating software. The role of consultant is evident in Dean Boyer's treatment of word processing: each step of the implementation of a word processing system is examined and critiqued, from the initial clarification of need to software upgrades after successful system installation.

The word processing hardware market is surveyed and divided into three categories: intelligent typewriters, microcomputers, and stand-alone dedicated word processors. A legal office system using a three-terminal dedicated machine is examined for productivity increases, security arrangements, formatting flexibility and shortcomings. On this last point, the author makes it clear that his book is not an apologia for word processing as a way of corporate life; some of the most valuable sections of COMPUTER WORD PROCESSING treat anti-word processing arguments. These include situations where correspondence is limited, or where the related productivity problems of slower startup, operation training or difficulty with "mode" changes precludes computerization.

The hardware survey is followed by a software overview. Generic software functions like editing, printing and word wraps are explained, with the most popular word processing packages evaluated. These include Apple Writer, Electric Pencil and Magic Wand. Criteria for additional user evaluation are presented in a function list for word processing software.

However, the software evaluation section falls short of a comprehensive analysis of available word processing packages. The user function list is never applied to the packages which are briefly discussed, for example. The author's recommendation is to refer to additional Que texts

for more detailed information (WordStar is reviewed only in CP/M WORD PROCESSING and Scripsit in yet another Que offering). This rather irritating bit of self-promotion is only outdone by the boldface listing of Que Corporation's phone number on the book's title page.

The businessperson to whom this book is addressed is precisely the reader who will most likely overlook this blemish in favor of the book's practical business overview. It is to this audience exclusively that we recommend COMPUTER WORD PROCESSING.

HOW TO BUY A WORD PROCESSOR

Steven Manus, Michael Scriven

Level: Novice Rating: **45**

62 Pages Alfred 1982 Paper
ISBN: 0-882-84222-6 4" x 11" \$2.95

Like much of the Alfred Handy Guide Series, this small and inexpensive volume contains a surprisingly detailed and comprehensive overview of word processing systems. It is not, however, tailored to the needs of the microcomputer buyer or user, who is primarily concerned with the ease with which a given microcomputer system can be used for word processing applications, as well as with the availability of word processing software for that system. Instead of addressing these questions, HOW TO BUY A WORD PROCESSOR surveys the entire spectrum of available word processing equipment ranging from electronic typewriters and personal computers at the lower end of the spectrum, to a variety of dedicated word processing systems at the upper end.

Despite the misplaced emphasis which makes this small volume of greater interest to corporate giants than to individuals who are tired of sitting at their typewriters typing and retyping pages of text, Manus and Scriven's survey of word processing features and functions, as well as their discussion of hardware components (and especially of keyboards and printers), still can be of use to the microcomputer enthusiast. In addition, the book's first appendix compares seven word processing packages for microcomputers (including Wordstar, Spellbinder, Easywriter II and Appewriter II and III) with respect to their features and price.

INTRODUCTION TO WORD PROCESSING

Hal Glatzer

Level: Novice Rating: **70**

210 Pages Sybex 1981 Paper
ISBN: 0-89588-076-8 6" x 9" \$12.95

The author is a journalist and television producer, which might explain this book's episodic, big-newspaper slant on word processing. The general discussion structures itself along "hardware" lines (one characteristic of INTRODUCTION TO WORD PROCESSING is its apparent lack of externally-imposed structure): the electronic typewriter, microcomputer, stand-alone word processor, minicomputer and mainframe are reference points for chapters treating the evolution and current use of word processing systems.

INTRODUCTION TO WORD PROCESSING is at its best when describing those generic functions which form the ease-of-use argument for word processing: scrolling, global searching and on-line editing, for example. The three chapters on actual word processor use give the reader a good overview of its most powerful features and how they are used in the typical business environment. The rest of the book is a rambling, unfocused soliloquy on anything that can be remotely construed as impinging on the author's topic. We are given the less-than-informative hypothetical views of the business and professional members of a Chamber-of-Commerce on word processing. In a chapter entitled "How to Get Hands-on Experience," we find sales brochures lauded as "quite useful" and "part of the (word processing) learning process." A "comparative costs" section presents only a token survey of available word processing packages, neglecting to name any specific package other than IBM's Advanced Text Management System, and even neglecting its own previously defined hardware categories (mainframe and minicomputer suddenly merge into "large computers").

INTRODUCTION TO WORD PROCESSING is distinguished only by an exceptionally superficial, diffuse treatment of its subject. While the average reader will certainly deduce some very valuable facts about word processing from its commentary, we suggest that he or she instead heed that boldface product of our own word processing system—and look for a book with a higher rating.

INTRODUCTION TO WORDSTAR

Arthur Naiman

Level: Novice Rating: **95**

Sybex 1982 Paper
 202 Pages ISBN: 0-89588-077-6 7" x 9" \$11.95

While Naiman believes that Micropro has designed one of the most impressive word processing packages available for microcomputers, he laments that the reference manuals accompanying WordStar have not been assembled with much attention to the needs of those trying to learn to use it. Naiman's own introduction to WordStar (as well as to MailMerge and SpellStar, two supplementary programs produced by MicroPro), however, fills this gap quite well.

Naiman first indulges the desire of an impatient reader to enter a document with the aid of WordStar, and then turns to a detailed and extremely well-organized step-by-step exposition of the features and commands available in WordStar. Since WordStar itself consists of a sequence of menu screens, Naiman takes the reader carefully through these, beginning with the editing menu to enter text and concluding with the menu for printing a document. In the process, he offers a comprehensive treatment of all but the most advanced features of WordStar—its block handling and global search and change capabilities, the fine points of handling and manipulating files and the procedures for both on-screen formatting and printing text in special formats.

Although Naiman has designed the text to be used during on-line practice with WordStar, he correctly notes that this is not altogether necessary; the potential purchaser of a word processing package who is having difficulty deciding which one to buy can also benefit from reading INTRODUCTION TO WORDSTAR.

LOW-COST WORD PROCESSING

Laurence Press

Level: Novice Rating: **70**

Addison-Wesley 1983 Paper
 227 Pages ISBN: 0-201-05735-2 7" x 9" \$10.95

Most introductory word processing books make the distinction between personal computers outfitted with software packages and more expensive dedicated processors. While Press observes this traditional distinction in the first chapter of LOW-COST WORD PROCESSING, even to the extent of urging readers to see both "low-cost and professional-quality systems" in computer stores, it is the last time that this common

conception of cost is used as a criterion for the book's content. This is a text about low, medium and high cost word processing. In terms of hardware, one is equally likely to see an illustration of a Xerox Star work station as an Apple II.

Given the fact that the author's brush paints a much wider picture than the title indicates, what remains is an adequate introduction to word processing. After a brief consideration of the spectrum of word processing applications, Press embarks on a component-by-component tour of word processing hardware. Dot-matrix and formed-character printers, ergonomically designed keyboards, full-page video displays and on-line storage capacity are all covered in this survey of the peripherals marketplace.

Software is first discussed in terms of generally desirable characteristics and then on a specific-feature basis. The editing function, for example, is analyzed not only in terms of its focal role in any word processing system, but also with regard to the flexibility of screen formatting, document assembly, and search and replace capabilities. An "other software" section uses the potential need for integrated data processing and word processing facilities as an excuse to digress into a description of VisiCalc.

A final chapter, which ties in to the book's title better than any other, treats the possible tax benefits of word processing purchases. (The author's advice on selection of specific systems never transcends the "Company A vs. Company B" comparison.)

LOW-COST WORD PROCESSING is a rambling discourse on a wide range of systems. While its market overview will certainly prove informative to the novice, we recommend any of the more highly rated treatments in this category.

THE WORD PROCESSING BOOK

A Short Course in Computer Literacy

Peter A. McWilliams

Level: Novice Rating: **85**

	Prelude 1982	Paper	
240 Pages	ISBN: 0-931580-98-6	6"x 9"	\$8.95

Virtually everyone who writes regularly, McWilliams argues in THE WORD PROCESSING BOOK, can benefit enormously from the purchase of a microcomputer for use as a word processor. Not only does he succeed in proving the point, but he goes on to make a number of recommendations about the relative merits of various microcomputer systems and word processing packages.

McWilliams offers a treatment which is primarily literary. The text is light and enjoyable to read, despite some verbosity, several tasteless jokes and occasional displays of egotism. The photographs of microcomputer systems are supplemented by century-old advertisements for "word processing" equipment, reproductions of traditional wood cuttings and illustrations of primarily Biblical themes. McWilliams curtails his discussion of microcomputer basics to a bare minimum, but still provides the reader which more than sufficient information to make a judicious choice; paradoxically, works which are much richer in technical detail are far more prone to muddy the practical issue of hardware and software selection.

While *THE WORD PROCESSING BOOK* excels in informing the reader what to look for in good microcomputer systems and word processing software, McWilliams goes beyond this to express a number of opinions about the relative virtues of various microcomputer systems, software packages and printers. Here we urge the reader to be careful; although some of his opinions are decidedly accurate, others are not. His endorsement of certain dot matrix printers, for example, is at variance with the experience of many personal computer users.

Although in some instances McWilliams should not be taken too literally, *THE PERSONAL COMPUTER BOOK* nevertheless provides a highly distinctive and outstanding guide to selecting a word processor.

WORD PROCESSING PRIMER

Mitchell Waite and Julie Arca

Level: Novice Rating: **90**

	Byte	1982	Paper	
188 Pages	ISBN: 0-07-067761-1		6" x 10"	\$14.95

For the author, attorney or administrative assistant who is considering the purchase of a personal computer primarily for word processing applications, *WORD PROCESSING PRIMER* provides an entertaining, structured approach to judicious selection. A highly readable history of word processing from intelligent typewriters to voice-recognition systems is followed by a succinct, comprehensible introduction to microcomputer functions. Both are supplemented by illustrations and photographs.

The real utility of this book lies in the authors' approach to hardware and software selection on an occupational level: one set of features is optimum for the secretary, another for the mass mailer. A survey of the ten most popular word processing packages also evidences this application-specific analysis of an individual's word processing needs. AppleWriter, for example, is recommended for "the home and light business user," while WordStar is advised for the author or technical writer.

Purchase of the WORD PROCESSING PRIMER is the first step in an intelligent word processing decision.

WORDSTAR MADE EASY

Walter A. Ettlín

Level: Novice Rating: **60**

Osborne 1981 Paper
125 Pages ISBN: 0-931988-69-1 7" x 10" \$11.95

Although some users have found that MicroPro has provided an altogether inadequate reference manual to accompany its otherwise excellent WordStar word processing package, Ettlín evidently does not share this opinion. WORDSTAR MADE EASY "is not intended to replace the WORDSTAR USER'S GUIDE published by MicroPro. The MicroPro manual will supplement the command descriptions presented here, as well as introduce any WordStar commands not covered in this book."(xi) All that Ettlín has to offer the reader is a somewhat more organized introduction to the WordStar manual; for this purpose, he includes page references to the manual for each feature of WordStar which he introduces. An additional notable feature of WORDSTAR MADE EASY is the large number of half-blank or completely blank pages interspersed throughout a text which is already quite short; blank pages alone (including those devoted to chapter headings) occupy over one quarter of the text. Finally, the exercises provided at the end of each chapter are extremely simplistic and do not contribute either to appreciably increasing the reader's grasp of WordStar's individual features, or to enabling the user to manipulate WordStar as a comprehensive word processing package.

WORDSTAR MADE EASY does not, in short, help to make using WordStar any easier. For those who find that a text to supplement MicroPro's reference manual is either useful or necessary, we recommend Sybex's more comprehensive INTRODUCTION TO WORDSTAR, written by Arthur Naiman.

WRITING WITH A WORD PROCESSOR

William Zinsser

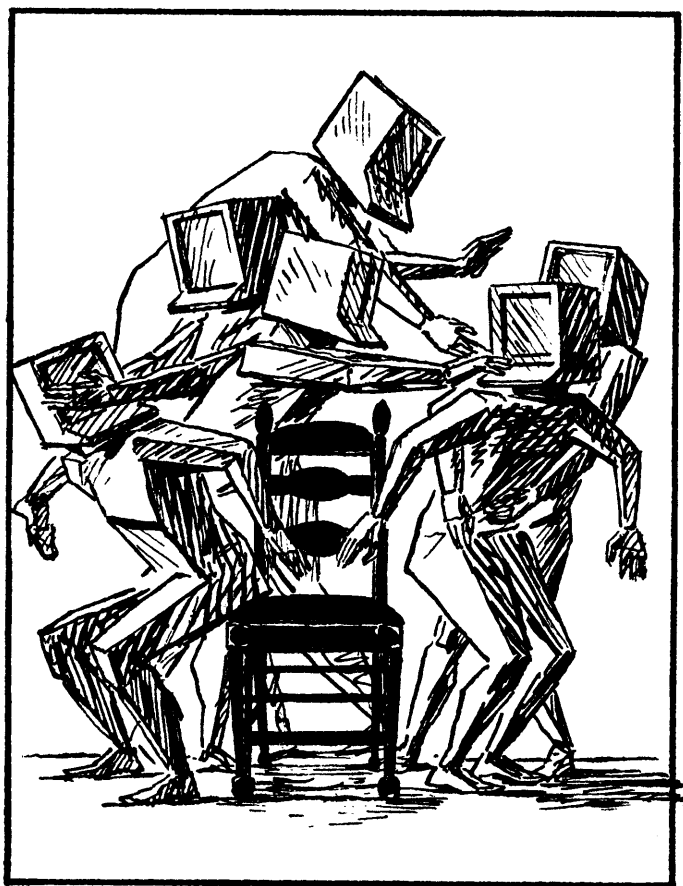
Level: Novice Rating: **90**

Harper & Row 1983 Paper
117 Pages ISBN: 0-06-091060-7 6" x 8" \$5.95

"Anyone who can operate a Venetian blind can operate a word processor" is the conclusion reached by William Zinsser after ninety-two pages of alternating clammy hands and technological epiphanies. The author of *ON WRITING WELL* has crafted, in his own immediately recognizable style, a contemporary saga of one writer's tumultuous affair with an IBM Displaywriter.

The transition from Underwood to IBM occurs in Zinsser's Book-of-the-Month Club office; it is marked by hardware and software errors, lost data and considerable psychological trauma. But the inimitable Zinsser prevails. As an author whose reputation is founded upon an ability to clearly relate sound advice, Zinsser's account of his word processing experience is filled with pragmatic how-to-cope suggestions not found in any of the blander discourses in the literature. Jargon is deflated and explained; psychological "hang-ups" are addressed; even the more mundane considerations of eyestrain, hot-line numbers and diskette duplication fall under Zinsser's palliative scrutiny.

This is not a book about word processing software or hardware selection. Except for cursory treatments of the hardware components of a word processing system, it is hardly a technical book at all. It is a noteworthy source of literary salve for computer-phobic writers confronted with the harsh realities of the microcomputer revolution.



6

Specific Microcomputer Systems

At some time during the frenetic selection process, the music stopped; this chapter is intended primarily for those who have thrown caution to the winds and actually purchased their own systems. We have included books on machines produced by the six leading microcomputer manufacturers: Apple, International Business Machines, the Tandy Corporation, Atari, Commodore Business Machines and Timex.

The profusion of microcomputer manufacturers has spawned an array of BASIC dialects whose incompatibilities confuse and frustrate the novice programmer. Most of the titles reviewed in this chapter focus on a particular dialect of BASIC for a specific system; these include excellent introductory treatments (such as David A. Lien's *LEARNING IBM BASIC*) as well as works for intermediate and advanced programmers (such as Lewis Rosenfelder's *BASIC FASTER AND BETTER* for the TRS-80). Other texts treat additional machine-specific languages, including assembly language; good examples are *APPLE MACHINE LANGUAGE* and *THE ATARI ASSEMBLER*, by Don and Kurt Inman.

Another category of books, by focusing on a system and its operation, attempts to supplement the system's operating manual, thus allowing the user to go far beyond simple programming in BASIC; works of this kind include the *VIC-20 PROGRAMMER'S REFERENCE GUIDE* or the *APPLE II USER'S GUIDE*.

Finally, a number of titles examine the uses to which these machines can be put. Characteristic are software directories (*THE BOOK OF APPLE SOFTWARE*, for instance) or guides to software selection and utilization (such as *APPLE II WORD PROCESSING* or the titles reviewed in the *VisiCalc*

section in Chapter Five). But most common are collections of published software which offer ready-to-run programs for a wide range of applications; exemplary compendiums of this sort include 32 BASIC PROGRAMS FOR THE APPLE or CIRCUIT DESIGN PROGRAMS for the Apple or TRS-80.

APPLE

THE ADDISON-WESLEY BOOK OF APPLE COMPUTER SOFTWARE 1983

Jeffrey Stanton, Robert P. Wells, et. al.

Level: All Rating: **90**

	Book Co.	1983	Paper	
491 Pages	ISBN: 0-201-10279-x		9" x 10"	\$19.95

All three of the most popular Apple software directories—THE ADDISON-WESLEY BOOK OF APPLE COMPUTER SOFTWARE, THE BLUE BOOK FOR THE APPLE COMPUTER, and VANLOVES APPLE II/III SOFTWARE DIRECTORY—purport to be definitive directories of existing Apple software. Addison-Wesley proclaims itself the "...one necessary book that should be on all Apple Computer owners' shelves." Vanloves is touted as the "...most accurate and comprehensive review of Apple Computer-compatible programs and accessories." THE BLUE BOOK bills itself as "...the complete 'where to find it' book of software, hardware, and accessories" for the Apple II.

The cheapest of these directories is the ADDISON-WESLEY BOOK OF APPLE COMPUTER SOFTWARE at \$19.95, and it enjoys a number of other distinctions. It is the only book that rates each software package in a number of key categories, including ease of use, visual appeal, documentation and reliability. In addition, unlike VANLOVES or THE BLUE BOOK, each entry has been meticulously tested, with final ratings the result of each package's combined "score." None of the commentaries which accompany each listing has been copied from publisher's promotional material (again, unlike both VANLOVES and THE BLUE BOOK). All software is grouped into four categories: business, education, utilities and games. Each entry lists the publishing company, the language in which programs are written, hardware requirements, suggested retail price and general availability. Unfortunately, the Addison-Wesley book manages to treat only a total of five-hundred software packages in this exemplary fashion.

VANLOVES, with 2,300 entries, is indeed the "most... comprehensive" of the Apple directories. The 1983 edition also qualifies as the most current guide to the mercurial Apple software market. However, unlike the Addison-Wesley or BLUE BOOK, it omits a standardized treatment of

hardware requirements and prices for individual packages. Moreover, of the three, VANLOVES is guilty of the lowest production quality: typewriter-quality typeface was used on a poor choice of paper stock, resulting in tedious reading. VANLOVES is further distinguished by its lack of a composite alphabetic index. With almost as many categories of software as the Addison-Wesley has listings, finding a given package in VANLOVES reminds the reader of one of the reasons for the incredible success of microcomputers—fast, accurate data retrieval. In most cases, the text which accompanies each package is copied directly from the publisher's 'label,' giving no real indication of its relative merit. VANLOVES also neglects a consistent mention of the hardware requirements of each entry, so that the possibility of running out of memory in the middle of 'Alien Lander' becomes uncomfortably real. THE BLUE BOOK also claims 2,300 listings, although a higher percentage than either VANLOVES or Addison-Wesley are non-software (approximately 10%). It does contain a composite alphabetic index and shares an acceptable production quality with the Addison-Wesley. Very much like VANLOVES, however, the text accompanying each entry is publisher copy (many of the commentaries in VANLOVES and THE BLUE BOOK are identical).

THE BLUE BOOK shares with Addison-Wesley an affinity for supplementary graphics for the more popular packages. Its non-software listings are an extra bonus: these include boards, accessories, peripherals, books and magazines.

In summary, Addison-Wesley provides the best guide to a limited number of software listings, while THE BLUE BOOK offers the most readable, non-evaluative directory to a far greater number of packages.

APPLE BACKPACK

Humanized Programming in BASIC

Scott Kamins and Mitchell Waite
Level: Intermediate Rating: **95**

McGraw-Hill 1982 Paper
181 Pages ISBN: 0-07-033356-4 8" x 10" \$14.95

While most books on coding style relate to refining the internal structure of BASIC programs, APPLE BACKPACK addresses an even more important aspect of programming—the interaction of user and microcomputer. This external interface, overlooked or abused by some very elegant, highly structured BASIC programs, is the subject of Kamins and Waite's own very elegant, effective book.

An elementary knowledge of BASIC is assumed. Each chapter proceeds by providing a sample program, analyzing its degree of user-

friendliness (usually brutally low), and modifying the program until it becomes "pro-human." The final version is included at the chapter's end with a brief summary of the specific techniques it demonstrates.

Five chapters each discuss a different facet of programming ergonomics; their self-explanatory titles are: "Screen Formatting for Clarity," "Crashproofing Programs," "Verification and Validation," "Directions on the Display" and "Hard-Copy Documentation." The hidden gems of this book are those "final version" programs listed at the end of each chapter: they contain some very useful general subroutines for validating user-entered data, and verifying correct data against program criteria. Routines for checking the entry of a telephone number, for example, can easily be incorporated into any user-interactive program which prompts for entry of data into a possibly error-laden field. While short elementary BASIC examples are included with each chapter, two appendices contain longer, more complex efforts which demonstrate the techniques espoused by the authors.

In one final note, the book itself is an excellent example of its own humanizing thesis. The production quality of typeface, graphics and even paper is uniformly high. Illustrations, half-tones and screened boxes are effectively used to highlight key concepts.

APPLE BACKPACK, like CP/M or the S-100 bus, should become the de facto industry standard for the design of user-friendly software. We highly recommend its purchase to any programmer who intends to have another human interact with the results of his coding.

APPLE BASIC: DATA FILE PROGRAMMING A Self-Teaching Guide

LeRoy Finkel and Jerald R. Brown

Level: Intermediate Rating: **95**

	Wiley	1982	Paper
303 Pages	ISBN: 0-471-0957-X	7" x 10"	\$12.95

Many APPLE users, having learned that cassette recorders were simply too slow and inefficient, instead bought floppy disk drives, which largely remained idle or under-utilized while their owners tried to figure out exactly what to do with them. For everyone in this unenviable position, Finkel and Brown's APPLE BASIC: DATA FILE PROGRAMMING offers an effective solution.

The book's treatment of the manipulation of sequential and random access data files is an extremely competent one. But more importantly, the authors focus on avoiding two correlaries of Murphy's Law for

programmers: whatever can go wrong will go wrong if left to chance; and the likelihood of such catastrophic errors increases geometrically when disk drives are used. Consequently, the authors emphasize adequate documentation with REM statements, easy to read pretty-printed programs and correct and consistent programming techniques (such as not exiting from a loop before its natural termination). In addition, this guide to data file programming stresses user friendliness in order to insure the integrity of the data entered and stored on disk. This means not only that program-supplied prompts should specify the exact kind and format of the data to be input, but also that the program itself should subject all input data to a series of tests in order to minimize the possibility of erroneous data creeping into a data file.

The text of APPLE BASIC is informative and lively at the same time that it avoids the obvious condescension which characterizes some of the other books in the Wiley series. It is, in short, a volume which will enable the Apple user to make the most of his or her disk drives.

THE APPLE CONNECTION

Techniques and Principles of Apple Computer Interfacing

James W. Coffron

Level: Novice Rating: **85**

Sybex 1982 Paper
263 Pages ISBN: 0-89588-085-7 7" x 9" \$14.95

THE APPLE CONNECTION is one of the few layman's introductions to interfacing techniques: the software and hardware required for microcomputer control of external devices is the primary subject of Coffron's text. Technical jargon is explained, simplified circuit diagrams are used, and real applications with which any reader can "connect" are employed in a treatment which succeeds in demystifying a complex topic.

After a lucid, conversational introduction to interfacing principles, Coffron uses a special Creative Microprocessor System board to illustrate the manipulation of input/output signals through PEEKS and POKES (the CMS board uses LED displays for output and switches for input). "Handshaking" is briefly discussed (Coffron deliberately avoids the related subjects of interrupts, direct memory accesses and standard interface busses in the interests of keeping the presentation simple). The author then progresses to a consideration of the actual hardware required to send and receive data from an external device. Schematics supple-

ment this treatment. A complete hardware and software system for a home security application combines these early experiments into a functional example of Apple interfacing.

The Apple's ability to control home appliances through an intermediary solid state relay is explored in the chapters which follow. The book concludes with examples of analog-to-digital conversions (an Analog Devices AD 570 is used), and digital-to-analog conversions (AD 558).

THE APPLE CONNECTION is an experimenter's sourcebook of interfacing projects, combined with clear explanations of principles and techniques. We found Coffron's text accessible to the novice with a minimal electronics background, who is willing to invest in the peripherals required by the author's examples.

APPLE FORTRAN

Brian D. Blackwood and George H. Blackwood

Level: Novice Rating: 45

Sams 1982 Paper
236 Pages ISBN: 0-672-21911-5 5" x 8" \$14.95

APPLE FORTRAN, according to the book's introduction, is written for the beginning Apple programmer who wishes to learn Apple's version of FORTRAN 77. Aside from covering the basic statements of the FORTRAN language, the Blackwoods have included chapters on the operating system (the Apple Pascal Operating System is used in conjunction with FORTRAN) and on using FORTRAN with multi-disk drives.

The book can by no means be regarded as an introduction to Apple's FORTRAN language for a beginning programmer, since it is completely lacking in any conception of instructional technique. Rather than presenting and thoroughly explaining new material, principles and statements as they go along, the Blackwoods begin by using virtually all available FORTRAN statements, which they explain only gradually in the course of the first twelve chapters. Nor do they attempt to elucidate for the beginning reader the principles underlying FORTRAN statements so that the reader might be able to apply them independently in FORTRAN programming. An array (which is used for the first time in chapter two), for example, is only explained to the reader in chapter six. Even then, it is defined as a "a collection of many data items of the same kind. An array is an arrangement of data items identified by a variable that has a subscript. The array is constructed in such a manner that the program can examine the specific elements in the list."(107) However unhelpful this definition may be, it is in any case doubtful that the beginning reader

will ever reach it, since the Blackwoods' practice of bombarding the novice with all and sundry FORTRAN statements will quickly cause the reader to do something sensible—like throw the book away.

But if Apple FORTRAN is not really for the beginning programmer, then it is unclear exactly for whom it has been written. However badly they fail, the Blackwoods go into just enough detail in teaching FORTRAN that the the intermediate or advanced programmer would not be interested in their work. The book, in addition, suffers from a lack of appendices or a synopsis of FORTRAN commands, making it virtually useless as a work of reference.

APPLE GRAPHICS & ARCADE GAME DESIGN

Jeffrey Stanton

Level: **Advanced** Rating: **90**

Book Co.	1982	Paper	
288 Pages	ISBN: (None)	6" x 9"	\$19.95

The programmer fluent in BASIC and assembly language who consistently falls short of accurately reproducing arcade-quality graphics on his or her Apple will marvel at Stanton's text. It rates as a minor masterwork on advanced Apple graphics techniques.

In a departure from the approach used in graphics primers, the author initially treats high resolution graphics. Three distinct coding approaches are presented: "flipping screens," that is, substituting a pre-drawn screen at animation speeds for the screen being viewed by the game-player; drawing and redrawing figures retrieved from shape table directories; and animating with machine language character generators. This section includes the BASIC program listing for a "Shape Creator" which facilitates the storage and retrieval of programmer-designed patterns.

Low resolution graphics, when used in a BASIC program, is usually the simplest form of Apple graphics. Stanton's twist is to manipulate its functions using assembly language code exclusively. A cursory, admittedly inadequate introduction to assembly language graphics programming precedes a demonstration of this technique in a low resolution version of "Breakout." In a similar vein, subsequent discussions focus on directly accessing high resolution graphics routines located in the AppleSoft ROM by assembly language calls.

Raster graphics—that is, the use of bit-mapped shape tables in microcomputer game animations—forms the main programming technique employed in the second half of APPLE GRAPHICS.

High-speed drawing routines are incorporated into treatments of three different classes of arcade games: the Invaders-type game where a gunner defends against an attacking enemy; the fully mobile, marauding spaceship-type game; and games that simulate horizontal and vertical motion by scrolling a background. Each is explored in considerable detail, with annotated program listings, flowcharts and special treatments of paddle and joystick input. All of the programs and a bit-mapping utility mentioned by Stanton are available on diskette.

APPLE GRAPHICS AND ARCADE GAME DESIGN is a turbo-charged tour of coding techniques for the advanced graphics programmer.

APPLE II ASSEMBLY LANGUAGE

Marvin L. De Jong

Level: Intermediate Rating: **75**

	Sams	1982	Paper	
334 Pages	ISBN: 0-672-21894-1		5" x 9"	\$15.95

In view of the profusion of excellent works on assembly language programming for the 6502 microprocessor in general and the Apple II in particular, it is unfortunate that De Jong did not delineate both his audience and his topic more carefully. APPLE II ASSEMBLY LANGUAGE is designed to appeal to assembly language programmers of all persuasions and levels of expertise. Similarly, it encompasses the entire gamut of programming topics, from two's complement arithmetic to interfacing.

The result is a competent work which is, however, characterized by a certain sketchiness of coverage and lack of adequate detail. With the exception of interfacing, De Jong's prime area of interest, topics are treated either chaotically or altogether insufficiently—or are not addressed at all, as in the case of floating point operations.

Consequently, we feel that the needs of the newcomer to assembly language programming can be met more adequately by reading a good introductory work, such as APPLE MACHINE LANGUAGE or USING 6502 ASSEMBLY LANGUAGE. Both novices and more experienced programmers interested in a general presentation of 6502 assembly language topics would do well to read PROGRAMMING THE 6502. We do, however, recommend APPLE II ASSEMBLY LANGUAGE to those hobbyists interested in learning assembly language to control a variety of peripherals which they might connect to their Apple.

APPLE II ASSEMBLY LANGUAGE EXERCISES

Leo J. Scanlon

Level: Novice Rating: **45**

Wiley 1982 Paper
205 Pages ISBN: 0-471-86598-2 9" x 11" \$9.95

The "easy-to-read," non challenging text which Scanlon has considerably provided succeeds only in withholding those very details which form the basis of successful assembly language programming. By opting for a simplistic treatment at the expense of a detailed, informative and well-written one, APPLE II ASSEMBLY LANGUAGE EXERCISES provides a woefully inadequate introduction to assembly language programming.

It is paradoxical that Scanlon's discussion of assembly language programming appears to assume the novice's familiarity with its underlying principles. While most authors of works on assembly language programming make some attempt to explain what a bit or byte is, then progress to the storage of data in binary form, and finally conclude with a discussion of mathematical operations (such as binary-hexadecimal and decimal-hexadecimal conversion or two's complement arithmetic), Scanlon chooses to ignore this essential background information altogether. The reader who is confused by Scanlon's jargon, or who overlooks Scanlon's initial mention of a particular term or concept, will find no help within the book, since Scanlon has not chosen to include either a glossary or an index.

If the book's treatment of the prerequisites of assembly programming is inadequate, so too is its discussion of assembly language programming itself. Scanlon typically notes the existence of instructions without examining how they are used, or presents them within a very limited and highly specific context.

The author relies on the reader's "hands-on" practice to bridge the numerous deficiencies of his book. This assumes, however, that those who do not know how to write assembly language programs can learn to do so by exercising their lack of knowledge, without any help from the author. Rather than suffering from the frustrations resulting from this idiotic approach, we recommend that those who want to learn programming for the 6502 instead read one of the highly-rated works on this subject.

APPLE II BASIC QUICK REFERENCE GUIDE

Gilbert Held

Level: All Rating: **85**

	Wiley	1982	Paper	
8 Pages	ISBN: 0-471-87039-0		6" x 12"	\$2.95

See our review of IBM PC BASIC QUICK REFERENCE GUIDE on page 276.

APPLE II PROGRAMMER'S HANDBOOK

Richard C. Vile, Jr.

Level: Intermediate Rating: **90**

	Spectrum	1982	Paper	
275 Pages	ISBN: 0-13-039198-0		6" x 11"	\$16.95

In a literature filled with disappointingly superficial treatments of popular programming topics, Richard Vile's book provides refined, rich insights into the coding capabilities of an Apple II. It assumes more than a rudimentary knowledge of three separate disciplines: BASIC (both Integer and AppleSoft), Pascal, and 6502 assembly language. The author takes running starts into each of these subject areas, with his early admonitions to beginners their only solace.

Programmers will be able to find more than a few original techniques revealed in Dr. Vile's text: from simple BASIC routines that capitalize on 'trapdoors' in DOS, to documentation time-savers which increase both coding and maintenance productivity (one suggests the substitution of array names for subroutine line numbers). Of particular interest to the 6502 programmer are the ampersand support routines which embellish the BASIC/DOS output hook mentioned earlier. The Pascal chapters are less accessible than the the BASIC or 6502 sections, with the reader being expected to infer BASIC/Pascal differences from an analysis of the same program written in each of these languages.

APPLE II PROGRAMMER'S HANDBOOK should be sold with a tamper-proof seal; it is a cogent, advanced treatise on some of the more esoteric facets of Apple II programming couched in a presentation which the novice programmer may very well find indigestible.

APPLE II USER'S GUIDE

Lon Poole, Martin McNiff and Stephen Cook

Level: Novice Rating: **95**

	Osborne	1981	Paper	
385 Pages	ISBN: 0-931988-46-2		7" x 9"	\$15.00

At the time of this writing, this book represents the definitive reference manual to Apple II operation and programming use. It's scope ranges from initial set-up to intrinsic machine language subroutines.

The operational chapters are supplemented by highly effective photographs of the circuitry "core" (it's difficult to resist) of the Apple II. Shots of actual monitor prompts are used to walk the reader through a successful boot of DOS. BASIC is explained in detail; most of this book is, in fact, a programming manual and reference guide. The distinctions between Integer and Applesoft BASIC parallel the general development of command syntax. While an introduction to assembly language is relegated to the reader's purchase of a few recommended manuals (the only Apple-related topic not covered), this does not preclude a fairly complex discussion of advanced programming subjects. Shape table construction is explained in a high-resolution graphics chapter. Mini-assembler manipulation of the machine-language monitor, as well as BASIC program interfacing, are additional topics presented in the calm, precise prose characteristic of the entire book. The appendices more closely resemble supplementary chapters which treat technical topics in even greater depth; for example, besides a compendium of BASIC statements and functions, these appendices include text-like discussions of memory architecture and disk formatting.

APPLE II USER'S GUIDE is a state-of-the-art treatment of how to use and "push" an Apple II. Besides the information a line-by-line reading imparts, it is a valuable reference manual for the BASIC and assembly language programmer.

APPLE II WORD PROCESSING

Carol Poling

Level: All Rating: **95**

	Que	1981	Paper	
250 Pages	ISBN: 0-88022-005-8		8" x 10"	\$19.95

This volume in Que's Business Software Evaluations Series provides an in-depth examination of word processing with the Apple. It is not a

"how to" book, but rather attempts to provide the potential purchaser of a word processing package for the Apple with the detailed information necessary to make an intelligent and informed selection.

While the book's major focus is on word processing software, it does examine the suitability of Apple hardware for word processing applications. Aside from examining the possibility of expanding memory to a full 64k (with a memory expansion card) or of increasing the size of mass storage (by using more than two mini-floppy disk drives, or substituting either 8-inch or hard disk drives for them), APPLE II WORD PROCESSING stresses two major drawbacks of the Apple as a word processor: its 40-column per line video display, and its upper-case alphabet. To overcome these, Poling notes the availability of 80 column video boards and/or a variety of methods to modify the shift key. A third obvious drawback—the fact that the central processing unit and the keyboard form a single component—is not discussed.

The heart of the book, however, presents Que's evaluation of nine word processing programs for the Apple (Apple Writer, Easy Writer Professional, Magic Window, Pie Format, Select, Spellbinder, Super Text II, VTS-80 and WordStar), as well as of two spelling correction packages (Micro Spell and Spellguard). Integrated word-processing spelling packages were just beginning to appear as the book went to press.

Que's evaluation of each of these programs is notable for its critical content, its level of detail and the ease with which it permits the reader to compare the various programs. The review of each package contains information on estimated prices, sales volumes and any additional equipment required by the programs. A brief text assesses the program's strengths and weaknesses. The major portion of each evaluation is devoted to feature-by-feature charts listing the presence or absence of particular options and their importance in various word processing applications, and the overall effectiveness with which these features have been integrated into the package as a whole. The software publisher's (sometimes irate) response to Que's evaluation is included.

Although a number of word processing and especially spelling correction programs were not surveyed in APPLE II WORD PROCESSING, presumably because they were released after the book went to press, Que's evaluation of Apple word processing software surpasses all other software guides in providing the reader with detailed information and specific criteria needed to select the right program. Its relatively high purchase price is more than justified by the care with which it is assembled.

1983 APPLE II/III SOFTWARE DIRECTORY - Volume II

VanLoves

Level: All Rating: **80**

Advanced Software Techn.	1982	Paper
950 Pages	ISBN: 0-94-152002-1	6" x 9" \$24.95

See our review of THE ADDISON-WESLEY BOOK OF APPLE COMPUTER SOFTWARE 1983, on page 238.

APPLE INTERFACING

Jonathan Titus, David Larson, Christopher Titus

Level: Intermediate Rating: **70**

Sams	1981	Paper
206 Pages	ISBN: 0-672-21862-3	5" x 9" \$10.95

This offering in the Blacksburg Continuing Education Series is intended "for an audience that has a wide range of backgrounds, from the beginner to the advanced user," and treats the use of additional peripherals under the the control of the user's BASIC program. But in fact the authors' assumptions about their audience are much more specific than they indicate, and effectively limit this book's interest to a comparatively small circle of microcomputer users. Having dismissed assembly language and binary arithmetic as too complex for the average reader in their first two chapters, the authors suddenly assume the reader's familiarity with the principles of gate logic when they come to their discussion of input/ output interfacing. From this point on, no attempt is made to use interfacing as a means of increasing the novice's understanding of computer technology.

The heart of the book consists of a series of experiments in which a variety of peripheral devices, such as a temperature sensor or an analog to digital converter, are interfaced to the Apple and placed under user control. But because of the book's lack of a detailed introductory discussion, these experiments can be undertaken only at the risk of the Apple user who lacks a substantial degree of technical sophistication.

APPLE LOGO

Harold Abelson

Level: Novice Rating: **60**

Byte 1982 Paper
 224 Pages ISBN: 0-07-000425-0 8 x 10" \$14.95

The programming language Logo has become famous for its turtle which, as it blunders across the video screen, leaves graphics in its wake. Logo also reflects a comprehensive philosophy of education which, through its use of computers, aims at developing the intellectual and creative skills of students. In view of the limited utility of Logo as a programming language, a discussion of this second point—its utility as an educational tool—would seem to be an important priority. It is one, however, which the author has carefully avoided.

The most widespread use of Logo has probably occurred in schools. In light of this, it is curious that APPLE LOGO is a book which is decidedly not aimed at teaching children or teenagers how to program with Logo. Stylistically, the book is written at a level which many young people will not be able to understand; and the text, aside from its inclusion of numerous examples of turtle graphics, does not harmonize with the needs or the interests of the young. It is a work written for adults.

Even assuming that the reader wants to learn Logo, it is unclear how effective APPLE LOGO will be as an instructional tool. The more complex concepts of the language, such as recursion, are discussed both far too early (pp.32-43) and insufficiently. In the absence of a discussion of modern programming concepts, many of the features of Logo, and especially its encouragement of structured programming and modular program development, are likely to seem incomprehensible. Finally, a brief text is literally crammed full of information, which does little to encourage the reader's retention of the material covered.

APPLE MACHINE LANGUAGE

Don Inman and Kurt Inman

Level: Intermediate Rating: **90**

Reston 1981 Paper
 296 Pages ISBN: 0-8359-0230-7 6" x 9" \$14.95

APPLE MACHINE LANGUAGE is a non-intimidating introduction to machine and assembly language programming for the relative novice to microcomputers. The authors presuppose some working knowledge of BASIC, although they do review the statements which are most important to the beginning machine language programmer in the book's early

chapters. While the explanations which the book offers on complex topics (such as binary subtraction) are frequently either simplistic or incomplete, this should not obscure the book's very substantial merits; APPLE MACHINE LANGUAGE introduces new material in a gradual yet very well-organized manner, and offers the machine language practitioner hands-on experience in programming as he or she learns the 6502 instruction set. At the same time, the Inmans effectively use operating system subroutines as a means of teaching the user about the complexities of machine language programming.

The book uses three graduated methods of entering machine language programs. The first, which the authors term "the BASIC Operating System," is a BASIC program which uses POKE statements to load machine language codes into designated memory locations, and a CALL statement to execute the machine language program. Later, when the authors feel that they have demystified machine language programming while giving the reader a firm understanding of its basic principles, they introduce the Apple System Monitor, which permits the programmer to enter machine code directly. Finally, the authors introduce the mini-assembler available on Apple II (but not on Apple II Plus) systems, which allows the reader to use assembly language mnemonics rather than the more cumbersome and error-prone machine language instructions.

APPLE PASCAL GAMES

Douglas Hergert and Joseph T. Kalash

Level: Novice Rating: **90**

Sybex 1981 Paper
371 Pages ISBN: 0-89588-074-1 7" x 9" \$14.95

APPLE PASCAL GAMES is a welcome exception to the general tendency of authors to overlook using their programs as a means of improving the user's programming skills as a whole. The volume is above all distinguished by the fact that the authors explicitly focus on those Pascal statements around which they have constructed their games. "Horserace," for example, illustrates the use of Pascal records, while "Wumpus" and "Keno" make use of sets. In this way, the user, by examining the program and reading the authors' explanation, can focus on those Pascal concepts which he or she does not fully understand. The format of each section, moreover, is fully consistent with this emphasis on game playing as a means of education: after presenting the rules of the game and showing a sample run, Hergert and Kalash discuss the program in general and its notable features in particular; this is followed by the invariably well-documented Pascal program itself.

The quality of the book's 27 games is variable. The nine games in the first chapter ("Simple Games") are largely dreary affairs, some of which are not even games at all. "Twinkle" for example, merely plots asterisks randomly on the screen, while "Block" features the computer's attempt to write characters input from the keyboard. While a few of the fifteen "MORE ADVANCED GAMES" are notable for their uniqueness, complexity or imagination (our favorites are "Horserace," "Kismet," and "Wumpus"), several are completely uninspiring; "Numconvert," for example, allows the reader to do binary-octal-decimal conversions, and "Pascal" simply prints a Pascal triangle. Of the two games in the "Turtlegraphics" chapter, one is an enjoyable hunting game ("Gunshot"), while "Picture" merely draws pictures on the screen and requires a game paddle. The book's final program, a computerized version of cribbage, however, more than atones for these previous failures.

All of these games were written in UCSD Pascal for an Apple, although they should run with no trouble on other UCSD Pascal compilers. We highly recommend APPLE PASCAL GAMES especially to those users who would like to play games and at the same time improve their ability to program in Pascal.

APPLESOFT LANGUAGE

Brian D. Blackwood and George H. Blackwood

Level: Novice Rating: **60**

	Sams	1981	Paper	
253 Pages	ISBN: 0-672-21811-9		6" x 9"	\$10.95

In their third attempt, the Blackwoods have finally produced a manuscript which, whatever its limitations, at least deserves more than to be immediately relegated to remainder bins. On the one hand, in contrast to INTIMATE INSTRUCTIONS IN INTEGER BASIC, the text is long enough to avoid a highly condensed and therefore superficial approach to BASIC; on the other, in contrast to APPLE FORTRAN, the Blackwoods have recognized the need to develop an instructional technique in order to meet the book's stated objective, which is to teach the beginning Apple user the rudiments of BASIC. The improved quality of APPLESOFT LANGUAGE appears to result, at least in part, from the growing programming proficiency of George H. Blackwood, whose preface and introduction to the text no longer convey his despair of ever learning how to use the Apple.

Aside from treating the basics of AppleSoft BASIC, the range of topics covered in the course of APPLESOFT LANGUAGE is fairly broad. Throughout the text, and especially in the second section, the Blackwoods emphasize good programming techniques such as flowcharting,

the development of flexible programs which are not tailored to a very limited set of data, the use of menus and the preparation of effective, well-organized output. The two chapters of the third section contain an introduction to graphics on the Apple.

The text is, however, replete with short, choppy sentences, many of which are unrelated to one another. Nor is the organization and presentation of material within a chapter always logical or coherent; in the middle of their discussion of arithmetic precedence, for instance, they suddenly include the BASIC ASC and CHR\$ functions, only later to resume their discussion of precedence. A good deal of material is also missing from the book; the treatment of string variables and string arrays is inadequate, the possibility of accessing data external to a BASIC program is never discussed, and Apple DOS—the Apple disk operating system—is never mentioned, much less related to AppleSoft. Finally, the book lacks an index or appendices containing summaries of BASIC commands, so that its utility as a work of reference, once the beginning programmer has read it and mastered the rudiments of AppleSoft BASIC, is sharply diminished.

While the Blackwoods' ongoing struggle to improve the quality of their books is commendable, the progress they have made to date is far from sufficient to distinguish APPLESOFT LANGUAGE from other far superior treatments of BASIC for the Apple II.

ASSEMBLY LANGUAGE PROGRAMMING FOR THE APPLE II

Robert Mottola

Level: Novice Rating: **50**

	Osborne	1982	Paper	
143 Pages	ISBN: 0-931988-51-9		7" x 9"	\$12.95

Like Randy Hyde's USING 6502 ASSEMBLY LANGUAGE (see our review), this volume attempts to teach the novice the basics of 6502 programming with the Apple's LISA assembler. But unlike Hyde's book, ASSEMBLY LANGUAGE PROGRAMMING FOR THE APPLE II provides only a superficial and perfunctory treatment of its topic.

According to the author, "all computer languages are alike.. Therefore, since you already know how to program in BASIC, you should find it easy to learn assembly language."(ix) While this approach may be useful in introducing the BASIC programmer to Pascal, for instance, it seems incongruous when a high-level language is used to lay the groundwork for proficiency in a low-level one. But even assuming that such a technique can succeed if handled properly, it is abundantly clear that

Mottola has not done so. His scattered references to BASIC serve no other purpose than to avoid providing the reader with that detailed knowledge necessary to program in assembly language.

An examination of the lines of code presented in the book indicates that the author uses assembly language programming for altogether minimal purposes: subroutines written in assembly language serve as intermediaries between BASIC programs and Apple ROM subroutines; the assembly language programs themselves are responsible only for passing a few parameters and calling the ROM subroutines. Given this limited objective, it is not surprising that Mottola has devoted one-third of his book to explaining how to use the LISA assembler (although even this has been inadequately explained). It accounts as well for his lack of attention to 6502 architecture and his cursory survey of the 6502 instruction set which, while it does mention most of the 6502's op codes, completely fails to note how they are used in assembly language programming.

The sole worthwhile feature of this book is its first appendix, which lists instructions not covered in the text. We suggest that its title be changed to "Topics Not Covered in This Book," and that the whole of the book's text then be moved into this appendix.

BAG OF TRICKS (Diskette Included)

Level: Novice Rating: **85**

Quality Software	1982	Paper
150 Pages	ISBN: (None)	6" x 9" \$39.95

See our review of BENEATH APPLE DOS on page 257.

BASIC APPLE BASIC

James S. Coan

Level: Novice Rating: **95**

Hayden	1982	Paper
237 Pages	ISBN: 0-8104-5626-5	7" x 10" \$12.95

The author of BASIC BASIC has surfaced with this encore presentation of Apple Integer and AppleSoft BASIC. In many ways, the structure and content of the introductory chapters are updates of original BASIC BASIC material: chapter two of both books, for example, discusses pro-

gram planning, REM and IF-THEN statements. Even some of the problems bear a striking similarity (the Euclidean algorithm for finding the greatest common factor is another source of *deja-vu*). Yet, despite any resemblance to his past work, James Coan has crafted a state-of-the-art guide to BASIC programming on an Apple II.

This is especially evident in the book's focus on graphics: not only are simple low-resolution graphics covered relatively early in the text (chapter three), but the high-resolution chapter is thoughtfully presented, with clear illustrations and sample programs.

The introduction to programming is closely bound to explanations of the Apple's hardware: key functions, as well as a chapter on DOS, enhance the reader's understanding of the total system. In addition, appendices include hints on initial set-up and saving and retrieving programs.

BASIC APPLE BASIC is much more than an introduction to BASIC; written by one of the more talented BASIC educators, it combines a superior programming treatment with valuable information about the Apple's hardware/software interface.

BASIC FOR THE APPLE II

A Self Teaching Guide

Jerald R. Brown, LeRoy Finkel and Bob Albrecht
Level: Novice Rating: **85**

Wiley 1982 Paper
410 Pages ISBN: 0-471-86596-6 7" x 10" \$12.95

Most Wiley Self-Teaching Guides are characterized by their consistent approach to different microcomputer topics. The technique employed in these texts borrows heavily from the strict frame-by-frame programmed-instruction method (see our review of CRASH COURSE IN MICROCOMPUTERS), but adds its own distinctive embellishments in the form of computer-assisted exercises, illustrations and analogies.

It is the active use of the computer which makes this presentation most enlightening. The progression from simple PRINT statements to string variables is reinforced by entering code into the microcomputer and noting the results. Of special interest to the reader eager to manipulate the graphics capabilities of his monitor is the chapter on low-resolution graphics (the authors leave the discussion of high-resolution graphics to a more advanced text).

BASIC FOR THE APPLE II lays the foundation for mastery of the language's nuances. Its progress-at-your-own-pace technique allows the presentation to be simple without the condescension apparent in similar

treatments. As such, this Wiley guide ranks as one of the better introductions to the BASIC programming language.

BASIC FOR THE APPLE II PROGRAMMING AND APPLICATIONS

Larry Joel Goldstein and Martin Goldstein

Level: Novice Rating: 70

Robert J. Brady 1982 Paper
250 Pages ISBN: 0-89303-189-5 7" x 10" \$14.95

The title BASIC FOR THE APPLE II is a misnomer; although the authors do attempt to teach the computer novice the essentials of programming in AppleSoft BASIC, at the same time they include discussions of such topics as the Apple II keyboard, the operation of the computer, the Apple editor and Apple DOS. The book provides, in short, a fairly complete, if sketchy, introduction to the Apple II for the uninitiated user. The book is in numerous respects identical to the authors' other work, THE I.B.M. PERSONAL COMPUTER: AN INTRODUCTION TO PROGRAMMING AND APPLICATIONS.

Needless to say, BASIC FOR THE APPLE II therefore suffers from virtually all of the limitations which are present in the Goldsteins' introduction to the IBM PERSONAL COMPUTER. The text is filled with numerous exercises, answers and sections to "test your understanding," which correspondingly limits the amount of actual text devoted to discussing the features of the Apple II. Many of these questions, moreover, are not especially useful. As a result, some of the chapters contain inadequate discussions of their subject matter; this is especially true of chapter five on Apple DOS and of chapter six on computer graphics.

The strong points of these two books are also largely identical. BASIC FOR THE APPLE II does provide a fairly comprehensive and intelligible overview for the new and potentially intimidated Apple II user. At the same time, the authors make it clear that the uses of the computer are limited only by the user's imagination. They themselves suggest a wide range of software applications such as word processing, computer games and simulations, graphics and VisiCalc.

BENEATH APPLE DOS

Don Worth and Pieter Lechner

Level: Advanced Rating: 90

Quality Software 1981 Paper
169 Pages ISBN: (None) 6" x 9" \$19.95

DOS Version 3.3 is the primary focus of this detailed analysis of the Apple operating system. The advanced programmer will find unique treatments of the evolution and structure of DOS, diskette formatting and organization, DOS program logic, and assembly language interfacing.

The diskette chapters are, in many ways, the Apple II counterparts of Pennington's TRS-80 DISK AND OTHER MYSTERIES (see our review). The role of self-sync bytes in synchronizing hardware to diskette data is explored. Data field encoding and sector-skewing techniques employed by the Apple are discussed as preliminaries to the analysis of the diskette directory. The dissection of the diskette Volume Table of Contents (VTOC) includes a byte-by-byte analysis of the meaning of each VTOC entry. Armed with this elementary knowledge of Apple II diskettes, the reader can use a diskette utility listed in one of the appendices to recover sectors of a damaged diskette.

Of special interest to DOS aficionados are the chapters which treat the evolution, structure, and customizing of the Apple assembly language operating system. The step-by-step explanation of what actually occurs during the boot process is fascinating (bootstrapping is revealed as a complex four-stage procedure). The assembly language programmer will find techniques which utilize DOS subroutines to access the disk directly. DOS modification through memory or diskette patches is explained, but not before the reader is forewarned that oversights "may result in an unreliable system." The remainder of the text consists of an analysis of each DOS assembly language subroutine.

Appendices include listings of five utility programs which can be used to scan and/or modify diskette data: DUMP is a track dump utility; ZAP is a diskette update utility; INIT reformats a single diskette track; FTS scans a damaged diskette for TRACK/SECTOR lists; and COPY converts diskette file-types.

BAG OF TRICKS, an extension of the programs listed in these appendices, is a software package consisting of four machine language subroutines available from the authors of BENEATH APPLE DOS. With documentation presented in a similar spiral-bound format, these four programs dump and examine raw diskette data, reformat individual tracks while attempting to preserve their data, update diskette data, and automate the process of recovering a damaged VTOC. Though it is, by definition, more "software" than "bookware," we have listed BAG OF TRICKS as an expansion of the assembly language listings of BENEATH APPLE DOS's diskette utilities.

It is BENEATH APPLE DOS, however, which provides the advanced Apple programmer with an invaluable analysis of both DOS and DOS-formatted diskettes. Even if its utilities are never used, the book's explanation of the intricacies of the Apple operating system is necessary for any full appreciation of DOS.

THE BLUE BOOK FOR THE APPLE COMPUTER

For The Apple Computer

WIDL Video

Level: Novice Rating: **80**

	Widl Video	1982	Paper	
240 Pages	ISBN: 0-684-17793-5		9" x 11"	\$24.95

See our review of THE ADDISON-WESLEY BOOK OF APPLE COMPUTER SOFTWARE 1983, on page 238.

CIRCUIT DESIGN PROGRAMS FOR THE APPLE II

Howard M. Berlin

Level: Advanced Rating: **95**

	Sams	1982	Paper	
129 Pages	ISBN: 0-672-21863-1		8" x 12"	\$15.95

See our review of CIRCUIT DESIGN PROGRAMS FOR THE TRS-80 on page 287.

COMPUTER PROGRAMMING FOR KIDS AND OTHER BEGINNERS

Apple II Edition

Royal Van Horn

Level: Novice Rating: **80**

Sterling Swift 1982 Paper
144 Pages ISBN: 0-88408-151-6 11" x 8" \$9.95

Most authors begin elementary BASIC texts by introducing the reader to the PRINT statement. Typically, the user is asked to enter several lines of code, and run the program, which merely prompts the user for his or her name and then prints it after "hello." This extremely uninspiring inauguration to BASIC instruction pervades even the literature intended primarily for adults. Van Horn, however, wisely avoids this approach, since she discovered that children merely find it laborious, boring and uninspiring.

Instead, after a very brief introduction to the components of a micro-computer system, Van Horn begins her BASIC instruction with elementary graphics statements (COLOR PLOT, and later HLINE and VLINE). The second half of the book then introduces the more standard BASIC statements including PRINT, GOTO, FOR/NEXT, REM, RND, INPUT, READ, DATA, IF/THEN and assignment statements. While it seems that Van Horn's approach to teaching BASIC does engage the attention and imagination of the child, an emphasis on computer programming as a purposeful activity is conspicuously absent from the text. Aside from showing children that computers can do "some super neat stuff,"(74) it is not apparent that children will find computers and programming to be immediately practical or relevant.

Although we are somewhat suspicious about who the "other beginners" of the title may be, COMPUTER PROGRAMMING FOR KIDS AND OTHER BEGINNERS is a valuable and engaging introduction to BASIC programming on the Apple for children at roughly the second through fourth grade levels.

COMPUTERS FOR KIDS-APPLE EDITION

Sally Greenwood Larsen

Level: Novice Rating: **90**

Creative Computing 1981 Paper
75 Pages ISBN: 0-916688-21-6 5" x 8" \$4.95

See our review of COMPUTERS FOR KIDS-ATARI EDITION on page 318.

THE CREATIVE APPLE

Mark Pelczarski and Joe Tate, Editors

Level: Intermediate Rating: **90**

Creative Computing 1982 Paper
448 Pages ISBN: 0-916688-25-9 8" x 11" \$15.95

The microcomputer market and personal computer magazine industry have engaged in a symbiotic relationship over the past decade: magazines have been used to popularize software and hardware developments whose purchase has, in turn, fostered the need for the timely, concise information provided only by magazines. Both partners have prospered.

The growth of CREATIVE COMPUTING, one of the few "pre-Apple" microcomputer magazines still on the newsstands today, is characteristic of this process. THE CREATIVE APPLE is an anthology of CREATIVE COMPUTING articles relating to the Apple II, published during the four year period from 1978 to 1982. Its treatments of various topics possess a consistently high quality of technical expertise and clear, reader-friendly presentation.

These topics are categorized into the following general subject areas: Graphics, Music, Education, Word Processing, Business, Apple Cart (an Apple potpourri), Software Reviews, Ready-to-run Programs, Programming Tips and Branches (Apple peripherals). Cartoons, code, and commentary all work to produce the distinctly magazine-like flavor of what is, ultimately, an Apple II sourcebook. Follow-up articles by the same author appear sequentially, so that a Chuck Carpenter series on 6502 Assembly Language Programming is elevated to the monograph status which it deserves.

THE CREATIVE APPLE is an enjoyable, informative collection of CREATIVE COMPUTING articles which will provide the Apple II user with a valuable addition to his or her microcomputer library.

THE ELEMENTARY APPLE

William B. Sanders

Level: Novice Rating: **85**

Datamost 1983 Paper
228 Pages ISBN: (None) 5" x 8" \$14.95

The contents of William B. Sanders' THE ELEMENTARY APPLE serve as a model for determining what topics should be treated in a general overview of a specific microcomputer written for the beginning user. Aside from the narrow aim of teaching the reader the rudiments of BASIC (which he performs competently, especially given the limited space which he has allotted himself for the task), Sanders extends his discussion to include many aspects of microcomputer use and programming which are usually totally ignored in introductory works. Entire chapters, for example, are devoted to the use of the printer and the manipulation of disk files (including EXEC files) as well as to low and high resolution graphics; portions of other chapters discuss calls to monitor subroutines and encourage the reader to use the BASIC PEEK and POKE functions for specific purposes.

But while the quantity of material covered in THE ELEMENTARY APPLE is impressive, its quality frequently leaves a great deal to be desired. Instead of providing detailed explanations in many of his discussions, Sanders makes a few cryptic comments, presents a program or command, and then passes on to another topic. The book's "this is the way it is done" approach is perhaps best reflected in its presentation of calls to monitor subroutines. Rather than presenting a memory map (or even explaining what one is), Sanders contents himself with providing the addresses of eleven subroutines. Sanders recommends that the reader "make or buy or somehow get your hands on" a more complete list; one is available in the Apple Reference Manual, but its addresses are in hex, whereas BASIC requires that the addresses be entered in decimal. The author, however, feels that "it would probably be more confusing that enlightening to go into...the conversion process."(6-14) Nor does he care to remove the aura of magic surrounding such a seemingly incomprehensible command as CALL -936.

Despite these failings, Sanders has a clear idea of his audience. The reader of THE ELEMENTARY APPLE is an intelligent person who is eager to learn how to use the Apple effectively and who is willing to experiment with the machine in order to expand his or her computer skills. While Sanders may not always succeed in doing this directly, his book acquaints the reader with a broad array of topics and, by introducing material usually considered too "advanced" for the beginner, promises that the reader will eventually be able to expand his or her knowledge of programming very rapidly.

GRAPHICS COOKBOOK FOR THE APPLE

Nat Wadsworth

Level: Intermediate Rating: **75**

	Hayden	1980	Paper	
71 Pages	ISBN: 0-8104-6278-8		8" x 11"	\$9.95

The glassy-eyed microcomputer game player, guilty about all those wasted hours he's spent enthralled by phosphorescent sights and electronic sounds, has, as a sort of perverted penance, focused his blurry vision on books which demonstrate the creation of graphics programs. GRAPHICS COOKBOOK, through its explanation of low-resolution Apple-type graphics, attempts to nurture this new interest.

This is a 71-page cookbook, however, with a single recipe. Wadsworth uses one 32-line drawing routine to create all the patterns and images presented in this book. He accomplishes this simply by feeding the routine different sets of coordinates in data statements (to the very practiced eye, DATA 55,5,44,0,0,1,0,3,2,4,2,6,0,7,66 is actually a small seagull).

While there is a nice utility to this technique, and the data sets provided are numerous, there is something disquieting about a book based on 32 lines of Basic code; it's almost as if Julia Child presented one all-purpose casserole recipe, and proceeded to fill hundreds of subsequent pages with lists of ingredients that could be substituted to produce a different version of the original (a diet not unfamiliar to many American households).

GRAPHICS COOKBOOK is literally a paint-by-number guide to creating shapes on your Apple monitor. The wider variety of techniques outlined in more highly rated, similarly priced graphics texts provides ample reason for passing it up as an initial addition to your microcomputer library.

A GUIDE TO PROGRAMMING IN APPLESOFT

Bruce Presley

Level: Novice Rating: **90**

	Van Nostrand Reinhold	1982	Paper	
216 Pages	ISBN: 0-442-25890-9		9" x 11"	\$12.95

See our review of A GUIDE TO PROGRAMMING IBM PERSONAL COMPUTER on page 275.

HANDS-ON BASIC

For The Apple II

Herbert Peckham

Level: Novice Rating: **80**

McGraw-Hill 1983 Paper
 320 Pages ISBN: 0-07-049179-8 6" x 9" \$22.95

This book is an adaptation of BASIC: A HANDS-ON METHOD, a student text which introduced BASIC as it is used on a number of timesharing computers; HANDS-ON BASIC continues this tradition of a machine-intensive presentation of the BASIC programming language. The coding examples have been specifically tailored for an Apple II Plus equipped with DOS 3.3.

Peckham, in a BASIC BASIC backlash, has deliberately avoided all but the most elementary levels of mathematical skills in his text. The BASIC introduction is straightforward. "Frames" of coding exercises develop a facility with commands and syntax. Graphics functions, including simple animation techniques, enliven Peckham's presentation. Particularly well-done is a chapter on user-defined functions as illustrated by the DEF statement. The author's emphasis on simplicity unfortunately extends to a cursory treatment of file access methods.

The text's spiral binding facilitates its use as a microcomputer workbook, with answers to odd-numbered exercises provided in an appendix.

HANDS-ON BASIC is a popularized treatment of a non-mathematical classroom approach to introductory BASIC programming. While the novice will undoubtedly learn simple BASIC coding, many of the nuances of the language have been pruned from this spartan presentation.

HOW TO WRITE AN APPLE PROGRAM

Ed Faulk

Level: Novice Rating: **50**

DataMost 1982 Paper
 205 Pages ISBN: 0-8359-2990-6 5" x 8" \$14.95

See our review of HOW TO WRITE A TRS-80 PROGRAM on page 292.

I SPEAK BASIC TO MY APPLE

Student Text

Aubrey B. Jones, Jr.

Level: Novice Rating: **45**

	Hayden	1982	Paper	
234 Pages	ISBN: 0-8104-6175-7		9" x 11"	\$7.45

This book is actually the student text for a classroom set which includes a teacher's manual and examinations. We decided to review it on its own merits, however, because its publisher has marketed it to retailers as a stand-alone volume. What remains of that original classroom set is a distilled, flash-card-type introduction to BASIC programming. Single short sentences fill an entire page. Pages containing those short sentences are repeated for review. Diagrams devoid of explanation show the components of a specific microcomputer. This is a Rod Serling version of the BASIC one would learn from a nightmarish sequence of billboards posted along the SPEAK BASIC-BY-ENCINO highway.

Few BASIC commands are used, and though subroutines and string variables are covered, programs never aspire to anything but the most rudimentary coding level. Concepts and naming conventions are presented on one page and used in a "hands-on" example on the next. While an argument could be made that a reader with previous programming experience might actually enjoy a BASIC introduction stripped of any condescending, convoluted text, the superficial treatment contained in its sparse 234 pages barely imparts a hint of the complexities and nuances of the language. Any real learning associated with this text comes from following its recommendation to read the Applesoft Tutorial manual.

In short, the sale of this student text without its accompanying volumes on a retail level should be confined to those individuals, and those individuals only, who can produce a note from their mother.

KIDS & THE APPLE

Edward H. Carlson

Level: Novice Rating: **90**

	Datamost	1982	Paper	
218 Pages	ISBN: 0-8359-3669-4		8" x 11"	\$19.95

It is ironic that a book intended for "seventh graders" actually provides an introduction to BASIC programming more lucid than many similar texts intended for "mature" audiences. Whether this is a com-

ment on the high intelligence level of twelve-year-olds or the lack of general quality in extant adult BASIC primers is open to question. What is beyond doubt, however, is the inherent merit of Carlson's text.

Structured around thirty-three lessons in BASIC programming, *KIDS AND THE APPLE* provides the student with a humorous, precise, comprehensive study of how to program a microcomputer. "Instructor's notes" precede each lesson and note the topics about to be covered, as well as questions the narrative will equip the students to answer. The actual lesson follows. Cartoons by Paul Trap, a member of that special breed of humorist/illustrators who understand the unique problems that confront the beginning programmer, enrich Carlson's book: a rather ridiculous-looking robot, for example, plays bingo on a monitor-like grid to demonstrate the X and Y coordinate conventions used by the PLOT function. A sample program and practice problems end each lesson (an answer key for each problem forms one of the appendices).

This is not a treatment of "minimal" BASIC limited to twenty commands and ten-line programs; the full repertoire of BASIC functions is explored, including low and high resolution graphics, arrays, debugging techniques and "humanized" interactive programming. Appendices treat the operational considerations of disk usage and storing programs on tape. The book's spiral binding itself "humanizes" the process of microcomputer-assisted learning.

KIDS AND THE APPLE, written by a physicist/programmer/father (not necessarily the order of the author's preference) is a text noteworthy for its exceptional teaching technique and detailed treatment of introductory BASIC programming. So rare is this combination that we recommend it to beginning BASIC programmers of all ages.

MICROCOMPUTER SYSTEMS AND APPLE BASIC

James L. Poirot

Level: Novice Rating: **50**

	Sterling Swift	1980	Paper	
150 Pages	ISBN: 0-88408-136-2		6" x 9"	\$8.95

As the title of James Poirot's book indicates, it attempts to treat both the Apple as a microcomputer system and Apple BASIC as a programming language; in addition, the book, through its emphasis on flowcharting, also purports to examine "problem solving on the the computer, independent of the language."(iii) Poirot telescopes his treatment of these

three broadly-defined topics into a mere 118 pages of text; while this is clearly insufficient to teach the reader anything at all, it at least offers the advantage of not requiring the investment of a great deal of time.

After a simplistic discussion of microcomputer systems which Poirot fails to even remotely relate to the hardware-specific features of the Apple, the author moves on to flowcharting, which for him is synonymous with sound programming technique. Poirot himself could have learned a good deal from those adherents of top-down, structured programming who reject flowcharting because it tends to obscure the earlier and more critical stages of program design; while some programmers immediately begin to write lines of code, others immediately begin to draft their flowcharts.

Having failed to adequately discuss programming methodology, Poirot finally blunders into BASIC instruction, where the results are equally unsatisfactory. He focuses on an altogether minimal subset of the language which excludes string manipulation, matrices, user-defined functions, as well as all of the fine points of BASIC programming. Poirot has, fortunately, included a brief discussion of debugging in his chapter on BASIC; this is a critical skill for anyone who really hopes to learn programming from MICROCOMPUTER SYSTEMS AND APPLE BASIC.

SCIENCE AND ENGINEERING PROGRAMS APPLE II EDITION

John Heilborn

Level: Novice Rating: **85**

	Osborne	1981	Paper	
223 Pages	ISBN: 0-931900-63-2		8" x 11"	\$15.99

SCIENCE AND ENGINEERING PROGRAMS features forty-six specialized BASIC programs, none of which requires over 6K of memory, (excluding the space occupied by whatever data the user might supply). Although the programs were written for the Apple II, an extremely superficial appendix provides some comments on conversion into other BASIC dialects. An examination of the actual lines of code suggests, however, that all programs should run under almost any BASIC; the authors have gone to considerable trouble to include only the most generalized of statements, as is evidenced by their tendency to avoid even the BASIC exponentiation operand and to substitute simple multiplication for it.

The programs are grouped into twelve areas of interest to statisticians (data analysis, regression analysis eigenvalues and eigenvectors—the first time we've seen useful programs of this sort in a published software anthology) and engineers (structural analysis, thermodynamics

and heat transfer). Program listings are accompanied by a brief description of the program and sample output. Where arrays are used to store data, the authors usually list the line number of the DIM statement and note how it might be adjusted to fit the user's data set. Prompts clearly inform the user what action is expected of him.

All programs analyze data input from the keyboard; unfortunately, disk handling subroutines are not included for the analysis of really large data sets. Graphics subroutines, which would have been useful for the bivariate correlation analysis program, for example, have not been included. More broadly, the introductions to each of the programs make no attempt to expand either the user's programming proficiency or his knowledge of what the programs are intended to accomplish.

SCIENCE AND ENGINEERING PROGRAMS provides a convenient collection of ready-to-run programs for a very specialized audience. Although the programs could in most cases have been made more flexible, we nevertheless recommend this volume.

SOME COMMON BASIC PROGRAMS- APPLE II EDITION

Lon Poole, Mary Borchers and David M. Castlewitz
Level: Novice Rating: **85**

	Osborne	1982	Paper	
200 Pages	ISBN: 0-931988-683		9" x 11"	\$14.99

See our review of SOME COMMON PASCAL PROGRAMS on page 212.

MOSTLY BASIC-APPLICATIONS FOR YOUR APPLE II/BOOK 1

Howard Berenbon
Level: Novice Rating: **75**

	Sams	1981	Paper	
158 Pages	ISBN: 0-672-2189-9		9" x 11"	\$12.95

MOSTLY BASIC -APPLICATIONS FOR YOUR APPLE II/BOOK 2

Howard Berenbon

Level: Novice Rating: **90**

Sams 1981 Paper
217 Pages ISBN: 0-672-21864-X 9" x 11" \$12.95

Whether the use of the term "Basic" in the title refers to the language used in writing these programs or to the rather elementary character of many of the programs themselves is sometimes difficult to determine from an examination of these volumes. All three unnumbered volumes contain the same programs, although a slightly modified dialect of BASIC has been used in each instance to allow them to run on an Apple II, a TRS-80 and a Commodore PET. Similarly, both volumes of MOSTLY BASIC, Book 2, contain identical programs which have been modified to run under AppleSoft and TRS-80 BASIC. The twenty-eight programs in the first volume (grouped into real-time applications, educational programs, business and investment, home applications, utilities and "the unusual"—indicating a single-program Tarot Card reader) are all primarily short, simple programs of comparatively little interest to the user. The quality of the programs (educational programs, home applications, money and investment, E.S.P. testing and one fantasy game) in the second volume is far more variable, with short and relatively uninteresting applications provided along with very well-conceived and well-programmed ones. Examples of the latter include the Time Dungeon, a fantasy game which is simultaneously a history teacher; the advanced math and state capitals educational programs; and the Dungeon of Danger fantasy game.

With only one exception (the digital stopwatch program in Book 1, in which the seconds column was not properly reinitialized to zero), the programs which we tested from the Apple II volumes ran without error. On the other hand, given the trouble Berenbon has taken to adapt programs to each of these best-selling home computers—thus relieving the user of the need to make modifications or conversions to the programs—it is surprising that he did not also take the time to make them more "user-friendly." In the State Capitals program in Book 2, for example, the user who responds that "BOSTON" is the capital of Massachusetts will be credited with a correct answer, while the unfortunate person who accidentally enters "BOSTON—" will not be. The elimination of superfluous blanks at the end of a character string is a simple operation requiring just one or two program lines immediately after each input statement, but one which Berenbon has inexplicably failed to include.

Despite these failings, Book 2 of MOSTLY BASIC is a software bargain, while the first volume of the series is ideally suited to the

hapless microcomputer user who either will not or cannot do his or her own programming.

PASCAL PROGRAMMING FOR THE APPLE

T.G. Lewis

Level: Intermediate Rating: **95**

Reston 1981 Paper
234 Pages ISBN: 0-8359-5454-4 6" x 9" \$14.95

The seductive elegance of Pascal's inherent structure has enthralled more than a few authors of introductory programming texts; such texts focus on the admittedly fascinating conceptual framework of Pascal programs to the exclusion of its practical application to real business or scientific problem. PASCAL PROGRAMMING FOR THE APPLE redresses this imbalance. Lewis adopts an approach familiar to readers of BASIC primers—that of introducing new concepts, keywords and functions as part of a series of increasingly complex application programs.

A brief historical review of the origins of the Pascal language precedes a two-chapter introduction to the UCSD p-System. The fact that the reader is shown how the Pascal operating system works before a single line of code is explained is characteristic of Lewis' pragmatic approach. One highlight of this p-System introduction is the illustration, using graphics, of the assembly language equivalents of some simple Pascal commands. It is one thing to describe p-System naming conventions; it is quite another to simulate its 16-bit word "stack" mechanism through individual PUSH and POP statements. This detailed operating introduction is exceptional.

Lewis then "fills-in" the shell of a typical Pascal program with descriptions of data definitions, and procedures and main execution modules. Instructions are categorized as relating either to sequence, choice or iteration, and are explained using the appropriate Pascal terminology.

With this rudimentary foundation established, a series of application programs are developed which emphasize additional Pascal reserved words, functions and coding techniques. A Joseph Granville-type stock market analysis is used to demonstrate window pointers. A Real Estate Cash Flow Analysis describes the use of overlays to fold segments of a given Pascal program into a small section of memory. Random access and hash methods of file data retrieval are noted in a program which translates certain French words into their English equivalents. Pascal's TURTLEGRAPHICS extension and its musical capabilities are explored in respective chapters on "Star-Spangled Graphics" and "Making Music."

Lewis' concluding sections treat advanced file access techniques and a syntactical comparison of the BASIC and Pascal languages. His narrative in these chapters is indicative of the light, humorous tone present throughout the book.

PASCAL PROGRAMMING FOR THE APPLE is a practical how-to handbook fully utilizing this powerful high-level language. It is filled with application programs which themselves are sources of immediately usable code. Introductory programming language authors would do well to follow Lewis' lead.

PRACTICAL BASIC PROGRAMS-APPLE II EDITION

Lon Poole

Level: Novice Rating: **90**

	Osborne	1981	Paper	
179 Pages	ISBN: 0-931988-66-7		9" x 11"	\$15.99

See our review of PRACTICAL PASCAL PROGRAMS on page 210.

SWIFT'S EDUCATIONAL SOFTWARE DIRECTORY

Level: All Rating: **80**

	Sterling Swift	1982	Paper	
358 Pages	ISBN: 0-88408-150-8		6" x 9"	\$14.95

SWIFT'S is a spiral-bound directory of over 800 educational software packages for the Apple II, ranging from administrative applications to elementary school teaching aids to programs for college-level courses. Entries are organized by software publisher or vendor, presumably to indicate the extent of product support provided by each publisher—although in fact the text rarely does this. Separate indexes provide an alphabetic listing of program names and a topical listing further subdivided by grade level. The categories used in this latter index, however, are broadly defined; the use of single sections for mathematics and social studies, for example, does not contribute to the directory's ease of use.

Each entry contains a brief description of the software, along with its price and minimum system requirements (number of disk drives, memory size, programming language) necessary for it to run successfully. The descriptions are not critical, nor do they compare and contrast a given

program with similar applications. The directory does provide two methods for the reader to evaluate a software package: especially noteworthy programs have been designated with a solid circle (although the reasons for this are not clearly specified, very few entries have been singled out in this way); and the description indicates whether a given entry has ever been reviewed in any of approximately fourteen educational periodicals.

This is, in short, a guide which requires further effort and research on the part of the user. Its relatively complete coverage of educational software for the Apple, however, makes this an indispensable directory for schools or individuals concerned with the application of microcomputers in education.

TECHNIQUES FOR CREATING GOLDEN DELICIOUS GAMES —FOR THE APPLE COMPUTER

Howard M. Franklin, Joanne Koltnow, LeRoy Finkel

Level: Intermediate Rating: **50**

Wiley 1982 Paper
150 Pages ISBN: 0-471-09083-2 7" x 10" \$12.95

There are two possible ways of learning to create computer games from this book, neither of which involve reading the author's text: either buy the disk (\$34.95) which contains the sample programs listed in its chapters and infer from their code the techniques employed; or buy a magnifying glass (\$3.95) to read the Epson-quality BASIC programs listed in its chapters, and infer from their code the techniques employed.

While the authors' outlines of low- and high-resolution graphics is diverting, neither imparts the technical knowledge needed to use them. Indeed, the high resolution chapter is notable in that it is actually introduced by a series of disclaimers about what it does not cover—the assembly language subroutine method, or the alternative shape table method, or, in short, high resolution graphics.

The method that the book itself uses is to present hundreds of lines of marginally-readable BASIC (we used the randomizing code on page 22 to figure out the punctuation of the rest of the book's programs), and then through slight modification of this code, to demonstrate the chapter's topic. Within this context, it is impressive that two areas actually do enlighten: the chapters on Hangman-type text manipulation programs, and the chapter which treats BASIC subroutines for data entry validity checking.

GAMES may be of marginal interest to the experienced programmer who is willing to learn as much from what is not explained as what is, a recommendation as strong as the book's treatment of its purported topic.

USING 6502 ASSEMBLY LANGUAGE

How Anyone Can Program the Apple II

Randy Hyde

Level: Novice Rating: **90**

Datamost	1981	Paper	
283 Pages	ISBN: (None)	5" x 8"	\$19.95

Although this introduction to 6502 assembly language programming for the Apple II is "intended for use by beginning, intermediate and advanced programmers,"(1-1) we feel that it provides far too elementary an approach for anyone but the newcomer to assembly language programming.

This volume is primarily intended as a guide to programming with LISA, a disk-based assembler for the APPLE II manufactured by DATAMOST, Inc., the book's publisher. Hyde notes that none of the Apple assemblers are compatible with one another, a problem magnified somewhat by LISA's inclusion of four mnemonics which are not part of the 6502 instruction set (although these instructions—BGE, BLT, BTR and BFL—are identical to BCS, BCC, BEQ and BNE, and have been added because they are easier to remember in certain situations). Neither the focus on the LISA assembler nor LISA's inclusion of these four new mnemonics, therefore, should diminish the appeal of this book to newcomers.

USING 6502 ASSEMBLY LANGUAGE in fact furnishes an outstanding introduction to Apple assembly language programming. Having assumed that his readers have at least some familiarity with BASIC, Hyde effectively contrasts BASIC statements with their assembly language counterparts throughout the text. The focus of this work is also solely on assembly language programming; machine language is not discussed. This permits a more detailed discussion of the format of source code, as well as the use of pseudo-op codes, than is generally found in most texts. This treatment in fact exemplifies the book's general approach, which emphasizes practical hands-on experience in writing and assembling source code as a means of quickly learning 6502 assembly language programming.

WILL SOMEONE PLEASE TELL ME WHAT AN APPLE CAN DO?

Glenn M. Polin

Level: Novice Rating: **60**

Sterling Swift 1983 Paper
136 Pages ISBN: 0-88408-152-4 6" x 9" \$12.95

WILL SOMEONE PLEASE TELL ME WHAT AN APPLE CAN DO? is a collection of eight essays of highly differing qualities which focus on applying the Apple in eight different areas: accounting, agriculture, financial planning, investment, education, music, simulations and word processing. The common denominator linking each contribution is an emphasis on applications software. (Even this generalization does not hold true for each article; the otherwise undistinguished section on word processing discusses word processing features and requirements, and relegates the list of three word processing packages to the end of the article.)

The book's focus on available software has at times led contributors to loudly beat their own drums. This is true of the book's focus on Apple computers as a whole. Time after time, software which is either available for other microcomputer systems (e.g., VisiCalc) or, worse, was originally written for other systems (e.g., WordStar) is intimately identified with the Apple, which comes to personify the ultimate in state-of-the-art microcomputer technology. The explanation for this is that the volume's editor, Glenn M. Polin, is currently Apple's Manager of Education Market Development. The most self-serving essay in the book, the one dealing with accounting, is notable for its emphasis on the fine software products from BPI Systems, Inc.; the authors are John A. Moss, the firm's founder, and Dick Shocket, a technical writer at BPI.

Curiously, the one contributor who had the greatest reason to promote his software to the exclusion of all else failed to do so. Don Williams' essay on financial planning presents an interesting and useful comparison of VisiCalc and DESKTOP/PLAN which stresses the value of each package; Mr. Williams is the author of DESKTOP/PLAN. The only other essay of interest in this otherwise dreary volume is Jim Salmon's contribution on simulations, which defines the concept and then provides six examples. Aside from these two articles, WILL SOMEONE PLEASE TELL ME WHAT AN APPLE CAN DO? provides free advertising for one computer manufacturer and a wide range of software houses at the reader's expense.

32 BASIC PROGRAMS FOR THE APPLE COMPUTER

Tom Rugg and Phil Feldman

Level: Novice Rating: **100**

	Dilithium	1981	Paper	
284 Pages	ISBN: 0-918398-34-7		6" x 9"	\$17.95

Like many other published volumes of microcomputer software, 32 BASIC PROGRAMS contains its share of public domain software (bio-rhythm, loan, checkbook, and arithmetic). Even where this is the case, however, Rugg and Feldman have assembled a collection of programs notable for its professional programming quality and its user-friendliness. Some of the programs in this volume, especially its games, are characterized not only by competent and professional programming, but also by their definite originality and inventiveness. Our favorites were Roadrace (in which the user must prevent his car from driving off a curving road) and Obstacle (a program, evidently based on the Walt Disney movie TRON, in which the user must keep from colliding with walls built either by himself or his opponent).

The quality of the documentation provided in this volume of software matches the quality of its programming. In numerous instances, the authors have included photographs of the video display. In all cases, they have written introductions to the programs showing how they operate and the principles upon which they are based. The variables used in the programs are itemized, and the function of individual lines of BASIC code is described for each program. Finally, for those users who would like to change either individual features of a program or add their own enhancements, Rugg and Feldman provide a list of easy changes (one-line adjustments to the program) and more ambitious suggestions for expanding the program.

For those who want to run the software but do not want to do the typing, Dilithium Press also offers these 32 programs on diskette for \$19.95. But whether the user chooses the book or the diskette, Rugg and Feldman's collection is one of the best software values available.

IBM

BASIC EXERCISES FOR THE IBM PERSONAL COMPUTER

Jean-Pierre Lamoitier

Level: Intermediate Rating: **85**

Sybex 1982 Paper
251 Pages ISBN: 0-89588-088-1 7" x 9" \$13.95

See our review of FIFTY BASIC EXERCISES on page 139.

A GUIDE TO PROGRAMMING IBM PERSONAL COMPUTER

Bruce Presley

Level: Novice Rating: **90**

Van Nostrand Reinhold 1982 Paper
219 Pages ISBN: 0-442-26015-6 9" x 11" \$16.95

One general criterion for selecting an effective programming introduction is that the author have microcomputer teaching experience: not only do such texts reveal a greater degree of reader sensitivity in content and production quality, but programming problems are well integrated into the teaching process, and not textual appendages. Bruce Presley, director of the Lawrenceville School, has crafted a superior BASIC introduction, based upon "experiences in the classroom over the past 15 years." Nowhere is the merit of this presentation more apparent than in the originality of its coding exercises.

This is a beginner's book. Short machine programming examples masterfully develop fluency in BASIC commands. Emphasis is placed upon graphics and games in reinforcing the learning process: trigonometric functions are used to snake messages across the screen, and the randomizing function is demonstrated in a single slot machine program which selects among CHERRY, PLUM, LEMON, BAR and ORANGE. While Presley does not avoid using some of the better traditional coding problems (the "Towers of Hanoi" puzzle), most of his chapter exercises are characterized by a plugged-in topicality (a daily schedule program is composed for a busy Marcus Welby). Answers to all exercises are included in an appendix.

The discussion of BASIC programming subjects progresses to include subscripted and string variables and elementary structured pro-

gramming techniques. While sequential and random access file methods are covered, the book falls short of adequately explaining advanced programming topics. Bubble sorts, for example, are described parenthetically in the course of a payroll application program.

We nevertheless recommend *A GUIDE TO PROGRAMMING IBM PERSONAL COMPUTER*, and each of the books in Presley's *Guide To Programming Series*, as quality beginners' texts on machine-specific BASIC dialects.

HOW TO WRITE AN I.B.M.-PC PROGRAM

Ed Faulk

Level: Novice Rating: **50**

	DataMost	1982	Paper	
129 Pages	ISBN: 0-8359-2991-4		5" x 8"	\$14.95

See our review of *HOW TO WRITE A TRS-80 PROGRAM* on page 292.

IBM PC BASIC Quick Reference Guide

Gilbert Held

Level: All Rating: **85**

	Wiley	1982	Paper	
8 Pages	ISBN: 0-471-87042-0		6" x 12"	\$2.95

For the user seated before a computer, nothing is more frustrating than entering ten variations of a single BASIC command, only to be greeted each time with the inevitable syntax error. When the user finally decides to resort to reference manuals, their tendency to close shut or fall to the floor just as he finds the necessary command adds to the aggravation. Before embarking on a drastic course of action, we recommend that the user consider purchasing any of the three *QUICK REFERENCE GUIDES* published by Wiley for the IBM, Apple II or Atari computers.

Although they are not books, these inexpensive eight-page cards will compete with low-end reference handbooks and dictionaries. They offer a summary of BASIC commands grouped by function (system commands input/output, intrinsic BASIC functions, etc.), as well as a listing of BASIC operators, file naming conventions and control codes.

The vast amount of information which they pack into a limited amount of space makes them convenient and useful for BASIC programmers seated before their CRTs.

The guides do contain, however, some errors of omission. Most disappointing—although understandable—is their failure to list DOS commands, especially for the IBM PC. Somewhat more conspicuously absent is a treatment of arithmetic precedence in the tables dealing with BASIC operators, or a note on variable naming conventions for the Apple II and Atari computers. Neither the Apple II user who is working with random access files nor the IBM user exploring the PC's sound capabilities will find their guides to be of much use. But in most cases, the guides do succeed in providing a concise and helpful summary of BASIC commands and syntax.

THE IBM/PC GUIDE (Diskette Included)

James E. Kelley, Jr.

Level: Novice Rating: **95**

Dell 1983 Paper
310 Pages ISBN: 0-440-03946-0 6" x 9" \$29.95

Among the microcomputer books spawned by the introduction of the IBM Personal Computer is this novel guide, which includes a 5 1/4" floppy disk. The disk does not merely contain copies of the book's programs, but actually supports the author's "interactive" approach to his topic—disk source is used as the "skeleton" for reader solutions to programming exercises. Stand-alone applications are developed using these successfully modified programs.

Disk, book and IBM reference manuals provide the total learning environment. From initial power-up to disk file manipulation, the reader is guided through operational and programming topics. For example, an activities list compiled by the author is loaded from disk and used to demonstrate PC/DOS editor capabilities.

In a tribute to "the virtues of plagiarism," Kelley shows how reader exercises can be merged into author code to produce a working application. Beyond the practicability of its presentation and the effectiveness of its disk file use, the text is notable for its enlightened approach to user-friendly programming. Applications are menu-driven. The importance of validating all INKEY\$ statements is emphasized. Many of the "pro-human" programming techniques popularized by books like APPLE BACKPACK (see our review) work their way into the natural flow of Kelley's programming discussions.

Unlike *LEARNING IBM BASIC*, Kelley's text rarely ventures into advanced BASIC functions (relatively little attention is given to color and high resolution graphics); special topics do include the use of the IBM 80 CPS and Epson MX-80G printers and the use of DIF files in printed reports.

The IBM/PC guide is one of the more original presentations of introductory BASIC programming. While not as comprehensive as similar BASIC manuals, it provides a clear, professional approach to Personal Computer operation and programming.

IBM PERSONAL COMPUTER

An Introduction to Programming and Applications

Larry Joel Goldstein and Martin Goldstein

Level: Novice Rating: 70

	Brady 1982 Paper		
302 Pages	ISBN: 0-89303-111-9	7" x 10"	\$14.95

This guide to the IBM Personal Computer attempts to address virtually every aspect of the computer's use—an enormous range of topics which includes the operation of the PC, mass storage devices, the IBM editor, programming in BASIC, computer graphics and computer applications. It does not succeed, however, in covering any of them well. The book's 302 pages are interspersed with numerous exercises for the reader (many of which are condescending or trivial), so that the number of pages actually devoted to the IBM Personal Computer and its operation is actually quite limited. On a number of occasions, the authors simply refer the reader who has an IBM PC with disk drives to the IBM reference manual to look up some of the not-so-fine points of manipulating data files.

Despite these very real limitations, the book is effective in whetting the appetite of the beginning PC user. This is especially true of the last half, which discusses possible microcomputer applications. The discussion of word processing is adequate (although the authors, for some inexplicable reason, attempt to write a simple BASIC word processing program), while the discussions of VisiCalc, computer games, computer simulations and graphics are quite good, if abbreviated.

The Goldsteins' book provides an adequate sampling of Personal Computer-related topics for the complete novice. In each of the categories which they hastily survey, far superior treatments are available.

THE IBM PERSONAL COMPUTER

Robert J. Traister

Level: Novice Rating: **65**

Tab 1983 Paper
 197 Pages ISBN: 0-8306-1496-6 8" x 9" \$10.95

Traister is an author who, judging from the list of books he has written, is interested in a wide range of gadgets; his other publications include HOW TO BUILD METAL/TREASURE LOCATORS, BUILD A PERSONAL EARTH STATION..., ALL ABOUT AIRGUNS, and 32 ELECTRONIC SUPPLY PROJECTS. This focus on the device rather than its application is characteristic of THE IBM PERSONAL COMPUTER and accounts for the book's limited appeal.

Aside from an eight-page chapter listing software available for the personal computer and the obligatory and strikingly superficial chapter devoted to BASIC, this book focuses primarily on providing a capsule history of IBM (which loaned Traister the Personal Computer so that he could write this book), on the hardware available for the Personal Computer and on its installation. Very rarely is this discussion of Personal Computer hardware or system design critical in its focus; among the few problems or criticisms mentioned by Traister are the keyboard space bar, which tended to be unresponsive in early models, and the limited possibility of expanding the system, since the Personal Computer contains only five expansion slots.

THE IBM PERSONAL COMPUTER is of primary interest to two groups of people: those considering purchasing the Personal Computer and those having difficulty setting it up. For the former group, Traister's treatment is vastly inferior to that provided in Que's IBM'S PERSONAL COMPUTER. As for setting up the Personal Computer, Traister himself notes that "the IBM Personal Computer is quite easy to set up if you use the IBM Guide To Operations manual, which is included."(61) Our own experience in setting up the Personal Computer corroborates this. For the second group of readers, then, we suggest that they confine themselves to the IBM manual, and not Traister's book.

IBM'S PERSONAL COMPUTER

Chris DeVoney and Richard Summe

Level: All Rating: **90**

Que 1982 Paper
 303 Pages ISBN: 0-88022-100-3 7" x 9" \$14.95

Unlike many other offerings on the IBM Personal Computer, Que's IBM'S PERSONAL COMPUTER does not attempt to teach the reader

BASIC, nor does it claim to inform the reader how to operate the Personal Computer. (Paradoxically, it fulfills this latter goal far more successfully than a number of other works which claim to treat precisely this topic.) Instead, the book presents a comprehensive assessment of the IBM Personal Computer by examining its significance for the growing microcomputer market and evaluating the design and utility of each of its components, as well as of the system as a whole. The book is intended for anyone interested in buying an IBM Personal Computer, without regard for their degree of knowledge of—or familiarity with—computers. The book is ideally written to appeal to such a disparate audience; while the novice may find individual portions of each discussion too highly specialized and somewhat incomprehensible, the basic line of argument which the authors pursue is invariably clear, and some useful information is accessible to all readers regardless of their level of expertise.

The topics treated in IBM'S PERSONAL COMPUTER are designed to provide the potential Personal Computer purchaser/user with a sense of how the IBM Personal Computer fits together as a total system, how it compares to other popular small computer systems and how it can be expanded. The evaluation of system hardware and peripherals, available operating systems (PC DOS, CPM-86, the UCSD p-System and UNIX), programming languages (IBM BASIC, Pascal and FORTRAN, UCSD Pascal and FORTRAN), and selected IBM software (IBM/VisiCalc, IBM/EasyWriter version 1.0 and IBM/Peachtree version 1.0) is highly critical, and designed to enable the reader to assess the strengths and weaknesses of the IBM Personal Computer in relation to his or her needs.

As a result, this Que publication is an essential introductory text for potential purchasers and an essential reference work for owners. In addition, because it provides a standard against which other microcomputer systems can be measured, the book can be profitably used by anyone considering the purchase of a microcomputer, IBM or otherwise.

LEARNING IBM BASIC FOR THE PERSONAL COMPUTER

David A. Lien

Level: Novice Rating: **95**

CompuSoft	1982	Paper	
425 Pages	ISBN: 0-932760-13-9	7" x 9"	\$19.95

The author of THE BASIC HANDBOOK has written a light, enjoyable BASIC introduction for the IBM Personal Computer. Each of the fifty-three chapters in this first of two volumes has been intentionally shortened (some are only three pages long) to avoid overwhelming the reader

with too many concepts or commands. Additionally, the author shuns what he describes as "national debt size words." A step-by-step approach concentrates on the successful completion of machine exercises. All programs are written using the Advanced BASIC module of PC DOS.

The treatment is so careful in avoiding potential obscurity that many readers, even novices, may find the early chapters a bit plodding. The tempo quickens, however, with exercises highlighting the interaction of the operating system and hardware with the programming process. For example, though the author's primary focus is on "pure" BASIC programming, incidental discussions treat the use of special function keys and DOS editing commands. Though the editor chapter omits some functions noted in IBM's own DOS reference manual, this blend of programming and system introduction is masterfully executed. The generally relaxed tone evident in Lien's conversational prose is enhanced in every chapter by poignant pen-and-ink illustrations.

"Intermediate" BASIC, that is, BASIC which treats functions less complex than sequential and random access disk file manipulation, forms the core of the author's syllabus (reader's of David Lien's LEARNING TRS-80 BASIC will notice a marked similarity in structure and treatment). Multi-dimensional arrays, flowcharting, debugging, and even a chapter—complete with memory map—of special PEEK and POKE addresses are included. Characteristic of the presentation is the fact that a chapter which discusses BASIC solutions to trigonometric problems first provides the reader with a brief review of high school trigonometry. Of special note is a chapter on the unique musical capabilities of the Personal Computer.

LEARNING IBM BASIC FOR THE PERSONAL COMPUTER is one of the best machine-specific BASIC introductions in the literature. We highly recommend this first volume, and eagerly await the publication of an equally expert presentation of advanced BASIC topics.

PRACTICAL BASIC PROGRAMS-IBM PERSONAL COMPUTER EDITION

Lon Poole

Level: Novice Rating: **90**

	Osborne 1982	Paper	
162 Pages	ISBN: 0-931988-80-2	9" x 11"	\$15.99

See our review of PRACTICAL PASCAL PROGRAMS on page 210.

SOME COMMON BASIC PROGRAMS -IBM PERSONAL COMPUTER EDITION

Lon Poole, Mary Borchers and Peter M. Burke

Level: Novice Rating: **85**

	Osborne	1982	Paper	
212 Pages	ISBN: 0-931988-83-7		9" x 11"	\$14.99

See our review of SOME COMMON PASCAL PROGRAMS on page 212.

UNDERSTANDING THE IBM PERSONAL COMPUTER VOL.I

Charles A. Calabrese

Level: Novice Rating: **55**

	Exposition	1982	Paper	
253 Pages	ISBN: (None)		5" x 8"	\$14.95

Calabrese has written this introduction to the IBM Personal Computer for the reader to whom "the workings of computers in general is a completely new and foreign topic."(I-1) The book is divided into twenty-three sections which teach the reader how to start up the system, to use some of the basic features of the IBM Disk Operating System (Calabrese assumes a Personal Computer with two disk drives), and to write a BASIC program which will access data stored on disk. The book is clearly written, and the explanations given are relatively complete and calculated not to confuse the novice computer user.

Despite this clarity, Calabrese's book suffers from a number of serious drawbacks. The first is its extremely low production quality. The typeface of the book is that of IBM's Epson dot matrix printer, which is hardly of correspondence quality. The result is a text which causes eyestrain, and in which it is very easy to accidentally skip lines. After the reader has finished the book, it becomes almost useless as a reference manual, since Calabrese has not included an index nor has he numbered pages consecutively from beginning to end. Even worse is the approach which Calabrese adopts in trying to teach the reader how to program in BASIC (which is the central topic of eighteen sections of the book). Users typically purchase microcomputers for specific applications, and their approaches to programming will frequently reflect specific needs and interests. Calabrese, however, ignores this diversity as he tries to show the reader how to write a complete program which will examine, add, delete or print records from a name and address file stored on disk; all

eighteen sections on BASIC are devoted to writing this program. Since the manipulation of string files is not a topic of intrinsic interest to all readers, the result is far more likely to be boredom than instruction.

USER'S HANDBOOK TO THE IBM PERSONAL COMPUTER

Jeffrey R. Weber

Level: Novice Rating: **85**

Weber 1982 Paper
294 Pages ISBN: 0-938862-13-8 5" x 8" \$13.95

Jeffrey Weber has left theoretical discussions of microprocessor architecture, structured programming and personal computer color selection to others; instead, he has written a how-to-unpack-it-and-make-it-go manual for one very specific, very real microcomputer. His book covers all of the important operational aspects of the IBM Personal Computer in a manner guaranteed to reassure even the most befuddled novice.

The handbook begins with a complete hardware overview, summarized in charts which describe the precise characteristics of each of the system's components. After clearly illustrated set-up instructions and an extensive problem determination guide, the reader is presented with the intricacies of PC DOS, IBM's operating system. Special emphasis is placed on the use of DOS commands, particularly EDLIN, LINK, and DEBUG. In keeping with Mr. Weber's "handbook" philosophy, the treatment of BASIC which concludes the book neglects any mention of programming techniques in favor of a purely operational approach to BASIC on the IBM Personal Computer; keyboard and naming conventions used by cassette, disk and advanced BASIC are highlighted.

If any real qualification can be made of the USER'S HANDBOOK, it is that the preoccupation with simplicity sometimes degenerates into needless redundancy. Sentences like the following are not unusual: "the main difference between the color/graphics and the display/printer adapter is that the color/graphics board can provide color, while the display/printer board cannot."(p.48)

However, we still recommend USER'S HANDBOOK TO THE IBM PERSONAL COMPUTER both as an initial nuts-and-bolts introduction and as a problem determination, DOS command and BASIC keyboard manual for future reference.

USING THE IBM PERSONAL COMPUTER

T.G. Lewis

Level: Novice Rating: **90**

Reston 1983 Paper
239 Pages ISBN: 0-8359-8140-1 6" x 9" \$14.95

A number of volumes dedicated to helping the novice use the IBM Personal Computer assume that its operation is so complex that the user will at best be able to grapple with turning the system on, loading BASIC, and running a few simple BASIC programs; any attempt to show the reader how to use the system's numerous features effectively is simply considered too taxing. Such an approach, we believe, helps to retard the development of computer skills.

In contrast, Lewis' USING THE IBM PERSONAL COMPUTER represents a refreshing departure from this kind of literature. Lewis has wisely relegated BASIC instruction to other, more detailed textbooks, and instead concentrates on providing a brief list of BASIC commands implemented on the Personal Computer and a discussion of the speed of the BASIC interpreter. Similarly, his treatment of Pascal explains the process of compiling Pascal source code and explains metacommands used to control the operation of the Pascal compiler. Aside from this discussion of programming languages, Lewis provides the reader with a very good preliminary introduction to IBM DOS (we found his treatment of batch files, and especially of AUTOEXEC.BAT, a collection of DOS commands which are automatically executed whenever DOS is booted, to be especially informative), and examines some software applications; these include word processing with EasyWriter, spread sheet calculations with VisiCalc and data base management using a program written by Lewis himself. While the chapter on data base management would be better if it focused on a real software package, the introductions to EasyWriter and VisiCalc are extremely informative and useful for a novice.

Aside from encouraging the reader to use the Personal Computer most effectively, Lewis evidently intends to use his discussion of the Personal Computer as a means of expanding the reader's general knowledge about computer technology. The book's beginning and end chapters—which have been included for this purpose—provide a superficial treatment which is largely unrelated to the book's intervening sections.

Nevertheless, in contrast to some other works which claim to help the user "understand" or "apply" the Personal Computer, we highly recommend T.G. Lewis' USING THE IBM PERSONAL COMPUTER to those beginning and intermediate users wishing to do exactly what his title claims.

USING THE IBM PERSONAL COMPUTER

Kenniston W. Lord, Jr.

Level: Novice Rating: **75**

Van Nostrand Reinhold 1983 Paper
319 Pages ISBN: 0-442-26078-4 6" x 9" \$12.95

In a departure from the traditional Ruy Lopez opening to BASIC introductions, the author leaps into commands and syntax from an oblique discussion of numeric variables. This is a gambit which ultimately fails. Though Kenniston Lord's introduction to BASIC leaves much to be desired, his later treatment of software system design is clear and comprehensive; a poor opening is salvaged by a strong endgame.

Reader and author thrash through an introduction to BASIC marked by barrages of technical data without examples (ten function keys are presented simultaneously) and with examples which illustrate poor programming techniques (one eighteen-line program contains three GOTO's). Fortunately, this programming discussion evolves into descriptions of systems application design. Here Lord is at his best. A modular approach to a menu-driven inventory data entry system incorporates early BASIC coding lessons into a professionally designed inventory control system. After digressions into graphics and DOS, Lord returns to this software design theme. The book's highlight is a complete small business system, replete with accounts receivable, accounts payable, inventory and sales analysis modules. Documentation is excellent. Many users will find this section useful not for its illustration of BASIC coding techniques, but as a source of raw, useable programs.

USING THE IBM PERSONAL COMPUTER meets with only limited success as either an operational manual or programming introduction; its greatest appeal is as a unique informative presentation of microcomputer software systems design.

USING YOUR IBM PERSONAL COMPUTER

Lon Poole

Level: Novice Rating: **95**

Sams 1983 Paper
326 Pages ISBN: 0-672-22000-8 8" x 9" \$16.95

Lon Poole, the author of YOUR ATARI COMPUTER, has written an outstanding operational guide to the IBM Personal Computer. The book closely focuses on the needs of the BASIC programmer: the first part includes a detailed discussion of PC DOS; the second consists of a comprehensive survey of IBM BASIC.

What is most distinctive about USING YOUR IBM PERSONAL COMPUTER is its successful integration of a vast array of topics into a single, well-focused, comprehensive discussion. Poole emphasizes above all those features which are of immediate operational concern to the BASIC programmer; hence, his discussion of PC DOS, for example, omits EDLIN (the Personal Computer's text editor) and its link loader, both of which are primarily of interest to the assembly language programmer. But unlike many other works which introduce the reader to BASIC for a particular microcomputer, this is not a trivial, watered-down treatment which presents only the bare rudiments of the language. Poole also includes sections on the use of sound and graphics, the manipulation of both sequential and random data files, and the technique of overlaying programs with the CHAIN statment. We can think of no topic Poole has overlooked, and very little that needs to be expanded or improved upon.

USING YOUR IBM PERSONAL COMPUTER serves as a model BASIC text for a particular microcomputer. The complete novice will undoubtedly find the book difficult, although challenging and rewarding. For more experienced programmers, it is indispensable. For all BASIC programmers who are serious about using their IBM Personal Computer, it is invaluable both as a tutorial and a work of reference.

TRS-80

BASIC FASTER AND BETTER & OTHER MYSTERIES

TRS-80 Information Series Vol.IV

Lewis Rosenfelder

Level: Intermediate Rating: **100**

	IJG	1981	Paper		
288 Pages	ISBN: 0-936200-03-0		9" x 11"		\$29.95

Once the aspiring BASIC programmer graduates from printing his/her name on the CRT to more ambitious programming tasks involving either the manipulation of large data sets or the execution of lengthy programs, two factors intervene to reduce the enjoyment derived from a microcomputer. The first of these, of course, is the limited memory capacity of many personal computers; the second is the frequently long execution time required by many complex programs. For those who are interested in confronting these problems head on, as well as for those who are simply interested in becoming advanced BASIC programmers, Rosenfelder's BASIC FASTER AND BETTER is made to order.

There are two factors which apparently limit the audience for this book. The first, which is a very real consideration, is that Rosenfelder presupposes that his readers are almost fully conversant with the BASIC programming language; no attempt is ever made to explain what a particular BASIC statement does, since the author has instead chosen to emphasize using these statements most effectively. But the second factor—that this is the fourth volume of the TRS-80 INFORMATION SERIES—should not deter non-TRS-80 users from buying or reading this volume; it is simply too good to overlook. While it is true that many of the specific features which Rosenfelder employs are peculiarities of TRS-80 BASIC, his ultimate programming refinements are sometimes generalized ones which can be used on other microcomputers (especially on those based on a Z80 microprocessor), or can be adapted to a specific machine or BASIC dialect by an enterprising programmer.

In his attempt to substitute speed and efficiency for awkward, muddling code, Rosenfelder emphasizes the replacement of some IF statements by equations with Boolean operators, the use of relocatable machine language subroutines (which can be included in a BASIC program) to speed execution time, an increased reliance on BASIC user-defined functions, and the overlaying of programs (i.e., using a master program to load and run other programs from diskette) in order to conserve memory space. Aside from presenting techniques to improve program execution, Rosenfelder also discusses methods for improving data entry by making programs more user-friendly and less user-frustrating, as well as for effectively displaying material on either the CRT or the printer. Virtually every facet of programming is in some way addressed in the course of BASIC FASTER AND BETTER, which is one of the best advanced BASIC programming texts we have read.

CIRCUIT DESIGN PROGRAMS FOR THE TRS-80

Howard M. Berlin

Level: Advanced Rating: **95**

Sams 1980 Paper
140 Pages ISBN: 0-672-21741-4 9" x 12" \$14.50

CIRCUIT DESIGN PROGRAMS is a rather imposing and formidable title which, unfortunately, probably serves to minimize the attention which an otherwise excellent collection of software applications deserves. After seeing the book's title, we assumed that it contained a collection of interfacing projects for the hobbyist; after cursorily inspecting the book, we thought that it presented highly specialized programs

permitting an engineer to design and analyze electronic networks. Although the bulk of the book's software does indeed deal with the analysis of electric signals and design of electronic circuits, the inclusion of programs for plotting graphs and conducting some basic statistical analyses makes it a rather specialized work, but one of interest to a broader reading public than its title indicates.

The quality of the programs developed by Berlin (which are divided into graph plotting, signals analysis, basic statistical analysis, networks and transforms, active filter design, solid-state devices and a miscellaneous chapter) are uniformly high, while most of the programs themselves are notable for their flexibility and user-friendliness. The graph plotting programs, for example, permit the user to determine the scale of the x and y axes, which permits the examination of a variety of plots for the same data or the display of a selected portion of a larger plot. Berlin's regression procedure allows the user to select the type of regression analysis desired (linear, polynomial, exponential, geometric or inverse), as well as to add to, delete from or modify the data set; in addition, data can be entered either from the terminal or from disk, a feature rarely encountered in software available only in published form.

Although essentially the same programs are contained in these two volumes, with the Apple II version written in AppleSoft and TRS-80 version written in TRS-80 BASIC, the programs for the TRS-80 in general provide the user with a far more detailed wealth of information; the graph and histogram plotting procedures for the TRS-80, for example, label the graph and provide values and tickmarks along the x and y axes, while the programs written for the Apple II do not.

CIRCUIT DESIGN PROGRAMS features a collection of applications software which is notable for its professional programming quality, flexibility and ease of use. This is the best volume of published programs for science and engineering which we have seen to date.

COMPUTERS FOR KIDS-TRS-80 EDITION

Sally Greenwood Larsen

Level: Novice Rating: **90**

Creative Computing 1981 Paper
53 Pages ISBN: 0-916688-20-8 5" x 8" \$4.95

See our review of COMPUTERS FOR KIDS - ATARI EDITION on page 318.

THE CUSTOM TRS-80 & OTHER MYSTERIES

TRS-80 Information Series Vol. III

Dennis Bathory Kitsz

Level: Advanced Rating: **85**

IJG 1982 Paper
335 Pages ISBN: 0-936200-02-2 8" x 11" \$29.95

Many microcomputer enthusiasts have an aversion to appliance computers because of the distance they put between the user and the technology he or she is using. Previously, two major alternatives to turnkey systems were available: a microcomputer could be purchased in kit form and then assembled; or, with the aid of a text such as Ciarcia's *BUILD YOUR OWN Z80 COMPUTER*, the hobbyist could assemble a microcomputer from individual components. Kitsz, however, has suggested a third course of action: the user can buy an appliance computer in order to rebuild and "customize" it. *THE CUSTOM TRS-80* is a guide for doing just this with the TRS-80 Model I.

Among the hardware modifications which Kitsz suggests are memory expansion, the installation of an expansion reset button, extra keyboard and video display monitor, the replacement of the numerical keypad with a hexadecimal one, and the addition of an 8-track cassette. For each of the numerous projects which he recommends, Kitsz provides relatively complete explanations, accompanied by schematics, diagrams and photographs. And the book's focus is not exclusively on hardware; the author also suggests a variety of changes and additions to TRS-80 firmware. These include a key bounce, audible beep, key repeat and upper/lower case driver routine (which is patched into the TRS-80 keyboard scan routine); customizing the BASIC interpreter (by diverting program execution before the BASIC interpreter proper can be entered); and the installation of Level I ROM (which Kitsz recommends as "an ideal teaching language"). Finally, Kitsz examines selected advanced programming topics such as disabling BASIC's LIST command or restoring a program after the NEW command has supposedly erased program memory.

THE CUSTOM TRS-80 is a volume we recommend for the large range of topics it covers. Performing any of the hardware modifications mentioned in the book, however, does require a certain degree of expertise. In addition, since we ourselves have not attempted any of these modifications, we would hope that the discussion of these projects has been presented somewhat more carefully and accurately than portions of the text indicate; the presence of such helpful comments as "see Table 3-(?)" indicate that the book has not been proofread especially carefully.

DATA MANAGEMENT TECHNIQUES

John P. Grillo and J.D. Robertson
 Level: Advanced Rating: **75**

Wm. C. Brown 1981 Paper
 193 Pages ISBN: 0-697-09954-7 8" x 11" \$16.95

The topic treated by DATA MANAGEMENT TECHNIQUES (the handling, retrieval and sorting of large blocks of data, whether in memory or on disk) is of interest to all programmers whose skill has evolved to the point at which they are capable of producing programs of some complexity. It is unfortunate that Grillo and Robertson's final volume in the Microcomputer Power Series for the TRS-80 does not contribute toward helping any but the advanced user to express that growing interest in programming proficiency.

This volume treats a wide range of important data management techniques: these include the use of pointers to sort data and to access elements of an array; the manipulation of strings; the use of stacks, queues, dequeues, and linked lists as pointers; and the handling of sequential and direct access files. But rather than provide any detailed explanation of these techniques, the authors simply allow the reader to glean whatever information he can from examining their lines of BASIC code.

At times, as in their treatment of linear, linked list or tree searches, the authors do present elements of a general discussion. In other cases, their programs are sufficiently engaging to rivet the attention of even the most confused reader; our favorites were the programs for word frequency analysis and for a computer-generated report on computer operations (less a commentary on the quality of the program than on the intelligibility of computerese). Despite these high points, however, DATA MANAGEMENT TECHNIQUES remains a work which is almost exclusively accessible to the advanced programmer.

FAST BASIC

Beyond TRS-80 BASIC

George A. Gratzer
 Level: Intermediate Rating: **75**

Wiley 1982 Paper
 278 Pages ISBN: 0-471-09849-3 7" x 10" \$14.95

FAST BASIC is in part a misnomer, since the author introduces two distinctive sets of methods in order to speed the execution of BASIC programs. The first, which he terms "Controlled BASIC," is intended

primarily for the advanced beginner, and relies heavily on PEEK and POKE statements. The second, "Fast BASIC," allows the intermediate programmer to write simple machine language subroutines which in turn call ROM subroutines. Controlled BASIC and Fast BASIC do not represent systematic approaches to programming, but rather combine a diverse (if chaotic) array of procedures designed to produce more efficient BASIC code.

The procedures available within BASIC to shorten program execution include packing variables into fewer bytes of memory than the BASIC interpreter would ordinarily allow; POKEing values into protected memory in order to pass them to a second program; merging separate programs; or PEEKing to determine if a given key has been pressed on the keyboard. Much of Gratzner's focus in Controlled BASIC is not so much on methods to make BASIC run more quickly as on making BASIC more versatile (sending output to the printer with the PRINT statement, for example) or averting catastrophe (preventing a program from hanging if the printer is not turned on or restoring a BASIC program supposedly lost when the NEW command was entered). Although much of the material is interesting, Gratzner's disorganized mode of presentation and the haphazard character of the procedures themselves make controlled BASIC somewhat less than enjoyable to learn.

FAST BASIC is similarly disappointing, if only because of its superficial treatment of Z80 assembly language programming and its hasty explanation of incorporating machine language programs into BASIC programs. The general techniques outlined by Gratzner (including the use of ROM subroutines to perform repeated arithmetic operations or to sort string variables and the replacement of FOR/NEXT loops with assembly language subroutines) do promise enormous reductions in program execution time. By providing a better organized and more comprehensive discussion of his topic, the author could have gone far beyond this to consolidate his readers' grasp of advanced programming principles.

FAST BASIC does fulfill its limited goal of making BASIC programs run faster. It is much less successful, however, in permitting the reader to combine efficiency with a mastery of BASIC programming.

THE GENIE IN THE COMPUTER

Easy Basic Through Graphics-TRS-80 Edition

Rachel Kohl, Laura Karp and Ethan Signer

Level: Novice Rating: **85**

Wiley 1982 Paper
169 Pages ISBN: 0-471-87049-8 9" x 11" \$12.95

BASIC instructors have long realized the merit of teaching the language through graphics—the computer's visual output provides a

powerful reinforcement for learning to code. The authors of **THE GENIE IN THE COMPUTER** have designed a course in elementary BASIC which revolves around embellishing a simple screen "face" (the book's "genie"). A PRINT statement places the simplest face on the monitor. A subroutine using the RND function makes the genie "wink." String variables, the CHR\$ function and data statements all combine to build a repertoire of genie features.

A workbook format is adopted to step the novice programmer through each successive addition to the original graphics program. A short narrative which explains a new BASIC command or function is followed by coding exercises which further modify the single program forming the core of each of the book's seven chapters. "The Genie's Appendix" provides summaries of TRS-80 Level II BASIC short-cuts, editing commands, error messages and special combinations of the CHR\$, STRING and PRINT commands.

Since many of the graphics instructions used in this book are specific to the TRS-80, users of other microcomputer systems may have difficulty learning BASIC from its coding examples. This is unfortunate; **THE GENIE IN THE COMPUTER** is one of the more original presentations of introductory BASIC programming.

A GUIDE TO PROGRAMMING IN LEVEL II BASIC

Bruce Presley

Level: Novice Rating: 90

Van Nostrand Reinhold	1982	Paper
190 Pages	ISBN: 0-442-25892-5	9" x 11" \$12.95

See our review of **A GUIDE TO PROGRAMMING - IBM PERSONAL COMPUTER** on page 275.

HOW TO WRITE A TRS-80 PROGRAM

Ed Frank

Level: Novice Rating: 50

DataMost	1982	Paper
218 Pages	ISBN: 0-8359-2992-2	5" x 8" \$14.95

HOW TO WRITE A TRS-80 PROGRAM contains very little BASIC code. Rather, it treats programming methodologies; from program 'ideas' through the design and planning stages, structured and modular program writing is discussed.

A highly illustrated, conversational text guides the programmer already familiar with BASIC syntax through elementary flow charting, pseudo-code overviews and modular coding techniques. A checkbook application is developed utilizing each of these methods (indeed, the only machine-specific aspect of this book is the BASIC dialect used in this application). In addition, common program errors and their quick detection are covered in a "testing and debugging" section. (These include unbalanced if/else's, improper "fielding" of file data and incorrectly sequenced file opens and closes). A concluding chapter discusses the necessity of clear documentation and offers suggestions for its standardization and readability both within and without the actual program.

The generally informative text suffers, however, from the content and quality of the cartoons interspersed throughout the book. (While illustrations can add a great deal to a microcomputer book, rarely are they so bad as to warrant special attention.) "Rhonda Routine," an overdone playmate-type character, parades different subject placards through the check writing program. A professor points to a bikini-clad woman in order to differentiate physical from logical structure. Still another character thinks of a partially dressed woman in order to take a picture of the cartoon of his thought, in a display of sexism not even remotely related to any microcomputer topic in the book.

HOW TO WRITE A TRS-80 PROGRAM simultaneously contains a pleasant rambling discourse on the use of efficient BASIC programming techniques and, perhaps, the most incongruous, blantly sexist illustrations in microcomputer literature. We recommend that, in this case, the baby isn't worth the bath water.

I SPEAK BASIC TO MY TRS-80

Student Text

Aubrey B. Jones, Jr.

Level: Novice Rating: **45**

	Hayden	1982	Paper	
219 Pages	ISBN: 0-8104-6174-9		9" x 11"	\$7.45

See our review of I SPEAK BASIC TO MY APPLE on page 264.

INTERMEDIATE PROGRAMMING FOR THE TRS-80 MODEL I

David L. Heiserman

Level: Intermediate Rating: **60**

Sams 1982 Paper
238 Pages ISBN: 0-672-21809-7 6" x 9" \$9.95

The computer programmer, Heiserman writes, is "chronically afflicted with a desire to acquire more equipment or find ways to use existing schemes more effectively." (5) For those whose learning curves have reached a temporary plateau and who are eager to learn how to exploit the capabilities of their computers more fully, Heiserman has written this book. But the TRS-80 owner will continue to suffer from a static learning curve even after reading Heiserman's book.

The discussion of BASIC language programming techniques concentrates on a variety of methods for formatting output to a CRT and on handling keyboard input, on memory organization (including the way BASIC programs are stored in memory), and on calls to machine language subroutines. Since much of this material is indeed beyond the scope of most introductory texts, INTERMEDIATE PROGRAMMING promises to expand the TRS-80 user's programming capabilities enormously. The text, moreover, is lively, well-written and informative. But although Heiserman's discussion of each point is detailed and thorough, he depends on brief program listings to illustrate them. These lines of BASIC code "are meant only to illustrate—to emphasize—one or two key points. So not many of the programs presented here are, in themselves, very useful." (14) As a result, the reader is left with no idea how to apply these new techniques.

Heiserman also uses his discussion of the BASIC USR function as a means of introducing assembly language programming (with the aid of Radio Shack's T-BUG monitor and the EDTASM EDITOR/ASSEMBLER). The inclusion of five chapters on assembly language programming is altogether incongruous, since the author does not make the least attempt to teach the reader how to program in Z80 machine or assembly language. Instead, he warns that "you will find no details concerning the actions of these instructions nor any special discussions of fundamental machine-language programming." (87-88) In this case, however, it is unclear why Heiserman has even bothered to include the last two-fifths of his text.

INTRODUCTION TO GRAPHICS

John P. Grillo and J.D. Robertson

Level: Novice Rating: **85**

Wm. C. Brown 1981 Paper
133 Pages ISBN: 0-697-09953-9 8" x 11" \$15.95

Three levels of graphics displays are discussed in this book: line printer, character and pixel graphics. The author defines "line printer" graphics as the display of sequential print statements on a monitor or printer; "character" graphics as the use of the TRS-80's special graphics character set in conjunction with advanced BASIC commands; and "pixel" graphics as the manipulation of the "bits" of visual information to produce medium resolution pictures. Within these categories, coding techniques are explored which use two-dimensional table data to "load" pictures, and which bestow "motion" to a stationary display. Examples of these techniques range from the ridiculous (a picture of the Charlie Brown character Woodstock) to the sublime (a movie marquee which surrounds its title with "moving" lights).

Each of the thirty-eight programs included in INTRODUCTION TO GRAPHICS is presented as the solution to some graphics-related problem. Displays of sample output are followed by the authors' suggestions for possible reader modification of certain graphics routines within programs. All programs are written in TRS-80 Level II BASIC.

The process of translating these programs to other BASIC dialects is complicated by the dependence of many of these programs upon TRS-80-specific commands and functions. While most of these commands can be mimed by some combination of equivalent BASIC statements, it is a problem which the authors recognize but never directly address.

INTRODUCTION TO GRAPHICS is an easy-to-use survey of different graphics techniques, highlighted by original applications programs. The user of TRS-80 Level II BASIC will find it a more than adequate graphics introduction, while programmers fluent in other BASIC dialects will be forced to rewrite a number of TRS-80 BASIC commands before realizing the book's true utility.

LEARNING TRS-80 BASIC FOR MODELS I, II/16, AND III

DAVID A. LIEN

Level: Novice Rating: **95**

CompuSoft 1982 Paper
544 Pages ISBN: 0-932760-08-2 7" x 9" \$19.95

The author of both the original TRS-80 LEVEL I USER'S MANUAL and LEARNING LEVEL II, has created a comprehensive BASIC introduction that spans the TRS-80 product line. As such, it is one of the few BASIC tutorials which treats Model II Level II BASIC. No familiarity with any BASIC dialect or the TRS-80 is assumed.

Emphasis is on the non-disk functions of BASIC, with the main narrative using Model III BASIC for programming examples and exercises, and marginal "asides" annotating coding variations for other TRS-80 BASIC dialects. Readers familiar with the "Lien" style will immediately recognize the characteristic blend of humor and text in a relaxed yet thorough presentation.

Initial chapters treat boot-up procedures for each model of TRS-80, simple BASIC commands, and the use of the editor. Through machine exercises, programming examples and "checkpoint" reviews, BASIC is slowly and painlessly developed. Strings, graphics (non-Model II) and advanced functions like USR, VARPTR and INP are also discussed. A special section entitled "Program Control" contains some uniquely Lienesque suggestions for structured programming and debugging techniques. The reader is further admonished to practice "defensive programs" (a Murphy's Law follow-up on possible input errors) and to avoid "creeping elegance" (not a Southern vine but the novice programmer's tendency to embellish code needlessly).

The appendices alone contain enough information for a separate reference manual. Topics include expansion interface considerations for the Model I, special Model II and Model III BASIC functions and an annotated glossary of TRS-80 BASIC error messages.

With this guide, and his already considerable success as the author of THE BASIC HANDBOOK and LEARNING BASIC FOR THE IBM PERSONAL COMPUTER, David Lien has established a reputation as one of the premiere educators in BASIC programming literature. We recommend this current offering along with all of his previous work.

MICROSOFT BASIC DECODED & OTHER MYSTERIES FOR THE TRS-80

TRS-80 Information Series Vol.II

James Lee Favour

Level: Advanced Rating: **90**

IJG Computer Services 1981 Paper
 310 Pages ISBN: 0-936200-01-4 8" x 11" \$13.95

All the volumes in the TRS-80 Information Series are of extremely high quality. Favour's MICROSOFT BASIC DECODED, which features a disassembled source code listing of the TRS-80 Model I's Level II ROM, is an especially noteworthy member of the series.

Level II incorporates a rudimentary operating system and the TRS-80 BASIC interpreter. The brief (57-page) discussion with which Favour introduces his disassembled listings focuses on the relationship of Level II to TRSDOS, a second operating system for computers with disk drives, as well as on Level II itself. The major portion of the book, however, is devoted to a listing of each instruction and operand contained in the Model I's 12K ROM, complete with Favour's own annotations. It is this feature, and the enormous amount devotion on the part of the author that it reflects, which give the volume its unique value in microcomputer literature.

Unfortunately, the production quality of the book is not commensurate with its overall quality. Favour's brief text has been hastily proofread so that typographical errors abound; at times, the meaning of phrases or even whole sentences is garbled. Favour's commentaries are on the reverse side of the page from the source code listings which they accompany, an inconvenient feature.

This should not, however, obscure the inherent merits of MICROSOFT BASIC DECODED. For assembly language or even BASIC programmers who include calls to ROM subroutines in their code, this book is indispensable.

MOSTLY BASIC-APPLICATIONS FOR YOUR TRS-80/BOOK 1

Howard Berenbon

Level: Novice Rating: **75**

Sams 1980 Paper
 168 Pages ISBN: 0-672-21788-0 9" x 11" \$12.95

MOSTLY BASIC -APPLICATIONS FOR YOUR TRS-80/BOOK 2

Howard Berenbon

Level: Novice Rating: **90**

	Sams	1981	Paper	
217 Pages	ISBN: 0-672-21865-8		9" x 11"	\$12.95

See our review of MOSTLY BASIC: APPLICATIONS FOR THE APPLE II, Books 1 and 2, on page 268.

PRACTICAL BASIC PROGRAMS-TRS-80 LEVEL II EDITION

Lon Poole

Level: Novice Rating: **90**

	Osborne	1982	Paper	
172 Pages	ISBN: 0-931988-57-5		9" x 11"	\$15.99

See our review of PRACTICAL PASCAL PROGRAMS on page 210.

PROGRAMS FOR BEGINNERS ON THE TRS-80

Fred Blechman

Level: Intermediate Rating: **85**

	Hayden	1981	Paper	
150 Pages	ISBN: 0-8104-5182-4		7" x 10"	\$8.95

"If BASIC were a foreign language," writes the author, "this might be considered as a course in conversational BASIC, not in BASIC vocabulary, syntax and grammar." The "conversation" revolves around twenty-one sample programs, explained line-by-line, in a thoroughly pragmatic approach to learning BASIC programming techniques. Both Level I and Level II TRS-80 BASIC are covered. Novice programmers interested in developing expertise in a particular technique can consult a master index which cross-references each program by the function it employs (e.g., timing loops, entry-error typing or screen painting).

The "beginner" of the book's title should have some familiarity with BASIC programming, despite the author's claims to the contrary. The

book's statement-by-statement "parsing" of a BASIC program provides valuable insights into how different commands interrelate to achieve some result, but contains no commentary on the "vocabulary, syntax and grammar" of the original individual commands.

Each application program begins with a brief commentary on the BASIC topics to be covered, as well as a detailed explanation of any formulae used to calculate results. The actual code follows. An in-depth analysis then walks the reader through each program step, explaining the flow and manipulation of data, and even suggesting user modification. An Interest Calculation and Tabulation Program, for example, computes the amount of interest for a given principle according to a number of user-defined variables (time, rate, etc.). The author's narrative highlights the conversion of formulas into BASIC code. Additionally, both the Level I and Level II techniques for handling integer exponents are explained.

Other sample programs include "Spirangle" (a graphics application), "Mortgage Loan Amortization," "B-I-N-G-O" and "Racing Alphabet." Appendices treat Level I/Level II conversion guidelines, cassette loading aids and Level II "Slick Tricks."

PROGRAMS FOR BEGINNERS ON THE TRS-80 is probably best suited for the intermediate BASIC programmer who understands the discrete "notes" of the BASIC octave but has difficulty using them to compose a coherent melody. We do not recommend it to beginners with little prior exposure to TRS-80 BASIC.

TECHNIQUES OF BASIC

John P. Grillo and J.D. Robertson

Level: Intermediate Rating: **85**

	Wm. C. Brown	1981	Paper	
256 Pages	ISBN: 0-697-09951-2		9" x 11"	\$18.95

Although contemporary home computers provide more computing power than the mainframes of the 1960s, many users content themselves with balancing their checkbooks, playing games and, in general, failing to fully exploit the full capability of their machines. Implicitly, Grillo and Robertson see this as a failure of education; "as educators...we are continually surprised at the variety of people who exhibit a talent for this science, or art, or craft (of programming)." (xiii) To redress this problem, they have written a work which attempts to pick up where the more elementary (and frequently oversimplified) introductory BASIC texts leave off. Their book is written for the microcomputer user who has at least a passing familiarity with BASIC; it is also, unfortunately, oriented heavily toward TRS-80 users, although owners of other machines can still profit enormously from reading it.

TECHNIQUES OF BASIC examines a broad spectrum of topics, including programming techniques, graphics, input/output and disk file manipulation. The quality of the authors' treatment of these subjects is highly uneven. Their discussion of PEEK and POKE statements, for example, is not much more informative than that of those (comparatively few) introductory texts which consider these functions to be within the grasp of the novice. In other cases, their presentation of new material (e.g., the VARPTR function, which returns the memory address of a variable) lacks the detail necessary to permit the reader to realize its full potential. Finally, in still other cases, Grillo and Robertson confine themselves almost exclusively to listing BASIC programs, from which the reader is evidently expected to draw the knowledge necessary to expand his or her programming ability; the chapter on TRS-80 graphics is of this type.

At the same time, large portions of the text are uniformly excellent. The initial chapter, which notes clearer methods of decision-making and branching under Extended BASIC and emphasizes the construction of Boolean variables, is extremely informative. While the chapter on sequential files leaves a great deal to be desired, the two chapters devoted primarily to the far more complex random access files make a major contribution toward enabling the user effectively to manipulate disk drives, which have traditionally been the most underutilized component of microcomputer systems. Finally, Grillo and Robertson's stress on "conversational" (i.e., user-friendly) programming, effective internal and external documentation, and more generally on structured programming techniques, clearly and persuasively introduces the reader to a set of methods which result in less programming frustration and more effective code. It is because of these sections that TECHNIQUES OF BASIC can help transform the novice programmer into an intermediate one.

TRSDOS 2.3 DECODED & OTHER MYSTERIES

TRS-80 Information Series Vol.VI

James Lee Favour

Level: Advanced Rating: **95**

IJG 1982 Paper
304 Pages ISBN: 0-936200-07-3 8" x 11" \$29.95

Whereas James Favour's MICROSOFT BASIC DECODED features a practical and operational approach to the TRS-80 BASIC interpreter, his interest in decoding TRSDOS (the disk operating system) for the MODEL

I is almost exclusively theoretical. His painstaking work in disassembling (i.e., converting from machine language to assembly language mnemonics) and then interpreting all of TRSDOS merits both praise and respect; this volume makes a very important contribution to the literature not only on the TRS-80 Model I, but also on operating systems generally.

TRSDOS 2.3 DECODED first provides the reader with an overview of the Model I microcomputer system, of operating systems in general and of TRSDOS in particular. Farvour then focuses on each of the components of TRSDOS: its nucleus, which is continually stored in approximately 7K of memory; its overlay modules (such as error-processing, debugging, and file opening and closing routines), which are brought into memory as needed; and its utility programs (like COPY or DUMP) which are also moved into memory when needed. Since TRS-80 computers with disk drives actually have two operating systems (TRSDOS and Level II), the relationship between them is also examined. Farvour then goes far beyond a general analysis of TRSDOS by presenting a documented assembly language listing of the entire operating system; it is this above all which makes the book invaluable.

Although Farvour never specifies how his work is to be used, a number of applications suggest themselves. Such a treatment is of obvious importance to systems programmers who are concerned with operating system design. But it is relevant as well to TRS-80 users; when the actual code of an operating system is known, its contents can be changed and modified to suit the needs of the user. Finally, Farvour's book appeals to those whose interest in operating systems has evolved from simply manipulating their features to attempting to understand how they work. We highly recommend TRSDOS 2.3 DECODED to all those whose interest in operating systems has reached this advanced level.

TRS-80 ASSEMBLY LANGUAGE MADE SIMPLE

Earles L. McCaul

Level: Intermediate Rating: **30**

Sams 1981 Paper
190 Pages ISBN: 0-672-21851-8 5" x 9" \$12.95

By some cruel twist of fate, the TRS-80 user who invests \$12.95 in TRS-80 ASSEMBLY LANGUAGE MADE SIMPLE will find that the book not only fails to make assembly language any simpler, but that it is not really about assembly language programming at all. McCaul has written

this book for the user "who is interested in writing short assembly language programs, but is not necessarily interested in learning assembly language programming. The goal of this book is to show how to USE assembly language programming, not how to become an assembly language programmer."(3)

In an effort to meet his broader goal of not teaching the user how to write assembly language programs, McCaul includes brief discussions of assemblers, hand assembly of source code, the use of the T-BUG monitor, Z80 architecture and the Z80 instruction set. These truncated discussions are of no particular value to anyone at all. The chapters which occupy almost the entire first half of an already short book (157 pp., without appendices) have probably been included because either the author or publisher did not consider a 90-page work marketable.

In the last half of his book, (pp. 70-157) McCaul finally turns to his central topic: the use of Level II BASIC ROM subroutines either in BASIC programs, or in machine language "programs" (which themselves do nothing other than call these subroutines). For this purpose, the author briefly examines the TRS-80 memory map (pp. 70-79), lists a wide range of ROM subroutines and includes a rather confused discussion of passing parameters between main programs and ROM subroutines. A far superior treatment of these topics can be found in Lewis Rosenfelder's BASIC FASTER AND BETTER (see our review).

We must concede that McCaul's book admirably succeeds in fulfilling its negative goal of not teaching the reader assembly language programming. Such success, it might be argued, merits a higher rating. Instead, we have chosen to supplement our abysmally low rating with the recommendation that this badly over-priced and mistitled volume be religiously avoided.

TRS-80 BASIC A Self-Teaching Guide

Bob Albrecht, Don Inman and Romon Zamora

Level: Novice Rating: 40

Wiley 1980 Paper
351 Pages ISBN: 0-471-06466-1 8" x 10" \$10.95

In their introduction to this volume in the Wiley Self-Teaching Guide series, Albrecht, Inman and Zamora succinctly formulate their approach to teaching BASIC programming to beginners:

We believe that learning to program should, and can, be an enjoyable experience. We believe that computer terminology and concepts can be introduced within a framework of fun and exploration, and that when we do this, TRUE learning takes place.(v-vi)

Whatever the merits of this argument, however, the authors' attempt to put it into practice has succeeded in utterly confusing fun and games with education. As a result, Albrecht, Inman and Zamora have written the Ultimate book for those who want to go through the motions of learning BASIC when they in fact have no serious intention of doing so.

While claiming to use "fun" as a means of promoting learning, the authors have instead relied on some not-very-funny attempts at jokes and wisecracks to conceal a variety of abuses. Their flippant writing style serves as a means of avoiding the task of carefully organizing the material which they are presenting. Most obvious is their inclusion of a discussion of the PRINT @ statement in the middle of a section dealing with FOR-NEXT loops; but even more irritating is their constant tendency to mention a BASIC statement or a programming technique, only to defer any in-depth discussion of it by noting that "all will be revealed." Secondly, this emphasis on "fun" de-emphasizes the fact that the authors are teaching a highly simplified version of BASIC. After introducing a single example of exponential notation (the number $9.22339e + 18$), for example, the authors urge the reader, "Don't worry about floating numbers now." (33) BASIC subroutines are discussed inadequately (see page 83), and functions are overlooked altogether, presumably because they would be too complex for the average reader.

In a literature which is all too frequently marked by the arrogance, condescension and elitism of its authors, this work stands out as one of its worst examples.

TRS-80 COBOL

Robert T. Grauer

Level: Novice Rating: **90**

Prentice-Hall 1983 Paper
305 Pages ISBN: 0-13-931204-8 9" x 11" \$16.95

COBOL compilers designed for specific microcomputer systems have not yet enjoyed the widespread popularity of their BASIC, FORTRAN and Pascal counterparts. The consequent lack of adequate treatments of microcomputer COBOL is a need directly addressed by TRS-80 COBOL. Grauer does an excellent job of introducing the Model II or Model III user to the richness and power of this high-level programming language.

The mystery surrounding COBOL programming is immediately dispelled by the dissection of a complete COBOL program in the book's first

chapter. The four main divisions and general program structure are noted. While certainly not a system-specific language, COBOL nonetheless does depend heavily upon the operating system under which its programs execute. The book's early discussion of elementary coding considerations is therefore paralleled by treatments of TRSDOS and CEDIT.

After a review of the steps required to enter, compile, debug and execute a sample program, Grauer begins a detailed discussion of COBOL programming. Data and Procedure Divisions are naturally emphasized, with demonstration programs explained using pseudocode and hierarchical diagrams (the author frowns on the over-use of flowcharts).

That Grauer is an experienced COBOL programmer who has spent more than a little time in the mainframe trenches is evident in the pragmatic tone of this portion of his programming presentation. His detailed analysis of debugging techniques and advice on proper structured programming style supplement a clear, well-developed introduction to the nuances of COBOL syntax. Arithmetic, conditional and I/O statements are all slowly added to the reader's COBOL vocabulary. Control breaks, sequential and non-sequential file access methods, and the use of subprograms are topics which employ this vocabulary in a number of programming examples.

All program listings are available on diskette. Appendices include an answer key to chapter exercises and quick reference guides to COBOL reserved words and statement syntax.

TRS-80 COBOL is a comprehensive and, at the time of this writing, unique handbook on microcomputer COBOL. We highly recommend it to mainframe programmers interested in a painless transition to personal computer coding.

TRS-80 COLOR BASIC

Robert L. Albrecht

Level: Novice Rating: **80**

Wiley 1982 Paper
 378 Pages ISBN: 0-471-09644-X 7" x 10" \$9.95

TRS-80 COLOR BASIC provides the novice user with an introduction to minimal BASIC for the TRS-80 color computer; it does not treat the commands available in extended BASIC. Although the book features the frame-by-frame instructional technique typical of Wiley Self-Teaching Guides, it departs somewhat from their traditional organization by presenting sound and color commands from the outset and treating

graphics in the middle of the text. This difference in emphasis is an appropriate one, since most users buy the Color Computer for both its graphics and sound capabilities.

But the most distinctive feature of the text is that the condescension with which the reader is treated in some of the Wiley BASIC introductions has been substantially downplayed in this volume. It does retain, however, the rather slow, repetitive and detailed approach to the BASIC statements it covers. Only the section on sound was far too cryptic: instead of presenting a listing of the codes needed to produce each note, it simply encourages the reader to try out a few numbers to see what might happen (an appendix does, however, provide fuller detail).

Admittedly, TRS-80 COLOR BASIC introduces the reader to only the bare essentials of the language, as well as to those features which account for the color computer's appeal. This is largely consonant with the needs of color computer users, and especially those who depend on its series of plug-in cartridges. We recommend this work to users who are somewhat intimidated by their new computers, as well as to those who want to try their hands at writing a few simple BASIC programs.

TRS-80 DISK & OTHER MYSTERIES

TRS-80 Information Series Vol.1

H.C. Pennington

Level: Advance Rating: **90**

IJG 1979 Paper
 136 Pages ISBN: 0-936200-00-6 9" x 11" \$22.50

In language as colorful as the cover of his book, H.C. Pennington has compiled the definitive treatment of the TRS-80-formatted diskette. Using commercially available diskette utilities, the author explains techniques for scanning and updating diskette data.

After a preliminary discussion of decimal to hexadecimal conversion, Pennington introduces SUPERZAP, a software product of APPARAT, Inc. of Denver, Colorado. SUPERZAP reads raw data from diskette, regardless of its protect status; it is Pennington's main tool in recovery from diskette error conditions. The reader is stepped through its use in copying, relocating, modifying and clearing diskette sectors. This demonstration is followed by the author's opinionated evaluations of comparable diskette utilities (RSM-2D, MONITOR, DEBUG, DIRCHECK, LMOFFSET) as well as TRS-80 operating systems (TRS.DOS 2.3, VTOS 4.1, LDOS 5.0, ULTRA DOS, DOS PLUS, NEW DOS 2.1, NEWDOS/80).

Pennington then describes both the structure of the diskette directory and the formatting differences among the various TRS-80 file types.

These discussions are marked by a byte-by-byte analysis of the raw data as it appears in a hex dump of a diskette sector. Pennington explains the role of relative byte positions in assisting the programmer to discern a particular file format (except for SYSTEM files, the directory gives no clue as to file format).

The development of this material in considerable detail leads to the discussions of data recovery techniques which make this text invaluable for the advanced TRS-80 programmer. Directories, whether "clobbered" or "unreadable," are miraculously revived; ASCII, BINARY and ELECTRIC PENCIL files are similarly recovered. Pennington's diskette pharmacopia extends even to electrically and physically damaged diskettes.

Each one of these topics is presented in one of the most "conversational" tones in the literature (it is not uncommon for the operating system to have "a bitch of a time" with diskette searches). The author's own illustrations are particularly poignant in supplementing this matter-of-fact narrative. The only qualification that can be made of this informative, popular text concerns its production quality: it is incomprehensible that a text whose fifth printing attests to its intrinsic value should continue to be published with a reader-unfriendly typewriter typeface.

TRS-80 DISK AND OTHER MYSTERIES is an unconventional discourse on diskette esoterica. It is required reading for any programmer interested in a deeper understanding of diskette formatting and data manipulation.

TRS-80 FOR KIDS FROM 8 TO 80 Volume I

Michael P. Zabinski

Level: Novice Rating: **80**

Sams 1982 Paper
 133 Pages ISBN: 0-672-22046-6 9" x 11" \$9.95

The "school-is-out" flavor of TRS-80 FOR KIDS FROM 8 TO 80 is perhaps due to the role of its author as the founder and director of the National Computer Camps. The book is purportedly "the result of his search for a suitable text for beginners to learn programming."

Adopting a hybrid Sesame Street/Highlights for Kids approach to computers, the specific BASIC dialects it discusses are Model I and Model II TRS-80 BASIC. Standard paragraphed narrative is the exception rather than the rule. Instead, small "capsules" of programming information engage the reader's attention through exercises and cartoons. "Brain food" sections remind the young reader about coding rules.

Crosswords reinforce keyword retention. "Discovering exercises," "experiments" and "Checkpoints for review" provide some special insight into computers or programming.

A concluding "Programs to go" section includes complete programs like "Etch-a-sketch" and "Shoot the Duck." These programs contain many BASIC commands and functions never mentioned in Volume I, and as such, emphasize the book's role as only the most elementary of BASIC introductions.

Though rudimentary BASIC is treated in a light, illustrated format, TRS-80 FOR KIDS FROM 8 TO 80 is a primarily conceptual introduction to BASIC programming for elementary school children. Despite this qualification, it is an excellent supplemental teaching aid for a pre-teen programming course.

TRS-80 GRAPHICS

For The Model I & Model III

David A. Kater and Susan J. Thomas
Level: Intermediate Rating: 85

Byte 1982 Paper
287 Pages ISBN: 0-07-033303-3 9" x 11" \$12.95

TRS-80 GRAPHICS is a comprehensive introduction to the Model I and III graphics repertoire. Written for the intermediate BASIC programmer, the book is divided into two main sections: the first treats the full TRS-80 graphics set, including machine language graphics; the second demonstrates its use in a number of graphics applications.

After a preliminary review of TRS-80 editor and BASIC commands, the authors delve directly into the use of strings in PRINT statements to produce the simplest form of screen displays. Specific graphics commands like SET, RESET and POINT are introduced. The memory maps of the Model I and Model III are explained as PEEKs and POKEs further manipulate simple graphics patterns. Considerations of speed in the animation of graphics displays leads to the incorporation of machine language routines into BASIC programs via the USR function. A few advanced coding techniques treat validating user input in interactive graphics programs and avoiding time delays caused by the automatic reorganization of string pool space in memory.

The second half of the book is devoted exclusively to applications examples. Programs are categorized into seven graphics areas: Geometric Shapes, Statistics, Computer Assisted Instruction, Visual Aids, Games and Animations, Figure Animations, and Art. Most programs are quite succinct (less than twenty statements), yet, using optimal coding tech-

niques, manage to produce elegant, effective displays. One program, entitled "Flashy," incorporates a machine language subroutine and is particularly original.

A series of appendices cover graphics and special character tables, graphics-related memory locations and a screen dump patch which facilitates printing displays on a TRS-80-compatible printer.

TRS-80 GRAPHICS is a well-illustrated, clearly presented text on the full graphics capabilities of the TRS-80. We recommend its purchase to the intermediate BASIC programmer familiar with assembly language code who is ready to sample everything from low-resolution to animated graphics coding techniques.

TRS-80 MORE THAN BASIC

John Paul Froehlich

Level: **Advanced** Rating: **90**

	Sams	1981	Paper	
220 Pages	ISBN: 0-672-21813-5		5" x 8"	\$10.95

What does an assistant professor in the Department of Electrical Engineering at the University of Hartford do when he becomes justifiably frustrated with the assembly language development systems available on the TRS-80, Models I and III? Two things: he builds his own system, and writes a book about it.

Froehlich's book treats the alteration of these machines, either by loading object code from cassette or disk or by physically removing ROM and replacing it with a pre-programmed EPROM. What results is something the author claims was not commercially available at the time of his writing—an efficient assembly language development system for the TRS-80.

Though the author never explicitly treats Z80 architecture (he instead recommends a few texts), the book does include detailed discussions of edge-card connections, port assignments, graphics capabilities and keyboard input for both the Model I and III. A mini-manual describes the naming conventions and functions specific to the "Froehlich Monitor." Assembly listings are provided for readers who intend to simply load the monitor from auxiliary storage. Additional EPROM listings are included for those die-hards who wish to follow the author's preferred path of removing and replacing the entire TRS-80 ROM (the fact that an EPROM pre-programmed with this code is commercially available will probably do little to dissuade users excited by the prospect of microprocessor incendiaries).

TRS-80 MORE THAN BASIC is an unusual approach to learning about assembly language development systems. If the advanced pro-

grammer can manage to overcome the momentary squeamishness that the complexity of Froehlich's project elicits, he or she will find an absolutely fascinating discourse on microprocessor architecture, assembly language and the virtues of doing it oneself.

WRITING BASIC ADVENTURE PROGRAMS FOR THE TRS-80

Frank DaCosta

Level: Intermediate Rating: **95**

222 Pages Tab 1982 Paper ISBN: 0-8306-1422-2 5" x 8" \$9.95

Like the adventure programs it describes, WRITING BASIC ADVENTURE PROGRAMS FOR TRS-80 lets the curious, stalwart reader explore the nooks and crannies of its chapters for treasures of novel coding techniques and insights into structured BASIC programming. Rarely do the contents of a book rise to the expectations proclaimed by its jacket copy; this book is such an exception.

Adventure programs are a remarkably well-defined, semi-literary game genre in which a player is placed in some labyrinthine scenario filled with obstacles, beasts and treasure. The computerized version of this game, as the author notes, dates back to the Crowther and Woods PDP-10 program which, in the last decade, spread across university computer centers. DaCosta extracts the major components of the typical game (rooms, player commands, creatures, etc.) and translates them into numerical equivalents that can be used as input to a BASIC program.

An elementary familiarity with BASIC is assumed. Since it is the author's stated intention to optimize execution time and conserve memory space without resorting to assembly language, more than token attention is given to incorporating structured programming techniques into the adventure code. An initialization routine sets the micro stage. A main "executive" section manages the activation of all subroutines. One ingenious technique used by DaCosta involves a routine for manipulating the TRS-80 data pointer in order to speed the program's access to individual data elements (without using repetitive READs and RESTORES). Unfortunately, some of the more original coding short-cuts are specific to the TRS-80 (the above routine, for example, uses locations found only in the TRS-80 memory map).

Two complete programs are discussed in WRITING BASIC ADVENTURES FOR THE TRS-80. The first, "Basements and Beasties," is a non-graphic program around which the first ten chapters of text revolve. The

second, "Mazies and Crazies," uses low-resolution TRS-80 graphics and receives a summarized treatment, including program listing, in the book's last three chapters.

Frank DaCosta has written a fascinating combination of adventure program analysis and introduction to advanced BASIC programming techniques. Game enthusiasts looking to sharpen their BASIC skills will enjoy reading WRITING BASIC ADVENTURES FOR THE TRS-80, regardless of the particular microcomputer they own.

80 PRACTICAL TIME-SAVING PROGRAMS FOR THE TRS-80

Charles J. Carroll

Level: Novice Rating: **85**

	Tab	1982	Paper	
252 Pages	ISBN: C	06-1293-9	5" x 8"	\$10.95

The rationale behind the publication of 80 PRACTICAL TIME-SAVING PROGRAMS is not that the programs are fun, or that they provide shining examples of imaginative or brilliant BASIC coding; instead Carroll has attempted to supply ready-assembled programs that might spare the user the loss of "precious time... spent looking for that elusive formula or its derivation"(5) and then translating it into BASIC statements.

The book's eighty primarily short programs are divided into five categories: numbers (polar/rectangular conversions, simultaneous equations, conversions to and from base 10), finances (compound interest, loan payments and balance), statistics (permutations, means, correlations), electronics (matching of antennae to transmission lines, attenuators) and geometry (hyperbolic functions, triangles). Although all have been written as stand-alone programs, they can be converted into sub-routines with minimal difficulty. Non-TRS-80 users, moreover, can also use these brief programs, since Carroll has almost entirely avoided using TRS-80-specific statements.

Although the inclusion of such commonly known formulae as factorials and arithmetic means indicates that Carroll did not always succeed in striking a balance "between simple, mundane programs and more complicated, obscure ones,"(5) this collection of programs is nevertheless both practical and useful. They could have been more useful still if Carroll had defined the variables used in the formulae, and if he had developed more refined algorithms for some of his programs (e.g., by having the chi-square program, rather than the user, calculate expected frequency distributions). But 80 PRACTICAL PROGRAMS is a valuable

addition to the libraries of those who use their microcomputers as number-crunching tools.

ATARI

THE ATARI ASSEMBLER

Don Inman and Kurt Inman

Level: Intermediate Rating: **90**

Reston 1981 Paper
270 Pages ISBN: 0-8359-0236-6 6" x 9" \$12.95

The authors of APPLE MACHINE LANGUAGE have written another introduction to machine and assembly language programming, in this case for users of Atari 400/800 computers. As they did in APPLE MACHINE LANGUAGE, the Inmans adopt a graduated approach to teaching assembly language programming, although it is one which reflects the unique features of Atari software: the reader rapidly progresses from machine language programming with a "BASIC operating system" (the authors having assumed that the reader is familiar with BASIC) to assembly language programming using the Atari Assembler Cartridge (which includes an editor, assembler and debugger). Use of this cartridge is also explained in detail.

The ATARI ASSEMBLER focuses on only a subset of the 6502 microprocessor's instruction set and addressing modes in order to insure the user's complete assimilation of the basics of assembly language programming; interrupt handling, bit manipulation with BIT, indirect, indexed indirect, and indirect indexed addressing, for example, are not treated in the text, while minimal attention is devoted to logical operations. Although the treatment of individual instructions is sometimes repetitive and confusing as well, the authors eventually succeed in developing sufficiently detailed explanations. At best, this repetition functions to encourage the reader's mastery of assembly language programming.

The actual programming applications treated in the text feature mathematical operations, which the authors explain clearly and (except in the case of two's complement arithmetic) fully. The text would be even better, however, had the Inmans devoted more attention to Atari-specific features; the memory map which is usually included for this purpose does not even show actual addresses.(28) Finally, the authors' "BASIC operating system" should have been more thoroughly explained for the benefit of beginning programmers, since it is both advanced and interesting in conception; it defines each op code or operand as an element of an

array in order to safeguard its location in memory, and determines the starting address of the assembly language program with the Atari ADR function.

The generally high quality of THE ATARI ASSEMBLER serves to highlight its limitations and to point out ways in which it could have been further improved. Nevertheless, the book succeeds in imparting the foundations of assembly language programming to the novice at the same time that it prepares him or her for more advanced programming texts. It is a work which we highly recommend.

ATARI BASIC

Robert L. Albrecht, LeRoy Finkel, Jerald R. Brown

Level: Novice Rating: 65

Wiley 1979 Paper
332 Pages ISBN: 0-471-06496-3 7" x 9" \$10.95

This introduction to BASIC programming for Atari 400 and 800 computers features the same frame-by-frame instructional technique and organization which characterizes the authors' other volumes in the Wiley Self-Teaching Guide series. In addition, much of the text and its examples are identical to those contained in their other BASIC introductions. To accommodate Atari users, however, Albrecht, Finkel and Brown have added a final chapter on Atari graphics and sound. Since most users purchase Atari computers for precisely these two features, this chapter should have appeared earlier—and even now, is not nearly as detailed a treatment of these topics as is mandated.

The volume is marred by a large number of condescending and trivial questions to the reader. Most egregious are questions centering on the return key. On page 12, the reader is asked to identify the key at which "a large black arrow" is pointing; the question is trite enough, and in this instance a typographical error omitting the arrow makes it insulting as well. On page 49, after the reader has already entered brief programs, the authors still insist upon emphasizing for their undoubtedly doltish readers the importance of pressing "return" after entering a line; their question is so obvious that many readers are likely to get it wrong! ATARI BASIC covers the bare rudiments of BASIC in a long (310-page), plodding text. Through pure repetition, the authors succeed in pounding familiarity with the material into the reader's brain. For an adult, such an approach is unjustified; for a child, it is hardly any better. We recommend ATARI BASIC to those intellectually limited readers who enjoy being insulted while they struggle with the process of learning.

ATARI BASIC QUICK REFERENCE GUIDE

Gilbert Held

Level: All Rating: **85**

Wiley 1982 Paper
8 Pages ISBN: 0-471-87041-2 6" x 12" \$2.95

See our review of IBM PC BASIC QUICK REFERENCE GUIDE on page 276.

ATARI GAMES & RECREATIONS

Herb Kohl, Ted Kahn, et. al.

Level: Novice Rating: **70**

Reston 1982 Paper
338 Pages ISBN: 0-8359-0242-0 7" x 10" \$14.95

Comic book illustrator Steve Olif has teamed up with the authors of ATARI GAMES AND RECREATIONS to present a guided tour of the Atari and its game-playing capabilities; the guides are the two alien characters who adorn the book's cover. The text itself is divided into three main sections: a BASIC introduction, a games section and a special graphics/color/sound section.

The BASIC introduction adopts the traditional approach of presenting increasingly complex programs which develop BASIC syntax and vocabulary. Detailed explanations of the individual parameters of the SOUND and SETCOLOR commands are missing from these early chapters, though both commands are used in sample game programs. The authors do give a detailed explanation of a special system of Atari graphics notation popularized by THE CODE WORKS, a company in Goleta, California. The introduction of a special notation and unexplained, complex commands (SOUND, SETCOLOR, PEEK, and POKE) in the middle of a presentation of elementary BASIC does little to enhance the clarity of this first section. (Only the alien guides remain unconfused.)

The second section, we are forewarned, contains many games and programs which "involve a deeper knowledge of programming than we can provide in this book." The authors advise that even if the games cannot be understood in light of their BASIC presentation, there is no reason not to enjoy them (a discussion which should have also preceded the first section). What follows is an anthology of popular number and word games: Nim and Anagrams are typical entries.

Color, sound and graphics combine in the final game section entitled "The Atari Special." Again, the emphasis is on recreation and not

education. Geometric patterns, high resolution drawings and a musical chord builder are subjects of some of the programs contained in these concluding chapters.

Contrary to its jacket copy, ATARI GAMES AND RECREATIONS is only marginally concerned with "introducing programming to the novice computer user." Rather, it is primarily a game program anthology which highlights the special arcade capabilities of the Atari 400 and Atari 800 personal computers.

ATARI SOUND AND GRAPHICS

A Self-Teaching Guide

Herb Moore, Judy Lower, and Bob Albrecht

Level: Novice Rating: 95

	Wiley	1982	Paper
234 Pages	ISBN: 0-471-09593-1	7" x 10"	\$9.95

The format of Wiley Self-Teaching Guides ranges from strict frame-by-frame programmed instruction to frameless conventional narratives. Normally, introductory texts adopt formats which present short sections of material interspersed with questions (i.e., programmed instruction), while books on more complex topics adhere to the standard chapter-by-chapter format. ATARI SOUND AND GRAPHICS is an exception to the Wiley rule: though purely introductory in content, this text is "Self-Teaching" only in that review questions mark the end of each chapter.

Through a series of machine-related exercises, the Atari owner learns BASIC. The authors' approach is simple and effective. Elementary graphics is introduced in one chapter, elementary sound in a subsequent chapter, and both are combined in a sound-and-graphics program. This "pendular" approach to teaching BASIC is repeated until more complex topics like the SETCOLOR command and musical string variables have been discussed.

Each of the Atari's graphics modes are explored in detail. Since no previous experience using BASIC is required, the introduction of graphics commands parallels explanations of rudimentary BASIC programming. Three-line programs typically consist of a FOR-NEXT loop bracketing a COLOR or PLOT command. A careful, deliberate pace combines with original program examples to make this simultaneous presentation easy to understand. Graphics modes 8 and 24, high resolution modes which require use of the SETCOLOR command, are saved for later chapters which treat advanced graphics functions.

That the Atari is a machine designed for arcade simulation is nowhere more apparent than in the chapters describing its sound-generat-

ing capabilities. It is to the authors' credit that they have explored this feature past the production of explosions and simple melodies to elementary music theory. The Atari's ability to mime musical instruments, for example, is approached by discussing the two factors which contribute to the "shape" of any musical note: attack (how fast the note reaches its maximum volume) and decay (the speed with which the loudness of the note falls off after reaching this maximum). This is also the only text which demonstrates BASIC's random number function by coding an aleatory music generator of which Stockhausen would be proud.

After swinging from graphics to sound to graphics enough times to cover most essential commands, the authors' pendulum comes to rest at a program which allows the reader/user to enter a sequence of notes stored by the Atari and simultaneously displayed on a staff and played as a melody. It is a most appropriate conclusion to the truly superior treatment of programming, graphics and sound found in ATARI SOUND AND GRAPHICS.

THE BOOK OF ATARI SOFTWARE 1983

Jeffrey Stanton, Robert P. Wells, et.al.

Level: All Rating: **95**

	Book Co.	1983	Paper	
347 Pages	ISBN: (None)		9" x 11"	\$19.95

THE BOOK OF ATARI SOFTWARE is, like THE BOOK OF APPLE SOFTWARE (see our review), one of the few critical guides to software packages. Not only are capsule reviews offered for each product, but evaluations according to standard criteria additionally assist the potential software buyer in making "...informed and intelligent decisions."

The general categories of software critiqued in this 1983 edition include "Games and Entertainment," "Business," "Education" and "Utility Programs." ("VCS Games," "The Atari 5200," and "Atari Hardware" are also covered.) Within each category, individual listings provide information essential to the prospective purchaser: retail price, program availability, memory requirements and the presence or absence of disk copy protect code are all treated. Though the criteria for each product's rating varies from category to category, the general grading divisions consist of creativity, controllability, graphics, skill, ease-of-use and originality. Sample monitor screens give the reader some idea of the quality of graphics resolution. In the case of business software, typical printer output is used to indicate the degree of application detail.

The reviews themselves display meticulous attention to detail, provide accurate synopses of software content (as revealed by our limited testing of some of THE BOOK's entries), and, in some instances, evolve

into minor monographs on game esoterica (the "Choplifter" and "Protector" reviews come to mind). Of special interest are those sections which explicitly compare one software product to another: three extensions of Atari BASIC and seven different Atari assemblers receive this in-depth analysis.

Any Atari software buyer, either at the retail or wholesale level, would do well to consult this thoughtfully produced "bookware" product. THE BOOK OF ATARI SOFTWARE 1983 is an invaluable consumer guide to the changing Atari software marketplace.

COMPUTE!'S FIRST BOOK OF ATARI

COMPUTE! Magazine

Level: Intermediate Rating: **90**

	COMPUTE	1981	Paper	
184 Pages	ISBN: 0-942386-00-0		6" x 9"	\$12.95

COMPUTE!'S FIRST BOOK OF ATARI is an anthology of articles published in Compute! Magazine in 1980. Its consistently high-quality selections cover the full spectrum of Atari-related subjects: chapter titles are "Getting to Know Your Atari," "Beyond the Basics," "Graphics," "Programming Hints," "Applications" and "Peripheral Information." The book's spiral binding and eminently readable articles combine to make it one of the more informative, reader-friendly offerings on the Atari 400/800.

An introductory section briefly reviews Atari BASIC, but not without a few poignant critiques of the Atari reference manual and its "less than perfect" version of BASIC. Advanced programming techniques follow: concise, informative essays treat memory structure, special POKE locations and a novel utility written in BASIC to scan memory for specific character strings. It should be noted that though assembly language routines are alluded to throughout the book, this is the one subject which the editors have not included.

The size and scope of the graphics chapter mirrors the unique capabilities of the Atari in this area. Topics range from the simple manipulation of SETCOLOR parameters to produce three-dimensional objects to a "player/missile graphics" article rivaling Lon Poole's treatment in YOUR ATARI COMPUTER. Ticker-tape messages and graphics card games are the subjects of two short, elegant program listings. An article by Craig Patchett on designing additional Atari BASIC graphics modes is exceptionally original.

Most of the remaining selections focus on enhancing Atari DOS or Atari BASIC either to compensate for inherent shortcomings (saving dimensional variables and error reporting), or to expand features far

beyond the limits of their original design (voice tracks and an interrupt-driven keyboard). Both Atari and non-Atari users will be intrigued by one article which lists a joystick-to-keyboard translation program.

COMPUTE!'S FIRST BOOK OF ATARI is, like the machine it describes, an impressive, easy-to-use sourcebook of valuable programming hints and techniques.

COMPUTE!'S SECOND BOOK OF ATARI

COMPUTE! Magazine

Level: Intermediate Rating: **85**

	COMPUTE!	1982	Paper	
250 Pages	ISBN: 0-942386-06-X		6" x 9"	\$12.95

Unlike COMPUTE!'S FIRST BOOK OF ATARI, this volume consists of previously unpublished Compute! articles. The five subject areas treated also reveal the more advanced scope of the SECOND BOOK: Utilities, Programming Techniques, Advanced Graphics and Games Utilities, Applications, and Beyond BASIC. While a familiarity with Atari BASIC is assumed, a knowledge of assembly language is helpful but not required. (Screen dumps and programs that load binary files from DOS are written in 6502 assembly language.)

Early articles focus on enhancing DOS or Atari BASIC. The lack of a TAB statement in Atari BASIC, for example, is remedied by a program which POKES a value into a single memory location. Another utility lets the user draw figures on any graphics screen using a joystick.

A series of articles by Phil Dunn in the Advanced Graphics section describes methods of filling shapes with color using the X10 FILL command, and of additionally creating "textured" graphics. Of the purely fun entries in the anthology, Sol Gruber's "Fast Banner," the Atari-version of a ticker-tape-run-amok, is an engaging example of animation and color-shifting graphics. Characteristic of the technically advanced articles in the Beyond BASIC chapter is instruction on bypassing DOS while still accessing data on diskette.

Like its predecessor, COMPUTE!'S SECOND BOOK OF ATARI is a compendium of informative articles on various aspects of Atari programming; its advanced scope makes it most suitable for the intermediate Atari BASIC programmer.

COMPUTE!'S FIRST BOOK OF ATARI GRAPHICS

COMPUTE! Magazine

Level: Intermediate Rating: **85**

COMPUTE!	1982	Paper	
248 Pages	ISBN: 0-942386-08-6	6" x 9"	\$12.95

"You have a palette that any artist would envy," comments David Diamond in his contribution to ATARI GRAPHICS. It is the exploration of the unique graphics capabilities of the Atari which forms the theme of this collection of published and unpublished Compute! articles. Each of the five subject areas included in the book rates as a minor dissertation on a different aspect of graphics esoterica; they include Graphic Fundamentals, Graphic Mode Customizing, Character Set Redefinition, Animation, and Advanced Graphics Techniques. Familiarity with Atari BASIC is assumed.

Deceptively simple programs demonstrate the differences between playfield and control graphics in the book's introductory chapter. Topics and programs become more complex as the reader advances to a discussion of shadow and hardware registers in the character set utilities section (the Atari uses the hardware register to locate the character set, but every sixtieth of a second "refreshes" hardware register data with the contents of a shadow register). Player/missile graphics ("P/M" graphics to jargon-lovers) is given a comprehensive, illustrated treatment by the author of ATARI BASIC, Bill Wilkinson.

Our particular favorite was the Advanced Graphics Technique chapter. In it, David Diamond produces elegant, effective examples of "artifacting"—a POKE-less method of producing four-color graphics in GRAPHICS mode 8.

Every Atari programmer will find COMPUTE!'S FIRST BOOK OF ATARI GRAPHICS a valuable sourcebook of introductory and advanced graphics programming techniques.

COMPUTERS FOR KIDS-ATARI EDITION

Sally Greenwood Larsen

Level: Novice Rating: **90**

Creative Computing	1981	Paper	
81 Pages	ISBN: 0-916688-22-4	9" x 11"	\$4.95

"How to get a date with Miss Piggy" is the very serious problem confronted by the elementary school programmer in the early chapters of

Sally Greenwood Larsen's book. Through flowcharting and the use of a "do-loop," the next generation of structured programmers learns that the first step in a possible solution is to continue moving until you find that she, too, lives in your neighborhood. The technique is characteristic of the effective use of humor and programming fundamentals in COMPUTERS FOR KIDS.

Through a large-print, illustrated format, children are taught the operation of each of the components of a microcomputer system. Special keys like "return" and "reset" are included in these lessons, along with reminders that there is to be "...no pounding, please." Students "graduate" to simple BASIC programs. Workbook-type exercises and actual machine practice reinforce the use of print statements, IF/THEN and FOR/NEXT loops, and graphics functions.

The author has also included an appendix suggesting general guidelines for parents and teachers, as well as specific microcomputer lesson plans to be used in conjunction with the book. One note explains the deliberate omission of more than a few sample programs in the text as an attempt to wean students away from dependence on someone else's programs.

COMPUTERS FOR KIDS, in each of its five microcomputer editions, is a funny, enjoyable BASIC manual which painlessly accomplishes its goal of teaching children fundamental programming skills.

INSIDE ATARI DOS

Bill Wilkinson

Level: Intermediate Rating: 65

	COMPUTE!	1982	Paper	
116 Pages	ISBN: 0-942386-02-7		6" x 9"	\$19.95

Wilkinson's text is actually an annotated listing of the assembly language program forming one of the four main elements of Atari DOS 2.0S. While Atari programmers will unquestionably find the author's analysis of the Atari File Management System useful, INSIDE ATARI DOS does not compare favorably with its more detailed, comprehensive counterparts, BENEATH APPLE DOS and TRS-80 DISK AND OTHER MYSTERIES (see our reviews).

The Atari Disk Operating System is composed of four discrete assembly language modules: DUP (Disk Utility Package), CIO (Central Input/Output), FMS (File Management System) and SIO (Serial Input/Output). After a cursory discussion of the interrelation of each of these programs, the author focuses on FMS diskette organizing conventions: the Disk Directory, Volume Table of Contents (VTOC) and Data Sectors are explained. The only intriguing piece of information divulged is the

improperly skewed bit map of the Atari 810 disk drive, which is unable to accept commands to access sector zero. The remainder of the book is a routine-by-routine description of the FMS assembly language program. Wilkinson's explanations of routine functions, address, and entry and exit registers supplement a 41-page listing of 6502 assembly language code. A single appendix provides a decimal/hex conversion program and some words of warning regarding careless DOS modification.

There is an interesting, informative book in the highly-regarded Atari Disk Operating System; upon reading *INSIDE ATARI DOS*, most Atari programmers will find that it has yet to be written.

MAPPING THE ATARI

Ian Chadwick

Level: Intermediate Rating: **85**

	Compute!	1983	Paper	
194 Pages	ISBN: 0-942386-09-4		6" x 9"	\$14.95

Chadwick's text contains an annotated listing of every significant location of the Atari memory map. Besides providing general insights into the Atari's use of available memory, this book is an essential reference manual for programmers whose code mandates memory mapped accesses.

The reader is introduced to the Atari's least significant byte/most significant byte storage conventions. Decimal/hexadecimal equivalents are discussed (a conversion program is included in one of the book's appendices), as is the role of Boolean operators in masking bytes of data.

The rest of the text is devoted to the memory maps. Each location is described in decimal, hexadecimal, and with the external label used by OS, DOS and DUP routines. The accompanying commentary frequently contains coding examples which either demonstrate or expand a particular location-specific function. The entry for MEMTOP, for example, after describing its primary function as a pointer to the top of available memory, includes a few lines of error-trapping code for possible use when reserving memory based on a MEMTOP value.

Chadwick's commentary on the RAM shadow registers used to produce the unique Atari graphics is particularly detailed. In this case, short, general purpose BASIC routines for rotating the colors in a given register set are included.

Appendices "geographically" depict the memory maps, compare old and new Atari ROM's, and additionally index the book's numerical listings by label and subject. *MAPPING THE ATARI* is a reference tool for

any intermediate programmer interested in the manipulation of specific memory locations.

SOME COMMON BASIC PROGRAMS- ATARI EDITION

Lon Poole and Mary Borchers

Level: Novice Rating: 85

Osborne 1982 Paper
202 Pages ISBN: 0-931988-53-5 9" x 11" \$14.99

See our review of SOME COMMON PASCAL PROGRAMS on page 212.

UNDERSTANDING ATARI GRAPHICS

Michael Boom

Level: Intermediate Rating: 85

Alfred 1982 Paper
48 Pages ISBN: 0-88284-224-2 4" x 11" \$2.95

For the experienced BASIC programmer interested in sampling the graphics capabilities of the Atari 400/800, this inexpensive Alfred Handy Guide is excellent. Rarely has fifty pages of microcomputer text been so filled with logically presented, useful information.

The different graphics modes of Atari BASIC are demonstrated in a sample program. Pixel concepts of low and high resolution graphics preface a discussion of each graphics command. The text is supplemented by annotated graphics programs, with illustrations depicting the programs' output. A special chapter treats the three graphics modes (9, 10, and 11) accessible through Atari BASIC on machines equipped with the GTIA chip. Four programs highlight its high resolution capabilities. Boom's quick tour of Atari graphics concludes with a few additional "tips and tricks" that can be incorporated into graphics code.

This is one of the few books where the appendices are as interesting as the text. One appendix contains charts of the color registers, pixel size and pixel load numbers that change in each of the Atari's twelve modes. Another lists the complete "ATASCII" character set. A final chart lists all of the hues available to the SETCOLOR command.

At \$2.95, UNDERSTANDING ATARI GRAPHICS represents one of the best buys in the Alfred Handy Guide series.

YOUR ATARI COMPUTER

A Guide to Atari 400/800 Computers

Lon Poole, Martin McNiff and Steven Cook

Level: Intermediate Rating: 90

	Osborne	1982	Paper
458 Pages	ISBN: 0-931988-65-9	7" x 10"	\$16.95

Lon Poole's book has yet to be displaced as the most comprehensive guide to Atari 400/800 operations and programming. YOUR ATARI COMPUTER is divided into three main sections: an operational guide, a treatment of Atari BASIC and a programmer reference manual.

Using detailed photographs to supplement the text, the brief (39 pages) operational section describes the Atari keyboard, booting of DOS, and loading and running of programs. An introduction to Atari BASIC follows, with immediate mode programming using PRINT statements providing the foundation for short BASIC programs.

Though better introductory treatments of Atari BASIC exist (see ATARI SOUND AND GRAPHICS), the advanced BASIC section of the book is without equal. It includes enlightened discussions of the use of input prompting, with an emphasis on masks and validity checking. Error handling routines are covered in detail, using combinations of TRAP and PEEK statements. The game programmer will find the authors' description of advanced PADDLE and STICK functions particularly illuminating. Poole's original claim that his discussions would occur in "a working environment, not an academic one," is amply supported in all BASIC chapters by the use of machine examples.

The use of Atari peripherals—program recorder, printer and disk drive—is covered from the programming perspective. File handling techniques (sequential and random) are given a comprehensive treatment in the disk drive section, with differences among the latest versions of DOS noted.

The unique graphics and sound capabilities of the Atari are highlighted in the last advanced BASIC section. A mode-by-mode description of Atari graphics prefaces a more detailed chapter on character set animation and player-missile graphics. Atari sound, especially the individual parameters of the SOUND statement, receives a similarly meticulous treatment.

The last section of the book is a programmer's quick reference manual. An alphabetic compendium of BASIC statements, an error message glossary and a listing of important PEEK and POKE locations are among the topics included in the final chapter and appendices.

YOUR ATARI 400/800 is a state-of-the-art overview of advanced programming techniques. Though intended for a broad audience which includes the non-programmer, we recommend it to the intermediate programmer searching for a definitive guide to advanced Atari BASIC functions.

COMMODORE

COMMODORE SOFTWARE ENCYCLOPEDIA 2nd Edition

Commodore Busn. Machines Software Group

Level: Novice Rating: **80**

	Sams	1981	Paper	
406 Pages	ISBN: 0-672-21944-1		8" x 11"	\$9.95

Written by Commodore Business Machines, Inc., this software encyclopedia is targeted primarily at PET/CBM software, though a single chapter does treat VIC-20 offerings. Hardware and firmware sections additionally provide sources of peripherals and/or modifications which permit Commodore upgrading or interfacing with other devices.

The sequence of the book's topics is rather peculiar. The categories containing exclusively software information are Business, Word Processing, Utilities, Engineering Aids, Personal Aids, Games and Education. The "extra" chapters on the VIC-20, Publications, Firmware, Hardware, Canada and Europe fall between "Games" and "Education."

This "misplaced" educational section is further sub-divided into Administrative Aids, Math and Science, and Verbal Skills and Social Studies. The fact that this one section constitutes over 30% of the entire encyclopedia is indicative of the Commodore PET's large share of the microcomputer educational market.

The individual listings include a brief summary of the product's features, its hardware requirements and its retail price. Commodore, after the obligatory legal disclaimers, has appended its "seal-of-approval" to many non-Commodore products which it believes "to be highly meritorious." An appendix lists all vendors, with addresses and phone numbers, for readers interested in following up one of these entries.

Whether looking for computer-aided chemistry exercises or the Commodore version of Startrek, the Commodore user will find the COMMODORE SOFTWARE ENCYCLOPEDIA a valuable reference guide. PET/CBM owners interested in expanding their libraries of educational software will find this book particularly useful.

COMMODORE SUPERPET COMPUTERS SYSTEM OVERVIEW

F.D. Boswell, T.R. Grove, K.I. McPhee, et. al.

Level: Novice Rating: **70**

	Sams	1981	Paper		
69 Pages	ISBN: 0-672-21903-4		6" x 9"		\$5.95

This introductory volume to the documentation for the Commodore SuperPET issued by Waterloo Computing Systems provides an overview of SuperPET hardware and of the software produced by Waterloo for it. The discussion of system hardware, however, is not intended to be a comprehensive one; rather, it focuses primarily on those hardware features which are essential to the operation of Waterloo software; these include microprocessor selection (the SuperPet incorporates a MOS Technology 6502, which is used by Commodore BASIC, and a Motorola MC6809, which is used by Waterloo software) and bank switching (the SuperPet can have 64k of auxiliary RAM, a single 4k block of which can be accessed by "main memory" at any given time).

Aside from this disappointingly brief (6-page) introduction to hardware, the book's focus is on Waterloo Computing Systems software. After a four-page advertisement/introduction to Waterloo software products for the SuperPET, the book concentrates on operating system-like commands to manipulate disk files and peripherals (although these are treated, in a more succinct form, in the programming language reference manuals issued by Waterloo) and on the operation of Waterloo's text editor.

This volume is less of a guide to hardware than a software overview. Its appeal is limited to those users who have purchased Waterloo systems software for their SuperPET.

COMMODORE SUPERPET COMPUTERS WATERLOO MicroBASIC

J. Wesley Graham and K. Ian McPhee

Level: Novice Rating: **25**

Sams 1981 Paper
179 Pages ISBN: 0-672-21904-2 6" x 9" \$10.95

COMMODORE SUPERPET COMPUTERS WATERLOO MicroFORTRAN

P.H. Dirksen and J.W. Welch

Level: Novice Rating: **25**

Sams 1981 Paper
179 Pages ISBN: 0-672-21904-2 5" x 8" \$10.95

The effort to standardize programming languages stems from an awareness of the numerous disadvantages which arise when a single high level language has too many variations and dialects. Language standards free the programmer from working with a particular computer; the essentials of a language, once learned, can be used on almost any microcomputer with comparatively minor variations. In terms of the dissemination and application of software, programs written on one machine can be used on others with only minor variations. Even these slight variations among the dialects of a language are problematical; hence, the existence of a variety of BASICs has caused a good deal of grumbling and led to the publication of many conversions handbooks to assist the disgruntled user.

Both Waterloo MicroBASIC and Waterloo MicroFORTRAN purport to follow the standards of their respective languages. In the case of BASIC, Waterloo MicroBASIC "includes" the 1978 X3.60 ANS standard, with "many important extensions." (iii) Waterloo MicroFORTRAN is similarly "a subset of the FORTRAN-77 standard language" which "has been extended in significant ways." (iii) The experienced BASIC or FORTRAN programmer, however, will find that these "enhancements" completely overwhelm the standard versions of these two languages, and that Waterloo MicroBASIC and MicroFORTRAN have more in common with each other than they do with their respective standards.

The aim of "improving" upon standard BASIC or FORTRAN is apparently to introduce the elements of structured programming into otherwise chaotic languages. While some of the innovations introduced by Waterloo Computing Systems may do this, others certainly do not. In our opinion, LOOP/ENDLOOP statements, for example, offer a far greater possibility for abuse than their rough counterparts, the BASIC FOR/NEXT and the FORTRAN DO statements.

Fortunately, the damage done by these departures from language standards is strictly limited by the inadequate explanation offered for them in the tutorial and reference manuals. The beginning programmer who turns to these to learn the details of a language will be sorely disappointed by these works' treatment of numerous commands (and especially of the "enhancements") in too few pages. Nor can the reader turn to other standard BASIC or FORTRAN texts to find out how such statements as GUESS/ADMIT/ENDGUESS are used, since they are unique to Waterloo Computing Systems software.

Both because of this practice of adding dubious "improvements" to established languages and the failure to adequately explain their use, we urge SuperPET users not only to forego the purchase of these manuals, but more generally to avoid Waterloo's highly eccentric programming languages altogether.

COMPUTE!'S FIRST BOOK OF PET/CBM

COMPUTE! Magazine

Level: Advanced Rating: **95**

	COMPUTE!	1981	Paper	
252 Pages	ISBN: 0-942386-01-9		6" x 9"	\$12.95

The process of compiling an anthology is fraught with pitfalls. The diversity of topics, approaches and levels of treatment frequently combine to produce a volume which is best left unnoticed. Given these potential problems, the editors of Compute! are to be congratulated for collecting forty-one articles which provide a sampling of topics of interest to advanced PET/CBM users.

The volume emphasizes both the effective use of machine-dependent features and the means of circumventing various limitations of PET/CBM hardware and software. An excellent example of the former is John Winn's article on using the Commodore 2022 printer to produce plots, graphs and charts; of the latter, Elizabeth Deal's article on passing parameters to BASIC utility-like subroutines.

But to single out individual contributions to this volume is somewhat unfair, since almost all are of extremely high quality. Topics treated include interfacing (the installation of joysticks), BASIC programming (the inclusion of BASIC code in machine language programs is a novel twist), machine language programming (and especially its use within BASIC programs), the manipulation of disk files (variable-length random access files), graphics (enhancing CRT resolution) and utilities (a section presenting ten extremely valuable or novel programs).

COMPUTE!'S FIRST BOOK OF PET/CBM is outstanding for its useful programs and discussion of advanced programming techniques. It

is an indispensable addition to the libraries of all serious PET/CBM users.

COMPUTE!'S FIRST BOOK OF VIC

COMPUTE! Magazine

Level: All Rating: **80**

COMPUTE 1982 Paper
212 Pages ISBN: 0-942386-07-8 6" x 9" \$12.95

COMPUTE!'S FIRST BOOK OF VIC features a collection of articles on Commodore's VIC-20, most of which originally appeared in COMPUTE! Magazine in 1981 and 1982. The articles have been grouped into six categories: an introduction to the VIC-20; BASIC and machine language programming; the VIC-20's memory map; and graphics and games on the VIC-20. While the thirty-six articles in this collection are highly diverse both in content and quality, a few generalizations can be drawn.

Most disquieting are the first two articles, which are more concerned with product promotion for the VIC-20 than with enabling the reader to use the VIC more effectively. Michael S. Tomczyk, Commodore's Product Marketing Manager, has contributed the opening article, which traces the conception, creation and marketing of the VIC-20. The second examines the history of computing from its earliest times to the appearance of the VIC-20, which may mark "the beginning of a new era in personal computing."(19)

Aside from these, however, each contribution tends to focus on practical applications. Some of them do so by berating the obvious: one article treats the use of BASIC direct mode statements to transform the computer into a calculator, an application which is as silly as it is inefficient; another provides a rather simple examination of BASIC commands for CRT output, a topic which should be the sole preserve of inferior BASIC texts. Some contributions are simply short programs which enable the user to play games (Zap!, Starfight3) or use a bubble sort. A number of articles, such as the one dealing with a customized character set, are heavily indebted to the much clearer treatment of the same topic in THE VIC-20 PROGRAMMER'S REFERENCE GUIDE (see our review). In general, the volume is quite distinctive; these are articles written by VIC-20 enthusiasts eager to share their knowledge about the VIC—but having some difficulty in doing so. They have opted to emphasize the practical (and sometimes mundane) by showing HOW something is done without adequately explaining WHY it can be done this way.

Aside from the ready-to-run programs, however, some of the articles are of more than passing interest; this is especially true of the contribu-

tions on condensing BASIC code in order to fit it into the VIC-20's limited memory, on the organization of memory, and on TINYMON1, a compact monitor for machine language programmers.

The volume is ideal for those users wishing to try out a sampling of applications about which they have little knowledge; for those who go beyond this to ask why it works, COMPUTE!'S FIRST BOOK OF VIC has some clear limitations, although it remains both interesting and useful.

COMPUTERS FOR KIDS-VIC-20 EDITION

Sally Greenwood Larsen

Level: Novice Rating: **90**

	Creative Computing	1983	Paper	
81 Pages	ISBN: 0-916688-42-9	9" x 11"	\$4.95	

See our review of COMPUTERS FOR KIDS -ATARI EDITION on page 318.

GETTING ACQUAINTED WITH YOUR VIC-20

Tim Hartnell

Level: Novice Rating: **75**

	Creative Computing	1981	Paper	
131 Pages	ISBN: 0-916688-28-3	6" x 9"	\$9.95	

Two different witnesses, we are frequently told, never see quite the same thing in the same way. The same holds true for reading or writing, and perhaps accounts for the disparity between the contents of Tim Hartnell's book and the description of it provided by the publisher's blurb writer. An elementary introduction to BASIC combined with a book of simple games software has been magnified into a work which slowly but surely "leads the reader... to writing complex, sophisticated programs." The scant attention paid to sound, music and color graphics by Hartnell becomes a thorough description in the mind of the blurb writer. A work which focuses exclusively on the rudiments of Vic BASIC is magically transformed into one "about the Vic, the BASIC language, and microcomputers in general."

In fact, aside from his perfunctory treatment of music, graphics and the VIC-20 character set, the BASIC language instruction as well as the game programs offered by Hartnell could have been applied to virtually any microcomputer. Both, however, remain at a very simple level. The

games—even the “longer” ones—are primarily short and uninspiring. The introduction to BASIC offers the reader only a simple subset of the language. The book lacks a table of contents; and while it does contain an index of games, an index of BASIC commands is conspicuously absent.

Although Hartnell's book is admittedly simplistic, it is not without merit. Most notable is the way in which its early portions combine game-playing with BASIC programming. After presenting the bare rudiments of a game, Hartnell points to its inadequacies and offers suggestions for improving it. The reader not only learns to write better, more well-conceived games, but how to use BASIC more effectively.

I SPEAK BASIC TO MY PET

Student Text

Aubrey B. Jones, Jr.

Level: Novice Rating: **45**

	Hayden	1982	Paper
219 Pages	ISBN: 0-8104-6174-9	9" x 11"	\$7.45

See our review of I SPEAK BASIC TO MY APPLE on page 264.

MOSTLY BASIC-APPLICATIONS FOR YOUR PET

Howard Berenbon

Level: Novice Rating: **75**

	Sams	1980	Paper
157 Pages	ISBN: 0-672-21790-2	9" x 12"	\$12.95

See our review of MOSTLY BASIC: APPLICATIONS FOR THE APPLE II, Books 1 and 2, on page 268.

SOME COMMON BASIC PROGRAMS-PET/ CBM EDITION

Lon Poole and Mary Borchers

Level: Novice Rating: **85**

	Osborne	1981	Paper	
200 Pages	ISBN: 0-931988-40-3		9" x 11"	\$14.99

See our review of SOME COMMON PASCAL PROGRAMS on page 212.

USER'S GUIDE TO PET/CBM COMPUTERS

Jeffrey R. Weber

Level: Novice Rating: **80**

	Weber	1982	Paper	
328 Pages	ISBN: 0-9604892-8-2		6" x 9"	\$13.95

Weber's *USER'S GUIDE* offers a comprehensive, concise and generally effective overview of a range of microcomputers manufactured by Commodore Business Machines; his generalized work treats the commodore PET (models 2001-8K, 8n, 16n, and 32n) as well as the CBM Series 4000 and 8000. The quality of Weber's treatment, however, is weakened by the fact that his book attempts to achieve three major goals, none of which have been clearly articulated or integrated. To the reader, the first of these is merely an annoyance: in a book devoted to the general handling of a microcomputer rather than to its use for specific applications, Weber has included a seemingly irrelevant chapter on "Using the CBM for Business Computing" (pp. 201-215) in order to insert an advertisement for Weber Systems software.

Weber has focused on the other two goals, however, in a way that makes them appear almost mutually incompatible. In a series of chapters devoted to Pet/CBM BASIC (pp.57-199), the *USER'S GUIDE* makes some attempt to teach the uninitiated reader the basics of BASIC at the same time that it aims at providing no more than a summary guide to BASIC as it is implemented on Commodore computers. The amount of space wasted by this unsatisfactory approach has evidently led Weber to curtail his discussion of PET/CBM system details, which are of interest to anyone wishing to obtain a deeper knowledge of how to use his microcomputer.

But while Weber's discussion of numerous important points may be too brief or lack the wealth of detail needed by a user eager to expand his knowledge of the PET or the CBM, Weber does succeed in at least

mentioning numerous topics (such as file handling and memory maps) which other authors have carefully avoided. The result is a work which, despite its imperfections, does offer its readers a useful and valuable introduction to Commodore microcomputers.

VIC-20 PROGRAMMER'S REFERENCE GUIDE

A. Finkel, N. Harris, P. Higginbottom, et.al.

Level: All Rating: **95**

Sams 1982 Paper
289 Pages ISBN: 0-672-21948-4 6" x 9" \$16.95

The VIC-20 PROGRAMMER'S REFERENCE GUIDE incorporates in a single volume a sampling of topics which are of interest to virtually any VIC-20 programmer. Whether the user is a BASIC programmer, an assembly or machine language enthusiast or a hobbyist interested in interfacing experiments, this reference guide helps to apply skills more effectively and to expand them into new areas.

The utility of this volume is most limited, however, for the complete novice. That portion of the text which is most relevant to the newcomer to programming—its BASIC language section—is not a tutorial, but rather a reference guide to BASIC commands and statements; the user will not derive the knowledge to wield them effectively for programming applications. This is true, in fact, for the bulk of the text: the basic level of explanation presupposes that the user has a minimal expertise in the areas covered.

This should not at all detract from the value or quality of this volume for the more sophisticated programmer. The range of topics covered includes, in addition to the BASIC reference guide, graphics, the generation of user-defined character sets (which is especially useful for those who wish to use non-Roman script), the use of music and sound (the latter, among other possibilities, to inform the end-user of the need for intervention during program execution), the BASIC interpreter, the organization of the VIC-20's memory, machine and assembly language programming and interfacing. For the user with some familiarity with computers and programming, these subjects are treated provocatively and effectively.

The production quality and user-friendliness of the book are high: both the guide to the 6502 instruction set and the BASIC reference guide are presented in a tasteful, easy-to-use spiral-bound format; memory maps of varying detail abound in the section which examines the organization of memory.

For all those who want to derive the greatest knowledge, expertise and enjoyment from programming their VIC-20, we strongly recommend this volume as a supplement to the VIC-20 owner's manual.

TIMEX/SINCLAIR

BYTEING DEEPER INTO YOUR TIMEX/ SINCLAIR 1000

Mark Harrison

Level: Intermediate Rating: **95**

Wiley	1982	Paper	
160 Pages	ISBN: 0-471-89888-0	7" x 10"	\$12.95

The use of a microcomputer with the meagerest of computing resources elicits either the best or worst from programming authors. Programs must be relatively concise (owing to the ZX-81's membrane keypad), memory-efficient (the ZX-81 has only 1K of memory, the Timex/Sinclair 1000, 2K), and of course, still be fun and easy-to-use. Mark Harrison has flourished in this desert of limited resources. In the process of providing thirty-seven programs which more than meet the stated criteria, he has also managed to present an introduction to ZX-81 operation and Sinclair BASIC.

Overlooking an initial chapter on "The Principles of a Computer System" which contains a mysteriously out-of-place discussion of binary arithmetic, the early sections of text treat a hardware overview of the ZX-81. A three-column illustration of the cursor, the membrane key to be depressed and its effects upon the display is particularly effective. The author categorizes ZX-81 function keys into groupings of system commands, assignment, I/O, and control statements. A general treatment of ZX-81 BASIC follows.

Though all topics are initially covered on an introductory level, the reader who will most benefit from Mr. Harrison's text will already possess some knowledge of Sinclair BASIC. BYTEING DEEPER includes a more detailed discussion of the ZX-81 memory map than is available in the reference manual, with the intention of using Z80A microprocessor architecture and specific memory addresses in machine-code programs. Little actual assembly language programming is done, with the author instead recommending the more comprehensive treatment provided by Zaks' PROGRAMMING THE Z80. Programming techniques designed to make the most efficient use of ZX-81 memory are explained in depth. Screen manipulation, for example, is facilitated by first "poking" data into special addresses. The novel use of REM statements to store program

data is explored. One three-page chapter on "conserving memory" contains some original hints which, at the time of this writing, are found nowhere else in ZX-81 literature.

The quality of the sample program of **BYTEING DEEPER INTO YOUR APPLE** surpasses those of books whose sole function is to provide an anthology of BASIC code. Trigonometric functions are used with the graphics capabilities of the ZX-81 to produce relatively short but elegant "pattern" programs. Of the pure "game" programs, "Black Holes," which included program modifications for 8K ROM ZX-80's, was our favorite.

BYTEING DEEPER INTO YOUR TIMEX/SINCLAIR 1000 is required reading for every intermediate ZX-81 programmer who is searching for a single definitive source of information on advanced Sinclair BASIC topics.

COMPUTERS FOR KIDS-SINCLAIR EDITION

Sally Greenwood Larsen

Level: Novice Rating: **90**

	Creative Computing	1982	Paper	
81 Pages	ISBN: 0-916688-32-1	9" x 11"	\$4.95	

See our review of **COMPUTERS FOR KIDS - ATARI EDITION** on page 318.

CRUNCHERS

21 Simple Games for the Timex/Sinclair 1000 2k

Yin Chiu and Henry Mullish

Level: Novice Rating: **75**

	Byte	1983	Paper	
137 Pages	ISBN: 0-07-010831-5	6" x 8"	\$8.95	

The games included in this book are of necessity "simple," for two reasons: first, they were all written for the 2K version of the Timex/Sinclair 1000; and second, anyone who has ever used the keypad on one of these micros knows that there is a limit to the number of BASIC statements that can be keyed at one sitting (yes, even using the two-thumb method). Given these limitations, however, the authors have still

managed to cover the gamut of computer gaming—from “High Roller” (a pseudonym for “craps”) to “Barriers” (space ship vs. the Forbidden Zone, our favorite).

Each entry is supplemented by a commentary which notes program and operational highlights. A “possible adaptations” section suggests areas for reader modifications.

With the availability of such games on cassettes which retail for about the cover price of this book, and with a keypad that is, at best, anti-ergonomic, the potential book buyer may be hard-pressed deciding whether to spend his time keying or shooting craps

GETTING ACQUAINTED WITH YOUR ZX81 3rd Edition

Tim Hartnell

Level: Novice Rating: **85**

Creative Computing Press 1982 Paper
120 Pages ISBN: 0-916688-33-x 6" x 9" \$9.95

Tim Hartnell, prolific ZX-81 author, as well as the editor of ZX COMPUTING and founder of the NATIONAL ZX USER'S CLUB, has compiled seventy new programs for the ZX-81. Though some concise, elegant graphs and an elementary “word processor” are included, this is first and foremost a game anthology. Typical entries are “Moon Lander” (a version of Lunar Lander), “Checkers,” “Dice Rolls,” “Racer” and “Blackjack.” Hartnell has gained a reputation for imaginative, simple code, and this current anthology is no exception. Each program is accompanied by explanatory text, which not only provides background and operating information, but also frequently suggests modifications. Interspersed throughout the book are additional author’s “asides,” in which he discusses his own programming techniques. Appendices include a program conversion guide for ZX-80 programs, and instructions for fitting the ZX-80 with the ZX-81 ROM.

GETTING ACQUAINTED WITH YOUR ZX-81 is an enjoyable, instructive collection of primarily game programs for the ZX-81 user. We recommend its purchase to anyone wishing to expand his library of quality ZX-81 programs.

MAKING THE MOST OF YOUR ZX-81

Tim Hartnell

Level: Novice Rating: **85**

Reston 1981 Paper
102 Pages ISBN: 0-8359-4188-4 6" x 9" \$10.95

Many of the ZX-81 publications displayed on the shelves of American book and computer stores are actually reprints of British texts, a natural consequence of Clive Sinclair's original introduction of the ZX-80 in England in February of 1980. This is true of MAKING THE MOST OF YOUR ZX-81. The distinctly British colloquialisms scattered throughout its chapters are certainly no problem for the microcomputer user who daily confronts the difficulties posed by different dialects of BASIC.

Tim Hartnell has compiled what is ostensibly another collection of programs for the ZX-81. It is much more. The programs themselves compare well with similar compendiums. "Lunar Landings," "Labyrinth," and an interesting version of a word-guessing game called "Hangcat" are all marked by a concise originality tailored to the ZX-81. The beginning programmer, however, will be especially pleased by the commentary which accompanies each program: these explanations aim at cultivating expertise in different areas of BASIC. "Lost in Space," for example, demonstrates the use of multidimensional arrays. "Snakes and Ladders" presents the technique used by experienced programmers working with a limited amount of memory of assigning constants to variables in order to conserve space. And instead of the stale random-number-generator program, Hartnell poses a novel variation of "Russian Roulette," which just as effectively introduces the RND function.

Part BASIC primer, part program collection, MAKING THE MOST OF YOUR ZX-81 is a perfect follow-up for beginning programmers who are eager to supplement their Sinclair reference manual.

MASTERING MACHINE CODE ON YOUR ZX-81

Toni Baker

Level: Advanced Rating: **85**

Reston 1981 Paper
180 Pages ISBN: 0-8359-4261-9 6" x 9" \$12.95

It is unfortunate for the ZX-81 user eager to learn machine-level code that the only available book on the subject is MASTERING MACHINE

CODE ON YOUR ZX-81; this is true not because of the book's content, which is comprehensive and entertaining, but because of the remarkably poor production quality of the text. Chapters resemble bound versions of individually typewritten monographs, with some sentences unintelligible and others marred by handwritten corrections.

The ZX-81 user who manages to wade through this tediously formatted text, however, will find his patience rewarded. Machine code is taught from its basics in binary arithmetic to register manipulation. The author additionally includes divergent ZX-80 instructions, where the differences in ROM mandate coding modifications.

Simple assembly language instructions are explained, along with an overview of Z80 microprocessor architecture. Programming examples consolidate these lessons; an anti-scrolling BASIC program, for example, uses a 10-line machine-code subroutine to reverse the display's normal scrolling direction. Techniques for storing machine-code above BASIC-addressable memory, in REM statements, and in the variables area are discussed. Stacks, registers and machine level op codes—all familiar subjects to assembly language programmers—are covered in detail.

A series of machine code games follow the "lesson" portion of the book. Checkers (or "draughts," as our English author prefers) evolves into a demonstration of techniques employed in complex heuristic searches. Less profound is the ZX-81 version of "Breakout," coded for speed and efficiency in hexadecimal. Another program translates return codes from letters on the ZX-81 keypad into two octaves of musical notes. A final chapter treats three techniques for disassembling the ROM, simultaneously changing hex codes into their equivalent assembly language op codes.

MASTERING MACHINE CODE ON YOUR ZX-81 is a state-of-the-art lesson in ZX-81 assembly language programming. The intermediate to advanced ZX-81 user, once accustomed to its paradoxically medieval production quality, will undoubtedly benefit from its presentation of the speed, efficiency, and flexibility of machine language coding.

MASTERING YOUR TIMEX/SINCLAIR 1000 PERSONAL COMPUTER

Tim Hartnell and Dilwyn Jones

Level: Novice Rating: **90**

Bantam 1982 Paper
208 Pages ISBN: 0-553-23241-x 5" x 7" \$3.95

While its mass market rival, the TIMEX PERSONAL COMPUTER MADE SIMPLE, concentrates on ZX81 operational details, this Tim Hartnell book focuses on developing programming skills in Sinclair BASIC.

The treatment of Timex/Sinclair 1000 keyboard usage in the introduction is barely adequate for the confused beginner; the author actually attempts to explain one of the more complex keyboards (five functions for every key) in five pages, without recourse to a single diagram or template.

It quickly becomes apparent that Hartnell has more important subjects to consider; among them, Sinclair BASIC programming. Discussions of individual BASIC topics are excellent. Sixty-five programs, written for the minimum 2K Timex/Sinclair 1000, are used as stepping stones across the whirls and eddys of BASIC commands and functions. Eye-catching graphics not only demonstrate the pure graphics functions, but also enliven the more mundane topics of loops and conditional statements. The chapter on PEEK and POKE commands uses a billboard analogy and memory map to provide one of the most lucid explanations in Timex/Sinclair 1000 literature. System constants, like error report codes and screen size, are changed using these commands in short BASIC programs. Hartnell additionally works them into an "advanced topic" treatment which considers the prevention of screen memory overflow and the insertion of noneditable lines into program listings.

Concluding chapters include a conversion guide to other BASIC dialects and some valuable hints to optimize the efficient use of the machine's limited memory.

MASTERING YOUR TIMEX/SINCLAIR 1000 PERSONAL COMPUTER is the best mass market treatment of Sinclair BASIC programming; while weak on operational topics, it is a comprehensive guide to developing a programming expertise on the Timex/ Sinclair 1000.

THE SINCLAIR ZX-81 Programming For Real Applications

Randle Hurley

Level: Novice Rating: **85**

Dilithium 1981 Paper
164 Pages ISBN: 0-88056-090-8 6" x 8" \$9.95

Unlike other ZX-81 offerings, Hurley's book is less an introduction to BASIC programming or the functions of the machine than a "cookbook" of different programming recipes to be tried by the ZX-81 owner. These include personal financial applications and rudimentary word processing.

The presentation is less palatable than the SAMS ZX-81 BASIC BOOK or the primer by Sinclair distributed with a ZX-81 purchase. (With the exception of one chapter which was set using the Hurley LX81 word

processing program, typewriter-quality typeface was used.) The content, however, is for the ZX-81 user who, having devoured the little introductory material which exists, now wants to push his micro to its limits and beyond. One of the most interesting chapters treats hardware modifications to the ZX-81 power supply to eliminate reported fuse problems; another discusses the electrical connections required to install a larger keyboard (the current ZX-81 keyboard being its primary drawback). Both subjects are indicative of the type of ZX-81 user who should buy Hurley's *SINCLAIR ZX-81*: one who is willing to forgo some of the niceties of presentation in exchange for real application knowledge and/or technical expertise.

TIMEX PERSONAL COMPUTER MADE SIMPLE

A Guide to the Timex/Sinclair 1000

Joe Campell, Jonathan D. Siminoff, Jean Yates

Level: Novice Rating: 75

	NAL 1982	Paper	
148 Pages	ISBN: 0-451-12138-4	4" x 7"	\$3.50

In many ways, this mass market offering is to other Timex/Sinclair 1000 books what the Timex/Sinclair 1000 is to other personal computers: significantly cheaper and harder to use effectively.

Of the three general subject areas it treats—operational guidelines, programming tips and a program anthology—the least emphasis has been placed on teaching the reader the fundamentals of Sinclair BASIC. The operational discussion, with its keyboard diagrams and informative text, does a noticeably better job than its mass market counterpart, *MASTERING YOUR TIMEX/SINCLAIR 1000 PERSONAL COMPUTER*. The “program” section categorizes simple programs according to their application in the home, at work or in school. These programs include “General Record-Keeping,” “Checkbook Balancing” and a “Scrambled Word Solver.” Though explanatory text is included with each program and some effort is made to introduce new programming functions and techniques, a comprehensive, cohesive programming presentation is not one of the book’s strong points. Graphics and arrays are briefly covered, PEEK and POKE commands not at all. Moreover, though the book contains more programs than the manufacturer’s manual, its mass market format makes it less convenient to use than its spiral-bound counterparts, especially for manual entry of thirty-line BASIC programs.

Given these shortcomings and variations in emphasis, the book does contain a unique "resource" appendix: publications, hardware and software suppliers, and national users groups are listed in a one-of-a-kind guide for Timex/Sinclair users.

THE TIMEX PERSONAL COMPUTER MADE SIMPLE is not one of the more outstanding treatments in this category; it does, however, possess certain "redeeming values" which may well justify its purchase to a segment of the Sinclair 1000 public.

THE TIMEX-SINCLAIR 1983 DIRECTORY

Eben Brown

Level: Novice Rating: **75**

E. Arthur Brown 1983 Paper
90 Pages ISBN: (None) 6" x 9" \$5.00

THE TIMEX-SINCLAIR 1983 DIRECTORY is one of the few sources of peripheral hardware and software packages for the Timex/Sinclair 1000. Though the general production quality of the book is far from superior, users will find comprehensive listings of disk drives, printers, memory expansion units, modems, keyboards and software. An alphabetical appendix of suppliers (some of which are understandably located in England) notes sources for all of the accessories contained in the book.

Some of the more novel products described in the DIRECTORY include a combination tape control module/RS-232-C interface, a non-membrane keyboard attachment which clips directly on to the existing keyboard, and one expansion board whose capabilities include "satellite tracking and irrigation control." A special "programming aids" section contains utility packages designed to enhance the graphics, assembly language, and debugging features of the original ZX-81. Six groups of software packages are reviewed: business and personal, education, science and engineering, games, graphics and sound, and miscellaneous. Many of these packages, such as a stock option analyzer and electronic spreadsheet, are scaled-down versions of products long popular with users of more powerful microcomputers. The periodicals and books which currently treat Timex/Sinclair topics are also listed in a supplemental chapter.

Like the machine it describes, THE TIMEX-SINCLAIR 1983 DIRECTORY is a no-frills tour of the growing market in Timex/Sinclair 1000 accessories. This sourcebook represents the most current, comprehensive guide to a wide range of Timex/Sinclair hardware and software products.

YOUR TIMEX SINCLAIR 1000 & ZX81

Douglas Hergert

Level: Novice Rating: **85**

Sybex 1983 Paper
 159 Pages ISBN: 0-89588-099-7 6" x 9" \$6.95

This late Sybex entry into the crowded ZX-81/Timex Sinclair 1000 field assumes no familiarity with either the operation of the machine or BASIC programming. It attempts to impart an understanding of both, in the process improving upon the ZX-81's succinct "official" reference manual. While the Hergert text is certainly more reader-friendly than the manual, it compromises the manual's comprehensive treatment of function keys, internal architecture, and advanced BASIC topics for a more rudimentary understanding of the Timex/Sinclair 1000. This is a simple, admittedly incomplete introduction for the easily-intimidated micro-computer user.

The book naturally divides into three main sections. The first is an operational guide to ZX-81/Timex Sinclair 1000 set-up and initial operation. The second is a general introduction to the machine's function-packed membrane keypad. Finally, function keys are used to enter progressively more complex BASIC programs, reinforcing keypad skills while teaching programming techniques. Graphics programs (which will function on the 2K Sinclair 1000 but not on the 1K ZX-81) are the primary tool for demonstrating the BASIC command set. Arithmetic functions and bar graphing are included in the book's "advanced" BASIC chapter.

The book's standard binding makes reading while keying code difficult; a spiral binding might have provided ZX-81 users with one of the easy-to-use amenities they so desperately deserve. YOUR TIMEX SINCLAIR 1000 AND ZX-81 is nevertheless a pleasant, lucid presentation targeted at first-time users in need of some literary hand-holding.

ZX-81 BASIC BOOK

Robin Norman

Level: Novice Rating: **80**

Sams 1982 Paper
 187 Pages ISBN: 0-672-21957-3 5" x 9" \$12.95

Norman's book coincidentally bears a striking resemblance to the introductory manual which accompanies each ZX-81 purchase. Between its spiral bindings, however, is a much more conversational text, written in a clearer typeface with a noticeable improvement in content as

well. One-upmanship appears to be the key. The sample programs, both in the example sections of each chapter and in a special program appendix, are simply more original than its Timex/Sinclair counterpart, and better illustrate individual BASIC commands and functions. Typical titles are "Electronic Dice," "Bar Charts" and "Drawing Pictures."

An overview of high level languages is followed by the introduction to Sinclair BASIC which forms the book's primary focus. All of the ZX-81 functions are covered. The graphics chapters, with their descriptions of PLOT, UNPLOT and PRINT AT commands, contain particularly enjoyable programming examples. Advanced programming topics include PEEKS and POKES into special ZX-81 memory locations.

Though better Sinclair BASIC books abound (see any of the more highly rated books in this category), Norman's ZX-81 BASIC BOOK represents a definite improvement over its Timex twin.

THE ZX81 COMPANION

Robert Maunder

Level: Novice Rating: **95**

Creative Computing 1981 Paper
131 Pages ISBN: 0-916688-26-7 5" x 9" \$9.95

If there is a single criterion for selecting a quality ZX-81 Timex/Sinclair text in order to avoid the the growing legion of pretenders, it is to first find a British author. Perhaps a function of their longer experience playing with Clive Sinclair's "toys," the British appear to have a monopoly on entertaining, informative ZX-81 handbooks. THE ZX-81 COMPANION, intended as a supplement to the manufacturer's manual, is no exception. Robert Maunder, author of ZX-80 COMPANION, has refined his original presentation and fashioned a readable, detailed sampling of ZX-81's capabilities.

A knowledge of Sinclair BASIC is assumed. The book's four sections treat graphics and real time programming techniques, information processing, education, and an analysis of the ZX-81 8K ROM Monitor. The programs which demonstrate each topic are structured with a description of the object of programs, a sample screen format, a sequential outline of the techniques used, a list of variables and, finally, the actual program listing. This standardization significantly adds to the ease with which programs can be keyed, understood, and enjoyed. The programs themselves reveal a uniquely British originality, with mazes, self-generating graphic patterns, and a cross between a math quiz and the arcade game "Frogger." Even "serious" subjects, like file processing, use as sample data the vital statistics of a character from another British import, THE HITCHHIKER'S GUIDE TO THE GALAXY.

Of particular note is the chapter which discusses the disassembled listing of the 8K ROM Monitor (which is printed in its entirety). While the author does not explain the nuances of Z80 programming (he, too, recommends Zaks' PROGRAMMING THE Z80), his annotated summary provides even the novice programmer with some information on how his ZX-81's operating system "thinks."

THE ZX-81 COMPANION is a masterfully crafted text. Every ZX-81 owner should own at least one copy.

101 TIMEX 1000/SINCLAIR ZX-81 PROGRAMMING TIPS & TRICKS

Edward Page

Level: Novice Rating: **60**

ArcSoft 1983 Paper
128 Pages ISBN: 0-866680-20-9 6" x 9" \$7.95

Contrary to its title, this book contains no explicit "programming tips and tricks" other than those that the reader may glean from the code of each program. Commentary which precedes each short section of code does provide some insight into the functioning of each program, but relates more about how the game is played than the programming techniques employed. Individual BASIC commands, syntax and program structure are never mentioned.

Simply, this is a collection of 101 ZX-81 programs—nothing more—despite publisher and author claims. These programs are classified in six categories: "Fun and Games," "Text on Text" (word manipulation), "Gee Why" (the neighbor's supposed response upon viewing the programs), "Number Crunching" (or "Number Chrunching," as listed in the table of contents), "Money Matters," and "Graphics." Programs average about twenty lines of code, contain no REM statements and frequently can only be ended by use of the BREAK key. Each provides a short, painless method of demonstrating various applications on the ZX-81.

101 TIMEX 1000/SINCLAIR ZX-81 PROGRAMMING TIPS AND TRICKS is a beginner's collection of simple BASIC programs; while it is not a book on BASIC programming techniques by any stretch of the imagination, it will provide quick, easy code in a wide range of ZX-81 applications.

37 TIMEX 1000 SINCLAIR ZX-81 PROGRAMS FOR THE HOME, SCHOOL, AND OFFICE

Edward Page

Level: Novice Rating: **50**

	ArcSoft	1983	Paper	
96 Pages	ISBN: 0-86668-021-7		6" x 9"	\$8.95

This is the "companion volume" to 101 TIMEX 1000/SINCLAIR ZX-81 PROGRAMMING TIPS AND TRICKS (see our review). Programs, which are generally longer than their counterparts in the original volume, are categorized as applicable to either the home, classroom or office.

The reader should be forewarned that some programs are identical to those published in the first book. ("Shopper's Friend" in 37 PROGRAMS is program #68 in 101 PROGRAMS; "Hourly Wages" is a clone of program #65, entitled "Wages and Hours.") What has not been copied from its companion volume, 37 PROGRAMS manages to copy from itself. The school section, for example, provides separate program quizzes on foreign capitals and state 'geographic centers' (capitals might have been a bit too obvious), arithmetic operations, and advanced arithmetic operations (even the most unperceptive reader begins to discern a pattern). Programs which may very well be original and unique to this book somehow find their way into rather incongruous categories. The "home" section, for example, not only contains a loan payment calculator, but also a slot-machine game (we assume for households located in either Las Vegas or Atlantic City).

Of special interest among the "business" programs is an "Executive Decision Maker" which bears a marked similarity to the "Superior Decision Maker" of 101 PROGRAMMING TIPS. In both programs, the randomizing function of the ZX-81 is used to make an entirely arbitrary decision from a set of limited alternatives, ranging from yes/no to "Fire Someone," or "Pass the Buck." We recommend that the ZX-81 owner unrandomize his selection criteria for program anthologies by assiduously avoiding this book.

49 EXPLOSIVE GAMES FOR THE ZX-81

Tim Hartnell

Level: Novice Rating: **80**

Reston 1981 Paper
140 Pages ISBN: 0-8359-2086-0 6" x 9" \$10.95

Many of the games included in this book are the ZX-81 versions of games long popular with users of "older," more powerful personal computers: "Star Trek," "Breakout," "Checkers," "Nim" and "Hangperson" are some examples. Though many of these programs will run in the 1K of a stripped-down ZX-81, we recommend the use of the 16K RAM attachment to fully appreciate their "explosive" power. Hartnell, an author whose early work includes ZX-80 books, has added an additional twenty-seven ZX-80 game programs to the ZX-81 portion of the text. With minor revisions, especially to machine-specific POKE statements, these programs will run on the new ZX-81 ROM. Readers of another Reston ZX-81 title, MAKING THE MOST OF YOUR ZX-81, will notice slightly modified versions of its games surfacing in this current anthology.

49 EXPLOSIVE GAMES FOR THE ZX-81 is an annotated collection of Sinclair BASIC programs which draws from a number of previously published books. We recommend its purchase to those ZX-81 users with both cassette tape deck and 16K RAM attachment who are looking for an enjoyable way to expand their personal games library.

Model Inventory Appendix

The following model inventory has been included for computer and bookstore managers interested in starting or expanding a microcomputer book section. Not only has quality of presentation been considered, but also the sometimes more important factors of subject balance, price, and current demand (oblations to that awful god of "turn"). While individual title selections will naturally be tailored to clientele and/or product mix, we think it is a good starting point for a personal computer book section that combines quality with broad appeal.

Model Inventory

TITLE	STOCK LEVELS		
	A	B	C
APPLE II USER'S GUIDE	1	3	6
APPLE BACKPACK		3	3
APPLE MACHINE LANGUAGE			3
ATARI ASSEMBLER			3
ATARI BASIC			5
ATARI GAMES & RECREATION			5
BASIC AND THE PERSONAL COMPUTER			4
BASIC APPLE BASIC		3	6
BASIC BASIC	3	5	8
BASIC COMPUTER GAMES	3	6	8
BASIC COMPUTER PROG. FOR BUSINESS VOL 1-2			5
BASIC FASTER AND BETTER		3	6
BASIC FROM THE GROUND UP			5
BASIC HANDBOOK	2	4	8
BASIC PROGRAMS FOR SCIENTISTS & ENGRS.			4
BASIC PROGRAMMER NOTEBOOK			3
BASIC PROGRAMMING PRIMER	3	5	8
BENEATH APPLE DOS			3
BOOK OF APPLE COMPUTER SOFTWARE 1983	2	4	6
BOOK OF ATARI SOFTWARE 1983	2	4	6
BOOLEAN ALGEBRA FOR COMPUTER LOGIC			3
BYTEING DEEPER INTO YOUR TIMEX		2	5
CIARCIA: VOLUMES 1-3			3
COMMODORE SOFTWARE ENCYCLOPEDIA	2	4	6
COMMON BASIC - IBM PERSONAL COMPUTER		3	6
COMMON BASIC - APPLE		3	6
COMPUTER DICTIONARY		3	4
COMPUTER GRAPHICS PRIMER		3	5
COMPUTERS FOR EVERYBODY	3	5	8
COMPUTERS FOR KIDS - APPLE & ATARI ED.		3	6
COMPUTERS IN NUMBER THEORY			3
CP/M HANDBOOK WITH MP/M	2	4	6
CP/M PRIMER			3
CP/M USER'S GUIDE		3	6
CRASH COURSE IN MICROCOMPUTERS	2	4	6
CREATE WORD PUZZLES WITH YOUR MICROCOMP.			3
CREATIVE APPLE			3
DATA FILE PROGRAMMING IN BASIC		3	6
DEVIL'S DP DICTIONARY			3

STOCK LEVELS

TITLE	A	B	C
DOING BUSINESS WITH VISICALC		3	6
DON'T	2	4	6
ELEMENTARY APPLE	2	4	6
ESSENTIAL COMPUTER DICT. AND SPELLER	2	3	4
EXECUTIVE COMPUTING	1	3	6
EXECUTIVE PLANNING WITH BASIC			3
FIRST BOOK OF MICROCOMPUTERS	2	4	6
FROM CHIPS TO SYSTEMS			5
GUIDE TO SYSTEMS APPLICATION			3
HOME COMPUTERS: 1024 Q'S & A'S, VOLS. 1-2			4
HOW TO BUY A PERSONAL COMPUTER	3	4	6
IBM/PC GUIDE	2	4	6
IBM'S PERSONAL COMPUTER			3
INSIDE BASIC GAMES		4	6
INTRODUCTION TO THE UCSD P-SYSTEM			3
INTRODUCTION TO WORDSTAR		3	5
KIDS & THE APPLE	2	4	6
LEARNING IBM BASIC FOR THE PERSONAL COMP.		4	6
LITTLE BOOK OF BASIC STYLE		4	8
MASTERING CP/M			3
MASTERING VISICALC			4
MASTERING YOUR TIMEX/SINCLAIR 1000	2	4	6
MICROCOMPUTER DICTIONARY			3
MORE BASIC COMPUTER GAMES	3	5	8
MOSTLY BASIC FOR APPLE, BOOKS 1-2	2	4	5
NAILING JELLY TO A TREE	2	4	6
PASCAL FOR BASIC PROGRAMMERS		3	5
PASCAL PROGRAMMING FOR THE APPLE		3	5
PERSONAL COMPUTER BOOK	3	6	8
PERSONAL COMPUTING 2ND EDITION	2	4	6
PRACTICAL BASIC - IBM PERSONAL COMPUTER		4	8
PRACTICAL BASIC - APPLE	2	4	8
PROBLEM SOLVING PRINCIPLES FOR PROGRAMMERS			4
PROGRAMMER'S GUIDE TO CP/M			4
PROGRAMMING THE Z80			4
PROGRAMMING THE 6502			4
READER'S GUIDE TO MICROCOMPUTER BOOKS	2	5	8
SO YOU'RE THINKING ABOUT-SMALL BUS. COMP.		3	6
UNDERSTANDING ATARI GRAPHICS	3	4	5
USING 6502 ASSEMBLY LANGUAGE			4
VIC-20 PROGRAMMER'S REFERENCE GUIDE	2	5	8
WEBSTER'S MICROCOMPUTER BUYER'S GUIDE	2	4	6
WORD PROCESSING BOOK	2	5	8

TITLE	STOCK LEVELS		
	A	B	C
WORD PROCESSING PRIMER	2	2	4
WRITING BASIC ADVENTURE PROG. FOR TRS-80		4	5
YOUR ATARI COMPUTER	2	4	8
YOUR FIRST COMPUTER	3	5	7
YOUR OWN COMPUTER	2	4	6
ZX-81 BASIC BOOK			3
ZX-81 COMPANION		3	4
32 BASIC PROGRAMS FOR THE APPLE COMPUTER	2	4	8
67 READY-TO-RUN PROGRAMS IN BASIC	3	5	8
8086/8088 MICROPROCESSOR PRIMER			3
<hr/>			
TOTAL UNITS	84	233	512
TOTAL TITLES	38	61	98
TOTAL RETAIL	A - \$1029.68	B - \$2992.96	C - \$6790.93

Title Index

TITLE	PAGE NO.
ADA: AN INTRODUCTION and ADA REFERENCE MANUAL	167
ADDISON-WESLEY BOOK OF APPLE COMPUTER SOFTWARE 1983	238
ADVANCED BASIC	122
ANS COBOL 2nd Edition	168
APPLE BACKPACK	239
APPLE BASIC: DATA FILE PROGRAMMING	240
APPLE CONNECTION	241
APPLE FORTRAN	242
APPLE GRAPHICS & ARCADE GAME DESIGN	243
APPLE II ASSEMBLY LANGUAGE	244
APPLE II ASSEMBLY LANGUAGE EXERCISES	245
APPLE II BASIC QUICK REFERENCE GUIDE	246
APPLE II PROGRAMMER'S HANDBOOK	246
APPLE II USER'S GUIDE	247
APPLE II WORD PROCESSING	247
APPLE INTERFACING	249
APPLE LOGO	250
APPLE MACHINE LANGUAGE	250
APPLE PASCAL GAMES	251
APPLESOFT LANGUAGE	252
ARE YOU COMPUTER LITERATE?	2
ASSEMBLY LANGUAGE PROGRAMMING FOR THE APPLE II	253
ATARI ASSEMBLER	311
ATARI BASIC	312
ATARI BASIC QUICK REFERENCE GUIDE	313
ATARI GAMES & RECREATIONS	313
ATARI SOUND AND GRAPHICS	314
BAG OF TRICKS	254
BASIC 2nd Edition	123
BASIC AND THE PERSONAL COMPUTER	125
BASIC APPLE BASIC	254
BASIC BASIC 2nd Edition	126
BASIC BASIC-ENGLISH DICTIONARY	126
BASIC BOOK	127
BASIC BUSINESS SOFTWARE	182
BASIC COMPUTER GAMES	182

TITLE	PAGE NO.
BASIC COMPUTER PROGRAMS FOR BUSINESS, VOL.1	184
BASIC COMPUTER PROGRAMS FOR BUSINESS, VOL.2	184
BASIC COMPUTER PROGRAMS FOR THE HOME	185
BASIC COMPUTER PROGRAMS IN SCIENCE AND ENGINEERING	186
BASIC EXERCISES FOR THE IBM PERSONAL COMPUTER	275
BASIC FASTER AND BETTER & OTHER MYSTERIES	286
BASIC FOR HOME COMPUTERS	129
BASIC FOR THE APPLE II	255
BASIC FOR THE APPLE II: PROGRAMMING AND APPLICATIONS	256
BASIC FORTRAN	147
BASIC FROM THE GROUND UP	130
BASIC PROGRAMMER'S NOTEBOOK	132
BASIC PROGRAMMING PRIMER	133
BASIC PROGRAMS FOR HOME FINANCIAL MANAGEMENT	187
BASIC PROGRAMS FOR SCIENTISTS AND ENGINEERS	188
BASIC WITH STYLE: PROGRAMMING PROVERBS	134
BASIC WORKBOOK	135
BASIC: ADVANCED CONCEPTS	123
BASIC: FUNDAMENTAL CONCEPTS	123
BEAT THE ODDS	189
BEGINNER'S GUIDE TO BUYING A PERSONAL COMPUTER	3
BEGINNER'S GUIDE TO COMPUTERS	3
BENEATH APPLE DOS	257
BIT OF BASIC	136
BLUE BOOK FOR THE APPLE COMPUTER	258
BOOK OF ATARI SOFTWARE 1983	315
BOOLEAN ALGEBRA FOR COMPUTER LOGIC	114
BRAIN TICKLERS: PUZZLES & PASTIMES	190
BUILD YOUR OWN Z80 COMPUTER	91
BUSINESS SYSTEM BUYER'S GUIDE	4
BYTEING DEEPER INTO YOUR TIMEX/SINCLAIR 1000	332
C PRIMER	169
CBASIC USER GUIDE	137
CELESTIAL BASIC	191
CHESS AND COMPUTERS	192
CIARCIA'S CIRCUIT CELLAR, VOL.1	92
CIARCIA'S CIRCUIT CELLAR, VOL.2	93
CIARCIA'S CIRCUIT CELLAR, VOL.3	93
CIRCUIT DESIGN PROGRAMS FOR THE APPLE II	258
CIRCUIT DESIGN PROGRAMS FOR THE TRS-80	287
COBOL WITH STYLE: PROGRAMMING PROVERBS	170

TITLE	PAGE NO.
CODES, CIPHERS AND COMPUTERS	193
COMMODORE SOFTWARE ENCYCLOPEDIA 2nd Edition	323
COMMODORE SUPERPET COMPUTERS SYSTEM OVERVIEW	324
COMMODORE SUPERPET COMPUTERS WATERLOO MicroBASIC	325
COMMODORE SUPERPET COMPUTERS WATERLOO MicroFORTRAN	325
COMPLETE BOOK OF HOME COMPUTERS	5
COMPLETE HANDBOOK OF PERSONAL COMPUTER COMMUNICATIONS	6
COMPUTE!'S FIRST BOOK OF ATARI	316
COMPUTE!'S FIRST BOOK OF ATARI GRAPHICS	318
COMPUTE!'S FIRST BOOK OF PET/CBM	326
COMPUTE!'S FIRST BOOK OF VIC	327
COMPUTE!'S SECOND BOOK OF ATARI	317
COMPUTER CAPERS	7
COMPUTER CONSCIOUSNESS	8
COMPUTER DICTIONARY & HANDBOOK 3rd Edition	9
COMPUTER DICTIONARY 3rd Edition	9
COMPUTER ESTABLISHMENT	11
COMPUTER GRAPHICS PRIMER	115
COMPUTER LANGUAGE REFERENCE GUIDE	11
COMPUTER LITERACY	12
COMPUTER PROGRAMMING FOR KIDS AND OTHER BEGINNERS	259
COMPUTER WORD PROCESSING	227
COMPUTERS AND EDUCATION	13
COMPUTERS AND PROGRAMMING GUIDE	14
COMPUTERS FOR EVERYBODY	15
COMPUTERS FOR KIDS - APPLE EDITION	260
COMPUTERS FOR KIDS - ATARI EDITION	318
COMPUTERS FOR KIDS - SINCLAIR EDITION	333
COMPUTERS FOR KIDS - TRS-80 EDITION	288
COMPUTERS FOR KIDS - VIC-20 EDITION	328
COMPUTERS FOR PEOPLE	15
COMPUTERS IN NUMBER THEORY	116
CONSUMER'S GUIDE TO PERSONAL COMPUTING & MICROCOMPUTERS	16
COURSEWARE IN THE CLASSROOM	17
CP/M HANDBOOK	94
CP/M PRIMER	95
CP/M REVEALED	95
CP/M SIMPLIFIED	97
CRASH COURSE IN MICROCOMPUTERS	18

TITLE	PAGE NO.
CREATE WORD PUZZLES WITH YOUR MICROCOMPUTER	194
CREATIVE APPLE	260
CRUNCHERS	333
CUSTOM TRS-80 & OTHER MYSTERIES	289
DATA BASE MANAGEMENT SYSTEMS	195
DATA FILE PROGRAMMING IN BASIC	137
DATA MANAGEMENT TECHNIQUES	290
DBASE II USER'S GUIDE	196
DEVIL'S DP DICTIONARY	19
DISCOVER FORTH	170
DISCOVERING BASIC	138
DOING BUSINESS WITH VISICALC	219
DON'T!	20
ELECTRONIC COTTAGE	21
ELECTRONICALLY SPEAKING	197
ELEMENTARY APPLE	261
ELEMENTARY BASIC	139
ELEMENTARY PASCAL	153
ESSENTIAL COMPUTER DICTIONARY AND SPELLER	22
EXECUTIVE COMPUTING	198
EXECUTIVE PLANNING WITH BASIC	199
EXECUTIVE VISICALC FOR THE APPLE COMPUTER	220
EXPERIMENTS IN ARTIFICIAL INTELL. FOR SMALL COMPUTERS	200
FAST BASIC	290
FIFTY BASIC EXERCISES	139
FIRST BOOK OF MICROCOMPUTERS	22
FIRST COURSE IN COMPUTER PROGRAMMING USING PASCAL	153
FORTH PROGRAMMING	171
FORTRAN FUNDAMENTALS: A SHORT COURSE	148
FORTRAN IV 2nd Edition	148
FORTRAN IV POCKET HANDBOOK	149
FORTRAN PROGRAMS FOR SCIENTIST AND ENGINEERS	200
FORTRAN WITH STYLE: PROGRAMMING PROVERBS	150
FOUNDATIONS OF COMPUTER TECHNOLOGY	23
FROM CHIPS TO SYSTEMS	63
FUN WITH MICROCOMPUTERS AND BASIC	201
FUNDAMENTALS OF DIGITAL COMPUTERS	65
FUNDAMENTALS OF MICROCOMPUTER PROGRAMMING	155
GAME PLAYING WITH BASIC	201
GENIE IN THE COMPUTER	291
GETTING ACQUAINTED WITH MICROCOMPUTERS	97
GETTING ACQUAINTED WITH YOUR VIC 20	328

TITLE	PAGE NO.
GETTING ACQUAINTED WITH YOUR ZX81	334
GIANT BOOK OF COMPUTER SOFTWARE	202
GRAPHICS COOKBOOK FOR THE APPLE	262
GUIDE TO PL/M PROGRAMMING FOR MICROCOMPUTER APPLICATIONS	172
GUIDE TO PROGRAMMING IBM PERSONAL COMPUTER	275
GUIDE TO PROGRAMMING IN APPLESOFT	262
GUIDE TO PROGRAMMING IN LEVEL II BASIC	292
GUIDE TO SYSTEMS APPLICATIONS	24
GUIDEBOOK TO SMALL COMPUTERS	25
HANDS-ON BASIC	263
HEXADECIMAL CHRONICLES	116
HOME COMPUTERS	26
HOME COMPUTERS VOL 1: HARDWARE	27
HOME COMPUTERS VOL 2: SOFTWARE	27
HOW TO BUY A PERSONAL COMPUTER	28
HOW TO BUY A PERSONAL COMPUTER (WITHOUT ANXIETY)	29
HOW TO BUY A WORD PROCESSOR	228
HOW TO BUY AND USE MINICOMPUTERS & MICROCOMPUTERS	29
HOW TO BUY THE RIGHT SMALL BUSINESS COMPUTER SYSTEM	30
HOW TO CHOOSE YOUR SMALL BUSINESS COMPUTER	31
HOW TO GET STARTED WITH CP/M	98
HOW TO PROFIT FROM YOUR PERSONAL COMPUTER	31
HOW TO PROGRAM MICROCOMPUTERS	65
HOW TO USE VISICALC SUPERCALC	221
HOW TO WRITE A TRS-80 PROGRAM	292
HOW TO WRITE AN APPLE PROGRAM	263
HOW TO WRITE AN IBM-PC PROGRAM	276
I SPEAK BASIC TO MY APPLE	264
I SPEAK BASIC TO MY PET	329
I SPEAK BASIC TO MY TRS-80	293
IBM PC BASIC QUICK REFERENCE GUIDE	276
IBM PERSONAL COMPUTER	278
IBM PERSONAL COMPUTER	279
IBM/PC GUIDE	277
IBM'S PERSONAL COMPUTER	279
INSIDE ATARI DOS	319
INSIDE BASIC GAMES	203
INSTANT FREEZE-DRIED COMPUTER PROGRAMMING IN BASIC	140
INTERMEDIATE PROGRAMMING FOR THE TRS-80 MODEL I	294

TITLE	PAGE NO.
INTERNATIONAL MICROCOMPUTER DICTIONARY	32
INTIMATE INSTRUCTIONS IN INTEGER BASIC	141
INTRODUCTION TO COMPUTER ANIMATION	204
INTRODUCTION TO COMPUTER PROGRAMMING	142
INTRODUCTION TO FORTH	172
INTRODUCTION TO GRAPHICS	295
INTRODUCTION TO MICROCOMPUTERS FOR THE HAM SHACK	33
INTRODUCTION TO MICROCOMPUTERS-VOL. 0	34
INTRODUCTION TO MICROCOMPUTERS-VOL. 1	66
INTRODUCTION TO MICROPROCESSORS	99
INTRODUCTION TO PASCAL	156
INTRODUCTION TO THE UCSD p-SYSTEM	99
INTRODUCTION TO WORD PROCESSING	229
INTRODUCTION TO WORDSTAR	230
INTRODUCTION TO 8080/8085 ASSEMBLY LANGUAGE PROGRAMMING	67
INVESTOR'S COMPUTER HANDBOOK	35
KIDS & THE APPLE	264
KIDS AND COMPUTERS	35
LEARNING IBM BASIC FOR THE PERSONAL COMPUTER	280
LEARNING TRS-80 BASIC FOR MODELS I, II/16, AND III	296
LEGAL CARE FOR YOUR SOFTWARE	205
LISP	173
LITTLE BOOK OF BASIC STYLE	143
LOW-COST WORD PROCESSING	230
MAKING THE MOST OF YOUR ZX-81	335
MAPPING THE ATARI	320
MASTERING CP/M	100
MASTERING MACHINE CODE ON YOUR ZX-81	335
MASTERING VISICALC	222
MASTERING YOUR TIMEX/SINCLAIR 1000 PERSONAL COMPUTER	336
MICROCOMPUTER BUYER'S GUIDE	36
MICROCOMPUTER DATA-BASE MANAGEMENT	117
MICROCOMPUTER DICTIONARY 2nd Edition	37
MICROCOMPUTER DISK TECHNIQUES	101
MICROCOMPUTER OPERATING SYSTEMS	102
MICROCOMPUTER PRIMER 2nd Edition	37
MICROCOMPUTER SYSTEMS AND APPLE BASIC	265
MICROCOMPUTERS AND THE 3 R's	39
MICROCOMPUTERS FOR BUSINESS APPLICATIONS	40
MICROPROCESSOR OPERATING SYSTEMS, VOL.II	103
MICROPROCESSORS	69

TITLE	PAGE NO.
MICROSOFT BASIC 2nd Edition	143
MICROSOFT BASIC DECODED & OTHER MYSTERIES FOR THE TRS-80	297
MODERN COMPUTER CONCEPTS	40
MORE BASIC COMPUTER GAMES	206
MORE CHESS AND COMPUTERS	206
MOST POPULAR SUBROUTINES IN BASIC	207
MOSTLY BASIC - APPLICATIONS FOR YOUR APPLE II/ BOOK 1	267
MOSTLY BASIC - APPLICATIONS FOR YOUR APPLE II/ BOOK 2	268
MOSTLY BASIC - APPLICATIONS FOR YOUR PET	329
MOSTLY BASIC - APPLICATIONS FOR YOUR TRS-80/ BOOK 1	297
MOSTLY BASIC - APPLICATIONS FOR YOUR TRS-80/ BOOK 2	298
MURDER IN THE MANSION 2nd Edition	208
NAILING JELLY TO A TREE	41
NCR DATA COMMUNICATIONS CONCEPTS	42
OSBORNE CP/M USER GUIDE	104
OSBORNE 16-BIT MICROPROCESSOR HANDBOOK	70
OSBORNE 4 & 8-BIT MICROPROCESSOR HANDBOOK	69
PASCAL	157
PASCAL FOR BASIC PROGRAMMERS	159
PASCAL FOR PROGRAMMERS	160
PASCAL FROM BASIC	160
PASCAL HANDBOOK	161
PASCAL PRIMER	162
PASCAL PROGRAMMING FOR THE APPLE	269
PASCAL PROGRAMS FOR SCIENTISTS AND ENGINEERS	209
PASCAL USER MANUAL AND REPORT 2nd Edition	163
PASCAL WITH STYLE: PROGRAMMING PROVERBS	164
PASCAL: A PROBLEM SOLVING APPROACH	158
PEANUT BUTTER & JELLY GUIDE TO COMPUTERS	43
PERSONAL COMPUTER BOOK	44
PERSONAL COMPUTERS HANDBOOK	105
PERSONAL COMPUTERS IN BUSINESS	44
PERSONAL COMPUTING 2nd Edition	45
PERSONAL GUIDE TO PERSONAL COMPUTERS	46
PLAYING THE STOCK & BOND MARKETS	209
POCKET COMPUTER PROGRAMMING MADE EASY	144
POCKET GUIDE TO FORTRAN	151
POCKET GUIDE TO PASCAL	164
POCKET GUIDE TO PROGRAMMING	118
PRACTICAL BASIC PROGRAMS - APPLE II EDITION	270

TITLE	PAGE NO.
PRACTICAL BASIC PROGRAMS - IBM PERSONAL COMPUTER EDITION	281
PRACTICAL BASIC PROGRAMS - TRS-80 LEVEL II EDITION	298
PRACTICAL PASCAL PROGRAMS	210
PRACTICE PROBLEMS IN NUMBER SYSTEMS, LOGIC & BOOLEAN ALG.	11
PROBLEM SOLVING AND STRUCTURED PROGRAMMING IN PASCAL	165
PROBLEM SOLVING PRINCIPLES FOR ADA PROGRAMMERS	175
PROBLEM SOLVING PRINCIPLES FOR PROGRAMMERS	120
PROFESSIONAL PROGRAMMING TECHNIQUES	145
PROGRAMMER'S GUIDE TO CP/M	105
PROGRAMMING A MICROCOMPUTER: 6502	71
PROGRAMMING PROVERBS FOR FORTRAN PROGRAMMERS	152
PROGRAMMING THE Z80	72
PROGRAMMING THE 6502	73
PROGRAMS FOR BEGINNERS ON THE TRS-80	298
RUNNING WILD	47
S-100 & OTHER MICRO BUSES 2nd Edition	106
S-100 BUS HANDBOOK	107
SCIENCE AND ENGINEERING PROGRAMS APPLE II EDITION	266
SINCLAIR ZX-81	337
SIXTY CHALLENGING PROBLEMS WITH BASIC SOLUTIONS	211
SMALL COMPUTERS FOR THE SMALL BUSINESSMAN SO YOU ARE THINKING ABOUT A SMALL BUSINESS COMPUTER	48
SOFTWARE ENGINEERING FOR MICROS	120
SOME COMMON BASIC PROGRAMS - APPLE II EDITION	267
SOME COMMON BASIC PROGRAMS - ATARI EDITION	321
SOME COMMON BASIC PROGRAMS - IBM PERSONAL COMPUTER ED.	282
SOME COMMON BASIC PROGRAMS - PET/CBM EDITION	330
SOME COMMON PASCAL PROGRAMS	212
SOUL OF A NEW MACHINE	50
SPEAKING PASCAL	165
STARTING FORTH	175
STIMULATING SIMULATIONS	213
STIMULATING SIMULATIONS ATARI VERSION	213
STRUCTURED COBOL	176
SUCCESSFUL SOFTWARE FOR SMALL COMPUTERS	214

TITLE	PAGE NO.
SWIFT'S EDUCATIONAL SOFTWARE DIRECTORY	270
TECHNIQUES FOR CREATING GOLDEN DELICIOUS GAMES	271
TECHNIQUES OF BASIC	299
TEN EASY PIECES	215
THE BASIC CONVERSION HANDBOOK	128
THE BASIC COOKBOOK	129
THE BASIC HANDBOOK	131
TIMEX PERSONAL COMPUTER MADE SIMPLE	338
TIMEX-SINCLAIR 1983 DIRECTORY	339
TRS-80 ASSEMBLY LANGUAGE MADE SIMPLE	301
TRS-80 BASIC	302
TRS-80 COBOL	303
TRS-80 COLOR BASIC	304
TRS-80 DISK & OTHER MYSTERIES	305
TRS-80 FOR KIDS FROM 8 TO 80 Volume I	306
TRS-80 GRAPHICS	307
TRS-80 MORE THAN BASIC	308
TRSDOS 2.3 DECODED & OTHER MYSTERIES	300
UNDERSTANDING APL	177
UNDERSTANDING ARTIFICIAL INTELLIGENCE	51
UNDERSTANDING ATARI GRAPHICS	321
UNDERSTANDING BASIC	146
UNDERSTANDING COBOL	178
UNDERSTANDING COMPUTER SYSTEMS	51
UNDERSTANDING COMPUTERS	52
UNDERSTANDING COMPUTERS	53
UNDERSTANDING DATA BASE MANAGEMENT	54
UNDERSTANDING FORTRAN	152
UNDERSTANDING LISP	178
UNDERSTANDING PASCAL	166
UNDERSTANDING THE IBM PERSONAL COMPUTER VOL.I	282
UNDERSTANDING THE MICRO	54
USER GUIDE TO THE UNIX SYSTEM	108
USER'S GUIDE TO PET/CBM COMPUTERS	330
USER'S HANDBOOK TO THE IBM PERSONAL COMPUTER	283
USING CP/M	109
USING MICROCOMPUTERS IN BUSINESS	55
USING THE IBM PERSONAL COMPUTER	284
USING THE IBM PERSONAL COMPUTER	285
USING YOUR IBM PERSONAL COMPUTER	285
USING 6502 ASSEMBLY LANGUAGE	272
VIC-20 PROGRAMMER'S REFERENCE GUIDE	331

TITLE	PAGE NO.
VISICALC BOOK - APPLE EDITION	224
VISICALC BOOK - ATARI EDITION	224
VISICALC FOR APPLE II PLUS	224
VISICALC FOR THE IBM PERSONAL COMPUTER	225
VISICALC FOR THE TRS-80 MODEL I & III COMPUTERS	225
VISICALC FOR THE TRS-80 MODEL II & 16 COMPUTERS	225
VISICALC HOME AND OFFICE COMPANION	226
VISICALC, ADVANCED VERSION	223
WHY DO YOU NEED A PERSONAL COMPUTER?	56
WILL SOMEONE PLEASE TELL ME WHAT AN APPLE CAN DO ?	273
WITHOUT ME YOU'RE NOTHING	57
WORD PROCESSING BOOK	231
WORD PROCESSING PRIMER	232
WORDSTAR MADE EASY	233
WRITING BASIC ADVENTURE PROGRAMS FOR THE TRS-80	309
WRITING WITH A WORD PROCESSOR	234
YOUR ATARI COMPUTER	322
YOUR FIRST COMPUTER	58
YOUR OWN COMPUTER	59
YOUR TIMEX SINCLAIR 1000 & ZX81	340
ZX-81 BASIC BOOK	340
ZX-81 COMPANION	341
Z80 AND 8080 ASSEMBLY LANGUAGE PROGRAMMING	74
Z80 ASSEMBLY LANGUAGE PROGRAMMING	75
Z80 INSTRUCTION HANDBOOK	77
Z80 MICROPROCESSOR PROGRAMMING & INTERFACING	77
Z80 SOFTWARE GOURMET GUIDE AND COOKBOOK	78
Z80 USERS MANUAL	79
Z8000 ASSEMBLY LANGUAGE PROGRAMMING	80
101 TIMEX 1000/SINCLAIR ZX-81 PROGRAMMING TIPS & TRICKS	342
1983 APPLE II/III SOFTWARE DIRECTORY -Volume II	249
32 BASIC PROGRAMS FOR THE APPLE COMPUTER	274
33 CHALLENGING COMPUTER GAMES FOR TRS-80/ APPLE/PET	216
37 TIMEX 1000 SINCLAIR ZX-81 PROGRAMS	343
49 EXPLOSIVE GAMES FOR THE ZX-81	344
50 PROGRAMS IN BASIC - 2nd Edition	216
55 ADVANCED COMPUTER PROGRAMS IN BASIC	217
57 PRACTICAL PROGRAMS & GAMES IN BASIC	218
60-MINUTE GUIDE TO MICROCOMPUTERS	60
6502 ASSEMBLY LANGUAGE PROGRAMMING	82
6502 SOFTWARE GOURMET GUIDE & COOKBOOK	82

TITLE	PAGE NO.
67 READY TO RUN PROGRAMS IN BASIC	219
6800 SOFTWARE GOURMET GUIDE & COOKBOOK	82
68000 ASSEMBLY LANGUAGE PROGRAMMING	84
68000: PRINCIPLES AND PROGRAMMING	83
80 PRACTICAL TIME-SAVING PROGRAMS FOR THE TRS-80	310
8080 SOFTWARE GOURMET GUIDE & COOKBOOK 2nd Edition	84
8080/Z80 ASSEMBLY LANGUAGE	84
8086/8088 PRIMER 2nd Edition	86
8086/8088 16-BIT MICROPROCESSOR PRIMER	87

Rating Index

RATING	TITLE	PAGE NO.
100	8086/8088 16-BIT MICROPROCESSOR PRIMER	87
100	THE BASIC HANDBOOK	131
100	32 BASIC PROGRAMS FOR THE APPLE COMPUTER	274
100	BASIC FASTER AND BETTER & OTHER MYSTERIES	286
95	COMPUTER DICTIONARY 3rd Edition	9
95	COMPUTER DICTIONARY & HANDBOOK 3rd Edition	9
95	CRASH COURSE IN MICROCOMPUTERS	18
95	GUIDE TO SYSTEMS APPLICATIONS	24
95	HOME COMPUTERS VOL 1: HARDWARE	27
95	KIDS AND COMPUTERS	35
95	MICROCOMPUTER DICTIONARY 2nd Edition	37
95	NAILING JELLY TO A TREE	41
95	PERSONAL COMPUTER BOOK	44
95	SOUL OF A NEW MACHINE	50
95	FROM CHIPS TO SYSTEMS	63
95	OSBORNE 4 & 8-BIT MICROPROCESSOR HANDBOOK	69
95	OSBORNE 16-BIT MICROPROCESSOR HANDBOOK	70
95	PROGRAMMING THE Z80	72
95	PROGRAMMING THE 6502	73
95	8080/Z80 ASSEMBLY LANGUAGE	84
95	BUILD YOUR OWN Z80 COMPUTER	91
95	CIARCIA'S CIRCUIT CELLAR, VOL.1	92
95	CIARCIA'S CIRCUIT CELLAR, VOL.2	93
95	CIARCIA'S CIRCUIT CELLAR, VOL.3	93
95	CP/M HANDBOOK	94
95	INTRODUCTION TO THE UCSD p-SYSTEM	99
95	MASTERING CP/M	100
95	BOOLEAN ALGEBRA FOR COMPUTER LOGIC	114
95	HEXADECIMAL CHRONICLES	116
95	BASIC AND THE PERSONAL COMPUTER	125
95	DATA FILE PROGRAMMING IN BASIC	137
95	INTRODUCTION TO PASCAL	156
95	PASCAL FOR BASIC PROGRAMMERS	159

RATING	TITLE	PAGE NO.
95	PASCAL HANDBOOK	161
95	BASIC PROGRAMS FOR HOME FINANCIAL MANAGEMENT	187
95	CHESS AND COMPUTERS	192
95	CREATE WORD PUZZLES WITH YOUR MICROCOMPUTER	194
95	ELECTRONICALLY SPEAKING	197
95	INTRODUCTION TO WORDSTAR	230
95	APPLE BACKPACK	239
95	APPLE BASIC: DATA FILE PROGRAMMING	240
95	APPLE II USER'S GUIDE	247
95	APPLE II WORD PROCESSING	247
95	BASIC APPLE BASIC	254
95	CIRCUIT DESIGN PROGRAMS FOR THE APPLE II	258
95	PASCAL PROGRAMMING FOR THE APPLE	269
95	IBM/PC GUIDE	277
95	LEARNING IBM BASIC FOR THE PERSONAL COMPUTER	280
95	USING YOUR IBM PERSONAL COMPUTER	285
95	CIRCUIT DESIGN PROGRAMS FOR THE TRS-80	287
95	LEARNING TRS-80 BASIC FOR MODELS I, II/16, AND III	296
95	TRSDOS 2.3 DECODED & OTHER MYSTERIES	300
95	WRITING BASIC ADVENTURE PROGRAMS FOR THE TRS-80	309
95	ATARI SOUND AND GRAPHICS	314
95	BOOK OF ATARI SOFTWARE 1983	315
95	COMPUTE!'S FIRST BOOK OF PET/CBM	326
95	VIC-20 PROGRAMMER'S REFERENCE GUIDE	331
95	BYTEING DEEPER INTO YOUR TIMEX/ SINCLAIR 1000	332
95	ZX-81 COMPANION	341
90	COMPLETE HANDBOOK OF PERSONAL COMPUTER COMMUNICATIONS	6
90	COMPUTER ESTABLISHMENT	11
90	COMPUTERS FOR EVERYBODY	15
90	DEVIL'S DP DICTIONARY	19
90	ESSENTIAL COMPUTER DICTIONARY AND SPELLER	22
90	FIRST BOOK OF MICROCOMPUTERS	22
90	MICROCOMPUTER BUYER'S GUIDE	36
90	MODERN COMPUTER CONCEPTS	40

RATING	TITLE	PAGE NO.
90	PERSONAL COMPUTING 2nd Edition	45
90	SO YOU ARE THINKING ABOUT A SMALL BUSINESS COMPUTER	49
90	UNDERSTANDING COMPUTER SYSTEMS	51
90	YOUR FIRST COMPUTER	58
90	YOUR OWN COMPUTER	59
90	INTRODUCTION TO MICROCOMPUTERS- VOL. 1	66
90	Z8000 ASSEMBLY LANGUAGE PROGRAMMING	80
90	68000 ASSEMBLY LANGUAGE PROGRAMMING	84
90	8086/8088 PRIMER 2nd Edition	86
90	CP/M PRIMER	95
90	OSBORNE CP/M USER GUIDE	104
90	PROGRAMMER'S GUIDE TO CP/M	105
90	COMPUTER GRAPHICS PRIMER	115
90	COMPUTERS IN NUMBER THEORY	116
90	PRACTICE PROBLEMS IN NUMBER SYSTEMS, LOGIC & BOOLEAN ALG.	119
90	PROBLEM SOLVING PRINCIPLES FOR PROGRAMMERS	120
90	ADVANCED BASIC	122
90	BASIC BASIC 2nd Edition	126
90	BASIC PROGRAMMER'S NOTEBOOK	132
90	BASIC PROGRAMMING PRIMER	133
90	LITTLE BOOK OF BASIC STYLE	143
90	PASCAL	157
90	PASCAL: A PROBLEM SOLVING APPROACH	158
90	PASCAL FOR PROGRAMMERS	160
90	PROBLEM SOLVING AND STRUCTURED PROGRAMMING IN PASCAL	165
90	C PRIMER	169
90	LISP	173
90	BASIC PROGRAMS FOR SCIENTISTS AND ENGINEERS	188
90	CELESTIAL BASIC	191
90	EXECUTIVE COMPUTING	198
90	EXECUTIVE PLANNING WITH BASIC	199
90	EXPERIMENTS IN ARTIFICIAL INTELL. FOR SMALL COMPUTERS	200
90	FORTRAN PROGRAMS FOR SCIENTIST AND ENGINEERS	200
90	LEGAL CARE FOR YOUR SOFTWARE	205
90	MORE BASIC COMPUTER GAMES	206

RATING	TITLE	PAGE NO.
90	PASCAL PROGRAMS FOR SCIENTISTS AND ENGINEERS	209
90	PRACTICAL PASCAL PROGRAMS	210
90	MASTERING VISICALC	222
90	VISICALC FOR APPLE II PLUS	224
90	VISICALC FOR THE IBM PERSONAL COMPUTER	225
90	VISICALC FOR THE TRS-80 MODEL I & III COMPUTERS	225
90	VISICALC FOR THE TRS-80 MODEL II & 16 COMPUTERS	225
90	WORD PROCESSING PRIMER	232
90	WRITING WITH A WORD PROCESSOR	234
90	ADDISON-WESLEY BOOK OF APPLE COMPUTER SOFTWARE 1983	238
90	APPLE GRAPHICS & ARCADE GAME DESIGN	243
90	APPLE II PROGRAMMER'S HANDBOOK	246
90	APPLE MACHINE LANGUAGE	250
90	APPLE PASCAL GAMES	251
90	BENEATH APPLE DOS	257
90	COMPUTERS FOR KIDS - APPLE EDITION	260
90	CREATIVE APPLE	260
90	GUIDE TO PROGRAMMING IN APPLESOFT	262
90	KIDS & THE APPLE	264
90	MOSTLY BASIC - APPLICATIONS FOR YOUR APPLE II/BOOK 2	268
90	PRACTICAL BASIC PROGRAMS - APPLE II EDITION	270
90	USING 6502 ASSEMBLY LANGUAGE	272
90	GUIDE TO PROGRAMMING IBM PERSONAL COMPUTER	275
90	IBM'S PERSONAL COMPUTER	279
90	PRACTICAL BASIC PROGRAMS - IBM PERSONAL COMPUTER EDITION	281
90	USING THE IBM PERSONAL COMPUTER	284
90	COMPUTERS FOR KIDS - TRS-80 EDITION	288
90	GUIDE TO PROGRAMMING IN LEVEL II BASIC	292
90	MICROSOFT BASIC DECODED & OTHER MYSTERIES FOR THE TRS-80	297
90	MOSTLY BASIC - APPLICATIONS FOR YOUR TRS-80/BOOK 2	298
90	PRACTICAL BASIC PROGRAMS - TRS-80 LEVEL II EDITION	298
90	TRS-80 COBOL	303

RATING	TITLE	PAGE NO.
90	TRS-80 DISK & OTHER MYSTERIES	305
90	TRS-80 MORE THAN BASIC	308
90	ATARI ASSEMBLER	311
90	COMPUTE!'S FIRST BOOK OF ATARI	316
90	COMPUTERS FOR KIDS - ATARI EDITION	318
90	YOUR ATARI COMPUTER	322
90	COMPUTERS FOR KIDS - VIC-20 EDITION	328
90	COMPUTERS FOR KIDS - SINCLAIR EDITION	333
90	MASTERING YOUR TIMEX/SINCLAIR 1000 PERSONAL COMPUTER	336
85	COMPUTER CAPERS	7
85	DON'T!	20
85	HOME COMPUTERS VOL 2: SOFTWARE	27
85	HOW TO BUY A PERSONAL COMPUTER	28
85	INTRODUCTION TO MICROCOMPUTERS- VOL. 0	34
85	MICROCOMPUTERS AND THE 3 R's	39
85	PEANUT BUTTER & JELLY GUIDE TO COMPUTERS	43
85	UNDERSTANDING COMPUTERS	52
85	USING MICROCOMPUTERS IN BUSINESS	55
85	HOW TO PROGRAM MICROCOMPUTERS	65
85	Z80 AND 8080 ASSEMBLY LANGUAGE PROGRAMMING	74
85	Z80 ASSEMBLY LANGUAGE PROGRAMMING	75
85	6502 ASSEMBLY LANGUAGE PROGRAMMING	82
85	MICROCOMPUTER OPERATING SYSTEMS	102
85	MICROPROCESSOR OPERATING SYSTEMS, VOL.II	103
85	PERSONAL COMPUTERS HANDBOOK	105
85	BASIC FROM THE GROUND UP	130
85	BIT OF BASIC	136
85	FIFTY BASIC EXERCISES	139
85	FORTRAN IV 2nd Edition	148
85	FIRST COURSE IN COMPUTER PROGRAMMING USING PASCAL	153
85	PASCAL PRIMER	162
85	PASCAL USER MANUAL AND REPORT 2nd Edition	163
85	BASIC BUSINESS SOFTWARE	182
85	BASIC COMPUTER GAMES	182
85	BASIC COMPUTER PROGRAMS FOR BUSINESS, VOL.1	184

RATING	TITLE	PAGE NO.
85	BASIC COMPUTER PROGRAMS FOR BUSINESS, VOL.2	184
85	BEAT THE ODDS	189
85	DATA BASE MANAGEMENT SYSTEMS	195
85	DBASE II USER'S GUIDE	196
85	MOST POPULAR SUBROUTINES IN BASIC	207
85	SOME COMMON PASCAL PROGRAMS	212
85	STIMULATING SIMULATIONS	213
85	STIMULATING SIMULATIONS ATARI VERSION	213
85	SUCCESSFUL SOFTWARE FOR SMALL COMPUTERS	214
85	67 READY TO RUN PROGRAMS IN BASIC	219
85	DOING BUSINESS WITH VISICALC	219
85	VISICALC, ADVANCED VERSION	223
85	WORD PROCESSING BOOK	231
85	APPLE CONNECTION	241
85	APPLE II BASIC QUICK REFERENCE GUIDE	246
85	BAG OF TRICKS	254
85	BASIC FOR THE APPLE II	255
85	ELEMENTARY APPLE	261
85	SCIENCE AND ENGINEERING PROGRAMS APPLE II EDITION	266
85	SOME COMMON BASIC PROGRAMS - APPLE II EDITION	267
85	BASIC EXERCISES FOR THE IBM PERSONAL COMPUTER	275
85	IBM PC BASIC QUICK REFERENCE GUIDE	276
85	SOME COMMON BASIC PROGRAMS - IBM PERSONAL COMPUTER ED.	282
85	USER'S HANDBOOK TO THE IBM PERSONAL COMPUTER	283
85	CUSTOM TRS-80 & OTHER MYSTERIES	289
85	GENIE IN THE COMPUTER	291
85	INTRODUCTION TO GRAPHICS	295
85	PROGRAMS FOR BEGINNERS ON THE TRS-80	298
85	TECHNIQUES OF BASIC	299
85	TRS-80 GRAPHICS	307
85	80 PRACTICAL TIME-SAVING PROGRAMS FOR THE TRS-80	310
85	ATARI BASIC QUICK REFERENCE GUIDE	313
85	COMPUTE!'S SECOND BOOK OF ATARI	317
85	COMPUTE!'S FIRST BOOK OF ATARI GRAPHICS	318

RATING	TITLE	PAGE NO.
85	MAPPING THE ATARI	320
85	SOME COMMON BASIC PROGRAMS - ATARI EDITION	321
85	UNDERSTANDING ATARI GRAPHICS	321
85	SOME COMMON BASIC PROGRAMS - PET/ CBM EDITION	330
85	GETTING ACQUAINTED WITH YOUR ZX81	334
85	MAKING THE MOST OF YOUR ZX-81	335
85	MASTERING MACHINE CODE ON YOUR ZX-81	335
85	SINCLAIR ZX-81	337
85	YOUR TIMEX SINCLAIR 1000 & ZX81	340
80	BEGINNER'S GUIDE TO COMPUTERS	3
80	BUSINESS SYSTEM BUYER'S GUIDE	4
80	COMPUTER CONSCIOUSNESS	8
80	ELECTRONIC COTTAGE	21
80	INVESTOR'S COMPUTER HANDBOOK	35
80	NCR DATA COMMUNICATIONS CONCEPTS	42
80	RUNNING WILD	47
80	SMALL COMPUTERS FOR THE SMALL BUSINESSMAN	48
80	UNDERSTANDING DATA BASE MANAGEMENT	54
80	UNDERSTANDING THE MICRO	54
80	60-MINUTE GUIDE TO MICROCOMPUTERS	60
80	INTRODUCTION TO 8080/8085 ASSEMBLY LANGUAGE PROGRAMMING	67
80	Z80 INSTRUCTION HANDBOOK	77
80	CP/M REVEALED	95
80	CP/M SIMPLIFIED	97
80	GETTING ACQUAINTED WITH MICROCOMPUTERS	97
80	INTRODUCTION TO MICROPROCESSORS	99
80	SOFTWARE ENGINEERING FOR MICROS	120
80	BASIC 2nd Edition	123
80	BASIC: ADVANCED CONCEPTS	123
80	BASIC BASIC-ENGLISH DICTIONARY	126
80	CBASIC USER GUIDE	137
80	DISCOVERING BASIC	138
80	INTRODUCTION TO COMPUTER PROGRAMMING	142
80	MICROSOFT BASIC 2nd Edition	143
80	PROFESSIONAL PROGRAMMING TECHNIQUES	145
80	PASCAL FROM BASIC	160

RATING	TITLE	PAGE NO.
80	ADA: AN INTRODUCTION and ADA REFERENCE MANUAL	167
80	STRUCTURED COBOL	176
80	UNDERSTANDING LISP	178
80	BASIC COMPUTER PROGRAMS IN SCIENCE AND ENGINEERING	186
80	BRAIN TICKLERS: PUZZLES & PASTIMES	190
80	CODES, CIPHERS AND COMPUTERS	193
80	INSIDE BASIC GAMES	203
80	INTRODUCTION TO COMPUTER ANIMATION	204
80	COMPUTER WORD PROCESSING	227
80	1983 APPLE II/III SOFTWARE DIRECTORY - Volume II	249
80	BLUE BOOK FOR THE APPLE COMPUTER	258
80	COMPUTER PROGRAMMING FOR KIDS AND OTHER BEGINNERS	259
80	HANDS-ON BASIC	263
80	SWIFT'S EDUCATIONAL SOFTWARE DIRECTORY	270
80	TRS-80 COLOR BASIC	304
80	TRS-80 FOR KIDS FROM 8 TO 80 Volume I	306
80	COMMODORE SOFTWARE ENCYCLOPEDIA 2nd Edition	323
80	COMPUTE!'S FIRST BOOK OF VIC	327
80	USER'S GUIDE TO PET/CBM COMPUTERS	330
80	ZX-81 BASIC BOOK	340
80	49 EXPLOSIVE GAMES FOR THE ZX-81	344
75	BEGINNER'S GUIDE TO BUYING A PERSONAL COMPUTER	3
75	COMPUTER LANGUAGE REFERENCE GUIDE	11
75	MICROCOMPUTER PRIMER 2nd Edition	37
75	WHY DO YOU NEED A PERSONAL COMPUTER?	56
75	Z80 SOFTWARE GOURMET GUIDE AND COOKBOOK	78
75	6502 SOFTWARE GOURMET GUIDE & COOKBOOK	82
75	6800 SOFTWARE GOURMET GUIDE & COOKBOOK	82
75	8080 SOFTWARE GOURMET GUIDE & COOKBOOK 2nd Edition	84
75	MICROCOMPUTER DISK TECHNIQUES	101
75	USING CP/M	109

RATING	TITLE	PAGE NO.
75	MICROCOMPUTER DATA-BASE MANAGEMENT	117
75	THE BASIC COOKBOOK	129
75	BASIC FOR HOME COMPUTERS	129
75	BASIC FORTRAN	147
75	UNDERSTANDING PASCAL	166
75	GIANT BOOK OF COMPUTER SOFTWARE	202
75	TEN EASY PIECES	215
75	33 CHALLENGING COMPUTER GAMES FOR TRS-80/APPLE/PET	216
75	VISICALC BOOK - APPLE EDITION	224
75	VISICALC BOOK - ATARI EDITION	224
75	APPLE II ASSEMBLY LANGUAGE	244
75	GRAPHICS COOKBOOK FOR THE APPLE	262
75	MOSTLY BASIC - APPLICATIONS FOR YOUR APPLE II/BOOK 1	267
75	USING THE IBM PERSONAL COMPUTER	285
75	DATA MANAGEMENT TECHNIQUES	290
75	FAST BASIC	290
75	MOSTLY BASIC - APPLICATIONS FOR YOUR TRS-80/BOOK 1	297
75	GETTING ACQUAINTED WITH YOUR VIC 20	328
75	MOSTLY BASIC - APPLICATIONS FOR YOUR PET	329
75	CRUNCHERS	333
75	TIMEX PERSONAL COMPUTER MADE SIMPLE	338
75	TIMEX-SINCLAIR 1983 DIRECTORY	339
70	COMPLETE BOOK OF HOME COMPUTERS	5
70	FOUNDATIONS OF COMPUTER TECHNOLOGY	23
70	MICROPROCESSORS	69
70	Z80 MICROPROCESSOR PROGRAMMING & INTERFACING	77
70	USER GUIDE TO THE UNIX SYSTEM	108
70	BASIC: FUNDAMENTAL CONCEPTS	123
70	BASIC WORKBOOK	135
70	POCKET GUIDE TO FORTRAN	151
70	GUIDE TO PL/M PROGRAMMING FOR MICROCOMPUTER APPLICATIONS	172
70	BASIC COMPUTER PROGRAMS FOR THE HOME	185
70	MORE CHESS AND COMPUTERS	206
70	PLAYING THE STOCK & BOND MARKETS	209

RATING	TITLE	PAGE NO.
70	EXECUTIVE VISICALC FOR THE APPLE COMPUTER	220
70	VISICALC HOME AND OFFICE COMPANION	226
70	INTRODUCTION TO WORD PROCESSING	229
70	LOW-COST WORD PROCESSING	230
70	APPLE INTERFACING	249
70	BASIC FOR THE APPLE II: PROGRAMMING AND APPLICATIONS	256
70	IBM PERSONAL COMPUTER	278
70	ATARI GAMES & RECREATIONS	313
70	COMMODORE SUPERPET COMPUTERS SYSTEM OVERVIEW	324
65	ARE YOU COMPUTER LITERATE?	2
65	COMPUTERS AND PROGRAMMING GUIDE	14
65	CONSUMER'S GUIDE TO PERSONAL COMPUTING & MICROCOMPUTERS	16
65	COURSEWARE IN THE CLASSROOM	17
65	INTRODUCTION TO MICROCOMPUTERS FOR THE HAM SHACK	33
65	PERSONAL COMPUTERS IN BUSINESS	44
65	PERSONAL GUIDE TO PERSONAL COMPUTERS	46
65	UNDERSTANDING ARTIFICIAL INTELLIGENCE	51
65	PROGRAMMING A MICROCOMPUTER: 6502	71
65	S-100 BUS HANDBOOK	107
65	ELEMENTARY BASIC	139
65	FORTRAN IV POCKET HANDBOOK	149
65	ELEMENTARY PASCAL	153
65	FUNDAMENTALS OF MICROCOMPUTER PROGRAMMING	155
65	POCKET GUIDE TO PASCAL	164
65	SPEAKING PASCAL	165
65	ANS COBOL 2nd Edition	168
65	HOW TO USE VISICALC SUPERCALC	221
65	IBM PERSONAL COMPUTER	279
65	ATARI BASIC	312
65	INSIDE ATARI DOS	319
60	COMPUTERS FOR PEOPLE	15
60	GUIDEBOOK TO SMALL COMPUTERS	25
60	STARTING FORTH	175
60	FUN WITH MICROCOMPUTERS AND BASIC	201
60	GAME PLAYING WITH BASIC	201
60	WORDSTAR MADE EASY	233
60	APPLE LOGO	250

RATING	TITLE	PAGE NO.
60	APPLESOFT LANGUAGE	252
60	WILL SOMEONE PLEASE TELL ME WHAT AN APPLE CAN DO ?	273
60	INTERMEDIATE PROGRAMMING FOR THE TRS-80 MODEL I	294
60	101 TIMEX 1000/SINCLAIR ZX-81 PROGRAMMING TIPS & TRICKS	342
55	HOME COMPUTERS	26
55	MICROCOMPUTERS FOR BUSINESS APPLICATIONS	40
55	68000: PRINCIPLES AND PROGRAMMING	83
55	BASIC WITH STYLE: PROGRAMMING PROVERBS	134
55	FORTRAN WITH STYLE: PROGRAMMING PROVERBS	150
55	PROGRAMMING PROVERBS FOR FORTRAN PROGRAMMERS	152
55	PASCAL WITH STYLE: PROGRAMMING PROVERBS	164
55	COBOL WITH STYLE: PROGRAMMING PROVERBS	170
55	55 ADVANCED COMPUTER PROGRAMS IN BASIC	217
55	UNDERSTANDING THE IBM PERSONAL COMPUTER VOL.I	282
50	COMPUTERS AND EDUCATION	13
50	HOW TO BUY A PERSONAL COMPUTER (WITHOUT ANXIETY)	29
50	HOW TO BUY THE RIGHT SMALL BUSINESS COMPUTER SYSTEM	30
50	WITHOUT ME YOU'RE NOTHING	57
50	FUNDAMENTALS OF DIGITAL COMPUTERS	65
50	POCKET GUIDE TO PROGRAMMING	118
50	BASIC BOOK	127
50	UNDERSTANDING BASIC	146
50	UNDERSTANDING FORTRAN	152
50	UNDERSTANDING APL	177
50	UNDERSTANDING COBOL	178
50	57 PRACTICAL PROGRAMS & GAMES IN BASIC	218
50	ASSEMBLY LANGUAGE PROGRAMMING FOR THE APPLE II	253
50	HOW TO WRITE AN APPLE PROGRAM	263
50	MICROCOMPUTER SYSTEMS AND APPLE BASIC	265

RATING	TITLE	PAGE NO.
50	TECHNIQUES FOR CREATING GOLDEN DELICIOUS GAMES	271
50	HOW TO WRITE AN IBM-PC PROGRAM	276
50	HOW TO WRITE A TRS-80 PROGRAM	292
50	37 TIMEX 1000 SINCLAIR ZX-81 PROGRAMS	343
45	COMPUTER LITERACY	12
45	HOW TO BUY AND USE MINICOMPUTERS & MICROCOMPUTERS	29
45	UNDERSTANDING COMPUTERS	53
45	Z80 USERS MANUAL	79
45	HOW TO GET STARTED WITH CP/M	98
45	FORTRAN FUNDAMENTALS: A SHORT COURSE	148
45	FORTH PROGRAMMING	171
45	INTRODUCTION TO FORTH	172
45	PROBLEM SOLVING PRINCIPLES FOR ADA PROGRAMMERS	175
45	50 PROGRAMS IN BASIC - 2nd Edition	216
45	HOW TO BUY A WORD PROCESSOR	228
45	APPLE FORTRAN	242
45	APPLE II ASSEMBLY LANGUAGE EXERCISES	245
45	I SPEAK BASIC TO MY APPLE	264
45	I SPEAK BASIC TO MY TRS-80	293
45	I SPEAK BASIC TO MY PET	329
40	INTERNATIONAL MICROCOMPUTER DICTIONARY	32
40	S-100 & OTHER MICRO BUSES 2nd Edition	106
40	THE BASIC CONVERSION HANDBOOK	128
40	TRS-80 BASIC	302
35	DISCOVER FORTH	170
30	HOW TO CHOOSE YOUR SMALL BUSINESS COMPUTER	31
30	HOW TO PROFIT FROM YOUR PERSONAL COMPUTER	31
30	MURDER IN THE MANSION 2nd Edition	208
30	TRS-80 ASSEMBLY LANGUAGE MADE SIMPLE	301
25	COMMODORE SUPERPET COMPUTERS WATERLOO MicroBASIC	325
25	COMMODORE SUPERPET COMPUTERS WATERLOO MicroFORTRAN	325
20	INTIMATE INSTRUCTIONS IN INTEGER BASIC	141
10	INSTANT FREEZE-DRIED COMPUTER PROGRAMMING IN BASIC	140

RATING	TITLE	PAGE NO.
10	POCKET COMPUTER PROGRAMMING MADE EASY	144
10	SIXTY CHALLENGING PROBLEMS WITH BASIC SOLUTIONS	211

Author Index

AUTHOR NAME	TITLE	PAGE NO.
Abelson, Harold	APPLE LOGO	250
Ahl, David H.	BASIC COMPUTER GAMES	182
	MORE BASIC COMPUTER GAMES	206
Albrecht, Robert L.	BASIC FOR THE APPLE II	255
	TRS-80 BASIC	302
	ATARI SOUND AND GRAPHICS	314
	BASIC FOR HOME COMPUTERS	129
	BASIC 2nd Edition	123
	ATARI BASIC	312
	TRS-80 COLOR BASIC	304
Alexander, Daniel E.	FORTRAN IV POCKET HANDBOOK	149
Arca, Julie	WORD PROCESSING PRIMER	232
Ashley, Ruth	USING CP/M	109
	STRUCTURED COBOL	176
	INTRODUCTION TO 8080/8085 ASSEMBLY LANGUAGE PROGRAMMING	67
	ANS COBOL 2nd Edition	168
Baker, Toni	MASTERING MACHINE CODE ON YOUR ZX-81	335
Barden, William	GUIDEBOOK TO SMALL COMPUTERS	25
	MICROCOMPUTERS FOR BUSINESS APPLICATIONS	40
	HOW TO PROGRAM MICROCOMPUTERS	65
	HOW TO BUY AND USE MINICOMPUTERS & MICROCOMPUTERS	29
Beech, Graham	SUCCESSFUL SOFTWARE FOR SMALL COMPUTERS	214
Beil, Donald H.	VISICALC BOOK - APPLE EDITION	224
	VISICALC BOOK - ATARI EDITION	224
Bennett, Bill	UNDERSTANDING THE MICRO	54

AUTHOR NAME	TITLE	PAGE NO.
Berenbon, Howard	MOSTLY BASIC - APPLICATIONS FOR YOUR APPLE II/ BOOK 1	267
	MOSTLY BASIC - APPLICATIONS FOR YOUR APPLE II/ BOOK 2	268
	MOSTLY BASIC - APPLICATIONS FOR YOUR TRS-80/BOOK 1	297
	MOSTLY BASIC - APPLICATIONS FOR YOUR TRS-80/BOOK 2	298
	MOSTLY BASIC - APPLICATIONS FOR YOUR PET	329
Berlin, Howard M.	CIRCUIT DESIGN PROGRAMS FOR THE APPLE II	258
	CIRCUIT DESIGN PROGRAMS FOR THE TRS-80	287
Billings, Karen	ARE YOU COMPUTER LITERATE?	2
Birnbaum, Mark	HOW TO CHOOSE YOUR SMALL BUSINESS COMPUTER	31
Blackwood, Brian D.	INTIMATE INSTRUCTIONS IN INTEGER BASIC	141
	APPLE FORTRAN	242
	APPLESOFT LANGUAGE	252
Blackwood, George H.	INTIMATE INSTRUCTIONS IN INTEGER BASIC	141
	APPLE FORTRAN	242
	APPLESOFT LANGUAGE	252
Blechman, Fred	PROGRAMS FOR BEGINNERS ON THE TRS-80	298
Boom, Michael	UNDERSTANDING ATARI GRAPHICS	321
Borchers, Mary	SOME COMMON BASIC PROGRAMS - IBM PERSONAL COMPUTER ED.	282
	SOME COMMON BASIC PROGRAMS - APPLE II EDITION	267
	SOME COMMON BASIC PROGRAMS - PET/CBM EDITION	330
	SOME COMMON BASIC PROGRAMS - ATARI EDITION	321
Boswell, F. D.	COMMODORE SUPERPET COMPUTERS SYSTEM OVERVIEW	324
Bosworth, Bruce	CODES, CIPHERS AND COMPUTERS	193
Bowen, Kenneth A.	SPEAKING PASCAL	165
Boyer, R. Dean	COMPUTER WORD PROCESSING	227
Bradbeer, Robin	BEGINNER'S GUIDE TO COMPUTERS	3
Brain Bank	THE BASIC CONVERSION HANDBOOK	128

AUTHOR NAME	TITLE	PAGE NO.
Brodie, Leo	STARTING FORTH	175
Brooner, E. G.	MICROCOMPUTER DATA-BASE MANAGEMENT	117
	BASIC BUSINESS SOFTWARE	182
Brown, Eben	TIMEX-SINCLAIR 1983 DIRECTORY	339
Brown, Jerald R.	BASIC 2nd Edition	123
	BASIC FOR HOME COMPUTERS	129
	DATA FILE PROGRAMMING IN BASIC	137
	INSTANT FREEZE-DRIED COMPUTER PROGRAMMING IN BASIC	140
	APPLE BASIC: DATA FILE PROGRAMMING	240
	BASIC FOR THE APPLE II	255
	ATARI BASIC	312
Brown, Peter	PASCAL FROM BASIC	160
Bryson, Susan M.	UNDERSTANDING APL	177
Buchsbaum, Walter H.	PERSONAL COMPUTERS HANDBOOK	105
Bui, X. T.	EXECUTIVE PLANNING WITH BASIC	199
Bukstein, Edward	PRACTICE PROBLEMS IN NUMBER SYSTEMS, LOGIC & BOOLEAN ALG.	119
Bunnell, David	INTRODUCTION TO MICROCOMPUTERS-VOL. 0	34
Burgess, Eric	CELESTIAL BASIC	191
Burke, Peter M.	SOME COMMON BASIC PROGRAMS - IBM PERSONAL COMPUTER ED.	282
Bursky, Dave	S-100 BUS HANDBOOK	107
Butah, Jon	INTRODUCTION TO THE UCSD p-SYSTEM	99
Calabrese, Charles A.	UNDERSTANDING THE IBM PERSONAL COMPUTER VOL.I	282
Campell, Joe	TIMEX PERSONAL COMPUTER MADE SIMPLE	338
Canning, R. G.	SO YOU ARE THINKING ABOUT A SMALL BUSINESS COMPUTER	49

AUTHOR NAME	TITLE	PAGE NO.
Carlson, Edward H.	KIDS & THE APPLE	264
Carr, Joseph J.	Z80 USERS MANUAL	79
Carroll, Charles J.	80 PRACTICAL TIME-SAVING PROGRAMS FOR THE TRS-80	310
Castlewitz, David M.	VISICALC HOME AND OFFICE COMPANION	226
	SOME COMMON BASIC PROGRAMS - APPLE II EDITION	267
Cater, John P.	ELECTRONICALLY SPEAKING	197
Chadwick, Ian	MAPPING THE ATARI	320
Chance, David	33 CHALLENGING COMPUTER GAMES FOR TRS-80/ APPLE/PET	216
Chew, Paul	CONSUMER'S GUIDE TO PERSONAL COMPUTING & MICROCOMPUTERS	16
Chirlian, Paul M.	UNDERSTANDING COMPUTERS	52
Chisausky, Lawrence J.	VISICALC HOME AND OFFICE COMPANION	226
Chiu, Yin	CRUNCHERS	333
Chmura, Louis J.	COBOL WITH STYLE: PROGRAMMING PROVERBS	170
	FORTRAN WITH STYLE: PROGRAMMING PROVERBS	150
Ciarcia, Steve	CIARCIA'S CIRCUIT CELLAR, VOL.1	92
	CIARCIA'S CIRCUIT CELLAR, VOL.2	93
	CIARCIA'S CIRCUIT CELLAR, VOL.3	93
	BUILD YOUR OWN Z80 COMPUTER	91
Clark, Roger E.	EXECUTIVE VISICALC FOR THE APPLE COMPUTER	220
Cleland, Pat	ATARI GAMES & RECREATIONS	313
Coan, James S.	ADVANCED BASIC	122
	BASIC APPLE BASIC	254
	BASIC BASIC 2nd Edition	126
	BASIC FORTRAN	147
Coffron, James W.	APPLE CONNECTION	241

AUTHOR NAME	TITLE	PAGE NO.
Cole, Jim	POCKET COMPUTER PROGRAMMING MADE EASY	144
	MURDER IN THE MANSION 2nd Edition	208
	50 PROGRAMS IN BASIC - 2nd Edition	216
Collins, Chuck	Z8000 ASSEMBLY LANGUAGE PROGRAMMING	80
Compute! Magazine	COMPUTE!'S FIRST BOOK OF ATARI GRAPHICS	318
	COMPUTE!'S FIRST BOOK OF VIC	327
	COMPUTE!'S SECOND BOOK OF ATARI	317
	COMPUTE!'S FIRST BOOK OF ATARI	316
	COMPUTE!'S FIRST BOOK OF PET/CBM	326
Cook, Steven	APPLE II USER'S GUIDE	247
	YOUR ATARI COMPUTER	322
Covvey, H. Dominic	COMPUTER CONSCIOUSNESS	8
Critchfield, Margot	BASIC AND THE PERSONAL COMPUTER	125
	BIT OF BASIC	136
Crowley, Thomas H.	UNDERSTANDING COMPUTERS	53
Dahmke, Mark	MICROCOMPUTER OPERATING SYSTEMS	102
Danley, William	NAILING JELLY TO A TREE	41
Davidson, Greg	PRACTICAL PASCAL PROGRAMS	210
DaCosta, Frank	WRITING BASIC ADVENTURE PROGRAMS FOR THE TRS-80	309
De Bono, Peter	BEGINNER'S GUIDE TO COMPUTERS	3
De Jong, Marvin L.	APPLE II ASSEMBLY LANGUAGE	244
Deken, Joseph	ELECTRONIC COTTAGE	21
Dennon, Jack D.	CP/M REVEALED	95
Desautels, Edouard J.	VISICALC FOR THE IBM PERSONAL COMPUTER	225
	VISICALC FOR THE TRS-80 MODEL I & III COMPUTERS	225
	VISICALC FOR THE TRS-80 MODEL II & 16 COMPUTERS	225
	VISICALC FOR APPLE II PLUS	224
DeVoney, Chris	IBM'S PERSONAL COMPUTER	279

AUTHOR NAME	TITLE	PAGE NO.
Didday, Rich	HOME COMPUTERS VOL 1: HARDWARE	27
	HOME COMPUTERS VOL 2: SOFTWARE	27
Dirksen, P. H.	COMMODORE SUPERPET COMPUTERS WATERLOO MicroFORTRAN	325
Doerr, Christine	MICROCOMPUTERS AND THE 3 R's	39
Dwyer, Thomas A.	BASIC AND THE PERSONAL COMPUTER	125
	BIT OF BASIC	136
Edwards, Raymond	8080 SOFTWARE GOURMET GUIDE & COOKBOOK 2nd Edition	84
Eisenbach, S.	PASCAL FOR PROGRAMMERS	160
Engel, C. W.	STIMULATING SIMULATIONS	213
	STIMULATING SIMULATIONS ATARI VERSION	213
Ennes, Harold E.	BOOLEAN ALGEBRA FOR COMPUTER LOGIC	114
Ettlin, Walter A.	WORDSTAR MADE EASY	233
Eubanks, Gordon	CBASIC USER GUIDE	137
Farvour, James Lee	MICROSOFT BASIC DECODED & OTHER MYSTERIES FOR THE TRS-80	297
	TRSDOS 2.3 DECODED & OTHER MYSTERIES	300
Faulk, Ed	HOW TO WRITE AN APPLE PROGRAM	263
	HOW TO WRITE AN IBM-PC PROGRAM	276
	HOW TO WRITE A TRS-80 PROGRAM	292
Feldman, Phil	32 BASIC PROGRAMS FOR THE APPLE COMPUTER	274
Fernandez, Judi N.	INTRODUCTION TO 8080/8085 ASSEMBLY LANGUAGE PROGRAMMING	67
	USING CP/M	109
Findley, Robert	8080 SOFTWARE GOURMET GUIDE & COOKBOOK 2nd Edition	84
	6800 SOFTWARE GOURMET GUIDE & COOKBOOK	82
	6502 SOFTWARE GOURMET GUIDE & COOKBOOK	82
Finkel, A.	VIC-20 PROGRAMMER'S REFERENCE GUIDE	331

AUTHOR NAME	TITLE	PAGE NO.
Finkel, LeRoy		
	BASIC FOR HOME COMPUTERS	129
	DATA FILE PROGRAMMING IN BASIC	137
	APPLE BASIC: DATA FILE PROGRAMMING	240
	BASIC FOR THE APPLE II	255
	TECHNIQUES FOR CREATING GOLDEN DELICIOUS GAMES	271
	BASIC 2nd Edition	123
	ATARI BASIC	312
Fishman, Katharine Davis		
	COMPUTER ESTABLISHMENT	11
Foster, Caxton C.		
	PROGRAMMING A MICROCOMPUTER: 6502	71
Fox, David		
	PASCAL PRIMER	162
Franklin, Howard M.		
	TECHNIQUES FOR CREATING GOLDEN DELICIOUS GAMES	271
Freiberger, Stephen		
	CONSUMER'S GUIDE TO PERSONAL COMPUTING & MICROCOMPUTERS	16
Freiling, Michael J.		
	UNDERSTANDING DATA BASE MANAGEMENT	54
Frenzel, Louis E.		
	CRASH COURSE IN MICROCOMPUTERS	18
	GETTING ACQUAINTED WITH MICROCOMPUTERS	97
Friedmann, Jehosua		
	FORTRAN IV 2nd Edition	148
Froehlich, John Paul		
	TRS-80 MORE THAN BASIC	308
Galanter, Eugene		
	KIDS AND COMPUTERS	35
Galbraith, Richard		
	PROFESSIONAL PROGRAMMING TECHNIQUES	145
Giarratano, Joseph C.		
	BASIC: ADVANCED CONCEPTS	123
	FOUNDATIONS OF COMPUTER TECHNOLOGY	23
	BASIC: FUNDAMENTAL CONCEPTS	123
	MODERN COMPUTER CONCEPTS	40
Gibson, Barbara		
	PERSONAL COMPUTERS IN BUSINESS	44
Gilder, Jules H.		
	BASIC COMPUTER PROGRAMS IN SCIENCE AND ENGINEERING	186
Glatzer, Hal		
	INTRODUCTION TO WORD PROCESSING	229

AUTHOR NAME	TITLE	PAGE NO.
Gloess, Paul Y.	UNDERSTANDING ARTIFICIAL INTELLIGENCE	51
	UNDERSTANDING LISP	178
Glossbrenner, Alfred	COMPLETE HANDBOOK OF PERSONAL COMPUTER COMMUNICATIONS	6
Goldsmith, W. B.	BASIC PROGRAMS FOR HOME FINANCIAL MANAGEMENT	187
Goldstein, Larry Joel	BASIC FOR THE APPLE II: PROGRAMMING AND APPLICATIONS	256
	IBM PERSONAL COMPUTER	278
Goldstein, Martin	BASIC FOR THE APPLE II: PROGRAMMING AND APPLICATIONS	256
	IBM PERSONAL COMPUTER	278
Goodson, Bobby	COURSEWARE IN THE CLASSROOM	17
Goodwin, James C.	S-100 & OTHER MICRO BUSES 2nd Edition	106
Graham, J. Wesley	COMMODORE SUPERPET COMPUTERS WATERLOO MicroBASIC	325
Grant, Charles W.	INTRODUCTION TO THE UCSD p-SYSTEM	99
Gratzer, George A.	FAST BASIC	290
Grauer, Robert T.	TRS-80 COBOL	303
Green, Adam B.	DBASE II USER'S GUIDE	196
Greenberg, Philip	FORTRAN IV 2nd Edition	148
Grillo, John P.	GUIDE TO SYSTEMS APPLICATIONS	24
	DATA MANAGEMENT TECHNIQUES	290
	INTRODUCTION TO GRAPHICS	295
	TECHNIQUES OF BASIC	299
Grove, T. R.	COMMODORE SUPERPET COMPUTERS SYSTEM OVERVIEW	324
Hancock, Les	C PRIMER	169
Harris, N.	VIC-20 PROGRAMMER'S REFERENCE GUIDE	331

AUTHOR NAME	TITLE	PAGE NO.
Harrison, Mark	BYTEING DEEPER INTO YOUR TIMEX/SINCLAIR 1000	332
Hartnell, Tim	GETTING ACQUAINTED WITH YOUR ZX81	334
	MAKING THE MOST OF YOUR ZX-81	335
	MASTERING YOUR TIMEX/SINCLAIR 1000 PERSONAL COMPUTER	336
	49 EXPLOSIVE GAMES FOR THE ZX-81	344
	GETTING ACQUAINTED WITH YOUR VIC 20	328
Hawkins, Doug	68000 ASSEMBLY LANGUAGE PROGRAMMING	84
Heilborn, John	SCIENCE AND ENGINEERING PROGRAMS APPLE II EDITION	266
Heiserman, David L.	INTERMEDIATE PROGRAMMING FOR THE TRS-80 MODEL I	294
Held, Gilbert	IBM PC BASIC QUICK REFERENCE GUIDE	276
	APPLE II BASIC QUICK REFERENCE GUIDE	246
	ATARI BASIC QUICK REFERENCE GUIDE	313
Helms, Harry L.	COMPUTER LANGUAGE REFERENCE GUIDE	11
	INTRODUCTION TO MICROCOMPUTERS FOR THE HAM SHACK	33
	BASIC BOOK	127
Herbert, Frank	WITHOUT ME YOU'RE NOTHING	57
Hergert, Douglas	MASTERING VISICALC	222
	YOUR TIMEX SINCLAIR 1000 & ZX81	340
	APPLE PASCAL GAMES	251
Higginbottom, P.	VIC-20 PROGRAMMER'S REFERENCE GUIDE	331
Hogan, Thom	OSBORNE CP/M USER GUIDE	104
	DISCOVER FORTH	170
Hollerbach, Lew	60-MINUTE GUIDE TO MICROCOMPUTERS	60
Horn, Bethold	LISP	173
Horn, Carin E.	COMPUTER LITERACY	12
Hueras, John F.	PASCAL WITH STYLE: PROGRAMMING PROVERBS	164
Hurley, Randle	SINCLAIR ZX-81	337

AUTHOR NAME	TITLE	PAGE NO.
Hyde, Randy	USING 6502 ASSEMBLY LANGUAGE	272
Hyndman, Brian	PEANUT BUTTER & JELLY GUIDE TO COMPUTERS	43
Inman, Don	APPLE MACHINE LANGUAGE	250
	TRS-80 BASIC	302
	ATARI ASSEMBLER	311
Inman, Kurt	APPLE MACHINE LANGUAGE	250
	ATARI ASSEMBLER	311
Jensen, Kathleen	PASCAL USER MANUAL AND REPORT 2nd Edition	163
Jones, Aubrey B.	I SPEAK BASIC TO MY APPLE	264
	I SPEAK BASIC TO MY TRS-80	293
	I SPEAK BASIC TO MY PET	329
Jones, Dilwyn	MASTERING YOUR TIMEX/SINCLAIR 1000 PERSONAL COMPUTER	336
Kahn, Ted	ATARI GAMES & RECREATIONS	313
Kalash, Joseph T.	APPLE PASCAL GAMES	251
Kamins, Scott	APPLE BACKPACK	239
Kane, Gerry	OSBORNE 4 & 8-BIT MICROPROCESSOR HANDBOOK	69
	OSBORNE 16-BIT MICROPROCESSOR HANDBOOK	70
	68000 ASSEMBLY LANGUAGE PROGRAMMING	84
Karp, Laura	GENIE IN THE COMPUTER	291
Kater, David A.	TRS-80 GRAPHICS	307
Keller, Arthur	FIRST COURSE IN COMPUTER PROGRAMMING USING PASCAL	153
Kelley, James E.	IBM/PC GUIDE	277
Kelly-Bootle, Stan	DEVIL'S DP DICTIONARY	19
Kidder, Tracy	SOUL OF A NEW MACHINE	50
Kimberley, Paul	MICROPROCESSORS	69

AUTHOR NAME	TITLE	PAGE NO.
Kitsz, Dennis Bathory	CUSTOM TRS-80 & OTHER MYSTERIES	289
Knecht, Ken	MICROSOFT BASIC 2nd Edition	143
	INTRODUCTION TO FORTH	172
Koffman, Elliot B.	PASCAL: A PROBLEM SOLVING APPROACH	158
	PROBLEM SOLVING AND STRUCTURED PROGRAMMING IN PASCAL	165
Kohl, Herb	ATARI GAMES & RECREATIONS	313
Kohl, Rachel	GENIE IN THE COMPUTER	291
Koltnow, Joanne	TECHNIQUES FOR CREATING GOLDEN DELICIOUS GAMES	271
Krieger, Morris	C PRIMER	169
Kruglinski, David	DATA BASE MANAGEMENT SYSTEMS	195
Krutch, John	EXPERIMENTS IN ARTIFICIAL INTELL. FOR SMALL COMPUTERS	200
Lamoitier, Jean-Pierre	FIFTY BASIC EXERCISES	139
	BASIC EXERCISES FOR THE IBM PERSONAL COMPUTER	275
Lancaster, Don	HEXADECIMAL CHRONICLES	116
Larsen, Sally Greenwood	COMPUTERS FOR KIDS - ATARI EDITION	318
	COMPUTERS FOR KIDS - APPLE EDITION	260
	COMPUTERS FOR KIDS - TRS-80 EDITION	288
	COMPUTERS FOR KIDS - VIC-20 EDITION	328
	COMPUTERS FOR KIDS - SINCLAIR EDITION	333
Larson, David	APPLE INTERFACING	249
Lathrop, Ann	COURSEWARE IN THE CLASSROOM	17
Laurie, Peter	BEGINNER'S GUIDE TO COMPUTERS	3
Lawson, Harold W.	UNDERSTANDING COMPUTER SYSTEMS	51
Lechner, Pieter	BAG OF TRICKS	254
	BENEATH APPLE DOS	257

AUTHOR NAME	TITLE	PAGE NO.
Ledgard, Henry F.		
	ELEMENTARY BASIC	139
	ELEMENTARY PASCAL	153
	ADA: AN INTRODUCTION and ADA REFERENCE MANUAL	167
	PASCAL WITH STYLE: PROGRAMMING PROVERBS	164
	BASIC WITH STYLE: PROGRAMMING PROVERBS	134
	FORTRAN WITH STYLE: PROGRAMMING PROVERBS	150
	PROGRAMMING PROVERBS FOR FORTRAN PROGRAMMERS	152
	COBOL WITH STYLE: PROGRAMMING PROVERBS	170
Ledin, George		
	PASCAL	157
	UNDERSTANDING PASCAL	166
Leeper, N. C.		
	SO YOU ARE THINKING ABOUT A SMALL BUSINESS COMPUTER	49
Leventhal, Lance A.		
	WHY DO YOU NEED A PERSONAL COMPUTER?	56
	Z80 ASSEMBLY LANGUAGE PROGRAMMING	75
	6502 ASSEMBLY LANGUAGE PROGRAMMING	82
	Z8000 ASSEMBLY LANGUAGE PROGRAMMING	80
	68000 ASSEMBLY LANGUAGE PROGRAMMING	84
Levy, David		
	CHESS AND COMPUTERS	192
	MORE CHESS AND COMPUTERS	206
Lewis, T. G.		
	HOW TO PROFIT FROM YOUR PERSONAL COMPUTER	31
	SOFTWARE ENGINEERING FOR MICROS	120
	PASCAL PROGRAMMING FOR THE APPLE	269
	USING THE IBM PERSONAL COMPUTER	284
Lewis, William E.		
	PROBLEM SOLVING PRINCIPLES FOR PROGRAMMERS	120
	PROBLEM SOLVING PRINCIPLES FOR ADA PROGRAMMERS	175
Libes, Sol		
	PROGRAMMER'S GUIDE TO CP/M	105
Lieff, Jonathan D.		
	HOW TO BUY A PERSONAL COMPUTER (WITHOUT ANXIETY)	29
Lien, David A.		
	THE BASIC HANDBOOK	131
	LEARNING IBM BASIC FOR THE PERSONAL COMPUTER	280
	LEARNING TRS-80 BASIC FOR MODELS I, II/16, AND III	296
Lindsay, Len		
	ATARI GAMES & RECREATIONS	313
Lord, Kenniston W.		
	USING THE IBM PERSONAL COMPUTER	285

AUTHOR NAME	TITLE	PAGE NO.
Lower, Judy	ATARI SOUND AND GRAPHICS	314
Ludwig, Herbert R.	UNDERSTANDING FORTRAN	152
Lundstrom, Peter	PERSONAL GUIDE TO PERSONAL COMPUTERS	46
Mansfield, Richard	BEGINNER'S GUIDE TO BUYING A PERSONAL COMPUTER	3
Manus, Steven	HOW TO BUY A WORD PROCESSOR	228
Marney-Petrix, Vicki	SOME COMMON PASCAL PROGRAMS	212
Mateosian, Richard	INSIDE BASIC GAMES	203
Mau, Ernest E.	CREATE WORD PUZZLES WITH YOUR MICROCOMPUTER	194
Mauder, Robert	ZX-81 COMPANION	341
McAlister, Neil Harding	COMPUTER CONSCIOUSNESS	8
McCaul, Earles L.	TRS-80 ASSEMBLY LANGUAGE MADE SIMPLE	301
McCracken, Daniel D.	GUIDE TO PL/M PROGRAMMING FOR MICROCOMPUTER APPLICATIONS	172
McGlynn, Daniel R.	PERSONAL COMPUTING 2nd Edition	45
	FUNDAMENTALS OF MICROCOMPUTER PROGRAMMING	155
McNiff, Martin	APPLE II USER'S GUIDE	247
	YOUR ATARI COMPUTER	322
	CBASIC USER GUIDE	137
McPhee, K. Ian	COMMODORE SUPERPET COMPUTERS WATERLOO MicroBASIC	325
	COMMODORE SUPERPET COMPUTERS SYSTEM OVERVIEW	324
McWilliams, Peter A.	PERSONAL COMPUTER BOOK	44
	WORD PROCESSING BOOK	231
Messer, Andrew C.	FORTRAN IV POCKET HANDBOOK	149
Meyer, Carl D.	TEN EASY PIECES	215
Miller, Alan R.	PASCAL PROGRAMS FOR SCIENTISTS AND ENGINEERS	209

AUTHOR NAME	TITLE	PAGE NO.
Miller, Alan R.	MASTERING CP/M	100
	8080/Z80 ASSEMBLY LANGUAGE	84
	FORTRAN PROGRAMS FOR SCIENTIST AND ENGINEERS	200
	BASIC PROGRAMS FOR SCIENTISTS AND ENGINEERS	188
Miller, Merl K.	COMPUTERS FOR PEOPLE	15
	HOME COMPUTERS	26
	COMPUTERS FOR EVERYBODY	15
Miller, Myron D.	BEGINNER'S GUIDE TO BUYING A PERSONAL COMPUTER	3
Moody, Robert	FIRST BOOK OF MICROCOMPUTERS	22
Moore, Herb	ATARI SOUND AND GRAPHICS	314
Morgan, Christopher L.	8086/8088 16-BIT MICROPROCESSOR PRIMER	87
Morse, Stephen P.	8086/8088 PRIMER 2nd Edition	86
Mottola, Robert	ASSEMBLY LANGUAGE PROGRAMMING FOR THE APPLE II	253
Moursund, David	ARE YOU COMPUTER LITERATE?	2
Mullish, Henry	CRUNCHERS	333
Murtha, Stephen M.	CP/M PRIMER	95
Nagin, Paul A.	BASIC WITH STYLE: PROGRAMMING PROVERBS	134
	PASCAL WITH STYLE: PROGRAMMING PROVERBS	164
Naiman, Arthur	INTRODUCTION TO WORDSTAR	230
Nevison, John M.	LITTLE BOOK OF BASIC STYLE	143
	EXECUTIVE COMPUTING	198
Newborn, Monroe	MORE CHESS AND COMPUTERS	206
Nichols, Elizabeth A.	Z80 MICROPROCESSOR PROGRAMMING & INTERFACING	77
Nichols, Joseph C.	Z80 MICROPROCESSOR PROGRAMMING & INTERFACING	77
Noonan, Larry	BASIC BASIC-ENGLISH DICTIONARY	126
Norman, Robin	ZX-81 BASIC BOOK	340
NCR	NCR DATA COMMUNICATIONS CONCEPTS	42

AUTHOR NAME	TITLE	PAGE NO.
Osborne, Adam	INTRODUCTION TO MICROCOMPUTERS-VOL. 0	34
	CBASIC USER GUIDE	137
	INTRODUCTION TO MICROCOMPUTERS-VOL. 1	66
	BUSINESS SYSTEM BUYER'S GUIDE	4
	OSBORNE 4 & 8-BIT MICROPROCESSOR HANDBOOK	69
	OSBORNE 16-BIT MICROPROCESSOR HANDBOOK	70
	RUNNING WILD	47
	Z8000 ASSEMBLY LANGUAGE PROGRAMMING	80
Packer, Rod E.	INVESTOR'S COMPUTER HANDBOOK	35
Page, Edward	101 TIMEX 1000/SINCLAIR ZX-81 PROGRAMMING TIPS & TRICKS	342
	37 TIMEX 1000 SINCLAIR ZX-81 PROGRAMS	343
Pardee, Michael	MICROCOMPUTER PRIMER 2nd Edition	37
	YOUR OWN COMPUTER	59
	BASIC PROGRAMMING PRIMER	133
Peckham, Herbert	HANDS-ON BASIC	263
Peddicord, Richard G.	UNDERSTANDING BASIC	146
	UNDERSTANDING COBOL	178
Pelczarski, Mark	CREATIVE APPLE	260
Pennington, H. C.	TRS-80 DISK & OTHER MYSTERIES	305
Poe, Elmer C.	S-100 & OTHER MICRO BUSES 2nd Edition	106
Poirot, James L.	COMPUTERS AND EDUCATION	13
	MICROCOMPUTER SYSTEMS AND APPLE BASIC	265
	COMPUTER LITERACY	12
Polin, Glenn M.	WILL SOMEONE PLEASE TELL ME WHAT AN APPLE CAN DO?	273
Poling, Carol	APPLE II WORD PROCESSING	247
Poole, Lon	APPLE II USER'S GUIDE	247
	PRACTICAL BASIC PROGRAMS - APPLE II EDITION	270
	SOME COMMON BASIC PROGRAMS - IBM PERSONAL COMPUTER ED.	282
	PRACTICAL BASIC PROGRAMS - IBM PERSONAL COMPUTER EDITION	281

AUTHOR NAME	TITLE	PAGE NO.
Poole, Lon	SOME COMMON BASIC PROGRAMS - APPLE II EDITION	267
	YOUR ATARI COMPUTER	322
	PRACTICAL BASIC PROGRAMS - TRS-80 LEVEL II EDITION	298
	SOME COMMON BASIC PROGRAMS - PET/CBM EDITION	330
	SOME COMMON BASIC PROGRAMS - ATARI EDITION	321
	USING YOUR IBM PERSONAL COMPUTER	285
Presley, Bruce	GUIDE TO PROGRAMMING IBM PERSONAL COMPUTER	275
	GUIDE TO PROGRAMMING IN APPLESOFT	262
	GUIDE TO PROGRAMMING IN LEVEL II BASIC	292
Press, Laurence	LOW-COST WORD PROCESSING	230
Remer, Daniel	LEGAL CARE FOR YOUR SOFTWARE	205
Ridler, Philip	POCKET GUIDE TO FORTRAN	151
Robertson, J. D.	GUIDE TO SYSTEMS APPLICATIONS	24
	DATA MANAGEMENT TECHNIQUES	290
	INTRODUCTION TO GRAPHICS	295
	TECHNIQUES OF BASIC	299
Rochowansky, Sandra	ADDISON-WESLEY BOOK OF APPLE COMPUTER SOFTWARE 1983	238
Rony, Peter R.	Z80 MICROPROCESSOR PROGRAMMING & INTERFACING	77
Rosa, Nicholas	SMALL COMPUTERS FOR THE SMALL BUSINESSMAN	48
Rosa, Sharon	SMALL COMPUTERS FOR THE SMALL BUSINESSMAN	48
Rosenfelder, Lewis	BASIC FASTER AND BETTER & OTHER MYSTERIES	286
Rugg, Tom	32 BASIC PROGRAMS FOR THE APPLE COMPUTER	274
Sadler, C.	PASCAL FOR PROGRAMMERS	160
Sagan, Hans	TEN EASY PIECES	215
	BEAT THE ODDS	189
Sanders, William B.	ELEMENTARY APPLE	261
Savage, Earl R.	BASIC PROGRAMMER'S NOTEBOOK	132

AUTHOR NAME	TITLE	PAGE NO.
Scanlon, Leo J.		
	68000: PRINCIPLES AND PROGRAMMING	83
	FORTH PROGRAMMING	171
	APPLE II ASSEMBLY LANGUAGE EXERCISES	245
Schmeltz, L. R.		
	PLAYING THE STOCK & BOND MARKETS	209
Schoman, Kenneth E.		
	BASIC WORKBOOK	135
Schueler, J. B.		
	COMMODORE SUPERPET COMPUTERS SYSTEM OVERVIEW	324
Scriven, Michael		
	HOW TO BUY A WORD PROCESSOR	228
Seiter, Charles		
	PASCAL FOR BASIC PROGRAMMERS	159
Shelley, John		
	POCKET GUIDE TO PROGRAMMING	118
Shrum, Carlton		
	HOW TO BUY A PERSONAL COMPUTER	28
	HOW TO USE VISICALC SUPERCALC	221
Sickman, John		
	HOW TO CHOOSE YOUR SMALL BUSINESS COMPUTER	31
Signer, Ethan		
	GENIE IN THE COMPUTER	291
Siminoff, Jonathan D.		
	TIMEX PERSONAL COMPUTER MADE SIMPLE	338
Simon, David E.		
	BASIC FROM THE GROUND UP	130
Singer, Andrew		
	ELEMENTARY BASIC	139
	ELEMENTARY PASCAL	153
Sipl, Charles J.		
	COMPUTER DICTIONARY 3rd Edition	9
	ESSENTIAL COMPUTER DICTIONARY AND SPELLER	22
	HOME COMPUTERS	26
	MICROCOMPUTER DICTIONARY 2nd Edition	37
	COMPUTER DICTIONARY & HANDBOOK 3rd Edition	9
Sipl, Roger J.		
	COMPUTER DICTIONARY 3rd Edition	9
	COMPUTER DICTIONARY & HANDBOOK 3rd Edition	9
Smith, Brian Reffin		
	INTRODUCTION TO COMPUTER PROGRAMMING	142
Smith, Noel T.		
	INTRODUCTION TO MICROPROCESSORS	99
Smith, Robert E.		
	DISCOVERING BASIC	138

AUTHOR NAME	TITLE	PAGE NO.
Smithy, Deborah	PEANUT BUTTER & JELLY GUIDE TO COMPUTERS	43
Smolin, C. Roger	HOW TO BUY THE RIGHT SMALL BUSINESS COMPUTER SYSTEM	30
Snover, Stephen L.	BRAIN TICKLERS: PUZZLES & PASTIMES	190
Spencer, Donald D.	COMPUTERS IN NUMBER THEORY	116
	COMPUTERS AND PROGRAMMING GUIDE	14
	FUNDAMENTALS OF DIGITAL COMPUTERS	65
	FUN WITH MICROCOMPUTERS AND BASIC	201
	GAME PLAYING WITH BASIC	201
	SIXTY CHALLENGING PROBLEMS WITH BASIC SOLUTIONS	211
Spikell, Mark A.	BRAIN TICKLERS: PUZZLES & PASTIMES	190
Spracklen, Kathe	Z80 AND 8080 ASSEMBLY LANGUAGE PROGRAMMING	74
Stafford, Irvin	WHY DO YOU NEED A PERSONAL COMPUTER?	56
Stanton, Jeffrey	ADDISON-WESLEY BOOK OF APPLE COMPUTER SOFTWARE 1983	238
	BOOK OF ATARI SOFTWARE 1983	315
	APPLE GRAPHICS & ARCADE GAME DESIGN	243
Steingraber, Jack	FORTRAN FUNDAMENTALS: A SHORT COURSE	148
Sternberg, Charles D.	BASIC COMPUTER PROGRAMS FOR BUSINESS, VOL.1	184
	BASIC COMPUTER PROGRAMS FOR BUSINESS, VOL.2	184
	BASIC COMPUTER PROGRAMS FOR THE HOME	185
Summe, Richard	IBM'S PERSONAL COMPUTER	279
Swanson, Paul	MICROCOMPUTER DISK TECHNIQUES	101
Tatchell, Judy	UNDERSTANDING THE MICRO	54
Tate, Joe	CREATIVE APPLE	260
Thomas, Rebecca	USER GUIDE TO THE UNIX SYSTEM	108
Thomas, Susan J.	TRS-80 GRAPHICS	307
Tiberghien, Jacques	PASCAL HANDBOOK	161

AUTHOR NAME	TITLE	PAGE NO.
Titus, Christopher	APPLE INTERFACING	249
Titus, Jonathan	APPLE INTERFACING	249
Townsend, Carl	HOW TO GET STARTED WITH CP/M	98
Tracton, Ken	THE BASIC COOKBOOK	129
	MOST POPULAR SUBROUTINES IN BASIC	207
	57 PRACTICAL PROGRAMS & GAMES IN BASIC	218
Traister, Robert J.	IBM PERSONAL COMPUTER	279
Trost, Stanley R.	DOING BUSINESS WITH VISICALC	219
Van Horn, Royal	COMPUTER PROGRAMMING FOR KIDS AND OTHER BEGINNERS	259
Veit, Stanley S.	USING MICROCOMPUTERS IN BUSINESS	55
Vile, Richard C.	APPLE II PROGRAMMER'S HANDBOOK	246
Wadsworth, Nat	Z80 INSTRUCTION HANDBOOK	77
	Z80 SOFTWARE GOURMET GUIDE AND COOKBOOK	78
	GRAPHICS COOKBOOK FOR THE APPLE	262
	INTRODUCTION TO COMPUTER ANIMATION	204
Waite, Mitchell	MICROCOMPUTER PRIMER 2nd Edition	37
	YOUR OWN COMPUTER	59
	8086/8088 16-BIT MICROPROCESSOR PRIMER	87
	CP/M PRIMER	95
	COMPUTER GRAPHICS PRIMER	115
	BASIC PROGRAMMING PRIMER	133
	PASCAL PRIMER	162
	WORD PROCESSING PRIMER	232
	APPLE BACKPACK	239
Waterford, Van	COMPLETE BOOK OF HOME COMPUTERS	5
Watson, Wm. Scott	67 READY TO RUN PROGRAMS IN BASIC	219
	55 ADVANCED COMPUTER PROGRAMS IN BASIC	217
Watt, David	POCKET GUIDE TO PASCAL	164
Weber, Jeffrey R.	USER'S GUIDE TO PET/CBM COMPUTERS	330
	USER'S HANDBOOK TO THE IBM PERSONAL COMPUTER	283

AUTHOR NAME	TITLE	PAGE NO.
Weber, Jeffrey R.	CP/M SIMPLIFIED	97
Webster, Tony	MICROCOMPUTER BUYER'S GUIDE	36
Weiss, Robert	PASCAL FOR BASIC PROGRAMMERS	159
Welch, J. W.	COMMODORE SUPERPET COMPUTERS WATERLOO MicroFORTRAN	325
Wells, Robert P.	BOOK OF ATARI SOFTWARE 1983	315
	ADDISON-WESLEY BOOK OF APPLE COMPUTER SOFTWARE 1983	238
Whiteside, Thomas	COMPUTER CAPERS	7
Widl Video	BLUE BOOK FOR THE APPLE COMPUTER	258
Wilkinson, Bill	INSIDE ATARI DOS	319
Willis, Jerry	COMPUTERS FOR EVERYBODY	15
	COMPUTERS FOR PEOPLE	15
	NAILING JELLY TO A TREE	41
	PEANUT BUTTER & JELLY GUIDE TO COMPUTERS	43
Winston, Patrick	LISP	173
Wirth, Niklaus	PASCAL USER MANUAL AND REPORT 2nd Edition	163
Wolverton, Van	VISICALC, ADVANCED VERSION	223
Worth, Don	BAG OF TRICKS	254
	BENEATH APPLE DOS	257
Yates, Jean	TIMEX PERSONAL COMPUTER MADE SIMPLE	338
	USER GUIDE TO THE UNIX SYSTEM	108
Zabinski, Michael P.	TRS-80 FOR KIDS FROM 8 TO 80 Volume I	306
Zaks, Rodnay	DON'T!	20
	YOUR FIRST COMPUTER	58
	FROM CHIPS TO SYSTEMS	63
	PROGRAMMING THE Z80	72
	PROGRAMMING THE 6502	73
	CP/M HANDBOOK	94
	INTRODUCTION TO PASCAL	156

AUTHOR NAME	TITLE	PAGE NO.
Zamora, Romon	TRS-80 BASIC	302
Zarrella, John	MICROPROCESSOR OPERATING SYSTEMS, VOL.II	103
Zinsser, William	WRITING WITH A WORD PROCESSOR	234
73 Magazine	GIANT BOOK OF COMPUTER SOFTWARE	202

Rating Index/By Subject

SUBJECT: Apple		PAGE NO.
RATING	TITLE	
100	32 BASIC PROGRAMS FOR THE APPLE COMPUTER	274
95	APPLE BACKPACK	239
95	APPLE BASIC: DATA FILE PROGRAMMING	240
95	APPLE II USER'S GUIDE	247
95	APPLE II WORD PROCESSING	247
95	BASIC APPLE BASIC	254
95	CIRCUIT DESIGN PROGRAMS FOR THE APPLE II	258
95	PASCAL PROGRAMMING FOR THE APPLE	269
90	ADDISON-WESLEY BOOK OF APPLE COMPUTER SOFTWARE 1983	238
90	APPLE GRAPHICS & ARCADE GAME DESIGN	243
90	APPLE II PROGRAMMER'S HANDBOOK	246
90	APPLE MACHINE LANGUAGE	250
90	APPLE PASCAL GAMES	251
90	BENEATH APPLE DOS	257
90	COMPUTERS FOR KIDS - APPLE EDITION	260
90	CREATIVE APPLE	260
90	GUIDE TO PROGRAMMING IN APPLESOFT	262
90	KIDS & THE APPLE	264
90	MOSTLY BASIC - APPLICATIONS FOR YOUR APPLE II/BOOK 2	268
90	PRACTICAL BASIC PROGRAMS - APPLE II EDITION	270
90	USING 6502 ASSEMBLY LANGUAGE	272
85	APPLE CONNECTION	241
85	APPLE II BASIC QUICK REFERENCE GUIDE	246
85	BAG OF TRICKS	254
85	BASIC FOR THE APPLE II	255
85	ELEMENTARY APPLE	261
85	SCIENCE AND ENGINEERING PROGRAMS APPLE II EDITION	266
85	SOME COMMON BASIC PROGRAMS - APPLE II EDITION	267
80	1983 APPLE II/III SOFTWARE DIRECTORY - Volume II	249
80	BLUE BOOK FOR THE APPLE COMPUTER	258

SUBJECT: Apple

RATING	TITLE	PAGE NO.
80	COMPUTER PROGRAMMING FOR KIDS AND OTHER BEGINNERS	259
80	HANDS-ON BASIC	263
80	SWIFT'S EDUCATIONAL SOFTWARE DIRECTORY	270
75	APPLE II ASSEMBLY LANGUAGE	244
75	GRAPHICS COOKBOOK FOR THE APPLE	262
75	MOSTLY BASIC - APPLICATIONS FOR YOUR APPLE II/BOOK 1	267
70	EXECUTIVE VISICALC FOR THE APPLE COMPUTER	220
70	APPLE INTERFACING	249
70	BASIC FOR THE APPLE II: PROGRAMMING AND APPLICATIONS	256
60	APPLE LOGO	250
60	APPLESOFT LANGUAGE	252
60	WILL SOMEONE PLEASE TELL ME WHAT AN APPLE CAN DO ?	273
50	ASSEMBLY LANGUAGE PROGRAMMING FOR THE APPLE II	253
50	HOW TO WRITE AN APPLE PROGRAM	263
50	MICROCOMPUTER SYSTEMS AND APPLE BASIC	265
50	TECHNIQUES FOR CREATING GOLDEN DELICIOUS GAMES	271
45	APPLE FORTRAN	242
45	APPLE II ASSEMBLY LANGUAGE EXERCISES	245
45	I SPEAK BASIC TO MY APPLE	264

SUBJECT: Atari

95	ATARI SOUND AND GRAPHICS	314
95	BOOK OF ATARI SOFTWARE 1983	315
90	ATARI ASSEMBLER	311
90	COMPUTE!'S FIRST BOOK OF ATARI	316
90	COMPUTERS FOR KIDS - ATARI EDITION	318
90	YOUR ATARI COMPUTER	322
85	STIMULATING SIMULATIONS ATARI VERSION	213
85	ATARI BASIC QUICK REFERENCE GUIDE	313
85	COMPUTE!'S SECOND BOOK OF ATARI	317
85	COMPUTE!'S FIRST BOOK OF ATARI GRAPHICS	318
85	MAPPING THE ATARI	320
85	SOME COMMON BASIC PROGRAMS - ATARI EDITION	321

SUBJECT: Atari		
RATING	TITLE	PAGE NO.
85	UNDERSTANDING ATARI GRAPHICS	321
70	ATARI GAMES & RECREATIONS	313
65	ATARI BASIC	312
65	INSIDE ATARI DOS	319
SUBJECT: Basic		
100	THE BASIC HANDBOOK	131
95	BASIC AND THE PERSONAL COMPUTER	125
95	DATA FILE PROGRAMMING IN BASIC	137
90	ADVANCED BASIC	122
90	BASIC BASIC 2nd Edition	126
90	BASIC PROGRAMMER'S NOTEBOOK	132
90	BASIC PROGRAMMING PRIMER	133
90	LITTLE BOOK OF BASIC STYLE	143
85	BASIC FROM THE GROUND UP	130
85	BIT OF BASIC	136
85	FIFTY BASIC EXERCISES	139
80	BASIC 2nd Edition	123
80	BASIC: ADVANCED CONCEPTS	123
80	BASIC BASIC-ENGLISH DICTIONARY	126
80	CBASIC USER GUIDE	137
80	DISCOVERING BASIC	138
80	INTRODUCTION TO COMPUTER PROGRAMMING	142
80	MICROSOFT BASIC 2nd Edition	143
80	PROFESSIONAL PROGRAMMING TECHNIQUES	145
75	THE BASIC COOKBOOK	129
75	BASIC FOR HOME COMPUTERS	129
70	BASIC: FUNDAMENTAL CONCEPTS	123
70	BASIC WORKBOOK	135
65	ELEMENTARY BASIC	139
55	BASIC WITH STYLE: PROGRAMMING PROVERBS	134
50	BASIC BOOK	127
50	UNDERSTANDING BASIC	146
40	THE BASIC CONVERSION HANDBOOK	128
20	INTIMATE INSTRUCTIONS IN INTEGER BASIC	141
10	INSTANT FREEZE-DRIED COMPUTER PROGRAMMING IN BASIC	140
10	POCKET COMPUTER PROGRAMMING MADE EASY	144
SUBJECT: Commodore		
95	COMPUTE!'S FIRST BOOK OF PET/CBM	326

SUBJECT: Commodore

RATING	TITLE	PAGE NO.
95	VIC-20 PROGRAMMER'S REFERENCE GUIDE	331
90	COMPUTERS FOR KIDS - VIC-20 EDITION	328
85	SOME COMMON BASIC PROGRAMS - PET/ CBM EDITION	330
80	COMMODORE SOFTWARE ENCYCLOPEDIA 2nd Edition	323
80	COMPUTE!'S FIRST BOOK OF VIC	327
80	USER'S GUIDE TO PET/CBM COMPUTERS	330
75	GETTING ACQUAINTED WITH YOUR VIC 20	328
75	MOSTLY BASIC - APPLICATIONS FOR YOUR PET	329
70	COMMODORE SUPERPET COMPUTERS SYSTEM OVERVIEW	324
45	I SPEAK BASIC TO MY PET	329
25	COMMODORE SUPERPET COMPUTERS WATERLOO MicroBASIC	325
25	COMMODORE SUPERPET COMPUTERS WATERLOO MicroFORTRAN	325

SUBJECT: Fortran

85	FORTRAN IV 2nd Edition	148
75	BASIC FORTRAN	147
70	POCKET GUIDE TO FORTRAN	151
65	FORTRAN IV POCKET HANDBOOK	149
55	FORTRAN WITH STYLE: PROGRAMMING PROVERBS	150
55	PROGRAMMING PROVERBS FOR FORTRAN PROGRAMMERS	152
50	UNDERSTANDING FORTRAN	152
45	FORTRAN FUNDAMENTALS: A SHORT COURSE	148

SUBJECT: General Programming

95	BOOLEAN ALGEBRA FOR COMPUTER LOGIC	114
95	HEXADECIMAL CHRONICLES	116
90	COMPUTER GRAPHICS PRIMER	115
90	COMPUTERS IN NUMBER THEORY	116
90	PRACTICE PROBLEMS IN NUMBER SYSTEMS, LOGIC & BOOLEAN ALG.	119
90	PROBLEM SOLVING PRINCIPLES FOR PROGRAMMERS	120
85	INTRODUCTION TO GRAPHICS	295
80	SOFTWARE ENGINEERING FOR MICROS	120
75	MICROCOMPUTER DATA-BASE MANAGEMENT	117

SUBJECT: General Programming

RATING	TITLE	PAGE NO.
50	POCKET GUIDE TO PROGRAMMING	118
SUBJECT: General Software & Application		
95	BASIC PROGRAMS FOR HOME FINANCIAL MANAGEMENT	187
95	CHESS AND COMPUTERS	192
95	CREATE WORD PUZZLES WITH YOUR MICROCOMPUTER	194
95	ELECTRONICALLY SPEAKING	197
90	BASIC PROGRAMS FOR SCIENTISTS AND ENGINEERS	188
90	CELESTIAL BASIC	191
90	EXECUTIVE COMPUTING	198
90	EXECUTIVE PLANNING WITH BASIC	199
90	EXPERIMENTS IN ARTIFICIAL INTELL. FOR SMALL COMPUTERS	200
90	FORTRAN PROGRAMS FOR SCIENTIST AND ENGINEERS	200
90	LEGAL CARE FOR YOUR SOFTWARE	205
90	MORE BASIC COMPUTER GAMES	206
90	PASCAL PROGRAMS FOR SCIENTISTS AND ENGINEERS	209
90	PRACTICAL PASCAL PROGRAMS	210
85	BASIC BUSINESS SOFTWARE	182
85	BASIC COMPUTER GAMES	182
85	BASIC COMPUTER PROGRAMS FOR BUSINESS, VOL.1	184
85	BASIC COMPUTER PROGRAMS FOR BUSINESS, VOL.2	184
85	BEAT THE ODDS	189
85	DATA BASE MANAGEMENT SYSTEMS	195
85	DBASE II USER'S GUIDE	196
85	MOST POPULAR SUBROUTINES IN BASIC	207
85	SOME COMMON PASCAL PROGRAMS	212
85	STIMULATING SIMULATIONS	213
85	SUCCESSFUL SOFTWARE FOR SMALL COMPUTERS	214
85	67 READY TO RUN PROGRAMS IN BASIC	219
80	BASIC COMPUTER PROGRAMS IN SCIENCE AND ENGINEERING	186
80	BRAIN TICKLERS: PUZZLES & PASTIMES	190
80	CODES, CIPHERS AND COMPUTERS	193
80	INSIDE BASIC GAMES	203
80	INTRODUCTION TO COMPUTER ANIMATION	204

SUBJECT: General Software & Application

RATING	TITLE	PAGE NO.
75	GIANT BOOK OF COMPUTER SOFTWARE	202
75	TEN EASY PIECES	215
75	33 CHALLENGING COMPUTER GAMES FOR TRS-80/APPLE/PET	216
70	BASIC COMPUTER PROGRAMS FOR THE HOME	185
70	MORE CHESS AND COMPUTERS	206
70	PLAYING THE STOCK & BOND MARKETS	209
60	FUN WITH MICROCOMPUTERS AND BASIC	201
60	GAME PLAYING WITH BASIC	201
55	55 ADVANCED COMPUTER PROGRAMS IN BASIC	217
50	57 PRACTICAL PROGRAMS & GAMES IN BASIC	218
45	50 PROGRAMS IN BASIC - 2nd Edition	216
30	MURDER IN THE MANSION 2nd Edition	208
10	SIXTY CHALLENGING PROBLEMS WITH BASIC SOLUTIONS	211

SUBJECT: IBM Personal Computer

95	IBM/PC GUIDE	277
95	LEARNING IBM BASIC FOR THE PERSONAL COMPUTER	280
95	USING YOUR IBM PERSONAL COMPUTER	285
90	GUIDE TO PROGRAMMING IBM PERSONAL COMPUTER	275
90	IBM'S PERSONAL COMPUTER	279
90	PRACTICAL BASIC PROGRAMS - IBM PERSONAL COMPUTER EDITION	281
90	USING THE IBM PERSONAL COMPUTER	284
85	BASIC EXERCISES FOR THE IBM PERSONAL COMPUTER	275
85	IBM PC BASIC QUICK REFERENCE GUIDE	276
85	SOME COMMON BASIC PROGRAMS - IBM PERSONAL COMPUTER ED.	282
85	USER'S HANDBOOK TO THE IBM PERSONAL COMPUTER	283
75	USING THE IBM PERSONAL COMPUTER	285
70	IBM PERSONAL COMPUTER	278
65	IBM PERSONAL COMPUTER	279
55	UNDERSTANDING THE IBM PERSONAL COMPUTER VOL.I	282
50	HOW TO WRITE AN IBM-PC PROGRAM	276

SUBJECT: Microcomputer Introductions

95	COMPUTER DICTIONARY 3rd Edition	9
----	---------------------------------	---

SUBJECT: Microcomputer Introductions

RATING	TITLE	PAGE NO.
95	COMPUTER DICTIONARY & HANDBOOK 3rd Edition	9
95	CRASH COURSE IN MICROCOMPUTERS	18
95	GUIDE TO SYSTEMS APPLICATIONS	24
95	HOME COMPUTERS VOL 1: HARDWARE	27
95	KIDS AND COMPUTERS	35
95	MICROCOMPUTER DICTIONARY 2nd Edition	37
95	NAILING JELLY TO A TREE	41
95	PERSONAL COMPUTER BOOK	44
95	SOUL OF A NEW MACHINE	50
90	COMPLETE HANDBOOK OF PERSONAL COMPUTER COMMUNICATIONS	6
90	COMPUTER ESTABLISHMENT	11
90	COMPUTERS FOR EVERYBODY	15
90	DEVIL'S DP DICTIONARY	19
90	ESSENTIAL COMPUTER DICTIONARY AND SPELLER	22
90	FIRST BOOK OF MICROCOMPUTERS	22
90	MICROCOMPUTER BUYER'S GUIDE	36
90	MODERN COMPUTER CONCEPTS	40
90	PERSONAL COMPUTING 2nd Edition	45
90	SO YOU ARE THINKING ABOUT A SMALL BUSINESS COMPUTER	49
90	UNDERSTANDING COMPUTER SYSTEMS	51
90	YOUR FIRST COMPUTER	58
90	YOUR OWN COMPUTER	59
85	COMPUTER CAPERS	7
85	DON'T!	20
85	HOME COMPUTERS VOL 2: SOFTWARE	27
85	HOW TO BUY A PERSONAL COMPUTER	28
85	INTRODUCTION TO MICROCOMPUTERS-VOL. 0	34
85	MICROCOMPUTERS AND THE 3 R's	39
85	PEANUT BUTTER & JELLY GUIDE TO COMPUTERS	43
85	UNDERSTANDING COMPUTERS	52
85	USING MICROCOMPUTERS IN BUSINESS	55
80	BEGINNER'S GUIDE TO COMPUTERS	3
80	BUSINESS SYSTEM BUYER'S GUIDE	4
80	COMPUTER CONSCIOUSNESS	8
80	ELECTRONIC COTTAGE	21
80	INVESTOR'S COMPUTER HANDBOOK	35
80	NCR DATA COMMUNICATIONS CONCEPTS	42

SUBJECT: Microcomputer Introductions		
RATING	TITLE	PAGE NO.
80	RUNNING WILD	47 ^a
80	SMALL COMPUTERS FOR THE SMALL BUSINESSMAN	48
80	UNDERSTANDING DATA BASE MANAGEMENT	54
80	UNDERSTANDING THE MICRO	54
80	60-MINUTE GUIDE TO MICROCOMPUTERS	60
75	BEGINNER'S GUIDE TO BUYING A PERSONAL COMPUTER	3
75	COMPUTER LANGUAGE REFERENCE GUIDE	11
75	MICROCOMPUTER PRIMER 2nd Edition	37
75	WHY DO YOU NEED A PERSONAL COMPUTER?	56
70	COMPLETE BOOK OF HOME COMPUTERS	5
70	FOUNDATIONS OF COMPUTER TECHNOLOGY	23
65	ARE YOU COMPUTER LITERATE?	2
65	COMPUTERS AND PROGRAMMING GUIDE	14
65	CONSUMER'S GUIDE TO PERSONAL COMPUTING & MICROCOMPUTERS	16
65	COURSEWARE IN THE CLASSROOM	17
65	INTRODUCTION TO MICROCOMPUTERS FOR THE HAM SHACK	33
65	PERSONAL COMPUTERS IN BUSINESS	44
65	PERSONAL GUIDE TO PERSONAL COMPUTERS	46
65	UNDERSTANDING ARTIFICIAL INTELLIGENCE	51
60	COMPUTERS FOR PEOPLE	15
60	GUIDEBOOK TO SMALL COMPUTERS	25
55	HOME COMPUTERS	26
55	MICROCOMPUTERS FOR BUSINESS APPLICATIONS	40
50	COMPUTERS AND EDUCATION	13
50	HOW TO BUY A PERSONAL COMPUTER (WITHOUT ANXIETY)	29
50	HOW TO BUY THE RIGHT SMALL BUSINESS COMPUTER SYSTEM	30
50	WITHOUT ME YOU'RE NOTHING	57
45	COMPUTER LITERACY	12
45	HOW TO BUY AND USE MINICOMPUTERS & MICROCOMPUTERS	29
45	UNDERSTANDING COMPUTERS	53

SUBJECT: Microcomputer Introductions

RATING	TITLE	PAGE NO.
40	INTERNATIONAL MICROCOMPUTER DICTIONARY	32
30	HOW TO CHOOSE YOUR SMALL BUSINESS COMPUTER	31
30	HOW TO PROFIT FROM YOUR PERSONAL COMPUTER	31

SUBJECT: Microprocessors

100	8086/8088 16-BIT MICROPROCESSOR PRIMER	87
95	FROM CHIPS TO SYSTEMS	63
95	OSBORNE 4 & 8-BIT MICROPROCESSOR HANDBOOK	69
95	OSBORNE 16-BIT MICROPROCESSOR HANDBOOK	70
95	PROGRAMMING THE Z80	72
95	PROGRAMMING THE 6502	73
95	8080/Z80 ASSEMBLY LANGUAGE	84
90	INTRODUCTION TO MICROCOMPUTERS- VOL. 1	66
90	Z8000 ASSEMBLY LANGUAGE PROGRAMMING	80
90	68000 ASSEMBLY LANGUAGE PROGRAMMING	84
90	8086/8088 PRIMER 2nd Edition	86
85	HOW TO PROGRAM MICROCOMPUTERS	65
85	Z80 AND 8080 ASSEMBLY LANGUAGE PROGRAMMING	74
85	Z80 ASSEMBLY LANGUAGE PROGRAMMING	75
85	6502 ASSEMBLY LANGUAGE PROGRAMMING	82
80	INTRODUCTION TO 8080/8085 ASSEMBLY LANGUAGE PROGRAMMING	67
80	Z80 INSTRUCTION HANDBOOK	77
75	Z80 SOFTWARE GOURMET GUIDE AND COOKBOOK	78
75	6502 SOFTWARE GOURMET GUIDE & COOKBOOK	82
75	6800 SOFTWARE GOURMET GUIDE & COOKBOOK	82
75	8080 SOFTWARE GOURMET GUIDE & COOKBOOK 2nd Edition	84
70	MICROPROCESSORS	69
70	Z80 MICROPROCESSOR PROGRAMMING & INTERFACING	77
65	PROGRAMMING A MICROCOMPUTER: 6502	71

SUBJECT: Microprocessors

RATING	TITLE	PAGE NO.
55	68000: PRINCIPLES AND PROGRAMMING	83
50	FUNDAMENTALS OF DIGITAL COMPUTERS	65
45	Z80 USERS MANUAL	79

SUBJECT: Operating Systems & Hdw Design

95	BUILD YOUR OWN Z80 COMPUTER	91
95	CIARCIA'S CIRCUIT CELLAR, VOL.1	92
95	CIARCIA'S CIRCUIT CELLAR, VOL.2	93
95	CIARCIA'S CIRCUIT CELLAR, VOL.3	93
95	CP/M HANDBOOK	94
95	INTRODUCTION TO THE UCSD p-SYSTEM	99
95	MASTERING CP/M	100
90	CP/M PRIMER	95
90	OSBORNE CP/M USER GUIDE	104
90	PROGRAMMER'S GUIDE TO CP/M	105
85	MICROCOMPUTER OPERATING SYSTEMS	102
85	MICROPROCESSOR OPERATING SYSTEMS, VOL.II	103
85	PERSONAL COMPUTERS HANDBOOK	105
80	CP/M REVEALED	95
80	CP/M SIMPLIFIED	97
80	GETTING ACQUAINTED WITH MICROCOMPUTERS	97
80	INTRODUCTION TO MICROPROCESSORS	99
75	MICROCOMPUTER DISK TECHNIQUES	101
75	USING CP/M	109
70	USER GUIDE TO THE UNIX SYSTEM	108
65	S-100 BUS HANDBOOK	107
45	HOW TO GET STARTED WITH CP/M	98
40	S-100 & OTHER MICRO BUSES 2nd Edition	106
SUBJECT: Other Programming Languages		
90	C PRIMER	169
90	LISP	173
80	ADA: AN INTRODUCTION and ADA REFERENCE MANUAL	167
80	STRUCTURED COBOL	176
80	UNDERSTANDING LISP	178
70	GUIDE TO PL/M PROGRAMMING FOR MICROCOMPUTER APPLICATIONS	172
65	ANS COBOL 2nd Edition	168
60	STARTING FORTH	175
55	COBOL WITH STYLE: PROGRAMMING PROVERBS	170
50	UNDERSTANDING APL	177
50	UNDERSTANDING COBOL	178

SUBJECT: Other Programming Languages

RATING	TITLE	PAGE NO.
45	FORTH PROGRAMMING	171
45	INTRODUCTION TO FORTH	172
45	PROBLEM SOLVING PRINCIPLES FOR ADA PROGRAMMERS	175
35	DISCOVER FORTH	170

SUBJECT: Pascal

95	INTRODUCTION TO PASCAL	156
95	PASCAL FOR BASIC PROGRAMMERS	159
95	PASCAL HANDBOOK	161
90	PASCAL	157
90	PASCAL: A PROBLEM SOLVING APPROACH	158
90	PASCAL FOR PROGRAMMERS	160
90	PROBLEM SOLVING AND STRUCTURED PROGRAMMING IN PASCAL	165
85	FIRST COURSE IN COMPUTER PROGRAMMING USING PASCAL	153
85	PASCAL PRIMER	162
85	PASCAL USER MANUAL AND REPORT 2nd Edition	163
80	PASCAL FROM BASIC	160
75	UNDERSTANDING PASCAL	166
65	ELEMENTARY PASCAL	153
65	FUNDAMENTALS OF MICROCOMPUTER PROGRAMMING	155
65	POCKET GUIDE TO PASCAL	164
65	SPEAKING PASCAL	165
55	PASCAL WITH STYLE: PROGRAMMING PROVERBS	164

SUBJECT: Timex-Sinclair ZX-81

95	BYTEING DEEPER INTO YOUR TIMEX/SINCLAIR 1000	332
95	ZX-81 COMPANION	341
90	COMPUTERS FOR KIDS - SINCLAIR EDITION	333
90	MASTERING YOUR TIMEX/SINCLAIR 1000 PERSONAL COMPUTER	336
85	GETTING ACQUAINTED WITH YOUR ZX81	334
85	MAKING THE MOST OF YOUR ZX-81	335
85	MASTERING MACHINE CODE ON YOUR ZX-81	335
85	SINCLAIR ZX-81	337
85	YOUR TIMEX SINCLAIR 1000 & ZX81	340
80	ZX-81 BASIC BOOK	340
80	49 EXPLOSIVE GAMES FOR THE ZX-81	344

SUBJECT: Timex-Sinclair ZX-81

RATING	TITLE	PAGE NO.
75	CRUNCHERS	333
75	TIMEX PERSONAL COMPUTER MADE SIMPLE	338
75	TIMEX-SINCLAIR 1983 DIRECTORY	339
60	101 TIMEX 1000/SINCLAIR ZX-81 PROGRAMMING TIPS & TRICKS	342
50	37 TIMEX 1000 SINCLAIR ZX-81 PROGRAMS	343

SUBJECT: TRS-80

100	BASIC FASTER AND BETTER & OTHER MYSTERIES	286
95	CIRCUIT DESIGN PROGRAMS FOR THE TRS-80	287
95	LEARNING TRS-80 BASIC FOR MODELS I, II/16, AND III	296
95	TRSDOS 2.3 DECODED & OTHER MYSTERIES	300
95	WRITING BASIC ADVENTURE PROGRAMS FOR THE TRS-80	309
90	COMPUTERS FOR KIDS - TRS-80 EDITION	288
90	GUIDE TO PROGRAMMING IN LEVEL II BASIC	292
90	MICROSOFT BASIC DECODED & OTHER MYSTERIES FOR THE TRS-80	297
90	MOSTLY BASIC - APPLICATIONS FOR YOUR TRS-80/BOOK 2	298
90	PRACTICAL BASIC PROGRAMS - TRS-80 LEVEL II EDITION	298
90	TRS-80 COBOL	303
90	TRS-80 DISK & OTHER MYSTERIES	305
90	TRS-80 MORE THAN BASIC	308
85	CUSTOM TRS-80 & OTHER MYSTERIES	289
85	GENIE IN THE COMPUTER	291
85	PROGRAMS FOR BEGINNERS ON THE TRS-80	298
85	TECHNIQUES OF BASIC	299
85	TRS-80 GRAPHICS	307
85	80 PRACTICAL TIME-SAVING PROGRAMS FOR THE TRS-80	310
80	TRS-80 COLOR BASIC	304
80	TRS-80 FOR KIDS FROM 8 TO 80 Volume I	306
75	DATA MANAGEMENT TECHNIQUES	290
75	FAST BASIC	290
75	MOSTLY BASIC - APPLICATIONS FOR YOUR TRS-80/BOOK 1	297

SUBJECT: TRS-80

RATING	TITLE	PAGE NO.
60	INTERMEDIATE PROGRAMMING FOR THE TRS-80 MODEL I	294
50	HOW TO WRITE A TRS-80 PROGRAM	292
45	I SPEAK BASIC TO MY TRS-80	293
40	TRS-80 BASIC	302
30	TRS-80 ASSEMBLY LANGUAGE MADE SIMPLE	301

SUBJECT: Visicalc

90	MASTERING VISICALC	222
90	VISICALC FOR APPLE II PLUS	224
90	VISICALC FOR THE IBM PERSONAL COMPUTER	225
90	VISICALC FOR THE TRS-80 MODEL I & III COMPUTERS	225
90	VISICALC FOR THE TRS-80 MODEL II & 16 COMPUTERS	225
85	DOING BUSINESS WITH VISICALC	219
85	VISICALC, ADVANCED VERSION	223
75	VISICALC BOOK - APPLE EDITION	224
75	VISICALC BOOK - ATARI EDITION	224
70	VISICALC HOME AND OFFICE COMPANION	226
65	HOW TO USE VISICALC SUPERCALC	221

SUBJECT: Word Processing

95	INTRODUCTION TO WORDSTAR	230
90	WORD PROCESSING PRIMER	232
90	WRITING WITH A WORD PROCESSOR	234
85	WORD PROCESSING BOOK	231
80	COMPUTER WORD PROCESSING	227
70	INTRODUCTION TO WORD PROCESSING	229
70	LOW-COST WORD PROCESSING	230
60	WORDSTAR MADE EASY	233
45	HOW TO BUY A WORD PROCESSOR	228

About The Authors

MICHAEL NICITA has been the Director of Data Processing/Accounting for the past five years at an installation which includes an IBM System 38, an IBM System 34, and sundry microcomputers. A member of COMMON, D.P.M.A., Guidance, and the New York Amateur Computer Club, he has spoken on minicomputer and microcomputer topics at seminars and conferences, and published in trade journals. He is currently working on the remote networking of microcomputers into the System 38 database.

RONALD PETRUSHA is a doctoral student in Russian history, specializing in the computerized quantitative analysis of historical data. As a research assistant at the W. Averell Harriman Institute for Advanced Study of the Soviet Union for the past seven years, his statistical research has involved the intensive use of micro and mainframe computers. His most recent paper on quantitative historical analysis was presented at the International Colloquium on Comparative Labor History, sponsored by the Maison des Sciences de l'Homme in Paris in June 1982.

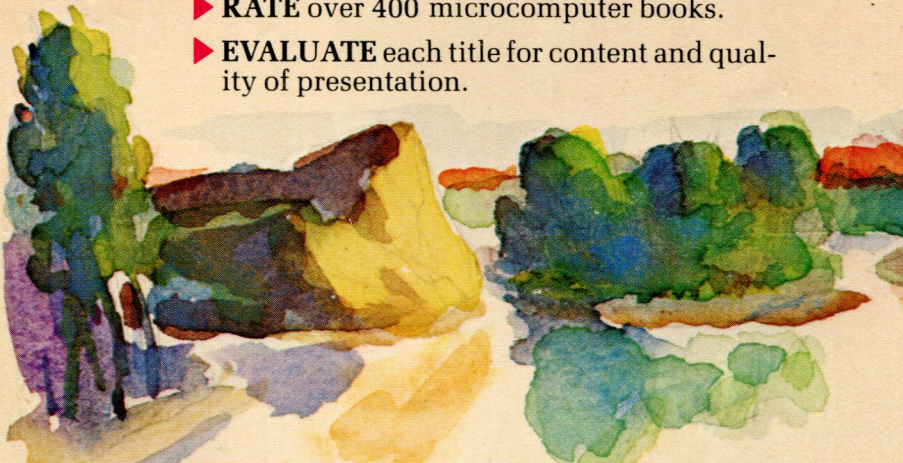
THE READER'S GUIDE TO MICROCOMPUTER BOOKS

DO YOU KNOW THE BEST BOOKS ON:

The Apple II®? CP/M®? BASIC? VisiCalc®? Pascal? The IBM Personal Computer®? Microprocessors? FORTRAN? The TRS-80®? The Timex® Sinclair 1000™? Hardware Design? Assembly Language? The Commodore Vic-20®? Word Processing? Forth? The S-100 Bus? The Atari 400/800®? Games? Software? General Introductions to Microcomputers?

Avoid costly mistakes. Find out in the New Reader's Guide to Microcomputer Books. Informative, capsule reviews:

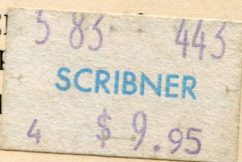
- ▶ **RATE** over 400 microcomputer books.
- ▶ **EVALUATE** each title for content and quality of presentation.



FOR BEGINNERS: An introduction to all other introductions.

FOR EXPERIENCED USE: Work as important as your own.

FOR ALL READERS: A unique book of titles on microcomputers.



THE DEFINITIVE GUIDE TO BOOKWARE