

First Edition

# 80micro's

REVIEW GUIDE



**Over 500 concise reviews of TRS-80\* software, hardware, and books, from the pages of *80 Micro*.**

**The most complete guide to TRS-80 products available.**

\* TRS-80 is a trademark of Radio Shack, a division of Tandy Corp.

80micro's  
REVIEW GUIDE  
★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★



**80micro's**  
**REVIEW GUIDE**  


---

Coordinating Editor—Janet Fiderio  
Book Design—Laura Landy  
Cover Design—Jonathan Graves and Howard Happ  
Illustrations—Phillip Geraci

80 Micro Books, a subsidiary of Wayne Green Inc.  
Peterborough, New Hampshire 03458



---

Published by *80 Micro*  
Peterborough, NH 03458

Copyright © 1983 by Wayne Green Inc.

All rights reserved. Printed in the United States of America. No part of this publication may be reproduced, stored in any retrieval system, or transmitted in any means—electronic, mechanical, photocopying, recording, or otherwise—without prior written permission of the publisher.

---

# CONTENTS

<b>Preface</b> . . . . .	vii
<b>Introduction</b> . . . . .	ix
<b>Acknowledgements</b> . . . . .	xi
<b>Software</b> . . . . .	xiii
Business/Professional Applications . . . . .	1
Compilers . . . . .	16
Data Base Managers . . . . .	26
Disk and Cassette Operating Systems . . . . .	52
Editor/Assemblers . . . . .	61
Education . . . . .	74
Games . . . . .	82
Graphics . . . . .	141
Home Applications . . . . .	146
Languages . . . . .	149
Monitors . . . . .	179
Music/Voice Synthesis . . . . .	191
Terminal Packages/Communication Software . . . . .	195
Utilities . . . . .	205
Word Processors . . . . .	238
Miscellaneous . . . . .	266
<b>Hardware</b> . . . . .	273
Disk Drives . . . . .	275
Disk Drive Controllers . . . . .	279
Expansion Modifications . . . . .	288
Modems . . . . .	301
Music and Voice Synthesizers . . . . .	305
Printers/Plotters . . . . .	313
Printer Spoolers/Buffers/Interfaces . . . . .	345
Speed-Up Kits . . . . .	349
Systems . . . . .	354
Upper- Lowercase Modifications . . . . .	362
Miscellaneous . . . . .	364
<b>Books</b> . . . . .	379
<b>Indexes</b> . . . . .	437
Software Products Index . . . . .	439
Hardware Product Index . . . . .	447

---

Book Index . . . . .	451
Color Computer Software Index . . . . .	453
Model II Software Index . . . . .	455
Software Manufacturers' Address Index . . . . .	457
Hardware Manufacturers' Address Index . . . . .	463



---

# PREFACE

**T**he microcomputerist has an insatiable need for information. Knowledge is his most important tool, for knowledge is what ultimately defines his computer's abilities and potentials. The more he knows, the greater his skills, and the more useful his machine.

It is not surprising, then, that product reviews are among a microcomputer magazine's most well-read features. *80 Micro's* surveys show, for instance, that reviews are second only to the new products section in popularity. Nearly everyone wants to expand his software library or is looking for a new peripheral to add to his system. Next to hands-on experience and the word of friends, the review is the computerist's best source of solid information.

But reviews are ineffective—indeed, potentially damaging—if they are not objective. Unfortunately, many magazines and reviewers have been so caught up in their enthusiasm for computing that they have overlooked the industry's blemishes. We believe that *80 Micro* has avoided this trap. As the largest independent source of information on the TRS-80, we have been acutely aware of our responsibility to print critical product evaluations. If a product has problems, we'll say it. This policy has resulted in a certain amount of criticism from manufacturers (and even some readers). But we feel that it is the correct one if we are to help the consumer make proper buying decisions.

This does not mean that our reviews are the final word on TRS-80 products. In fact, the consumer who decides to buy or not buy a product strictly on the basis of a review is asking for trouble. A software package might, for instance, meet the needs of the reviewer, but lack some important features that the reader's application requires. Or a reviewer might criticize a product for failing to have capabilities that the reader doesn't need.

Remember that each review, no matter how objective and fair, is only one person's opinion. The discriminating buyer will read other reviews, seek the judgement of users, and, most important, test the product out, either at a computer store or on a friend's machine.

Nevertheless, if you are an active TRS-80 computerist, we feel that *80 Micro's Review Guide* can be an important tool. Nowhere else will you find a book with as much comprehensive information on TRS-80 products. If you want to find out the features of a software package, or compare the prices of two similar products, or learn what problems you might run into with a piece of hardware you've been looking at, this will be the first place you'll turn to.

One last word. Write and let us know what you like and dislike. With your input, we'll make future editions even more useful.

Eric Maloney  
Managing Editor  
*80 Micro*



---

# INTRODUCTION

You're the owner of a TRS-80 microcomputer. What information will you get in this book that you can't get somewhere else?

Most important, you'll find almost every review *80 Micro* published in its first three years (1980-1982). You can even read our original review of the Model I. We've condensed these reviews to make them more accessible, yet have tried to retain the review's essential information and the reviewer's opinions about the product. Reviews are cross-referenced and are in alphabetical order by category.

Each review also includes the following:

- A rating of one to five stars.

- Updated information on the manufacturer's name and address. If the manufacturer's name has changed, the original will appear at the top of the review and the new one at the bottom.

- Updated information on the product's name. Again, if the product has a new name, the original is at the beginning and the new name at the end.

- Model compatibility. All software and most hardware reviews list the models the product worked with at the time the review was written. If the product is now available for other TRS-80s, the models are listed at the close of the review.

- Product availability. If a product is no longer available, or if it is available from a source other than the original, it will say so at the end of the review.

- A list of manufacturers' addresses.

- Current price information.

- The date the original review was published, and the author of the review.

We contacted every company listed for updated information. If we could not reach the company, we say so at the end of the review.

And to help you along, we've also included complete indexes for software (with separate indexes for the Color Computer and the Model II/12/16), hardware, and books, and a list of manufacturers' addresses.

Most of the reviews in this book are for the Models I and III. If you own a Model 4, you should be able to run the majority of Model III (and converted Model II) programs. We recommend, however, that you first contact the manufacturer. We suggest the same to Model 12 or 16 owners who are interested in Model II packages.

You might wonder why we've included reviews of products that are no longer being marketed. We had several reasons. First, many out-of-print programs are still available from other sources, including user groups and bulletin boards. Second, we felt that these reviews

---

would help give computerists—particularly new ones—an overview of the market. Thus, we think you'll find this book not only informative, but an interesting glimpse at the history and evolution of the TRS-80.

So there you have it. Whatever your microcomputing interests, we think you'll enjoy browsing through our collection.

Janet Fiderio  
Coordinating Editor



---

# ACKNOWLEDGEMENTS

Many people have contributed their talents to this guide and deserve thanks. Foremost among them are the authors of the reviews compiled here.

My personal thanks to the *80 Micro* editors who painstakingly condensed each review and were helpful in innumerable ways: Eric Grevstad, Eric Maloney, Peter McKie, John Meilo, Michael Nadeau, Lynne Patnode, Lynn Rognsvoog, Mary Ellen Ruth, Deborah Sargent, and Stephen Tomajczyk. Special thanks go to Terry Kepner, our technical editor.

I would also like to thank Michele DesRochers, who typeset the guide; Peter Bjornsen, Louis Marini, and the proofreading staff; Nancy Salmon, David Wozmak, Karen Wozmak, and the production department.

Thank you to Laura Landy for her time and effort, and to Phil Geraci for his fine illustrations. To all the others who contributed, my thanks and appreciation.

J.F.  
May 1983







**BUSINESS/  
PROFESSIONAL  
APPLICATIONS**

★ ★ 1/2

**Ann Rose—Our Accounts-  
Receivable Clerk  
Sturdivant & Dunn  
Model I or III**

Ann Rose—Our Accounts-Receivable Clerk was designed specifically for applications where the majority of the billings are repetitive standard amounts.

The Model I version requires two disk drives; one for the program and data, and one for data exclusively. You can modify the Model III version to run with one disk.

Ann Rose is a bill-and-post system. You add a new account by coding it according to a table that you have set up. When you select the automatic billing feature, these codes identify the standard rate to charge.

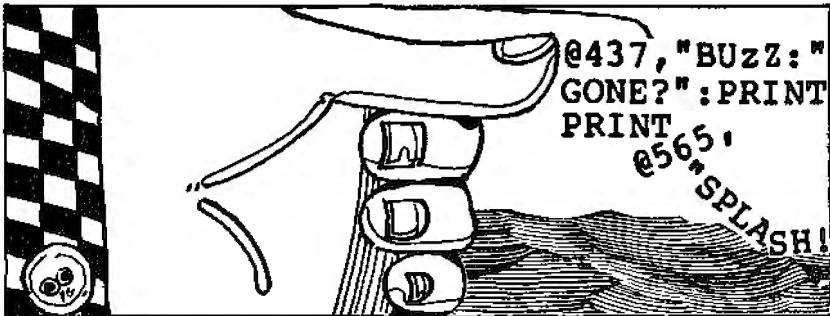
You can establish multiple standard rates and apply them to different customers as needed. You can also bill directly in addition to the standard amount. A

code of 1 is used for accounts that only incur direct charges.

The system uses a clever technique for posting specific items to an account. It assumes that all positive numbers are charges and that all negative numbers are payments. The correction routine changes negative amounts from payments into returned items or discounts, and positive amounts from charges into service charges. You must also use the routine to correct account number, invoice number, or dollar-amount problems.

In reality, Ann Rose is a 17-line computerized ledger card. Unfortunately, since the system can't key payments against specific charges, problems could develop if you had to reconcile a high-volume account. Because the program prints only 17 transactions and the net balance due, the amount paid could be for details previously purged.

The monthly report of customer balances requires some explanation since at first glance it appears to be out of balance. The detail columns provided—service charge, 30-60 days, and over 60 days—do not necessarily total to the balance-due column. The difference is the current amount due, found



on the posting-to-accounts-receivable report.

Since the system is set up to print at 10 pitch on 8 1/2 by 11-inch paper, space is at a premium and some information was omitted. When Peter C. Dunn, a CPA with the firm of Sturdivant and Dunn, developed the program, he should have included the total balance due as a single number or used a short name for the customer so that the report would crossfoot.

The program provides an accounts-receivable list with the account number, code, name, phone number, and blank spaces in the columns labeled charge and payment. At first it appears that it should contain a recap of the financial figures, but it is intended as a worksheet for the bookkeeper.

The system's major weakness is its documentation. The manual starts out by providing a step-by-step list of instructions for formatting new disks and backing up and copying the program from the nonsystem disk provided. The documentation explicitly reminds the user to make adequate back-up copies every time he runs the system. (After terminating processing, the system even displays a parting message of "MAKE BACK-UPS.")

Unfortunately, there is very little follow-up documentation on how to run the system. Essentially, the manual says to experiment—an excellent way to learn, but reading about it first would have been better. A new and expanded version of the documentation should be available soon.

The system has other deficiencies. For example, bills are

printed without account numbers and there is no provision for sales or commission analyses.

I would recommend this system only under certain circumstances. At present Ann Rose could be ideal for a small business with repetitive standard billings. However, the lack of comprehensive documentation limits its usefulness to someone already familiar with data processing.

*(Sturdivant and Dunn Inc., Box 277, 124 Washington St., Conway, NH 03818; \$150, 48K required.)*

Frederic S. Goldstein  
May 1982

★ ★ ★ ★

**Check Register Plus  
Softco  
Model I and III**

★ ★ ★ ★

**Checkwriter-80  
Tandy/Radio Shack  
Model III**

★ ★ ★ ★

**Maxi Check Register  
Accounting System  
Exador Inc./Adventure  
International  
Model I, II, and III**

Small businesses considering TRS-80 bookkeeping software have many programs from which to choose. Rather than buy a full-fledged business system including payroll, general ledger, accounts payable and accounts receivable, many of these businesses might be better served by some type of check register program, which is basically an elementary general ledger.

These three programs can be used by individuals as well as businesses. However, all are more complicated than simple in-memory household checkbook programs. These programs do much more than balance the checkbook; they are sophisticated recordkeeping systems. If you normally itemize deductions, or have special financial problems such as rental income/expenses, these programs may be well worth the time and expense. Otherwise, they may be more program than you need or want.

Two of the programs, Maxi Check Register Accounting System (Maxi CRAS) and Checkwriter-80, print the checks, if desired, to avoid the double work of typing an entry into the computer and typing it again on a check. Both programs also balance the checkbook, cancel checks, and reconcile the bank statement. Maxi CRAS has the additional feature of interacting with VisiCalc.

Check Register Plus (CRP) is quite different from the other two since it does not balance the checkbook, write checks, or reconcile the bank statement. It does maintain payroll records and totals for tax and FICA purposes, and can even print W2 forms, with the purchase of an extra module.

Maxi CRAS requires a Model I or III with 48K and two drives, or Model II with 64K and one drive. Printers which require special drivers may be used. Its capacity is 223 accounts, divided as needed between expense accounts and income accounts. Several disks may be used, if necessary, to store all the transactions for a year.

Checkwriter-80 is for the

Model III only. It requires 48K and two drives. Printers that require special drivers are not supported and a parallel printer must be on-line before the program will run. The program is written in Cobol, and allows 33 expense accounts, 33 income accounts, 75 payee names/addresses, and nine bank accounts.

Check Register Plus needs a Model I or III, 48K and two drives. Special printer drivers are not supported.

Its capacity is 400 accounts, divided as needed between income and expense. A Model II version is also available. The Model II version needs 64K, one drive, and includes check reconciliation.

### **Checkwriter-80**

Checkwriter-80 is a well written, easy-to-use program. Screens in various parts of the program all use the same control keys, and errors are simple to correct. A minor annoyance is that control-key entries must be in uppercase. The manual is well organized and easy to understand.

Checks are numbered internally by the program, and the actual check number is not used until the checks are printed. At that time, the starting number on the forms may be used. If the check is written manually, there is no way to enter the actual check number into the program. A check which is cancelled accidentally may be restored.

One deposit is allowed per account per day. When you receive bank statements, the program reconciles the balance. This is the only point at which it is easy to obtain a checking account

balance. Normally, to find the balance you must note the beginning balance, obtain a printout of the check register, and manually add the total deposits and subtract the total checks. This is a lot of work for such an important piece of information.

After the monthly statement is reconciled, the cancelled checks and deposits are cleared off the disks. Only the totals are retained in the income and expense accounts, not the actual transactions. At the end of the year, a final clearing gives totals for the whole year in each account.

### **Maxi CRAS**

Like Checkwriter-80, Maxi CRAS is initialized by deciding on expense and income account titles. It is wise to include a few empty accounts in case you need more later. After you enter accounts, obtain a printout for reference when entering transactions.

A unique feature is this program's ability to allocate a check, in one transaction, to more than one expense category. You may also allocate to accounts by percent or by fractions, rather than actual dollar amount. Furthermore, individual expense entries are saved by the system for the entire year, so that the final summary lists every item in each expense category. This is not true of the other two programs.

Check entry involves typing the payee, amount and purpose of the check. You type a check number if you write the check manually. For printed checks, you enter the starting number at printout time. The expense ac-

counts are chosen by account number, but their names do not print on the screen, so it is easy to make an error and not notice it. To correct an error, the entire transaction is erased and you must begin again.

During check entry, the balance is available at any time—an important convenience. Check entry is made as rapid as possible by liberal use of default values when you press enter.

If you find an error after a check has been written to disk, it is not possible to alter or delete the check. A memo transaction is created to fix the error. If you do not find the error until after the check is printed, the check may not be reprinted. The advantage of this system is that all entries are permanent and cannot be maliciously or inadvertently altered. For personal use, however, the system is not as simple as Checkwriter-80's check edit function.

The program maintains a payee and address file. You may use either file, both or none, depending on your needs.

### **Check Register Plus**

CRP is actually a general ledger program whose function is solely to chart the flow of money, not to balance the checkbook. As such, it does not have any of the checkbook features of the other two, and it does not print checks.

Surprisingly, this is the only program of the three which has password security. The security is on three levels, and passwords may be easily changed by the person having the top-level password.

This is also the only program that does not allow backup of



the program disk. CRP is shipped with a backup disk provided by Softeo. If the original disk is harmed, the backup may be used while you wait for the free replacement. The programs also may not be listed. I would feel more comfortable with this system if the program disk could be write-protected. However, during operation the program disk is used as a backup for data before posting. Since the program disk is used regularly for write as well as read, and must remain in the drive whenever the program is in use, eventual damage to the disk seems likely. As long as Softeo will replace it at no charge, however, it is difficult to object to their policy.

The CRP manual is well written and nicely presented, with important points well marked. However, at the end the reader may still have some unanswered questions, such as, "How do you handle income?" It is never mentioned in the manual.

CRP lets you choose your own account categories, or use the samples provided. The program automatically sorts alphabetically on the account number, so it is helpful to group similar expenses and give them sequential account numbers.

Because the program does not maintain a checkbook balance, cash entries can be handled as easily as checks.

The program has a particularly effective method of handling dollar entries. As you type each digit, it is printed on the right side and pushed left by the next digit. The decimal point is already on the screen and does not need to be typed. Errors may not be corrected at the time of check entry (except for payroll

checks) but may be corrected later.

CRP maintains payroll totals. Each payroll check must still be calculated by hand, but you may then enter the check and its deductions in the program, where totals accumulate for quarterly reports.

At regular intervals the checks must be posted to the accounts. A backup of the check data is automatically made on the program disk before posting occurs. As in Checkwriter-80, posting accumulates totals in each expense account. The individual entries are not retained.

### Reports

All three programs print a list of the accounts for reference, in order by account number.

Checkwriter-80 provides: a list of payee names and addresses, sorted by number and by name; a check-register statement; a list of each check and deposit; a checkbook reconciliation statement; and a year-end expense category list.

Maxi CRAS provides: a check-register statement; accounts file; payee file; address file; account distribution statement; monthly account subtotals; individual account statements; and a bank reconciliation statement.

Check Register Plus provides: account reports with user-selectable headings and contents; a check-register statement; and a quarterly payroll report.

A very welcome feature of CRP's report section is the ability to abort a printout. In the other programs, use reset or break at your own risk.

### Conclusion

Assuming all three programs are compatible with your equipment, which you purchase will be dictated by your needs. If you need a program to maintain a checkbook, choose between Checkwriter-80 and Maxi CRAS. Of these two, bear in mind that Maxi CRAS allows many more income/expense accounts, whereas Checkwriter-80 allows multiple bank accounts. It is possible to use Maxi CRAS for more than one bank account.



however, by maintaining separate sets of disks for each account.

If you want quarterly payroll data maintained, the logical choice is CRP. It also allows the greatest total number of accounts and has password security. (Checkwriter-80, Tandy/Radio Shack, Fort Worth, TX 76102; Model III 48K, \$100. Maxi CRAS, Exador Inc./Adventure International, P.O. Box 3435, Longwood, FL 32750; Model I and III 48K, Model II 64K,

\$99.95. Check Register Plus, Softco, 1516 South Orchard, Boise, ID 83705; Model I and III 48K, \$100.)

Wynne Keller  
June/July 1982

★ ★ ★

### Dynamic Report Generator Dynamic Software Model I or III

The Dynamic Report Generator (DRG) is an inexpensive program that generates columnar reports of data within specified parameters. The package includes a program disk and a manual. The disk includes two programs—RUN/CMD, a run-time module for the compiled program, and DRG/CHN.

The program is easy to use, and the manual is composed primarily of a long example that uses all the program's commands.

The Dynamic Report Generator is menu-driven and allows little flexibility in the form of the data output. This makes manual writing easier. The flexibility is limited to varying the number of columns of data, their titles, data type (string, numeric, or result), and the relationships among them.

The program is not perfect, but quite good considering the cost. A few things that might be added include better error-trapping, a sort command independent of the Terminate option, and a way to browse through the data without sending it to the printer. The program bombs on several types of input errors, requiring you to reload it, and it doesn't support any printer options other than

number of lines per page, a title, and page numbers.

Overall, the Dynamic Report Generator is quite a good deal for the money. There is nothing I know that can compare for its low cost. If you need simple reports of the type that DRG can generate, you certainly would be hard put to find a better way to spend your \$14.95.

*(Dynamic Software, 58-04 208 St., Bayside, NY 11364; \$14.95.)*

Bruce Powel Douglass  
October 1982

★ ★ ★ ★

**Electric Spreadsheet**  
**Dan G. Haney and Associates**  
**Inc.**  
**Model I and III**

Electric Spreadsheet turns your TRS-80 into a large worksheet on which you can perform complex calculations in integer, single-precision, or double-precision number representations. I reviewed the Exatron Stringy Floppy version. If you want to use the Electric Spreadsheet with a 32 or 48K machine, a \$9.95 program named @Freeze must first be purchased from Exatron.

The program is easy to use. You can select the calculation to be done, enter the number of columns and number of lines the worksheet will contain, either a one- or multiple-page format, and the number of decimal places for each column.

You can easily move around the worksheet by using the arrow keys. You can change pages by pressing the @ key. Output can be prepared at any time by pressing the question-mark key, and you can go to the menu by

pressing the slash key.

Once in the menu mode you can do the following: output to screen, output to printer—after screen output, input to screen, input to printer—after screen input, change the format, save data, find out how much string space is left, or you can just quit.

Electric Spreadsheet is quite useful and worth the price. I would rate everything about this package excellent. It is worth the asking price for the superb documentation alone.

*(Dan G. Haney and Associates Inc., San Mateo, CA 94401. This company could not be reached for an update.)*

Mark D. Goodwin  
January 1982

★ ★ ★ 1/2

**Finance—Loans and**  
**Investments**  
**Computerware**  
**Color Computer**

If you'd like your computer to figure interest on your loans or calculate how much you need to save to reach a goal, consider this program. Only three functions require a printer, so this utility is excellent for the beginner.

Finance has two separate programs: Loans and Investments. Loans has nine separate functions dealing with often used loan calculations. Most functions calculate the missing variable. This works for any missing variable by supplying the remaining variables.

You can also calculate commercial paper discount, if you can find a bank these days with a discount. You need a printer for receiving a mortgage amor-

tization table indicating payment, interest and principal. You also have the option of a printed declining-interest loan table. This program includes most of the common calculations involving loans. You can even show how much of a loan is interest as opposed to principal in the first years of home buying.

Investments, the second program in this package, contains nine separate calculations used in investments. You can calculate the future value of an investment by giving initial investments, interest rate, compounding periods, and number of years. This is handy for calculating the value of your IRA in 2098, or determining how much you can withdraw from an annuity without damaging the principal.

Another function calculates the effective interest rate given the current and future value and period. The only function that requires a printer is an earned-interest table.

The program comes on a quality cassette with a program on each side or a disk. It loads with no problem and the programs are easily written and easy to understand. There's an excellent introduction discussing copyrighted material in an intelligent fashion as opposed to stern warnings about duplication.

If you have a need for calculating loans and investments, I recommend these programs.

*(Computerware, Box 668, Encinitas, CA 92024; \$17.95 cassette, \$22.95 disk, 16K Extended Color Basic required.)*

Mark E. Renne  
November 1982

★ ★ ★ ★

### **General Accounting Package Microed Model II**

I received a product announcement from a company in California called Microed, offering both the CP/M system and an accounting package for the Model II. It appeared to be what I needed. The software package contained manuals and two disks. The first disk contained all the CP/M programs and 11 general-ledger programs. The 11 ledger programs alone occupied 218K bytes of space, in executable code, not Basic.

The first Microed manual was a summary of CP/M programs and a description of Microed's written programs for CP/M. The accounting manual is written in a self-teaching style. It is intended to be used as a step-by-step guide in setting up the accounting system on the computer.

The data disk contains all the accounting-program results. This disk also receives all the newly created files. It is similar to the data-base concept where the data disk becomes the data base.

When I first found out that the program was written in Fortran and I would not have access to the source code, I was somewhat disappointed—I would have no way to modify it for my own situation. I see now that the capability of this accounting package far exceeds my needs, and it may be quite some time before I need to modify it, if ever.

Transactions are entered using the Add General Journal Transactions program. Another program posts transactions and gives you a report containing

those transactions. Two additional programs give you a general-ledger detail report and a general-ledger summary. Several others provide for various accounting information such as annual summaries, accounts receivable, and accounts payable. My package also came with what Microed called security programs. These programs allow me to use a password for entry into the system, and hence the accounting information.

After using this package for several weeks, I have become more and more convinced that it is one of the best bargains available in off-the-shelf software. It is successful, in my eyes, because it is a complete package: system software and application software combined to run together.

*(Microed, 3910 Bandini St., San Diego, CA 92103, could not be reached for an update.)*

Hellen Huffman  
June 1981

★ ★

### **In Memory Information 3.0 Tandy/Radio Shack Model I and III**

The In Memory Information System stores information on pages of memory called cards. Each card may contain up to 14 fields of one line (41 characters each) that you define. The card has a maximum length of 255 bytes, which can limit the amount of data per field.

The package consists of three programs: Initialization, Sort, and Update and Retrieval. The Initialization program sets up the number and the length of fields and creates the initial data

file. The Sort program sorts the file in ascending or descending order, prints the data file on an optional 80-column printer, and optionally splits a long data file into two or more sections to avoid over-running available memory. The Update and Retrieval program adds, deletes, and updates data cards. Updating can be quite tedious.

Each program allows for the repetition of key entries by holding down the proper key until the desired number of characters has been printed. This has been a mixed blessing, although it allows the rapid entry of multiple characters, if you are a heavy-handed typer like myself, it acts like a badly bouncing keyboard.

The program's only real operating bug is that if the amount of available memory drops below twice the length of a data card, all of the data in memory will be lost.

This program is exactly what Radio Shack advertises, a computerized card filing system. It has potential for those who wish to keep brief records of customer payment or status in very small businesses. For most small businesses though, Radio Shack's Versafile would be better suited.  
*(In Memory Information 3.0, Tandy/Radio Shack, Fort Worth, TX 76102; cassette \$19.95.)*

Fritz Milhaupt  
November 1981

★ ★ ★ ★

### **Litigation Support System Tandy/Radio Shack Model II**

With Radio Shack's Model II

Litigation Support System, savvy lawyers no longer need to waste time searching old files. After an hour of learning the system, anyone (even a lawyer) can create, store, and retrieve files.

The Litigation Support System is composed of two files: a Clients file and a Forms file. The Clients file can hold up to 375 files. You can use the client record to store information such as the client's personal background, case history and prior correspondence. The Forms file is used to conduct legal research. The user enters the area of law, case number, keyword, or any other criteria desired, and the computer searches up to 570 records and displays any matching records.

The Clients and Forms files can also generate reports. The Client Representative report prints the current date, client's name, client number, type of case, attorney of record and other relevant information for each client selected. The Client Personal Data report prints the name, record or case number, address and phone number of each client selected. The Forms file report generates a list of cases, subjects, forms or keywords based on any criteria you select.

Perhaps the most impressive feature of this program is its documentation. The instructions are clear, concise, and easy to read.

The manual also provides meticulous instructions for printing client reports, using various formats: alphabetical order, chronological order, by attorney's name, or by just about any other criteria you

desire. You can also adjust the page length and paper width to accommodate your printer.

The Litigation Support System is designed both for expansion and for use with other Radio Shack programs. The manual explains how to merge Scriptit with it to create form letters, leases, and powers of attorney. This feature alone can save a law office much time and effort. Additionally, the system has the capacity to print mailing labels.

The Litigation Support System can be used with Radio Shack's Time Accounting System. If you have this system, you can give clients the same client numbers in both systems. This way, users of both systems can maintain the continuity, accuracy, and accessibility of the client record system.

This program works so well that you wish you had more storage capacity so you could store complete opinions, excerpts from legal treatises, or even legal briefs.

*(Tandy/Radio Shack, Fort Worth, TX 76102; \$299.)*

Edward D. Young III  
June/July 1982

### **Maxi Check Register** (See Page 2)

★ ★ ★ ★  
**Microstat**  
**Ecosoft**  
**Model II**

★ ★ ★  
**Statistical Analysis**  
**Tandy/Radio Shack**  
**Model II**

These two packages contain common statistical procedures.

Statistical Analysis runs under TRSDOS 2.0A, while Microstat runs under Pickles and Trout CP/M and MBasic. Both manuals are more than satisfactory, although the Shack manual is slicker and more tutorial. Both contain sample printouts and are easy to follow. The Microstat package, however, contains sample data that can be run to confirm the accuracy not only of the programs, but of how you are responding to the prompts.

Data files in both packages are random access. Each contains a data-handling utility to create new files and edit old files. Microstat also lets you specify the precision of the files, a desirable feature. I like Radio Shack's data entry better, however, because you can review an observation and reenter it if there is an error. With Microstat's package, you must enter the entire data file and then edit it by observation number; this is not as convenient as reentering observations before they are recorded. Microstat also claims to save every observation as it is input, although my disk drive did not run after entering every observation. Radio Shack did not provide a way to recover data if you accidentally depress Break.

Statistical Analysis enters data a little more conveniently. However, this is more than offset by Microstat's ability to transform data.

A serious deficiency in Radio Shack's data handling utility is the lack of a satisfactory method for subsetting observations in the data set. Radio Shack allows specifying up to 10 values of a particular variable that the program uses to select observations

for analysis. While this is better than nothing, it makes it difficult if you are dealing with a large data set and impossible if you wish to subset on more than one variable.

Microstat solves this problem by providing file moving and merging utilities as well as a sorting utility. You can sort on multiple keys and then subset by either copying part of a file to another file or specifying that only certain observations in the sorted file be used.

Both packages contain programs for descriptive statistics (mean, standard deviation, range, and so on) and histograms. The Radio Shack package has separate programs, while Microstat combines them. Statistical Analysis offers an option to let the computer determine the intervals based on the data, while Microstat requires that you enter the lower limits of each class.

In addition to descriptive statistics and histograms, each package offers analysis of variance, correlation analysis, regression analysis, and time-series analysis. In general, the programs do what they are supposed to. Radio Shack's ANOVA offers one-way, randomized block and two-way, factorial design, while Microstat offers simple one-way as well as the other two. Microstat's correlation program is more sophisticated in that it provides the sum-of-squares-and-crossproducts matrix as well as the variance-covariance matrix and the correlation matrix. Again, Microstat has added an extra touch that could be valuable to those doing more sophisticated research projects.

The difference in sophistication between the two programs shows up especially well when comparing the regression packages. The Shack offers multiple regression in one subprogram and simple regression within the correlation program, a satisfactory arrangement. The regression is full model with complete analysis of the variance table. Unfortunately, it does not display (or apparently compute) the *t* ratios that indicate whether a particular coefficient is significantly different from zero. You are left to do this from the output of the regression slopes and their standard deviations.

There is no Durbin-Watson statistic and no residual analysis. Microstat, on the other hand, offers both stepwise and full-model regression and provides an option to compute and plot the residuals as well as the Durbin-Watson statistic. In my estimation, these are necessities for serious research.

Time-series analysis is provided in both packages and both contain serious flaws. Statistical Analysis offers moving averages, analysis of seasonality based on the ratio of actual to moving averages, and a time-series regression program. However, there is no way to combine the time-series regression and the seasonal indices to obtain a true forecast of a seasonal series. This is a serious flaw and renders the time-series package useless.

Microstat provides moving averages, centered moving averages, deseasonalization (or the computation of the seasonal indices), and simple exponential smoothing. However, they succeed in frustrating the time-series forecaster again, since

there is no way to compute the time-series trend, much less a way to combine the trend and the seasonal indices to compute a forecast. Worst of all, the program displays the preliminary seasonal indices instead of final seasonal indices, which are normalized so that their mean value is one. Simple exponential smoothing provides a way of predicting data without a trend and works as it is supposed to.

Unfortunately, there is no computation of mean square errors or any other indication of fit, effectively eliminating the use of the program in searching for the optimal smoothing constant. This again makes this part of the program effectively useless.

Both programs generally do their jobs, but Microstat's output is more sophisticated. The one exception to this is the time-series package, which I regard, in both instances, as completely inadequate.

Programs contained in the Statistical Analysis not included in Microstat are analysis of covariance, a test-scoring package and a random-sample generator. For those who use ANCOVAR, and those who do questionnaire research, these programs are quite valuable.

Programs that Microstat offers not available in the Radio Shack package include nonparametric statistics, factorials, permutations, combinations, several probability distributions, and test for differing proportions and means.

In the final analysis, a package is only as good as its reliability and customer service. In using Microstat, there were three problems that I encountered—an illegal function call, a hang-up



when I tried to go to the printer in the probability distribution program, and the failure to read a double-precision file in the descriptive statistics program. When I notified Ecosoft of the first two problems, they asked me to return my disk for an updated version. The package is now working fine.

In summary, the Radio Shack package, although not as sophisticated as Microstat, performed as it was supposed to with no errors. Microstat, on the other hand, had errors that were satisfactorily remedied. The Radio Shack package is certainly a bargain for the price if one can live with its limitations. But for those users who require more sophisticated types of analysis and data handling capability, especially if one already owns CP/M and MBasic, the Microstat package is worth the extra money.

*(Statistical Analysis. Tandy/Radio Shack, Fort Worth, TX 76102; Model II, \$99. Microstat, Ecosoft, Indianapolis, IN 46206; Model II, \$250.)*

L. H. Zincone  
June/July 1982

★ ★ ★

### **Spectaculator Tandy/Radio Shack Color Computer**

Spectaculator is an electronic spreadsheet for performing budgetary or other tabular calculations on the Color Computer. It comes in the familiar ROM-pack format, and can be used with either 4, 16, or 32K machines. You need a cassette recorder if you want to reuse your work, and a printer is handy.

The manual leads you along very gently and can be read quickly. Almost half of it consists of sample sessions and command summaries. The Spectaculator vocabulary has 17 two-character commands, plus redefined meanings for a few individual keystrokes.

When you turn the power on, the upper left-hand section of the sheet is shown on the screen. Normally, you will see 13 rows and as many columns as will fit the video display. You can specify column widths; the default value is seven spaces. When you move the cursor the row and column ID numbers change.

A few peculiarities of the program become evident when you set out to spruce up a worksheet with labels. Numerical entries are right-justified when entered, while text entries are left-justified. Judicious use of the space bar is sometimes necessary to properly align column labels with the data they refer to. Text entries may exceed the width specification of the column they start in without affecting subsequent numerical entries. You can create new blank rows and columns at any point, so you always have the option of adding labels or data to an existing worksheet.

What makes electronic worksheets so special is being able to use predefined formulas, with numerical entries, to compute new entries, all of which can be easily updated or manipulated. This program uses a convenient, easy-to-learn syntax for setting up formulas and is geared to straightforward financial calculations, not elaborate analysis of scientific data.

A typical Spectaculator application involves identical calculations on the corresponding elements of rows in a worksheet, the results of which are displayed in a particular column. Calculations can also be carried out vertically, with elements in different rows being operated upon to yield a results row.

Spectaculator will add, subtract, multiply and divide as well as calculate square roots. Parentheses may be used freely. It also offers a SUM and SMT function. The first is a short way of forming, for each row, the sum from a designated starting column up to the column to the left of the marker's position. The SMT function, followed by a column designation, calculates the total of a specified column, and displays the running subtotals in the column where the marker is positioned.

Normally, calculated results are displayed in a dollars and cents format with two decimal places. The D causes calculations to be carried out to six decimal places, but only displays the results up to the first trailing zero. The program is inconsistent on this point: there are situations in which the first zero is displayed, and others in which it is suppressed.

Spectaculator will automatically adjust the width of a results column, to fit as many as 10 digits, to accommodate calculated values, even if you have predefined such a column's width. All Spectaculator calculations can be carried out vertically across columns, as well as horizontally. The two types of calculations can be used together in all sorts of financial work.

Spectaculator worksheets can

accommodate nearly 14,500 characters in a 16K computer, and nearly 31,000 in a 32K machine. You can save a worksheet to tape at any point, and reload it to resume work later. Data, labels, and defined formulas are all saved. You can assign a name to the tape file, and either use the name or ignore it when reloading. One annoying point is that Spectaculator doesn't give any indication of a file name during the loading process.

Worksheets can be listed on a printer; the row and column numbers are automatically suppressed when this is done, making for a more polished printout. You can select any portion of the worksheet for printing, consistent with an 80-character line.

I found Spectaculator to be useful for fairly simple budgeting, in which a few columns of results are derived from several columns of data and both horizontal and vertical summations are required. Because of its limited mathematical functions, calculations requiring operations such as exponentiation become awkward. It also prevents the taking of roots other than the square root.

Several aspects of the system only emerge when you give it an actual test case. It wasn't until I had it sum 22 rows of data that I realized there is no way to exclude individual columns from the sum if they contain numerical data. Not only did I get the total annual budgets for my 22 programs, I also got the sum of the 22 program ID numbers! There is an easy fix for this particular case: Just enter such identifying numbers as text.

Another shortcoming is the lack of a desk calculator mode for changing individual entries in the middle of a worksheet.

Spectaculator's operation has an asymmetry that deserves mention. That is, you can erase a derived column, formula and all, you can erase the data in a column, but you can't leave the data and just erase the formula. Spectaculator also lacks the ability to copy a single row or column. I also miss something like Basic's Print Using command. There really are times when you'd like calculated results to be displayed with exactly one decimal place, trailing zeroes or no!

It is not for the elaborate analysis of scientific data, but if you need straightforward manipulations for home, business, or organization, give Spectaculator a try.

*(Tandy/Radio Shack, Fort Worth, TX 76102; \$39.95.)*

Scott L. Norman  
November 1982

### **Statistical Analysis**

(See Page 10)

★ ★ ★ ★

#### **Visigraph Micro Software Systems Model I or III**

Visigraph is a TRS-80 program intended primarily to produce graphs and charts from data generated by the popular VisiCalc program. Designed to work with the Epson MX-80 and MX-100 printers (Model III), or the Epsoms plus the IDS Paper Tiger 440-G or 445-G (Model I), Visigraph offers high resolution,

the ability to title and label graphs and charts, and a user-definable character set of plotting symbols.

Visigraph also permits the graphing of your own program's data, using VisiCalc's DIF (data interchange format).

Visigraph is distributed for the Model I on single-density, 35-track, NEWDOS-formatted disks. Model III disks are available for the TRSDOS 1.3 or the DOSPLUS 3.4 user. Both versions require 48K of memory.

This is not a program for the novice computer user. Visigraph requires a working knowledge of the computer system and VisiCalc. Visigraph does not allow the plotting of VisiCalc files at the mere touch of a button; you have to install it and set up each graphform first. But, the program has many excellent features that make it a valuable tool for the VisiCalc user.

Visigraph has two features that make it especially interesting—symbol definition and auto scaling. The program uses 12 symbols for making its graphs and charts, and, if you prefer, you can create your own symbol. The auto-scaling function automatically calculates the most readable increment for any scale you attempt to plot.

Most of Visigraph's features are well-planned and executed. The documentation is well-written and complete.

If Visigraph has a fault, it is simply that it works with only certain printers. However, since it is written mostly in Basic, you can alter the program to drive any printer that recognizes CHR\$ codes.

*(Micro Software Systems,  
1815 Smokewood Ave., Fuller-*

ton, CA 92631; \$79 Model I, \$89 Model III.)

G. Michael Vose  
October 1982

---

## COMPILERS

---

★ ★ ★ ★

**ACCEL2**

**Southern Software**

★ ★ ★ 1/2

**Basic Compiler**

**Microsoft**

**Model I and III**

Here are two programs that will help you speed up your TRS-80 without resorting to hardware modifications or complete program rewrites in machine code.

The Basic Compiler from Microsoft and the ACCEL2 compiler from Southern Software can each compile Basic programs written on a disk-based TRS-80.

Microsoft's compiler appears not to have been written specifically for the TRS-80. It comes with a manual for Basic-80, which is similar to the TRS-80 implementation of Basic, but with a few enhancements (Swap, While/Wend, and Call to name three) that can be used in the compiled programs, but cannot be tested in the interpreter mode.

The Microsoft compiler produces a relocatable binary file of machine code that runs from the DOS mode. The procedure for using it is quite involved, not automatic. The operator must swap disks and interact with the computer at certain points dur-

ing the compile and link operations. The Basic program to be compiled must first be saved on disk as an ASCII file.

Southern Software's ACCEL2 is easy to use. Once loaded, the compiler need only be called by the command "/FIX" to automatically carry out the four-pass compile operation. It can be called over and over to compile different Basic programs.

ACCEL2 takes about 30 seconds to compile 200 lines of code. The Microsoft system takes several minutes to compile even a short program due to the swapping of disks and loading of library routines.

ACCEL2 does not produce a machine-code module, but operates in the Basic interpreter environment with hooks to machine-code routines. It can selectively compile specific types of statements under programmer control. It has statements to save, load, or run compiled programs, and even a limited chaining capacity that can load and run both compiled and noncompiled programs from a program that is running.

With ACCEL2, noncompiled statements are processed at runtime by the normal interpreter. This does not mean that a program compiled by ACCEL2 necessarily runs slower than the same program compiled by the Microsoft compiler. It depends on the complexity of the program.

To increase the speed of Set, Rset, and Point, the Microsoft compiler produces only slightly higher increases than ACCEL2. In my tests I found purely graphic programs increased by as little as 1.5:1, to as much as 13:1 using the Microsoft system.

The speed-up factor using ACCEL2 varied from 4:1 to 8:1.

On the other hand, programs with much single-precision number crunching may experience only a 2:1 speed-up using ACCEL2, but can exhibit as much as a 5:1 speed-up if compiled via the Microsoft Basic compiler.

A Star Trek program I have occupies exactly 10,002 bytes in Basic. When compiled by the Microsoft compiler it occupies over 27,000 bytes.

The same program compiled by ACCEL2 takes up 16,997 bytes. ACCEL2 needs 5,632 bytes of RAM at compile time, and can use as little as 1,280 bytes at run-time. The TRS-80 ROM routines are used whenever possible.

The compiled code may be larger than the Basic code it replaces, but not nearly as large as the same program compiled by the Microsoft system. For many short programs, the Microsoft system produces at least 10,000-byte modules, whereas ACCEL2 may need only 2,000 to 3,000, including its run-time component.

Both compilers may require that the original Basic program be cleaned up before compilation. To obtain a speed improvement, your code must be logical.

What may seem perfectly logical for the TRS-80 Basic interpreter may not be logical for a compiler.

There are more restrictions on the program source code (the original Basic listing) with either compiler. ACCEL2 has a compiler option called NOARRAY that prevents compilation of all array variable references and causes all arrays to be processed

by the interpreter at run-time. There are no array restrictions if ACCEL2 is used with this option, but as a consequence, the program runs more slowly.

ACCEL2 has another option: EXPR or NOEXPR. These commands control the compilation of Let, For, POKE, Set, Reset, and If. EXPR causes compilation, NOEXPR prevents it. These can be used throughout the program to selectively turn compilation on and off. If not compiled, these statements are executed by the interpreter at run-time. This slows the operation, but if planned properly, the speed loss will be minimal and the space saved significant.

The Microsoft compiler has many compile options (called switches), but none of them do the things described for the ACCEL2 compiler. There are about 20 switches in all; if you like many possibilities, the Microsoft compiler has them.

The Microsoft offering is part of a system of advanced programs allowing many combinations of languages and functions to interact. It is not for beginners: it is complex and requires a minimum of 48K in a disk-based system. It offers higher speed-up for mathematical operations and more flexibility for the advanced programmer than ACCEL2.

ACCEL2 is easier to use, requires only 16K of memory, and offers excellent speed-up performance for logic and graphics programs. It can be used by anyone who has mastered TRSDOS, and operates on Models I and III. There is also a tape version available, called ACCEL.

ACCEL2 has another advan-

tage for those of you who like to sell or trade software you have written: No royalty payment is required if you distribute programs compiled by ACCEL2. ACCEL2 allows you to make either tape or disk copies of compiled software, along with the necessary run-time component. Southern Software requests that you put a copyright notice on the tape or disk label, but does not request a royalty.

Microsoft requires 9 percent of the end-user price per copy, or a flat \$195 per year (cheaper if you sell over \$2,166.67 worth of software a year).

I feel that each compiler offers truly improved capabilities for the TRS-80, ACCEL2 being the lower cost alternative (and best for compiling programs of logic and graphics routines). The Microsoft system is aimed at the more advanced programmer, allows one to interlace with other languages, and is best at speeding up number-crunching programs.

*(ACCEL2, by Southern Software, Eastleigh, Hants, England, is now ACCEL3 and 4 and is distributed by Allen Gelder Software, Box 11721 Main P.O., San Francisco, CA 94101, \$99.95. Basic Compiler, for the Model I only, by Microsoft, 10700 Northup Way, Bellevue, WA 98004; \$329.)*

Dennis Wilkins  
May 1982

★★★★

**BASCOM**  
**Basic Compiler for Model I**  
**Microsoft**  
**Model I**

Microsoft promised that their

compiler, BASCOM, could increase speed anywhere from 3 to 30 times over the interpreter's execution. To test this claim, I ran a statistical program that I occasionally use with the interpreter. With the interpreter, the computations were completed in 23 seconds. The compiled version ran in less than a second!

To use BASCOM, you must write a normal program for the TRS-80 Disk Basic interpreter. There are only a few unimportant incompatibilities. You also need 48K RAM with at least one disk drive; two drives, especially 40-track ones, are better.

You can only run programs of about 16K or less in length. The compiler reads source code from disk, but the linking loader loads the program into memory. The object code and the run-time system require substantially more memory than the original source code. With memory area taken up by the loader, little room is left for lengthy programs.

The only other limitation is the inability of the `USR(n)` function to pass arguments to machine-language subroutines. But you can fix that.

Included with the BASCOM documentation is a Basic-80 Reference Manual. Basic-80 costs about the same as a new disk drive, but purchasers of Microsoft's Basic Compiler get it at no extra cost.

This Basic-80 has the ability to reset the width of your screen or printed display. While . . . Wend loops, variable names with 40 significant characters, SWAP, and CALL (a fix for the `USR(n)` problem). However, you don't get the interpreter, meaning that you have to write the source

code and then compile it before you can run it.

Errors in standard Basic can be detected by running the program using the interpreter, and then correcting any detected errors. Mistakes in either Disk Basic or Basic-80 will show up, when being compiled, with error codes, the line containing the error, and an arrow pointing to the error.

Considering this compiler is the same software that is being sold for Model II and CP/M users for about twice that price, we Model I users are getting a real break! I heartily recommend BASCOM. It's like going from a cassette player to a disk drive again.

*(Microsoft, 10700 Northup Way, Bellevue, WA 98004; \$195.)*

Bill Sholar  
May 1981

★ ★ ★ 1/2

**Basic Compiler  
Microsoft  
Model II**

I bought Microsoft's Basic Compiler to make more memory available on my Model II and to make the programs execute faster. Although it will run on one drive, you need two drives to transfer the files to a system disk.

The documentation is divided into several sections. The first and largest section deals with the commands available in the Microsoft version of the Basic language. The information dealing with the Model II version of the compiler is contained in six pages appended to the front of the manual. The appended information deals mainly with the commands (instructions) that are not legal in this compiler

version.

The system's greatest deficiency is its documentation.

Microsoft has fallen into the trap of trying to make one documentation package serve for every conceivable application. A compiler is fundamentally different from an interpreter and deserves separate instructions. Five or six pages of addenda cannot begin to provide the detail required to effectively use this powerful tool.

The manual states that dashes must be substituted for slashes; :TT (for CRT display), not TTY.; and :LP for line printer. I typed the command line to compile, and the disk drive commenced to spin furiously, but only a few lines of output appeared on the CRT.

After struggling awhile, I decided that the compiler might not accept the compressed format for Basic. I loaded the Basic interpreter, read my program from the disk, and saved it in ASCII using the ".A" extension. One appendix does state that the compiler will accept ASCII files.

I returned to TRSDOS and entered the Basic compiler command. I had to eliminate several errors, including one labeled LL—line too long. I discovered that it wasn't the indicated lines that were too long but the lines following the indicated lines.

Once I figured out how to use the compiler, I decided to perform a few experiments to determine what kind of increase in processing speed I could expect. After I had a small Basic program running in the interpreter mode, I made the changes required for the compiler (such as deleting the Clear statement) and saved an ASCII version to compile. I then ran both versions

of the program and, using the internal clock in the Model II, printed the time required to perform the various functions.

As I expected, there was little difference in the screen display time (although the compiled version appeared slightly faster) or the printout. A significant difference in the string-search times was indicated.

The string-search test consisted of establishing a 2,000-item array with a five-character pattern in each item. A loop was established to search the array using the INSTR\$ function. I set the last item in the array to the pattern I was searching for. The interpreter version took 16 seconds to find the required item. The compiler version took just four seconds, a 4-to-1 advantage in speed.

I noticed I had used a single-precision variable for the loop counter, so I changed the variable to an integer-type variable in both versions of the program, recompiled, and ran the test again. The interpreter version run-time was reduced by a second or two, but the compiled version now ran in only two seconds. This would indicate that programs requiring searches of large numbers of string items would run seven or eight times faster using compiled versions.

I have since received Version 5.2 of Microsoft's Compiler.2. This version adds the include/ function that lets a programmer create subroutines and add them to a program without reentering the code (a very handy function).

Version 5.2 also eliminates the step of dumping the object code to a command file during the load process. A new compile

switch has also been added to allow compiling using one set of Basic conventions and execute using another. Considering all the improvements, the latest version is a significant step forward. (*Microsoft, 10700 Northup Way, Bellevue, WA 98004; \$400.*)

James L. Waggoner  
May 1981

★ ★ ★ 1/2

### **Microsoft Basic Compiler Microsoft Model I**

★ ★ ★

### **Radio Shack Basic Compiler Tandy/Radio Shack Model I and III**

Microsoft's Basic Compiler is almost totally compatible with Level II and Disk Basic (when running TRSDOS 2.3). You can run and debug your program using the interpreter and then compile it for additional speed. This means all the excellent debugging tools Basic offers are available, plus such utility programs as Boss to make debugging even easier. Debugging compiled Basic is difficult and there is a long delay while the program compiles again each time you change the source code.

The Radio Shack compiler can only run programs after compilation. It has some debugging features (such as tracing line numbers, checking variable values, though this is awkward, and stepping through a program line by line), but this is not as convenient as using the interpreter. Documentation of how to access these features is poor and checking variable values is awkward. The biggest problem with



this computer is that you must compile the corrected program again.

The Radio Shack compiler generates shorter object code programs than Microsoft's. Both compilers require a runtime module to be used with the program's object file. Thus the object code is not pure machine code and cannot be used on just any Z80 computer. The Radio Shack compiler loads and saves source code very, very, very slowly!

Both compilers offer sequential and random access files. Radio Shack's Compiler Basic permits adding new data to a sequential file after closing it, and lets you use variable length random access records. The Microsoft compiler must use the 256-byte random access records TRSDOS 2.3 allows. Generally, Compiler Basic offers easier file handling.

An indexed file is a Radio Shack option; it lets you tie a key to each record. You can then use the key word or number to access that record in the future. The Call statement allows you to organize the program in modules, carrying data directly to and getting it back from modules very easily. To speed up debugging, you can write a short program to test each module before you incorporate it into the main program--short programs compile much faster than long ones. You can create a library of modules in either source or object code and append or incorporate them into your programs as required.

Neither system permits using variables or expressions in a DIM statement. Only constants are allowed.

The Radio Shack compiler per-

mits long variable names. The first six characters are significant. It uses integer or double precision (called REAL in Radio Shack Compiler Basic), not single precision, variables. All the usual Basic statements are present in Compiler Basic, with a few additions.

A potentially useful statement is On Break Goto. . . . You cannot, however, use this statement in a main program as it is not recognized in any of the modules; if you use it in a module it is only recognized in that module.

Editing is a sore point in the Radio Shack compiler; once you are used to the character editing features in Microsoft Basic you will find Radio Shack editing very primitive. You can change a character or group of characters in a line, but it is all very awkward.

The Radio Shack program compiles in one step. You do not have to worry about the linking loader and subroutine library. Thus compilation is much easier, but takes about the same length of time for long programs.

The Radio Shack compiler runs on either the Model I or Model III; both versions are supplied when you purchase the program. As this is written, the Microsoft compiler runs only on the Model I, though I suspect there will be a Model III version by the time you read this.

Both compilers come with a fat loose-leaf manual. If you are fluent in Microsoft Basic and TRSDOS Disk Basic the Microsoft manual is better; the Radio Shack manual leaves some points rather obscure.

Programs created by the Microsoft compiler run much

faster when you use integer variables, and about the same speed when you use single and double-precision variables, with infrequent integer use. If you use the optional Basic-80 syntax with the Microsoft program you gain most of the special features Radio Shack's Compiler Basic offers, but lose the use of the interpreter. Debugging becomes much harder.

Please note this warning not found in Radio Shack's documentation: if you have source code in memory and compile it all goes well if there is room in memory for the object code. If you run out of memory the system freezes up; your own recourse is to reboot, losing the source code forever if you did not save it before trying to compile.

If you are fluent in Microsoft Level II Basic and TRSDOS Disk Basic then you will find the Microsoft compiler easier to use. However, Radio Shack's Compiler Basic is not much different, especially if you disregard the differences in file handling. A very nice feature of the Microsoft compiler is that most Basic programs you already have will probably compile with few or no changes. The Radio Shack compiler accepts regular Basic programs saved in ASCII (.A option), but you will still have more changing to do to get the program to run than you would using Microsoft's program. This can be difficult with Radio Shack's poor editor.

The Radio Shack compiler is \$50 cheaper. I prefer it for the kinds of programs (business, data base) I write. If I were doing games or using a lot of graphics I would prefer Microsoft's,

because of the ability to use the interpreter for debugging. (*Microsoft, 10700 Northrup Way, Bellevue, WA 98004; Model I and III, \$329. Tandy/Radio Shack, Fort Worth, TX 76102; \$149.*)

Ken Knecht  
September 1982

★ ★ ★  
**UCSD Pascal Compiler**  
**FMG Corporation**  
**Model I**

The FMG Pascal Compiler package is on three disks. The modules are arranged to allow for maximum free space on the disk, which is usually about 30K.

The Pascal Compiler gives you several modules with which to work, including the linker, compiler, execute, run, file, and edit. All the modules work well and generate nice, compact code.

The sore point of the package is the documentation. The manual was written to supplement a teacher's instruction and does not teach you Pascal. You must learn it from another source to use this compiler. (*FMG Corporation, Fort Worth, TX. The UCSD Compiler is no longer available.*)

Dennis Thurlow  
October 1981

★ ★ ★  
**ZBasic Compiler**  
**Simutek**  
**Model I**

ZBasic comes in two packages; the first contains 16K and 32K versions for tape-based systems and the second has 32K and 48K versions for disk-based systems.

On the positive side is the small size of the compiler, the awesome speed with which it compiles, the run-time speed of the programs it compiles, the interactive nature of the compiler, the fact that you can compile a 4K Basic program in 16K, and that the compiler is yours when you buy it (no royalties to pay).

ZBasic sits in a fixed block of RAM, and its ORG depends on the version you use. The 32K version resides from 8680H to 9580H. Section 9200-9580 contains the subroutine package that gets tacked onto your program and speeds compiling as well as increasing run time speed. You could conceivably move the compiler anywhere you like by using an editor/assembler.

The program compiles faster than a speeding daisy wheel. I compiled a 2.5K Basic program in less than five seconds. I'm impressed with the speed of this program.

The ZBasic manual lists run times for various commands in Basic and ZBasic. The increases in speed are from six times faster for AS=INKEY\$ to 12 times faster for Set and Reset and up to 288 times faster for jump commands such as GOTO and GOSUB. Using the Set command in a tight double loop, Level II Basic requires 50 seconds to white out the screen. In its compiled form, it takes three seconds!

ZBasic syntax is slightly different than Level II. A very nice feature of the compiler is its ability to jump back and forth between the Basic program, ZBasic, and DOS, enabling you to check for syntax problems as you tidy up your debugged Basic



program.

The compiler checks for errors during compile time and if it finds one, will return the type of error and the line number where it occurs. If you attempt to jump to a nonexistent line, you get a peculiar error message, like line error in line 67757, which may cause some consternation if the highest line number in your program is 200.

The manual lists some potential causes and corrections of errors. If you get an error message giving a line number that doesn't exist, try renumbering your program in high memory, and you will be able to locate the problem.

Another advantage of this program is that you can compile a 4K program in 16K. Note, however, that you can only compile for a 16K machine using the 16K version, as the compiled program resides in a fixed block of RAM and the larger versions won't run in 16K.

Finally, I like the fact that when I buy this program, it is mine and mine alone. I don't have to pay anyone any money (except Uncle Sam) when I sell programs I write with ZBasic, unlike Microsoft's compiler.

This is not the ideal compiler for all applications. It only handles integers, does not work with arrays, limits your variables considerably, does not support all Level II commands and starts, and compiles into fixed RAM locations.

Part of the reason the compiler works so fast is that it only handles integers. This is also why the compiled programs run so quickly. Multiple precision takes time and memory space; the compiler would have to be larger and slower to handle nonintegers. However, you do need greater than integer accuracy for lots of applications.

The compiler also doesn't handle arrays. They can be simulated by fixing a block of RAM somewhere and using PEEK and POKE to store data. This requires some thinking on the user's part and rewriting most programs, but the method works.

Normally in Level II you have many variables: AA to ZZ, where the second letter can be anything from any letter to a single digit number. ZBasic uses fixed RAM locations for its variables and limits your variables to save space. You can use 26 string variables, A\$ to Z\$, each of 31 characters. If the string is longer, you will overwrite the string above it. Numerical variables run from A-Z, A1-Z1, and A2-Z2. This is a fair number of variables, but it is inconvenient to rewrite programs changing all instances of several variables.

Some Level II commands are not supported, and the manual contains a long alphabetical list of these commands. It does give short routines to simulate SIN

and COS and various string functions. The meanings of some other Basic commands are changed slightly as well. For example, you cannot use the logic operators AND and OR in conditional If. . . Then statements, and if you use logical math operators, syntax must be closely watched.

One major fault of this program is that it fixes the RAM locations of its compiled programs. It would be nice to be able to use the 32K version to write programs for a 16K machine. In the 32K version, the program begins at 9200H (the subroutine package); 846 bytes later, the compiled version of your program is tacked on. Variable memory for the 32K version begins at BC00H. To move the program to another memory location is a lot of work, using a disassembler and an editor/assembler.

The manual does give valuable information, but I don't feel it goes far enough. An alphabetical listing of commands supported by ZBasic, an index, a more complete explanation of syntax differences, and error codes are all things the manual should include. It does show how to link compiled programs (NEWDOS only), and gives some routines to get around or simulate normal Basic functions in ZBasic, and gives a memory map for the different versions, including the various fixed RAM locations for the variables. A complete map for the subroutine package would be nice, so that you could easily link different machine-language programs together with your ZBasic program. As it is, you must use the Level II USR format even with

the disk versions.

I am impressed with several aspects of ZBasic. Its most unfortunate aspect is the fixed-memory ORGs for compiled programs. Lack of higher-precision arithmetic can be a major problem. (*ZBasic Compiler, now version 2.2, is sold by Simutek, 4897 E. Speedway Blvd., Tucson, AZ 85712; cassette \$79.95, disk \$89.95.*)

Bruce Powel Douglass  
June 1981

★ ★ ★ ★  
**ZBasic 2.0**  
**Simutek**  
**Model I and III**

ZBasic is back and is vastly improved. In a previous review I said that it had some major problems, including integer math only, non-relocatable programs, and that it did not support all variable names. Well, these and other flaws have been corrected.

ZBasic is easier to use than ever. When calling ZBasic from disk (or loading from tape), it interfaces with the keyboard device control block. The compilation speed is slightly slower than the old version, but it is still incredibly fast. Compiled program speed is fast, too. Basic took over 50 seconds to white out the screen using a tight SET loop; ZBasic whites it out in under a second.

ZBasic still lets you jump back and forth between your Basic and compiled programs, making it the only interactive compiler available for any machine, to the best of my knowledge. ZBasic also allows TRON/TROFF to trace the execution of compiled

programs.

After loading the compiler, ZBasic asks if you wish to change any parameters. If you do not, the compiler loads normally; if you do, you may choose from a variety of options.

The manual is also much improved. Simutek has added an index and a command description section. It is easier to read and contains more useful information.

ZBasic 2.0 supports high-precision math. But the compiler is available with and without the high-precision math routines. The routines use about 700 bytes of memory; if you need this space, you can simply use the version without them.

ZBasic 2.0 has several improvements over Basic, including: the Block move of memory (10,000 bytes per second); the Compare command (to find byte sequence in memory); the Invert memory command (great for graphics); the Tone generator (for external speaker); High-speed multiply and divide; Cassette I/O; Single-byte disk I/O; Enable/disable interrupts; Read stack pointer; 16-bit PEEK; and improve user-defined machine-language routine calling commands (pass four or six registers).

ZBasic is an interactive compiler, meaning your source, object (compiled) and compiler programs are all coreident within memory simultaneously. When you activate the compiler, ZBasic compiles your programs and gives you the options of returning to Basic, running the compiled program, or saving the program to disk or tape. If you set the configuration to relocate your object program elsewhere, you can't execute it, but you can

save it. If you have dreams of writing 16K machine games on your 48K computer, ZBasic may prove invaluable.

When the compiler detects an error, it returns an error message, error line number, and makes it relatively easy to remove errors quickly, and it minimizes the pain of correcting syntax differences.

Missing in the manual is a subroutine map of the run-time system.

Simutek thoughtfully included a Misosys program called CMDFILE, useful for appending two or more CMD files or System tapes into one file; offsetting a tape or disk file so it loads into an area other than where it was meant to execute; appending machine-language programs with "patched" code without reassembling the program; single-drive copy of CMD files; and creating System tapes from non-continuous blocks of memory.

ZBasic 2.0 is a powerful, interactive compiler possessing capabilities beyond those offered by any other compiler on the market. Although it generally requires rewriting of source programs to compile them with ZBasic, the compiler works incredibly quickly, as do the object programs.

Best of all, no royalties need be paid for selling compiled programs. ZBasic 2.0 has my highest recommendation.

*(Simutek, 4897 E. Speedway Blvd., Tucson, AZ 85712. ZBasic version 2.2 is now available for \$99.95 cassette, \$89.95 disk.)*

Bruce Powel Douglass  
May 1982

---

## DATA BASE MANAGERS

★ ★ ★  
**Aids II**

★ ★ ★ 1/2  
**Aids III**  
**Meta Technologies**  
**Model I and III**

Aids II, a program that writes programs, has answered my business needs. I've used Aids II to write some useful data-base management software.

I received the software along with a 36-page manual that was clearly written and concise. It contained explanations of the various functions of the Aids II system and also a sample program that gave me hands-on experience without risking valuable real data.

As the weeks went by, I created a number of programs with the Aids II systems. At that time I realized I needed more than Aids II's 12 user-defined fields.

Aids III answered my problems. It features 20 user-defined fields and includes a machine-language sort, search and delete. It also includes Maps, which produces custom reports either horizontally or vertically and a very flexible label routine.

A few weeks after I received Aids III, Calcs was released. It can be used for any type of numeric manipulation. The combination of Aids III, Maps, and Calcs satisfies my business program generation needs. What impresses me the most about this system is its flexibility. By changing only a few lines I can

write any program I desire and manipulate data quickly.

The machine-language sort can sort by any combination of five fields in ascending or descending order, or in any combination. Aids III was also compatible with files created under the Aids II.

Many data-base systems presently market at over \$100. The data bases I have created with the Aids system has outperformed them all. The Aids III system is worth every dollar for business as well as personal use. (*Meta Technologies, 26111 Brush Ave., Euclid, OH 44132. Aids III sells for \$79.95, also see the April 1983 issue of 80 Micro.*)

David E. Wareham  
April 1981

★ ★ ★ 1/2  
**Aids III with Calcs III**  
Meta Technologies

★ ★ ★ 1/2  
**CCA Data Manager**  
Personal Software

★ ★ ★ ★  
**Maxi Micro Manager**  
Exador Inc./  
Adventure International  
Model I or III

Meta Technologies' Aids III is a sequential data base; it contains all the records in the computer's memory at one time. The number of records that will fit depends upon the size of the machine and how many fields are used.

A direct access data base brings the records into a computer's memory from a disk one

at a time. The limit is not machine capacity but disk size. Personal Software's CCA Data Manager is a direct access data base in which no one file can be larger than a single disk.

Maxi Micro Manager System (MMS) from Exador Inc. is also a random data base. However, MMS can span disk drives, allowing one file to carry over to more than one disk. Thus, it has the largest capacity of the three programs.

CCA gave the best first impression. The manual is printed on heavy paper and is contained within a three-ring binder. It is able to withstand the constant use to which most manuals are subjected.

Unlike CCA, the Aid's manual assumes the user has some computer knowledge. The inexperienced user is referred to the DOS manual. The Aids manual explains how to use the program and includes many examples and flowcharts.

The MMS manual, a paper-bound, indexless booklet, is more concerned with the theory of operation than specific details. It does cover file structure quite thoroughly, as well as the Basic words created for the program.

### Initializing and Input

CCA capacity on a single-drive system is quite small—unless the program and data are kept on separate disks. I had only about 100 financial transactions on a disk, but it was impossible to sort the data due to lack of disk space. Removing the program helps considerably. Single-drive owners can only half-fill a disk if sorting is desired.

With Aids III the program is initialized by editing data lines

within the program. Errors are easy to correct; just edit the line again. Field errors are not available instantly, as in CCA and MMS, but may be obtained by using Calcs III.

In overall display effectiveness and user interaction techniques, Aids and MMS are about equal.



and are much better than CCA.

CCA screen displays are poor; however, CCA is the oldest of the three programs. The input screens scroll from one record to the next, and there is no graphics indication of field length as a guide to correct entry.

The input screens for MMS are nicely designed and functional. Dots indicate the size of each field and graphics blocks show the field being typed.

The Aids input screen, like MMS, clearly shows the length of each field, using graphics blocks.

Input is very flexible. You don't need to finish the record before correcting an error. The up and down arrows move the cursor to any field for data correction. Since Aids is a sequential data base, records are not written to disk after each entry. Therefore, adding records is a fast process.

Aids is the only program of the three that allows you to quit easily.

Aids and MMS are both slow during input. Typing must be done at moderate pace, with frequent glances at the screen to be sure all data is accepted. The flashing cursor is responsible for this slow pace.

CCA, with its more primitive style, allows faster typing but is less clear. The addition of records takes time, and certainly calls for making backup copies.

### Update and Sort

MMS and Aids assume the operator won't always know the record number when he requests an update. They provide the search function as a subcommand for any update. Aids finds records so quickly you don't need record numbers. MMS requests a number if you know it, to save time. Both programs provide nice screens for updates with field length indicators. MMS requests the number of the field to change. Aids uses arrow keys to move the cursor to the desired spot, and uses shift/up or down arrow to page forward or back to the next record in the search. MMS allows paging back or forward to the next record in the file, rather than the next record in the search.

CCA again suffers from sloppy screen display in this mode. The request for editing a record re-



quires a record number. And when the selected record is displayed, CCA prompts "Is this the record?", to which the user must reply yes or no. Simply typing Y or N defaults to menu and the search must be started over.

CCA can easily handle more than one field at a time. This is vital for some uses. For example, you may need to sort an inventory by vendor, then by item number.

CCA's sort can easily handle multiple fields. This feature alone may make it the best data base to choose. There are, however, some drawbacks. The sort is the slowest of the three programs, since it is entirely in Basic. Another is that disk space is limited. Sorting creates another full file of data, as well as a working file. For this reason, a disk must be less than full before you try a sort on a one-disk system.

The Aids program has the fastest sort since it is in machine language, and holds all the records in memory. Like CCA, Aids can sort more than one field at a time. It can't sort within a slated range, but since printouts are available on a range basis, this feature is not vital.

The MMS program can only sort one field at a time. While sorting, MMS is more efficient with disk space than CCA; it is a machine-language sort.

### **Print and Calculate**

The quick printout for Aids goes to screen if the printer is shut off. As in all Aids commands, any range of records may be selected. User-selected fields are printed across the

page. The Aids printout is the simplest of the three to operate.

In CCA, there is no on-screen help for using the print function. You must refer to the manual and type in a long list of parameters. Errors usually require you to begin over.

One complication to printing a CCA file is that a sorted file cannot be used until it is renamed, by exiting the program and using DOS. If you want a printout of a new sort, both the new and the original file must be specified. The print file can't be saved after the sorted file is renamed. CCA's printout section seems unnecessarily difficult to use.

MMS printouts are even more complicated to generate. The manual is vague at crucial points; nonetheless, if you can figure it out, the result is a sophisticated printing system that will merge data base files with ASCII files.

All three programs are able to calculate new fields based on the entries in other fields. MMS has the most sophisticated calculation abilities. In CCA, calculations are not visible at the time the record is added. In Aids, the calculations are done by a separate program and are not as easily accessible.

Choosing the right data base depends on the application. For any involving relatively few records, Aids is my preference.

For larger applications, the choice between CCA and MMS is more difficult. CCA would benefit from some modernization. But, if the operator is inexperienced and multiple sorts are needed, the ease of running printouts and its clear documentation make it a good choice.

If handshaking with a word

processor is needed for form letters, if calculations of more complexity than +, -, \*, and / are needed or if files are longer than one-disk capacity, MMS would be the choice. MMS is clearly the more sophisticated program overall.

*(Meta Technologies, 26111 Brush Ave., Euclid, OH 44132; Aids III \$79.95, Calcs IV \$39.95. For an indepth look at Aids III, see the April 1983 issue of 80 Micro. The CCA Data Manager has been replaced by VisiFile. Personal Software of Sunnyvale, CA, is now named VisiCorp. Maxi Micro Manager, now named Maxi Manager, is available from The Business Division: Adventure International, P.O. Box 3435, Longwood, FL 32750; \$149.)*

Wynne Keller  
August 1981

★ ★ ★

**The Business Mail System  
The Bottom Shelf  
Model I**

★ ★ ★

**Mail/File List  
Galactic Software  
Model I, II, or III**

★ ★ ★

**Name and Address System  
Small Business Systems  
Group  
Model I**

The Business Mail System by The Bottom Shelf (TBS) is not the easiest of these programs to use. The instructions say the program will handle a mailing

list of 150,000; this may be true, but you can put only 500 on a single disk.

The program comes with two disks—a security disk and a program disk. The security disk protects the TBS software from unauthorized copying. Before using the program, you must transfer TRSDOS 2.2 to the program disk.

You can enter up to 100 names and addresses before saving them, and when the first disk is full, the program tells you to insert the next one.

After you've entered your list, you can print mailing labels. The flexible TBS printing routine is the best part of the program. You can print up to four labels across the page and you have numerous options for selecting the names and addresses to be printed. The program has three four-digit fields and a three-character alphabetical field that you can specify. This could give you over 100 billion selection codes for your printout!

You can add more names to an existing list by using option 1 on the menu. The program expands your list and inserts the additions to the proper data disk.

If you have a large list, the program tells you which disk to write the additions to; you might have to use several disks. The program has the usual routines to correct an entry or delete an inactive one.

The Mail/File List by Galactic Software Ltd. is the most unusual of the mailing-label programs. While this program will support a maximum of only 600 names and addresses on a two-disk system, it maintains both an alphabetic and a zip-code file

under constant sort. A list entry is automatically placed into proper sequence in both files.

The program has two label-printing routines, the standard three- or four-line label, and a user-designated label. In the user-designated label you can print any of the nine fields (in any order) over three or four lines.

You can print a 30-character message as the first line on a three-line label, making it, in effect, a four-line label. You can print labels with messages such as "Attn: Store Manager" on the first line. In addition to labels, the program prints a directory in either alphabetical or zip-code order. The program also has the usual edit routines.

The Name and Address System by Small Business Systems Group (SBSG) is available in two versions. One is a normal label program, and the other has a form-letter routine. The two programs are the same except for the form-letter routine, so I'll review the second one.

This program is the only one of the mailing programs that has its own DOS and is ready to run when you receive it. SBSG included a sample data file on the disk so you can familiarize yourself with the program's functions before you enter your own data. Your data is written to the disk as an input file in the order you enter it. You must create an output file to print your labels, which is done by sorting your input file.

When you need to add names, you must create a second input file and merge it with the first. Then, sort the merged files to create a new output file. This is the major fault of the program,

since it uses too much disk space for the number of records in the file.

The label-printing routine prints up to three labels across in a five-line format. Two select fields (one alphanumeric and one alphabetic) allow you to print any part of your list.

The best feature of this program is the Form Letter routine, which you use with Electric Pencil. It lets you type individual letters to everyone on your mailing list. Use special variables for the name and address fields; the label program changes them to the appropriate entries when the routine is run. All the other label printing options function in this mode, so you can send the letter to a selected few recipients.

If you have a large list, the TBS Business Mail should be your choice. Once you've entered your data, the printing routine prints over 100 labels per minute.

The Galactic Software Mail List offers the advantages of both an alphabetical and a zip-code file ready to print, but it is slow. It prints only about 20 labels per minute. On the other hand, it is the only one that prints more than one copy of a label at a time.

Without its Form Letter routine, the SBSG Name and Address System is not the best of the three, but that routine is worth the price of the program. *(The Bottom Shelf, P.O. Box 49104, Atlanta, GA 31359. Business Mail System is no longer available. Galactic Software, 11520 N. Port Washington, Mequon, WI 53092; \$159 Model I and III, \$199 Model II. Small Business Systems Group, 6 Carlisle Road, Westford, MA*

01886; \$125 for an updated version.)

Reese Fowler  
February 1980

### **Calcs III**

(See Page 27)

### **CCA Data Manager**

(See Page 27)

★ ★ ★ ★ 1/2

### **Data Ace**

**Computer Software Design  
Inc.**

**Model I, II, or III**

The Data Ace data-base program is a "relational systems language and operating system." Understanding relational data-base structure is important if you want to grasp the power that Data Ace offers.

The secret behind powerful data bases seems to be flexibility, exactly what Data Ace's relational format offers. Unlike the popular indexed sequential-access memory systems (ISAM), your applications are separate from the actual data storage.

Once you know the kinds of information you are going to store, the rules of relational data-base structure dictate how it is organized. When you go to use the data, programs are written to give you the desired results. The existing data can be easily reorganized to meet the needs of a new program.

Data Ace is a completely integrated package. It features its own operating system, data-management software, a query language allowing you to examine files directly, a structured programming language for

manipulating data, a full-screen editor to write programs, and a catalog mode to store help files, programs, and procedures. A second disk contains a collection of utilities for tasks like back-up, data recovery, and data-base initialization.

Data Ace is self-contained and provides no means for backing up the system disk: you must take care to prevent damage for the two copies provided, although Computer Software Design offers replacements for \$30.

The power of Data Ace's DIL (Data Interrogation Language) is substantial and can be taken advantage of with a minimum of training. If you need to do something several times, you can store a string of DIL commands as a procedure using the catalog mode.

The biggest drawback is the inability to customize reports that often call for complex manipulations of numbers, not to mention special printing. In this respect, the DIL mode of Data Ace is inferior when compared to the overall power of many other data-base managers.

Data Ace is vitalized by its unique programming language, otherwise known as Data Manipulation Language (DML). Here you have total control. You can open up to 12 different relations at any one time, shuffling data back and forth, inputting data when necessary, and printing it out just the way you want it.

DML is a derivative of Polyforth (the language in which Data Ace is written). If you're used to languages like Fortran and Basic, you have a lot of learning to do.

Data Ace's catalog mode lets you add programs and pro-

cedures (made up of DIL commands), and edit, list, or delete programs and procedures already on disk. A full-screen editor is available to assist you in writing and debugging programs and procedures. The most recent release of Data Ace features word-processing capabilities.

Data Ace has the distinction of allowing programs to be run with either a compiler or an interpreter. Source code for the interpreted programs is stored on the data disk and can be quickly altered in the catalog mode and rerun.

The power and efficiency of Data Ace's Forth-like language is evident when you compile a routine. The compiled code for Data Ace is about 135K (said to be 900K of source code), leaving room for more than 400K of applications software on the system disk.

What about speed? Suffice it to say that Data Ace is fast. Speed is enhanced by storing an index in memory that shows the cylinder location for each record. It only takes one seek operation to locate a record on the disk. Efficiency is also increased by the use of notional relations, which have data that is identical to the original relation except that the order is different.

Data Ace is no longer limited to the Model II. In the spring of 1982, Computer Software Design released a version for the Models I and III. To run the program on these machines, you need Miller Microcomputer System's Forth. The cost for the Model I/III package, including MMSForth, is \$345.

The best way to learn about Data Ace is through their demonstration package. For

\$19.95 you get a one-disk system with sample relations and full DIL and DML capability. There are even 20 pages of documentation on the disk to get you started. This low price makes it easy for you to decide if Data Ace will fit your needs.

My experience with Data Ace has convinced me of its tremendous power and flexibility. It has taught me the virtues of relational data structure. I have also learned that a truly versatile data base is probably going to take a knowledge of programming to be mastered. I'm still looking for the perfect data manager, but for now, Data Ace will remain in my collection of survivors.

*(Computer Software Design Inc., Anaheim, CA 92855; \$850 Model II, \$345 Model I/III, \$1,400 Model 16.)*

Tim Daniel  
October 1982

★ ★ ★ 1/2

**The Databandler  
Miller Microcomputer  
Services  
Model I and III**

★ ★ ★

**IDM-V  
Micro Architect Inc.  
Model I and III**

★ ★ ★

**Maxi Manager A.3.1  
Adventure International  
Model I and III**

This review outlines two database managers. The Databandler and the Interactive Data Manager, and gives an update of a third, Maxi Manager, which was reviewed in August 1981.

Although The Datahandler and the Interactive Data Manager (IDM-V) are both data-base managers, at first glance a comparison between them seems unfair as one is nearly twice the price of the other.

However, because The Datahandler is written in Forth, it must be run with MMSFORTH. Therefore the total package costs \$190. (If we allot half the cost of FORTH to the data-base application, the package would cost about \$125, which is comparable in price to IDM-V.)

These two programs use different data storage approaches. The Datahandler is a sequential data base, meaning each file's size is limited by the computer's memory. All records in a file are in memory at the end of a session. The capacity of the program in a 48K machine is about 24K of data, or roughly 300 records in a typical application.

The IDM-V program uses random-access files. This means records are written to disk as they are added, and brought back from disk on request. Only one or two records are normally in memory at any given moment. The size limit is the number of records that fit on the disk, not the number that fit in memory.

All three programs deserve praise for their documentation, which is written clearly and should offer few problems to a beginner.

The Datahandler actually has two manuals. The program's screen displays and command set are based on the PIMS data base, originally published by Scelbi Inc.; hence this manual is included. The PIMS manual is outstanding for beginners

because it gives assistance in how to organize a data base. The Datahandler manual describes the changes in some of the commands, and includes sample files.

The IDM-V manual does not explain how to organize a data base, but it explains well how IDM-V functions in the simplest possible terms. A full-page troubleshooting guide at the end of the manual outlines common beginner errors.

The most obvious change in the new version of Maxi Manager is the manual. It is now in an attractively printed 7 by 9 looseleaf binder. The contents have been completely rewritten, with greater attention paid to the beginner. It is now a professional manual, both in appearance and content.

### **Initialize**

When you initialize a data-base program you prepare it for its intended application. It is here that you establish the numeric and string fields and lengths needed for your data base. Once this is set, you usually cannot change it except by starting over.

The initialization section of IDM-V is a separate program that you call from the disk just for this function. Initialization is part of the main Datahandler program.

Initializing The Datahandler is easy because you do not specify each field's length in advance. Simply state the field name and whether it is numeric or string. As provided, The Datahandler allows up to 10 fields and a record length of about 255 bytes.

IDM-V limits the length of a record to 255 bytes, and the

number of fields to 40 (20 string, 20 numeric). The contents of a key field must be unique for each record. The manual explains the drawbacks to accessing the data by key, but not the advantages, e.g., namely that it is not necessary to know the record number to access a record.

With IDM-V, when you have defined the fields, you must specify the total anticipated number of records. This permits the program to write all the records to disk as blanks and zeros; such a procedure ensures there will be no lost files later in the event of a system crash.

There are two problems with IDM-V's initialization section. The first is if you specify too large a total, the file will not fit on the disk and a disk-full error occurs, making it necessary to start initialization over. The manual explains how to calculate how large a file will fit on the disk, but it is a lot of work; the computer should calculate free disk space and size of each record, and then inform the user how many records will fit.

The second problem is if you use key access, you must specify a prime number for the total number of records. The program should calculate the prime number nearest to the total number of records planned.

### **Input and Add**

IDM-V has an easy-to-use add function. Dots on the screen show the field length. You can modify the record before it is written to disk if you have made a mistake. IDM-V allows a fast typing speed. You cannot duplicate a field from a previous record by pressing a single key

using the add function.

The Datahandler's add function does not show the length of each field because the field length can be any length you wish. It is pleasant not having to determine field length in advance and be limited by whatever you choose. There is one drawback. If the total length of all fields in one record exceeds about 255 bytes, The Datahandler generates an error message that wipes out any file in memory. A stop message is needed. In actual practice, however, most users will not encounter this problem, as 255 bytes is sufficient for most applications and is the record size limit in most data-base programs.

Unlike IDM-V, The Datahandler's add function lets you duplicate the contents of a previous record by pressing a single key. Maxi Manager also has this convenient feature.

With The Datahandler, pressing a hyphen backs up on field at a time to correct errors. Once you complete the record, you cannot change it; the program goes on to the next record. To make changes you must use the change mode.

You can enter the change (Modify) mode in IDM-V both from the menu, by selecting "Inquiry," or during the add section if you note an error. Before you can alter a record you must find the record number using the search function, unless you are using key fields. If this is the case, the record number is based on the keyfield contents, so just typing the key entry finds the record.

It wasn't feasible to test IDM-V with a large number of records.

but the manufacturer claims it can pull any record from the disk, using a key file, in 0-3 seconds, no matter how large the file.

The Modify function scrolls through the fields. To change an item, retype it, or press enter to leave it unchanged. This method is simple, and the cursor quickly jumps from field to field.

IDM-V lets you make changes during the printing phase. This global change is very powerful. The results of calculations can be written to the disk all at once. This feature requires some careful planning at the initialization stage and great care in actual use. Do not use this function without complete backup disks.

The Datahandler has complete search capabilities within the change mode, hence it is not necessary to know the record number before asking for changes. You can make an identical change in more than one record by pressing a single key. This very nice feature is similar to the one-key command to repeat a field in the add mode. Also, because the number of the field to be changed is selected in advance, the record will be drawn on screen with the cursor already in place at the proper field.

### **Search and Sort**

For The Datahandler, the most time-consuming part of the search is defining the search requirements. Once you have defined these, the record is found almost instantly. The search's main limitation is that you may search only one field at a time. Aside from that, The Datahandler covers every conceivable search need, including a

masked search, which ignores certain parts of a string.

Both Maxi Manager and The Datahandler can find a match in any type of search even if the search string does not duplicate the uppercase/lowercase configuration of the original entry.

IDM-V does not support as many search functions as The Datahandler; consequently, there are fewer questions to answer before the search begins, and the ability to press Enter for default search parameters also speeds the process. The program searches only the first three characters on a perfect match, greater-than or less-than basis.

The search function prints each record's compared field as it scans, so it is possible to see where the search goes wrong if it doesn't find the record. Searching is a slow process if you do not use a key field. You must exactly duplicate the original record's uppercase/lowercase configuration.

IDM-V has a fast, capable machine-language sort of up to four fields, but there is a large drawback: Once sorted, the records are not written to disk. The sort is only for a printout. A second drawback is that all records to be sorted must be in memory. While this is true of all data bases, Maxi Manager and CCA (August 1981) handle it automatically whereas IDM-V requires the user to split the file and bring it in a portion at a time to be sorted.

The sort for The Datahandler is very fast because it is an in-memory system, and because of the capabilities of Forth. The speed is comparable to Aids (August 1981) and any number of fields can be sorted—ascend-



ing or descending.

Maxi Manager can sort on three fields at a time. The sort is in machine code and as fast as can be expected for a random-access data base. Unlike IDM-V, Maxi Manager stores the results of the sort on disk.

### Printouts

Usually, the most important purpose of a data base is printing data in the needed format. Since the whole process can be very troublesome, it is desirable to be able to save print commands onto the disk. IDM-V has this capability with room to store 10 different printer command files. The Datahandler cannot store print commands.

IDM-V provides less control over the printout format than The Datahandler; however, it is fairly simple to format a printout that would be adequate for many uses. IDM-V always prints record numbers whether you want them or not. Worse still, the column for record numbers wastes 10 spaces of the total printout.

You cannot align numeric fields to the right and left of the decimal point with IDM-V. It prints the name given to the file that stores the print commands as the title, so be sure to use a meaningful file name. If the length of the printer line is exceeded, the excess wraps around. Unfortunately, there is no control over the alignment of the second line of data or the number of blanks between fields. You cannot abort printout except with the Break key.

You can access the IDM-V data base with Micro Architect's word processor, WORD-V, for use with form letters. The Lister com-

mand provides a quick and easy printout.

The IDM-V report section search is much more sophisticated than the search in the main program. You can search up to four fields with a choice of logical AND or OR.

Like IDM-V, The Datahandler report formatting section is fairly easy to use. The features available, though, are quite different. You can choose whether or not to print record numbers. There is no wraparound, so it is possible to control the appearance of the second data line. Unfortunately, no math is available for a printout, but full search functions are provided.

The Datahandler has a quick, easy printout via the "standard" mode of the Report command. If you desire a customized format, you can specify which fields are to be printed, and full search functions are available for selecting records. You can change file parameters if desired. Very few commands provide a surprisingly flexible number of printout choices; you can manipulate the maximum field length, the spaces between fields, and the size of the line to obtain the desired result. The program also remembers your last requested printer layout.

You can also obtain continuous printer interaction with the program so everything you type goes to the printer. Unfortunately, this function causes the line printer to form feed whenever the screen is cleared. This severely limits the option's usefulness.

The Datahandler has an inventive mailing list printout option that examines the name field. If it sees a comma, it assumes the

last name is first and it reverses it for the printout, eliminating the comma and printing one space between last and first names. This space insertion means none is needed when you enter the name, thereby saving one byte per name.

The newest version of Maxi Manager includes a utility program to create printer command files. You don't have to learn all the print-format commands, as the utility program creates a file of the proper commands based on your answers to various questions.

At first, you could not use Scripsit with the Model III version of Maxi Manager, since Scripsit is on a TRSDOS disk and Maxi for the Model III is supplied on DOSPLUS. Fortunately, a patch is now available from Adventure International to overcome this problem. The patch will be included on all future Maxi Manager disks.

### **Calculate**

Perhaps The Datahandler's greatest flaw is the lack of many math functions. Addition is the only feature, and it is not available on printouts. The addition can be performed within a record as well as between records, and like all Forth commands is quite rapid. IDM-V, on the other hand, has standard addition, subtraction, multiplication, and division functions that are also available on Maxi Manager. IDM-V math is only available with a printout, but Maxi Manager provides mathematical functions on screen as well as in printouts.

### **Special Features**

The Datahandler's special

features stem from the fact that it is written in Forth rather than Basic. Actually, The Datahandler is not a program so much as a new set of Forth words, defined to perform various functions. As such, a Forth programmer can easily modify it to include any new functions that might be needed.

When using The Datahandler, you are in the command mode much of the time, rather than in the program. To make a choice, type a command such as Add, Change, or List. There is no menu, but the word Help lists all the commands the program understands. A directory is always available because of being in Command mode, and it is possible to stop using The Datahandler temporarily, if desired, to ask Forth to do a few calculations and then go right back into the program.

IDM-V has made every effort to speed interaction between program and user by the liberal use of default values for any choice. In formatting a printout, for example, pressing enter quickly bypasses the mathematics questions. For every question in any section, pressing enter automatically selects the most commonly used choices. It is a great feature for beginners who aren't sure how to answer many questions.

IDM-V is advertised as bug free and reliable. It has been released for the Model I, II, and III, CP/M, and Heath computers. While I haven't used it long enough to categorically state that it has no errors, it does appear to be a solid, dependable program. Many precautions have been taken to protect files from accidental erasure or damage.

A new feature in Maxi Manager is a utility program that lets it read Aids III files. An Aids user who has outgrown that in-memory system can convert to Maxi's random access without retyping his files.

### Conclusion

There is no best program. You must choose the one best suited to a particular application. Maxi Manager is tops for sophistication and file capacity, but it is the most complicated to use. IDM-V has powerful global change and delete, fast sort, and good search with printouts. Its greatest flaws are the lack of full control of printout appearance, the need to break files up for sorting, and the inability to save records in sorted order. IDM-V is easy to use, but is expensive.

Some programs are better than others for the single-drive owner. On a double-density drive, any of the programs would be adequate since the program and data could all be on one disk. On a single-density drive, there is often insufficient room for both program and data. Having a separate data disk allows for a larger file, but then the program and data disks must be switched. Maxi Manager requires this switching whenever sorting or printouts are desired, but does prompt for the switch. IDM-V also does not prompt for the switch, but requires fewer switches than Maxi Manager.

For an in-memory system, The Datahandler is pleasant to use because of its great speed in calling files in and out and for adding and changing records (faster typing is allowed, and the cursor

is at the correct field automatically for changes).

Despite the fact that The Datahandler is extremely weak in some important areas, I am fascinated by it. After a few years at a computer keyboard, one begins to feel half a life has been spent waiting for the computer to load and save files. Forth and The Datahandler end all that. I recommend the program for anyone willing to take the time to learn Forth or willing to pay to have The Datahandler customized as the program can be modified to do most of the things a good data base needs to do. (*The Datahandler*, Miller Microcomputer Services, 61 Lake Shore Road, Natick, MA 01760; \$59.95. *IDM-V*, Micro Architect, Inc., 96 Dothan St., Arlington, MA 02174; \$129. *Maxi Manager*, version A. 3.1., Adventure International, P.O. Box 3435, Longwood, FL 32750; \$149.99.)

Wynne Keller  
June/July 1982

★ ★ ★ ★

### **Date-O-Base Calendar Custom Software Engineering Color Computer**

Date-O-Base is an inexpensive program that turns the Color Computer into an electronic datebook able to search and display short memos previously stored for a particular date. It also includes a calendar-page display mode that takes advantage of the Color Computer's high-resolution graphics capabilities.

A disk can hold more than 4,000 one-line memos of up to 28 characters each. There are

restrictions, though: no more than 12 memos for any single day, no more than 300 for any month. This should be more than adequate for any household or small office. The program even allows multiple lines for complicated messages.

When Date-O-Base is run, a master menu containing six options presents itself. This is home base; you can get back to it by hitting enter once or twice. The options available to you include: Normal exit from DOB; Display Month, which prompts the month and year and then draws the appropriate calendar page; Display/Change Memos, which permits you to specify a date and jump right to the text screen; Search Memos, which display or print memos falling within a specified period of time; Delete Memos, a global delete function; and Data Computation, which computes the elapsed time between two specified dates.

Date-O-Base fills a genuine need at a reasonable price. If you can live with the short memos imposed by the program, it can

do a good job for you. (*Custom Software Engineering, 807 Minutemen Causeway, Cocoa Beach, FL 32931; \$19.95 disk, \$16.95 cassette.*)

Scott L. Norman  
December 1982

★ ★ ★ ★  
**dBASE II**  
**Ashton-Tate**  
**Model II**

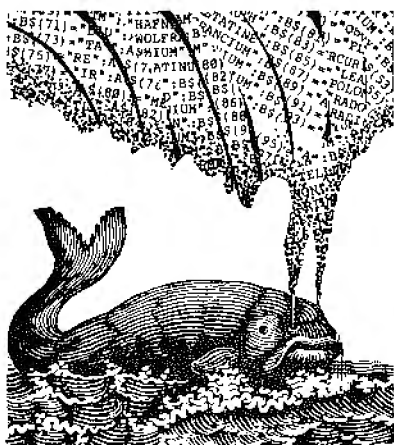
dBASE II, written under CP/M to reach the widest possible market, is a sophisticated database manager.

It provides two entirely separate modes of operation. An interactive mode, similar to other database managers, allows data manipulation in discrete steps under end-user control.

It is the command mode, however, that makes dBASE II unique. Like a high-level language, it allows complex sequence functions to be programmed in a Pascal-like structured language.

Many complex routines, that would require numerous Basic statements to accomplish, can be called using dBASE II with a single command: Create creates an entire file structure; Sort sorts a file; Total sums portions of a file; and so on. Through the use of a wide range of qualifiers to the commands, diverse and very sophisticated programs can be written.

The two modes of operation function in a relational management environment, meaning data elements are represented internally in a two-dimensional table. Without the internal complexity of the usual DBM hierarchical management environment



(where linked lists and pointers maintain data relationships), indexing speed and efficiency are not affected by file length.

Impressively quick operations are a result of the relational system used in the program. An indexed record can be located in a 100 or 65,000-record file in the same time—two seconds. Each file can handle up to 65,535 records of 1,000 characters each in 32 fields. The only limit to the number of files contained within the system is imposed by the available storage capacity.

Non-dBASE II programs and files can be used directly under the command mode control, allowing the incorporation of the dBASE power within existing routines and data files.

Included in the newly released updated version is a respectable text editor (sorely lacking in the previous version) and several file-handling enhancements including a fascinating "window-like" Browse command.

Problems? The system suffers from inadequate documentation of its abilities only partially satisfied by the newly updated manual. Be ready to spend hours learning the system's hidden abilities by trial and error.

I was also disappointed in the interactive mode report writer. To generate truly customized report formats, plan on writing the formats yourself using the command mode.

One more sore point—a file cannot be split between drives.

Overall, this is a very powerful, fast, and sophisticated database manager. I found the efficiency of file organization, speed of execution, and the command mode of operation most impressive. In the most rigorous

and complex string manipulation operations, where I have found most serious DBM problems, data integrity was never compromised under dBASE II.

dBASE II will let the novice create useful database applications almost immediately. The competent user can expect, with trial and error, to master the command mode functions. This system is capable of fulfilling almost any database requirement conceivable in the micro environment.

*(Ashton-Tate, 10150 W. Jefferson Blvd., Culver City, CA 90230; Model II \$700.)*

Craig Hilton  
June/July 1982

### ★ **Disk Mailing List Tandy/Radio Shack Model I**

After installing Radio Shack's disk mailing list program, I found many errors. The problems are not only in the cosmetics of the program, but go deeper into its basic organization.

The program menu displays five options, each a module: add names, edit the list, list names to the screen, print mailing labels, and recover space. The space recovery is a utility module run irregularly. Records that are deleted in the editing module are only flagged and the physical deletion does not take place until recovery is run. After either adding records or recovering space, a sort module is run producing an index of the records in their correct alphabetical order.

The edit module lets you

search through the list of names and addresses one by one; the records are examined in order and you can change the record in several ways. However, the records are always rewritten to disk, whether or not you have changed any information. Since the reliability of mini-floppy disks is in question, rewriting good records seems to be asking for trouble.

The recover-space module has a similar problem. The disk is overwritten in the same order in which the good records reside with the flagged deletions deleted. Rewriting all records beyond the first deletion not only requires confidence in the disks, but it also takes longer.

The most serious problem is in the sort routine and its use. Consider sorting a mailing-list file. You can't assume that the records are already sorted. Some segments might be sorted, but the file itself won't be.

The second consideration is the volatility of the file. How long can you expect the index file to be reasonably accurate? Should you re-sort the records from scratch each time, or merely update the existing index?

The Radio Shack program uses an insertion sort to order the records. An insertion sort has a  $O(N$  to the power of 2) worst-case running time. This is the same as a bubble-sort and essentially the worst possible time of any of the standard sort methods.

Only if the file is largely sorted and few additions are made is the insertion sort effective. The current sort index is ignored in resorting, so that the heavy initial cost is paid each time the file is sorted.

The sort routine not only has the potential for making  $O(N$  to the power of 2) comparisons, but  $O(N$  to the power of 2) disk accesses. All comparisons are made between one record currently in memory and one record brought in off disk.

Performance is poor. I put 210 records in the file, and sorting took nearly an hour. The disk holds about 990 records, but I shudder to think how long it would take to sort them all.

The add-names module does not refuse new records when a fixed number has been entered but when the disk-full error message is trapped by the program. This does not guarantee that there will be space for the index file. If you add several records when the disk-full error is trapped, the index file requires more sectors than it currently does and the extra space simply won't exist.

*(Tandy/Radio Shack, Fort Worth, TX 76102; \$19.95.)*

Duncan Buell  
February 1980

### ★ ★ ★

#### **File Management System Exatron Corp. Model I**

If you have a cassette-based TRS-80, you know how slowly any data-base program saves and loads records on cassette. A floppy disk solves this problem, but you need an expansion interface, more memory, and at least one disk drive—about \$900. The Exatron Stringy Floppy (ESF) costs \$250 and saves and loads programs at 7200 baud, 14 times faster than a cassette. A special input/output program al-

lows you to read and write data to the ESF tape wafer at speeds that make data base management programs practical.

The FMS (File Management System) is a Basic data-base program that offers far more flexibility than any cassette data base I've seen. It allows you to use up to nine fields for each record; each field can be either string, integer, or single-precision, with prompting labels for each. Records can be added, changed, or deleted. A sorting subroutine lets you sort the entire record alphanumerically on any field.

You can find selected records in the file equal to, less than, or equal to or greater than any one of the fields. You can right justify numeric fields (single precision can have dollar signs and two decimal places, or zero to seven decimal places) and total fields for all or part of the file. A map function lets you establish the display format, even hiding selected fields from the display or printer. You can even leave the program to Basic and return without losing the file data (unless you edit the program itself).

Records are entered from the keyboard, then saved on an ESF wafer. Entering the file from a previously made wafer is very fast compared to cassette load; I was able to load a file of 45 names, addresses, and zip codes at almost three records per second.

The 30-page FMS manual is very well written and specific. Like any data-base program, it takes some reading and practice—but then it's a breeze! The program is menu driven and easy to use. The break or enter keys almost always return to the

menu if you get confused, and I found the program virtually crashproof.

I found only one negative aspect about this program, and that's more a result of memory limitation than the program itself. FMS, despite multiline statements, performs so many functions it takes 10K of programming including a required data I/O program. This doesn't leave much room in memory for string space and arrays used for records. In my 16K machine, the name-and-address program would hold only 50 records.

The FMS is an outstanding data-base program. The prompting and error trapping routines are extensive and user-oriented. It's easy to learn to use, and the flexibility and versatility are limited primarily by your imagination. I consider it an outstanding buy.

*(Exatron Corp., Sunnyvale, CA 94086. FMS is now available in two versions; \$19.95 and \$80.)*

Fred Blechman  
March 1981

★ ★ ★ 1/2  
**IDM-IV**

★ ★ ★ ★  
**Word-IV**  
**Micro Architect Inc.**  
**Model I**

Word-IV is a disk-based, Basic word processor. Text and format control information are entered exactly like a Disk-Basic program.

Features of the system are: page length, page width, beginning point of page, automatic page numbering, page centering,

left and right justification, line spacing, title generation for each page, special character generation, and space reservation on succeeding pages.

Word-IV fully supports the upper- and lowercase capabilities of your printer. To produce an uppercase letter, you must press the up arrow preceding the letter. The text and format files are stored as a separate file on the disk using an ASCII extension. The length of the text is limited only by memory and the disk storage capacity. Editing is very easy using the Edit command from DOS Basic. A unique feature of this program is the ability to run more than one of the same program or a combination of a couple of programs.

There are several problems with Word-IV. Basic is very slow and you cannot load a text program saved in ASCII back into DOS Basic for further editing. This means that if you anticipate further editing you must save the text program also in Basic. The program also has difficulties generating a question-mark symbol. These problems are minor and correctable.

IDM-IV is a disk-based, Basic data-base manager. It consists of three separate programs: Initialization, Data Base Manager, and Report.

Initialization requires that you name the string and numeric fields (up to 10 numeric and 10 string), determine the number and size of the fields, choose the key field, specify the number of records, and name the file. This information is then formatted to disk.

String fields can contain up to 255 bytes, while numeric fields are limited to four characters.

The total record is limited to 255 bytes. Record totals are limited only by your disk storage and memory capacity.

The Data-Base Manager program is next loaded into Disk Basic. By using the Add facility, you now enter information. Once data entry has been completed, the report writer segment of the Data Base Manager program is run. Up to 10 different reports are possible.

Report is the last program to be entered into the computer. It has two options: data-base listing, which prints the entire data base in file-card format, and format report, which lets you select one of the reports that you created with the report writer.

IDM-IV has two problems in common with Word-IV: too much disk I/O, and the characteristic slowness of Basic. In spite of these two problems, it is a fantastic word processor. *(Revised versions of Word-IV and IDM-IV are available from the Micro Architect Inc., 96 Dothan St., Arlington, MA 02174; \$49 and \$69 respectively.)*

Don DeJarnette  
November 1980

**IDM-V**  
(See Page 33)

★ ★  
**Information Storage and Retrieval (ISAR)  
The Alternate Source  
Model I**

The basic ISAR program is made up of 10 modules for the Model I that consist of the following: a driver or menu module, create a file, add



records, change or delete records, sort, screen scan or search, and format hardcopy reports.

ISAR does have some problems. It can't add, delete, or modify entries during the same pass through a file. ISAR's sort is frustrating because it won't do a multiple field sort, and its print-out formats entries but not pages. It has yet another short-coming—it has an in-memory sort. A file with many records will overwhelm memory if the field being sorted is longer than 18 characters. Although the sort is fast, a slower disk sort would be more flexible.

For the occasional user who won't do a lot of file manipulation, ISAR will be very useful, at a good price and, to be fair, ISAR is not advertised as a solution to business or bibliographic problems. It was "originally designed to provide personal users with a low cost data management system." As critical as I am, it has proven valuable, but for large and potentially complicated files, ISAR is too difficult. In this case, my advice would be: Spend more money for a more flexible program.

*(The Alternate Source, 704 N. Pennsylvania Ave., Lansing, MI 48906; version 2.0 is now available for the Model I and III for \$39.95.)*

Robert L. Zeppa  
December 1980

★ ★ ★ 1/2

**ISAM**

**Johnson Associates  
Model I or II**

ISAM (Index Sequential Access

Method) is a set of Basic subroutines and utilities that allows the programmer to create and maintain direct-access files using keys, rather than the hardware-dependent numbers used by Radio Shack. ISAM requires 32K; the routines that must be merged into the application require about 4.5K. There are also minor restrictions on variable names used by ISAM and the line numbers used by the routines. In return for these restrictions, ISAM pays off with greatly increased disk space efficiency.

Johnson has included a demonstration program—a mailing list system that is more than adequate for my personal use and, I suspect, for some small businesses. ISAM does everything it promises very well. *(ISAM is sold by Johnson Associates, Redding, CA, \$50.)*

William L. Colsher  
September 1980

**Mail/File List**  
(See Page 30)

★ ★ ★ 1/2  
**Maillist  
DAR Sales  
Model I**

According to Richard Alva of DAR Sales, with Maillist software and his business tutelage, you will be able to establish and maintain mailing lists for local businesses and organizations. The result: instant money for you and your computer. Sound too good to be true? Maybe it is, since the success is in the sell-

ing, and Alva leaves that to you.

The Maillist package comes with software, documentation, and 100 promotional letters. Alva's intention is that these will be used to drum up interest in the service. He even includes an instruction manual and suggested price list detailing how to set up your business.

You say you haven't got a printer? Well, don't worry. Alva has thought of that too. Just copy your data tapes or disks and send them to DAR. For a nominal fee (3 cents per label), Alva will do the printing for you. DAR will also handle sorting, again for a nominal fee.

The program cassette supplied has two program dumps. The first is a 32K disk-based Maillist; the second is a 16K tape-based version. The disk version requires only one drive; most of the additional memory required is taken up by DOS instructions.

The program creates three files on a disk. Each file is capable of holding up to 125 names and addresses, and entries are coded to the disks. Each disk can hold 375 entries. The manual provides detailed instructions for opening, changing, and deleting file entries; they seem clear enough for the most novice of operators.

If you think you have a knack for selling and want to get a sideline going, the Maillist package can give you a start. It comes with a 30-day money-back guarantee, and a promise from Richard Alva that you will earn at least \$10 an hour for your trouble. If he can sell you, maybe you can sell others. (*DAR Sales, Sacramento, CA 95812; 16K cassette or 32K disk, originally \$39.95. This*

*company could not be reached for updated information.*)

Chris Brown  
October 1980

**Maxi Manager A-3.1**  
(See Page 33)

**Maxi Micro Manager**  
(See Page 27)

**Name and Address System**  
(See Page 30)

★ ★ ★  
**Newtrieve**  
**Unique Printing and**  
**Stationery Co. Inc.**  
**Model I or III**

Newtrieve is advertised as "the programmer's program." It is an in-memory, recursive, sequential-search program of an array of 500 by 40 characters. It is meant to be used by programmers in data-base-management systems for in-memory data structures. It performs very fast sequential searches of your data by key words. In fact, a 25,000-character array of data can be sequentially read in about one second.

The program disk contains a demonstration program called NT/DEM, a machine-language driver (which is the main part of the package), and two empty data bases of 500 entries of 40 characters each. A routine called ENHAN/BAS is provided and can be merged with NT/DEM to give additional commands.

The manual provides enough information to operate the program. Some examples for using the program NT/DEM are given. Since this program is directed to programmers, it is interesting that only a single license is pur-

chased. It appears, therefore, that Newtrieve is meant only for in-house applications, and not for programmers to develop marketable products.

Newtrieve is composed of two parts: an index and a Basic program that references the index via *USR* calls. The index must all reside in memory at once, and it allows 500 records with entries up to 40 characters. The Basic program is easy to use and reasonably powerful. It is also well-documented, so you can develop your own application programs.

You also get a merge module with Newtrieve that provides two additional commands. The first command is used if a search does not find an entry, and provides you with the opportunity to enter it into the data base. The second command lets you include data into the search string to execute a more specific search.

Limitations include the lack of a sort for printout, and the lack of specific information on interfacing the indexes with preexisting data bases, for both input and output. It would also be nice to have a program to generate the custom data bases along with Newtrieve, rather than be stuck with the 500 records, 40 characters long, or one of the optional ones that you can buy from Unique.

The searches are fast and allow simple Boolean logical operators (*AND* and *NOT*) to be used. The program is easy to use and well-documented. I don't recommend it for a general-purpose data-base referencing system for nonprogrammers, but a programmer should be able to use Newtrieve to his ad-

vantage in designing a data-base referencing system.

*(Unique Printing and Stationery Co., Inc., 11 Maiden Lane, New York, NY 10038; \$49.95.)*

Bruce Powel Douglass

October 1982

★ ★ ★ 1/2

**Profile II**

★ ★ ★ ★

**Profile Plus**

**Tandy/Radio Shack**

**Model II disk**

A data base manager (DBM) is a program that allows the user to easily manipulate a collection of data. We tell the DBM what type of data we want manipulated and in what form, and the DBM tells the computer how to go about doing it.

The ability to develop ideas into usable programs depends on the sophistication of the DBM and the user. The simplest DBM may function as a cross-reference; more advanced ones can integrate a filing operation with mathematical functions, word processing, and sub-program integration.

Profile II works more as a reliable filing cabinet with cross-references. It has limited mathematical capabilities (addition and subtraction); fields cannot be altered once created, and one file cannot access another. You cannot chain together parts of a block to create one record. There are no high-speed index capabilities, non-DBM programs cannot be run in conjunction with DBM programs, and the system will not support single-sheet feed for word processor printers. The first-time user should purchase the auxiliary

manual with cassette tape instructions along with the program. Serious first-time users should consider the extra \$220 for the Profile Plus additions.

Profile Plus adds some of the features needed to move Profile II into the true DBM class. It can do the things mentioned above that Profile II can't, has 15 data field types to II's six, and supports 2,400 records of 500 characters each versus II's 1,800. The system is well priced; Profile Plus may be the least expensive DBM in its class. The system works quickly. With better documentation, a layered sort capability, subtotal options, and tie-in capabilities to non-DBM file data, it could be a first-class system.

*(Tandy/Radio Shack, Fort Worth, TX: Profile II sells for \$179 and Profile Plus sells for \$299.)*

Craig Hilton  
April 1982

★ ★ ★ 1/2

**Profile III+**  
**The small Computer**  
**Company**  
**Tandy/Radio Shack**  
**Model III**

Profile III+ data base comes on two disks: the creation disk, which initializes the data base, and a run-time disk. If the creation disk is damaged it can usually be restored from a back-up, but the back-up disk itself is unusable. Run-time disks are not protected.

The maximum number of characters per record is 1,020; up to 255 of these can be in the key segment. The key segment

includes all the fields for sorting or searching. All other fields go to different segments. Total capacity depends on the number of drives available.

Segments must be carefully allotted to the various drives to fill them uniformly. If any one drive fills, the empty space in the rest becomes unusable. Total four-drive capacity with 255-character records is 2,000. For some applications, this might not be enough.

Unfortunately, the program is only compatible with TRSDOS, and cannot be used on hard-disk drives.

Sort is in memory, and the capacity depends on the length of the field chosen. If the records don't fit, select a portion to sort at one time. Although this is annoying, what the sort lacks in capacity it gains in speed.

Profile III+ allows math formulas using add, subtract, multiply, or divide. Up to 16 formulas can be written, using up to 20 fields per formula. Used properly, formula fields are very powerful.

The manual assumes the reader is a business user who is approaching data-base software for the first time. Worksheets are provided to help the user determine what fields are needed, and includes typical field lengths. The manual, although explicit in some areas, is vague in others. It includes a helpful glossary.

The most striking features of the program are speed and flexibility. Program sections load rapidly, and all programs are in machine code. You can design as many as five different screens for adding and viewing records.

Add, update, and search are all available from the same

master program, so there's no delay. The program appears forgiving of user errors. The break key doesn't cause lost records.

A potentially serious flaw in the program is the search on numeric fields. Numeric entries are right justified after they are typed. If the field is 10 characters long and you type five, the five characters move over when you press enter. If the search data is only five digits, the record is not found. Type five blank spaces ahead of the number for a match to occur.

Designing report formats is easier with Profile III+ than any other data base I've tested. The program allows you to visualize the report on the screen. A grid across the top marks print positions up to 132 characters. One part of the screen is for column headings, and the part directly below is for the field number to go with the heading. Up to five different report formats are allowed per file.

Profile III+ has features designed to simplify use or increase flexibility, including connections to SuperScripts and VisiCalc. You can also customize menus. Mass operations are very powerful; they include mass recalculate, delete, printout, and purge.

This program has more features than the best I have previously reviewed, and also offers greater speed and less aggravation. The convenience, of course, has a higher price tag, but convenience is not all you're buying. There are some new features not previously available in Model I or III data-base software.  
*(The Small Computer Company,  
230 West 41st St., Suite 1203,*

*New York, NY 10036; \$300.  
Tandy/Radio Shack, Fort  
Worth, TX 76102; cat.  
#26-1592, \$199.)*

Wynne Keller  
October 1982

### ★ ★ ★ **Radex-10** **IJG** **Model I**

Radex-10 (Random Access Data Executive) creates and maintains random access files easily and can generate reports with parameters that you create. Thirty-one searches are made per report, with each variable and its relationship to the others infinitely selectable.

Any number of fields can be output to the report and printed in any order. Fields can be used more than once, and formatted either vertically or horizontally. Mailing labels are automatically printed using the first five fields.

The six control programs are stored in Basic in Drive zero, along with your DOS. All files are stored in Drive 1, requiring only a formatted disk with no tracks locked out. When this disk fills up, you are prompted to insert another and continue. This disk-spanning method enables up to 10,199 records of 155 bytes each to be stored and accessed.

To begin, the package asks you to choose a function from its menu: 1) create data base, 2) file maintenance, 3) create reports, 4) run reports, 5) print file parameters, or 6) end program. When you create a report, it stays on file for future use. All you need do when using it is provide a new set of parameters to suit your current needs.

Radex-10's documentation is simple, well written, and very easy to understand. It includes six pages of definitions, which I found more enlightening than Radio Shack's TRSDOS manual. There is also a section on file structure and data bases in general, and even a list of suggested reading. The manual is a real treat for those who are not too technically minded.

Similarly, the program instructions mercifully avoid the use of abbreviations. Sometimes it even tells you why it is doing a particular thing, which is quite a departure from the "push the X button now" syndrome. (LJC, 1953 W. 11th St., Upland, CA 91786. The original version of Radex-10 sold for \$99; it is no longer available. A revised version of the book and software is available.)

Dave Orozco  
July 1980

★ ★ ★ 1/2

### **Scriinput ACR Consultants Model I and III**

Imagine being able to sit down at your computer and enter data by simply filling in the blanks. Imagine that once filled in, you could edit that data until it was perfect. What if you could also define the field lengths, specify what type of data each field accepts, and could control a high-speed, nondestructive cursor during data entry? With data input like that you could let anyone fill in data and get perfect results every time. Scriinput does all that and more.

Other programs may use a similar input routine but they do

not measure up. On many, you can out-type the cursor; others' fields are inaccessible for alteration once completed, or tied to a sort routine that makes editing slow. With the Scriinput utility, you can set up your CRT just like a data input form, store formatted data directly on the screen, and then write the entire screen onto disk or tape, send it to a printer, and manipulate or use it in calculations.

Although Scriinput is written in Assembly language (source code is supplied with the documentation), the user enters it in a Basic program. This makes it easily usable by the non-machine-language programmer.

Simple PRINT statements build a video display, and minor instructions to Scriinput inject the cursor into the first of up to 80 logical fields on the screen. You can enter data into each field with complete cursor control. When the data entry for the entire display is complete, Enter or another control key returns the program to Basic. From this point on, the Basic program handles the data as if it had been entered with an Input statement.

When the CRT is fielded the cursor can appear only within the data fields; shift right arrow skips tab to the next logical field, shift left arrow to the previous one. There is complete wrap for the skip-tab feature. Under non-shift conditions the arrows move the cursor either one space left, right, up, or down. They do not destroy the data they pass over. The space bar spaces and removes any character in its path.

During field specification you can tell Scriinput what type of data to accept in a given field (only numbers in a zip-code

field, for example). This eliminates many common data-entry errors. You can search for punctuation, uppercase alpha, lowercase alpha, and numerical entry in any combination. The program ignores all keystrokes that are not specified as acceptable.

While Enter returns control to the Basic program, you can use other control keys—Break breaks, Clear clears all data in the fields without emptying the screen—or use shift down arrow and any letter A-Z held at the same time as a control. Control P, for example, turns on the printer for data output from the display.

Two sample programs show Scriinput at work, and two utility programs aid data storage and recovery. A complete source code with comments and flowcharts is provided.

While thorough, the documentation does not lead you by the hand through Scriinput's operation; some unclear points may cause trouble for inexperienced Basic programmers. A number for technical assistance is supplied; that should be enough to help anyone get on the right road to proper Scriinput use. *(Electronic Display Technology, 3200 Polaris, Suite 3, Las Vegas, NV 89102; As of December 1982, an updated version of Scriinput has been on the TRS-80 market. The package sells for \$49.95.)*

Richard C. McGarvey  
April 1982

★ ★ ★ 1/2

## Sort-II Northeast Microwave Model I

Sort-II provides sophisticated sorting capability. It has excellent documentation, and it sorts alphanumeric as well as numeric data. Sort-II looks at data as a series of multifield records. It handles and sorts records containing up to 20 fields and can sort key on any five of them.

This program is great for maintaining a mailing list that can be sorted by name, phone number, or zip code. It can also sort such items as your phonograph collection by name, composer, type of music, performer, or location.

*(Northeast Microwave, P.O. Box 6153, Syracuse, NY 13217. Sort-II is no longer available.)*

Rod Hallen  
February 1980

★ ★ ★ ★

## Special Delivery Software Etc. Model I

Among professional mailing-list processors, Special Delivery represents the current state of the art. If this package had reached the market a year ago, it could have become the Electronic Pencil of mailing lists—and still may do so.

As well as three programs on disk, three files are supplied that allow a newcomer to experiment before lurching headlong into his own lists. This is a thoughtful idea, because the documentation, though well written, is so exten-



sive that it becomes too much to digest.

All three programs are written in machine code and leave any equivalent Basic program miles behind. If you have the Basic mailing list program from Radio Shack (TM., bow, scrape, face East, etc.), but want to change to Special Delivery, your current files can be quickly and painlessly converted by Convert.

The main part of the package, Mailform, lets you create and edit mail list files flexibly. After checking to see if you have a lowercase video chip enabled (if not, everything typed will be taken as uppercase regardless of whether you use shifted characters), it displays a form that makes it clear which field goes where and how many characters each allows.

The keyboard routine has been well thought out. A flashing cursor lets you know where you are, an optional beeper wired to the cassette jack warns that you've reached the end of a field. Shift/down-arrow is used as the control key, so no keyboard modifications are required.

The whole concept of Mailform is to do all of its work in memory, rather than using time-consuming disk I/O. This allows extremely fast searches and sorts. When you have completed a file (up to 316 records), that's when you save it to disk, or re-sort using another key and save it to a different file.

Even if you hit the reset button after typing in 315 records, all is not lost. Just type MAILFORM\*. The appended asterisk will see to it that memory is not initialized and your records are safe.

The Mailrite program creates personalized letters, labels, and

envelopes from Mailform address files. Flags such as L for last name are replaced with fields from your address file; a letter file containing only flags yields a printout of fields from Mailform. Eight flags allow printer control from within the text, though pressing Break instead of Enter to abort a printout results in a spectacular crash.

At a price of \$125 this package is not cheap. But any businessman using mail lists (and that doesn't leave out many!) would be hard put to find a better one. (*Special Delivery is now being marketed by Software Concepts, 105 Preston Valley Shopping Center, Dallas, TX 75230. It is available for the Model I and III with 32K and one disk drive for \$125 or for the Model II with Scriptsit 2.0 for \$199. An updated version for the Model I and III, XTRA Special Delivery, is now available for \$199.*)

Jake Commander  
July 1980

## — DISK AND CASSETTE — OPERATING SYSTEMS

### DOSPLUS

(See Page 279)

★ ★ ★ ★

### DOSPLUS 3.3

**Micro-Systems Software Inc.**  
**Model I or III**

At this writing (October 1981), only one serious double-density DOS is available: DOSPLUS 3.3. It not only reads single-density TRSDOS, LDOS, and NEW-DOS80 disks, but also reads standard Model III TRSDOS



disks. In addition, the Model III DOSPLUS reads and writes disks that are compatible with the Model I.

Specialized disk I/O files and hardware-dependent features, however, cannot be shuttled between the machines. The little gem that lets the Model I read Model III TRSDOS disks is the Convert utility.

DOSPLUS 3.3 for the Model I currently supports only three devices: keyboard, video monitor, and printer. The RS-232 is not defined at this time.

DOSPLUS 3.3 allows mixing of various track drives. The Format utility formats disks for the desired track count and storage mode (double or single density). Then DOSPLUS 3.3 takes care of all housekeeping.

If specifying the track on each format is disturbing to you, Config will let you preselect the track count and stepping rate, operate with a clock speed modification installed, configure a drive as double sided, and, if your printer can handle them, send unaltered graphics codes to the printer. However, Config can only configure the system drive to a certain track count.

You might want to transfer your program library from single- to double-density format; DOSPLUS does this with the Transfer utility. A special utility, Copy1, allows single-drive copying of all file types. Create allocates disk space for a specified file.

Chaining is available with the Build . . . Do commands. Build lets you create a sequential set of tasks for DOSPLUS to perform. A Do file can be created with Build to get a directory, display the Free map, load a printer

driver, set Basic memory size, and jump to Basic.

Do checks high memory (4049H-404AH) and reserves about 300 bytes at the top of memory for itself and a small buffer. After Do executes, this memory space is returned to the system.

A problem arises when a high-memory program occupies the same memory area that Do usurps. Trying to Build a file that first loads the GSF (Racet Computes) sort module into high memory is disastrous. When Do loads GSF, it obliterates its buffer and all Build instructions are lost.

About a dozen programs are listed in the utilities section of the DOSPLUS 3.3 manual—all extend the versatility of the system. By being discrete about which utility functions become programs, you can create relatively small system disks. This provides the single-disk user with maximum storage space per disk.

CLRFILE zeros a file but does not reallocate its storage area. Copy1 is a necessity for single-drive users. It permits single-drive disks to transfer all types of files. Crunch is a utility that eliminates remarks and extra spaces in Basic programs. DiskZap has been amended to read single- or double-density disks, but it does not generate hash codes or passwords.

Conspicuous by its absence is Superzap's DFS function, which permits examining or modifying by file name rather than by sector. Micro-Systems Software provides this ability in a stand-alone machine-language DiskDump program.

Micro-Systems also provides an

adequate spooler with DOSPLUS 3.3. The spooler uses an operator-defined memory buffer and an optional disk file for printer output. Spoolong becomes a semi-background task allowing the host program disk I/O. If you are too conservative in allocating spooling memory space, the spooler temporarily seizes CPU control until the spooler buffer is cleared of its backlog.

DOSPLUS Extended Disk Basic Version 1.4 is authored solely by Micro-Systems Software, but I have not experienced a compatibility problem between DOSPLUS Basic and the MicroSoft/Tandy Extended Disk Basic. The only incompatibilities encountered were with the specialized disk I/O file structure of NEWDOS80 and with entry into Disk Basic itself.

When entering DOSPLUS Basic, no file buffers (channels) are allocated. With other DOSes, the default value is three buffers reserved. You must explicitly reserve I/O buffers when entering DOSPLUS Basic.

The SRCMD module is unique in that it allows the programmer to search and replace a string variable or expression.

File-handling enhancements with DOSPLUS include variable-record-length files, making ISAM techniques possible. With variable-record-length files, you don't have to calculate subrecords.

Micro-Systems Software reworked Basic in DOSPLUS 3.3 quite extensively. Actually, two Basics are provided: a full-feature Basic and an abbreviated TBasic. TBasic is meant to be used after a program is debugged and the various debug-

ging utilities are not needed. This provides the user with more memory space for data.

DOSPLUS 3.3 does have some shortcomings. It lacks a DOS high-memory command as well as complete device handling. It has no DOS Boot command for a warm reboot and the Basic2 library command has been eliminated.

It has only fair documentation. The literature is adequate for an experienced hacker but might not suffice for an entry-level user. There is also some conflict between DOSPLUS's keyboard routine and some Level II programs.

Overall, DOSPLUS 3.3 is a stable, efficient, fast, and easy-to-master DOS. However, if you have mastered NEWDOS80, you will miss many of Apparat's extra enhancements.

The major factor in weaning me away from NEWDOS80 was DOSPLUS's portability from Model I to Model III, and the ease with which it interchanges single- and double-density disks. The standard features of DOSPLUS have been the basis for many stand-alone utilities. If nothing else but this is considered, DOSPLUS 3.3 is a bargain. A business can do little better than DOSPLUS.

*(Micro-Systems Software Inc., 4301-18 Oak Circle, Boca Raton, FL 33431. DOSPLUS 3.3 is up to version 3.5; \$149.95.)*

James LaSalle  
October 1981

★ ★ ★ ★  
**DOSPLUS 3.4**  
**Micro-Systems Software**  
**Model I or III**

Several new concepts are evi-

dent in DOSPLUS 3.4. One of these is the cylinder idea, replacing tracks on disks. This idea comes from hard-disk technology.

On DOSPLUS 3.4, both sides of a drive are seen as one drive and files can span both sides. Track 1 goes around the first side of the disk and then continues on the back side, so there are six granules per track (with six sectors per granule).

DOSPLUS 3.4 can also read and write 40-track disks in an 80-track drive (with some restrictions). This is done by skipping every other track, and guidelines are given in the manual for carrying this out successfully.

This removes one difficulty in using 80-track drives: compatibility with the more common 40-track drives. The only real difficulty remaining is its inability to read self-booting 40-track disks in an 80-track drive.

Another new concept implemented in DOSPLUS 3.4 is that of complete device handling. The Force and Join commands allow you to tie devices together or redirect them. Even more important is that it allows you to use disk files as devices, so all printer or video output could be sent to a disk file for later examination, for instance.

A third new concept introduced is that of the wild-card disk file specification. When converting, transferring, or purging, you can use a wild-card mask so all fields meeting the desired conditions (such as beginning with CRS or ending with /DA1) will receive the specified action.

There are also two significant additions to Basic. The first is the INPUT@ command, which

vastly improves input formatting. The second is CMD "O, an array-sorting utility. Its syntax is compatible with CMD "O in Model III TRSDOS, but it is far more flexible.

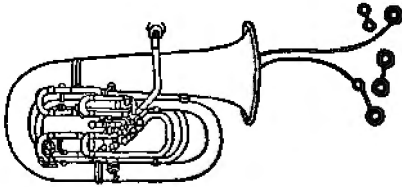
The DOSPLUS manual is vastly improved, and it now includes a 42-page technical section detailing the available DOSPLUS system calls.

DOSPLUS 3.4 does have some drawbacks. The authors have reverted to the ROM screen printer in the Model III version (in the interest of saving space), which means that graphics will not be sent to the printer. Diskzap, the disk editor, isn't set up under the cylinder concept. To examine the back side of a disk, you must ask for drive OB—a fact not mentioned in the manual.

Unfortunately, the price of DOSPLUS 3.4 has risen with its new capability; its \$150 price tag puts it in the same category as NEWDOS80 and LDOS, thus removing one of the arguments in its favor. However, with its new features, along with the DOSPLUS 3.3 strong points such as Model I-to-Model III compatibility, automatic density recognition, variable-save option when chaining programs, all DOS commands available from Basic, print spooling, Basic shorthand commands, Diskzap, and so on, combined with its tremendous speed and ease of use, it is one of the best operating systems available for the TRS-80.

*(Micro-Systems Software Inc., 4301-18 Oak Circle, Boca Raton, FL 33431; version 3.5; \$149.95.)*

John Ratzlaff  
October 1982



### **DOSPLUS 3.4** (See Page 283)

### ★ ★ ★ 1/2 **KWICOS** **KWIK Software** **Model I**

KWICOS is a cassette operating system that converts your 500-baud Model I with 16K to a high-speed system for loading and saving Basic or System programs.

KWICOS is a System program using about 1500 bytes of low user RAM. It lets you use your regular high-memory utilities as usual. KWICOS provides disk-like loading and saving commands in addition to your regular Basic commands. Keyboard debounce, operating speed, and the speed of loading and saving programs are controlled with KWICOS commands.

KWICOS's features include program loading and saving up to 4.5 times faster than Level II, with unique screen graphic symbols to indicate lock-on and transfer status; an active break key during program transfer; a Merge command allowing you to append Basic programs; program identification with a file name and comments of up to 32 characters; optional password protection; verification of any file saved in KWICOS format;

display of Basic program length; display of start, end, and entry addresses for System programs; display of file name and parameters of all KWICOS programs on tape; controllable keyboard debounce; and slowed program execution and scrolling of listing or tabulations for any program in memory.

For all its versatility, KWICOS is simple to load and use, and it doesn't need reserved memory space. You can add another KWICOS-format Basic program to the one in memory with Merge or MERGE "file name" in memory; KWICOS appends the second program to the first by moving the appropriate program pointers with no regard to line numbers.

KWICOS is easy to use, and it has no bugs. The user must identify KWICOS cassette programs with the data-rate number; the data-rate selected on program initialization must match the KWICOS recording, and the data-rate cannot be changed without reloading KWICOS. The preliminary documentation I received with KWICOS was well written but left out some essential details. This is being corrected, and some more program features are being added.

If you do not have a high-speed tape device and do not expect to get disk drives soon, KWICOS will save you time and frustration, and provide keyboard debounce and slow-down for program development and debugging.

*(KWIK Software, P.O. Box 328, Bolivar, MO 65613; \$26.)*

Fred Blechman  
February 1982

★ ★ ★ ★

## **LDOS**

**Logical Systems Inc.**

**Model I**

First of all, let me relieve the suspense by stating that, in my opinion, LDOS is by far the best disk operating system (DOS) currently available for the Model I TRS-80.

The current official version of LDOS documentation (Ldocs) is 253 pages. It's broken down into sections which describe the features and commands of the system for the everyday user, and sections with specialized technical information for the systems level programmer. Every command has its own section in the documentation and each of these sections has its own sequence of page numbers. The writing style is a little more technical than it needs to be, but no one should be stymied by the language, thanks to the numerous examples. Explanations are provided for key LDOS concepts.

A sampling of information in the Ldocs technical section includes such goodies as maps of system entry points (including some in Level II ROM), explanations of the directory, device control blocks (DCBs), file control blocks (FCBs), and file formats. This section also contains explanations of the more arcane features of the DOS. Finally, Ldocs contains a five-page glossary and a six-page error dictionary.

LDOS offers extensive customer support for registered purchasers of LDOS. The LDOS development team is constantly working on upgrades and patches. Any certified user may

send in his or her master disk, plus return postage, and, for no additional charge, receive the latest version by return mail. A newsletter is mailed quarterly to registered users who also have access to an LDOS bulletin board on CompuServe. A toll free number exists for the sole purpose of user support, not ordering.

LDOS offers various changes and improvements to the TRS-80 keyboard, including de-bounce and auto-repeat. A key-stroke multiply (KSM) package lets you custom define the keys.

System reconfiguration refers to the System command and its various parameters. This command allows you to alter certain aspects of the system's intrinsic performance. For example, you can tell the DOS that you only have one disk drive. Another System command, Alive, keeps a graphics block wriggling at the top right corner of the screen whenever the interrupt-task processor is active. Other System parameters let you tell your computer what you want the cursor to look like, whether or not it should blink, invoke lower-case display (if you have a lower-case modification), or tell the system if you want a screen print option.

LDOS provides a number of enhancements to Basic (Basic must be copied over from a TRSDOS disk). These include new file modes, blocked (variable length) records, a program single stepper, and several new CMD "n" statements. CMD's O, P, N, and X, respectively, turn off Break's ability to send you to Dcbug, for a screenprint, renumber a program, and provide a variable cross reference

utility.

LDOS also provides an extended debugging package that goes beyond the capabilities of TRSDOS's Debug. It may be used on code either in memory or on disk. Yct, it is not quite as handy for disk editing as one of the ZAP family monitors because you must load the information off disk, alter it in memory, and then write it back to disk.

A mini-monitor, one of LDOS's fine touches, is available right from the DOS, via the Memory command. LDOS honors HIGHS, which means that it protects programs in high memory, as does Basic. Memory tells you the current HIGHS (Memsize reserved) or you can use it to set HIGHS to a new value. You can use Memory to jump directly from DOS to any specified address.

LDOS also offers a surprisingly comprehensive job control language (JCL), extended device independence, and a wide spread acceptance of partial filespecs (with wild-card characters).

Another aspect of the new LDOS design philosophy is manifested in its upward compatibility with TRSDOS. It does not promise to mesh perfectly with NEWDOS files, but its PROT command does attempt to render alien disks readable by LDOS without diminishing their readability by the other system.

LDOS comes with the best spooler I've yet seen for the TRS-80 Model I. LDOS's spooler allows you to specify whether the buffer is to be in memory, on disk, or both.

One feature that may not be popular is the absence of Disk

Basic from the LDOS master disk. You must copy Basic over from a TRSDOS 2.2 or 2.3 disk. However, once this is done, LDOS patches Basic and offers several enhancements.

Other features available on competing DOSes but not offered in the current release of LDOS include ULTRADOS's selection of three Basics with three different trade-offs of features versus memory and NEWDOS-80's MINIDOS. At this time, LDOS does not support the Percom double-density board, but plans to in the future.

The most serious problem I've come across is the possibility of losing data if you kill a file while it's still open. The consequences of killing an open file in LDOS are nowhere near as dire as they were under early releases of TRSDOS. A similar problem is that innocent files may be overwritten if a disk fills up while the printer is routed to a file.

On the whole, I've found LDOS to be as error-free as any DOS I've ever worked with. In human engineering, system-integration, and flexibility it runs cylinders around the others. I think it's well worth its price.

*(LDOS is sold by Logical Systems Inc., 11520 N. Port Washington Road, Mequon, WI 53092; LDOS 5.1, Model I and III, \$129.)*

Paul Wiener  
June 1981

★ ★ ★ ★ 1/2

**LDOS 5.1**

**Logical Systems Inc.  
Model I, II, and III**

LDOS 5.1 is the state of the art in operating systems for the TRS-80. It supports more

features and different kinds of hardware than any of its competitors and possesses a user friendliness not found elsewhere.

LDOS allows any device to be opened as though it were a file; for instance, you can transmit different lines of a program to serial and parallel printers. An optional disk (\$70) contains many routines for device filtering, such as removing or translating codes that make your printer do funny things. One of the filters provided makes printers which do not slash their zeroes do a backspace and print a slash mark. Also on the filter disk is a complete EBCDIC translator and a Dvorak keyboard filter.

A MINIDOS filter is provided with LDOS 5.1 that allows you to kill files, check free space, send characters to the printer, turn the clock on or off, enter the debugger, display a disk directory, or repeat the last DOS command. All this is possible from within your application, provided the application has not disabled the interrupts. All properly written filters and device drivers may be loaded into memory and then SYSGENed to be loaded automatically and quickly at each reboot.

A TRSDOS user can step right into LDOS with minimal effort. Although some of its advanced features require study to understand, full mastery is not needed for successful operation of the system. The command syntax is almost identical to TRSDOS, though every command has been greatly expanded.

Another example of user friendliness is the Percom Doubler installed in the expansion interface. The default im-

mediately becomes double density for the format operation, and the track count for the drive is used as a default also. The LDOS user simply specifies the kind of format he wants or presses enter to use the defaults (which are user definable) and the system takes care of the rest. LDOS knows what kind of disk it is trying to read and reads or writes to it automatically, recognizing a single- or double-density disk without operator intervention.

The Copy command under LDOS is greatly expanded. It copies any file from one disk to another, but also copies a file to a device such as the RS-232 (useful to me when I transferred a number of files to an Apple). The command is not intended for multiple file transfers or disk backup; a Backup command is provided for this purpose. Unlike other systems, Backup cannot reproduce to a disk that has not been formatted, but I like this as it has saved me from accidentally ruining a valuable disk on more than one occasion. Backups may be done by file, date, range of dates, or whether or not they exist on the destination disk.

LDOS comes with a complete job-control language and 26 pages of documentation. The former is a complete chaining language in itself; though not as powerful as Basic, it is much more powerful than the simple chaining facilities offered on other systems. The documentation is the clearest, best written I have ever seen. More help is only a phone call away from the LDOS support team.

Included with LDOS is a printer spooler that operates with complete invisibility. It

spools the printer from 5K of memory and, when this is filled, can use up to 50K of disk space before hanging up the system to wait on the printer. It is one of the best spoolers I've seen, including those offered as stand-alone programs. It works from within Scripsit and any other program that does not disable the interrupts.

The KI/DVR device driver enables features such as keystroke type-ahead and the JKL screen-print function. Type-ahead is the feature I like most; it allows you to key in your input before the computer is ready for it. When the computer is ready, the information is relayed to it without delay. The type-ahead buffer can be emptied at any time. The JKL function, though still called by that name, is activated by pressing shift-down arrow-\* at the same time on the Model I and III. This prevents the J, K, or L key from echoing to the screen and messing up the printout—a thoughtful and needed change.

The most powerful feature in LDOS is the System (SYSGEN) command. This command causes all your system configuration, drivers, filters, routing, linking, and so on, as well as Spooler, Verify, and Clock, to be written to a file called CONFIG/SYS on drive 0. Each time the system is booted, it loads the user's configuration from this file. All system changes are made first in memory and do not become permanent until you execute a SYSGEN. This lets you experiment with different configurations without making a special System disk.

The LCOMM utility is an ad-

vanced communications package for use with the RS-232. Because it is so flexible (you can even use it to communicate directly with your printer, though not very well), it is difficult to learn; sit down with the documentation for at least half an hour and save yourself a lot of on-line frustration. The patch utility allows changes to the system or to other files.

The repair utility brings disks created on certain other systems up to LDOS standards. It updates TRSDOS's data address mark to an F8H instead of the FAH, making single-density disks directly usable on the Model III with LDOS. In fact, if your Model I is double density, you can switch disks back and forth between models without even knowing what density the disk is! It also performs additional correction on the disk to ensure reliable operation. The HITAPE utility permits the use of high-speed cassette I/O on a Model III.

The Basic provided with LDOS, LBasic, is a complete language rather than a patch to Radio Shack's Basic as was the case in the past. It offers several enhancements over RS Basic and is completely compatible with it.

Using Scripsit under LDOS with the LSCRIPT/FIX patch is a pleasure. The patch allows any ASCII character to be entered from the keyboard, including the tilde, left and right braces, and underline character. Also provided is a Scripsit\* text recover re-entry that works with as much ease as Basic\* does under TRSDOS. To accommodate this, the control key and a couple of other keys have been redefined.



There is also a Scrip/FIX for those who do not need the advanced features or do not wish to redefine their keyboards; VisiCalc has had a couple of new commands added and made Model I and III transportable.

An extra feature is a quarterly newsletter mailed to all registered LDOS owners. System module patches are included in this newsletter as well as many user-contributed utility routines.

LDOS is the best value today in an operating system for the TRS-80. LDOS 5.1 comes with so many utility programs, drivers, and filters that it requires two disks to hold it all on a single-density, 35-track Model I. If you need additional computing power. LDOS 5.1 is the best choice you can make! (*Logical Systems Inc., 11520 N. Port Washington Road, Mequon, WI 53092. LDOS 5.1 is currently available for the Model I and III, and is priced at \$129. A Model II version is being developed.*)

Charles D. Knight  
September 1982

**LDOS 5.1**  
(See Page 283)

**MULTIDOS**  
(See Page 283)

★ ★ ★ ★  
**NEWDOS80 2.0**  
**Apparat Inc.**  
**Model I and III**

With the release of NEWDOS80 2.0 for both the Model I and III, Apparat has produced an extensively revised

system that can meet any challenger head-on! NEWDOS is a new operating system rather than a debugged and enhanced TRSDOS.

Apparat supplies seven other programs on the NEWDOS disk. Superzap allows direct disk access for applying corrections. Chainbid provides a mini-text editor for writing chain files. EDTASM is Radio Shack's Editor/Assembler with disk capabilities.

LMOFFSET loads machine-language programs from tape to disk and vice-versa, and applies a relocating appendage to keep them from crashing DOS while loading. Dircheck reads a directory to provide essential repair information. Disassem, a machine-language disassembler, writes a source file readable by EDTASM to disk.

H. S. Gentry's automatic spooler ASPOOL feeds a print file from disk to the printer while the computer processes another program. Version 1 owners note that none of these programs as supplied with Version 1 operate under Version 2. Read the documentation carefully when converting.

Version 2 and Version 1 incompatibilities make some programs inoperative unless you incorporate changes. The same applies to TRSDOS, mainly with machine-language programs that call certain DOS routines.

Apparat also supplies zaps for Scripsit, VisiCalc, APL80/CMD, and Racet's DSM and Infinite Basic so they function properly. As with other operating systems differing from TRSDOS, not all programs from other sources run correctly.

Technical jargon can be con-

fusing even for knowledgeable operators. Because the word "track" implies that a track could span several physical tracks on the disk, Apparat created "lump." A lump can span several tracks, and a granule can start in one track and end in another.

Double-density and eight-inch disks can use the maximum sector/track count and maintain the same directory structure.

In NEWDOS80 2.0, Apparat has modified Copy format 6 to recognize a copy-by-file (CBF) parameter. Files are copied one at a time rather than as a full disk back-up. This copy format recognizes two new parameters: the destination file only (DFO) parameter, and the inclusion list file (ILF) parameter.

NEWDOS does not provide a back-up function; instead, six different forms of the Copy command are available. The first four provide for single-file copy, while five and six are for full disk copies.

For the Model III only, you can specify printer lines per page and characters per line with a Forms command. Setcom activates and deactivates the RS-232 interface, and can set or change word length, baud rate, stop bits, and parity. Setcom directs whether the input routine should wait until an input byte is received or an output byte is sent.

Apparat has enhanced older Basic abilities and added new ones—specify the number of files, memory protect size, and a Basic command sequence from DOS or by an Auto command. Two direct editing commands have been added, and a Renew command recovers programs

lost by an accidental New.

NEWDOS80 2.0 incorporates Basic CMD functions similar to Model III TRSDOS. Apparat provides a greatly enhanced Basic sort routine with command O. You can implement a direct sort on up to nine single- or multi-dimensional arrays.

In addition, an indirect sort builds a new integer array that forms a table of pointers to elements of other arrays in sorted order without changing the other arrays. You can then use the new arrays to read the other arrays in sorted order. Both sorts function in ascending or descending mode.

Further refinements to delight any veteran programmer are available through nine CMD F functions. These help you bail out of complex programming and prevent program crashes.

TRSDOS defined sequential and random disk-file structures; NEWDOS80 1.0 redefined these files and added two other types with a total of five subtypes. Sequential files became print/input files; random files became field-item files. Apparat added marked-item files with subtypes MI, MU, and MF, and fixed-item files with subtypes FI and FF.

NEWDOS80 2.0 leaves all five file types. The new documentation is much better than the old one. Understanding these file structures is not easy and requires time and effort, but no more than the original TRSDOS manual.

NEWDOS80 2.0 supports up to four physical drives, and it supports standard single-density disks along with PERCOM, LNW, and Apparat double-density boards on 35-, 40-, 77-, and 80-track single- or dual-

headed five-inch or eight-inch drives.

Double density presents certain problems with data transmission. To ensure that a formatted disk is stress tested, Apparat supplies two optional zaps using the byte 6DB6 for formatting in place of the standard Model I E5E5 or Model III 5B5B patterns.

The manual states that up to 30 percent of disks not certified for double density might fail using the worst-case 6DB6 formatting. Since I just added double density, none of my old disks are DD certified. Using 6DB6, NEWDOS80 2.0 and the new Percom Doubler II, only 3 percent failed. Apparently, NEWDOS and the Doubler II are a perfect team.

Model III TRSDOS disks cannot be read by NEWDOS80 2.0 due to a difference in disk structure. However, the Model III Convert utility works on 2.0 five-inch single-density disks if the directory is properly laid out.

Also, the Copy command, combined with the proper PDRIVE specification, lets you transfer files between the two systems. You can swap disks between the Model I and Model III if you follow certain limitations and specifications.

NEWDOS80 2.0 represents an improvement over Version 1.0 and is a worthwhile investment. Its extensive additions to Basic editing and programming features provide my Model I with most of the Model III capabilities.

Apparat goes to a great deal of time and effort to remove bugs from NEWDOS, but does not supply phone support personnel. Apparat also charges for new on-disk copies of NEWDOS zaps.

There are trade-offs in any operating system. The best way to choose between all the alternatives would be to use them, but expense prohibits this. The NEWDOS routing capabilities should suit most needs although they do not approach the niceties provided by the LDOS phantom-device routing.

However, to my knowledge, no one supplies a facility in any way similar to NEWDOS Mini-DOS, or the various disk file structures of NEWDOS. NEWDOS80 2.0 represents the state of the art.

*(Apparat Inc., 4401 S. Tamarac Parkway, Denver, CO 80237; \$149.)*

Paul R. Prescott  
February 1982

**NEWDOS80 2.0**  
(See Page 283)

—EDITOR/ASSEMBLERS—

★ ★ ★ 1/2  
**CC EDTASM +**  
**Tandy/Radio Shack**  
**Color Computer**

EDTASM + for the Color Computer is the offspring of EDTASM + for the Model I. It has a subset of the features found in its dad, but unfortunately, it isn't much like the old man.

EDTASM + is reasonably well done and comes in a ROM cartridge. It is made up of three parts: the editor, the assembler, and a debug package called ZBUG.

The editor contains standard Radio Shack editing commands, as well as two interesting commands not seen on other Radio Shack editors: C, which copies a block of lines; and M, which moves a block of lines.

The assembler uses 6808E mnemonics only. This isn't a detriment unless you have existing 6800 code that could be converted to 6809E system, and that isn't easily done given the configuration differences between systems.

There are a large number of operators that you can use in the assembler. You can use addition and subtraction, division and multiplication, modulus, and logical operations.

The most powerful feature that the assembler portion gives you is the ability to do in-memory assembly. This allows you to take the source code in the edit buffer, assemble it, and store the resulting object code in memory. The object code is stored after the edit buffer and symbol table; it is properly relocated without your having to specify any absolute origins.

This in-memory assembly makes for rapid debugging. You can go directly from one function to another. You don't have to laboriously patch the program; you simply type E, enter the editor, make the changes, type A/M to assemble in memory, type Z to enter ZBUG, and voila! The object is there.

I do wish the beautiful macro capability of the Model I ED-TASM+ had been left in, though. You can get along well without macros, but I'd be willing to pay double to have the capability.

Another thing that would have

been nice is conditional assembly. It lets you assemble bracketed code segments conditionally.

Pretty printing is another desirable capability. It enables you to use things like Title, Page (to skip a page), and other amenities.

The third segment of ED-TASM+, ZBUG, gives you a disassembly capability that allows you to list any data area in memory, with the assumption that it contains 6808E instructions; the output is the equivalent 6809E mnemonics.

ZBUG allows a great deal of flexibility in input and output formats. You can display memory in the byte mode (1 byte at a time), word mode (2 bytes at a time, as in addresses), mnemonic mode (disassembly), or ASCII mode, and in numeric or symbolic form. The input and output number bases can be octal (base 8, not too useful), decimal, or hexadecimal. You can enter data in mixed formats. ZBUG shines in this area.

I would like to see a Find command included that would enable you to find a specified byte or address value.

If you have a cassette-based system, I recommend ED-TASM+ without hesitation. It is sure to become the standard for the Color Computer.

*(Tandy/Radio Shack, Fort Worth, TX 76102; \$49.95.)*

William Barden, Jr.  
November 1982

**Color Diagnostics**  
(See Page 184)

★ ★ ★ 1/2

## EDAS Misosys Model I and III

EDAS is a disk-based editor/assembler. Tape I/O is not supported and EDAS does not support macros, relocatable code generation, or conditional assembly. The next implementation of EDAS will support some conditional assembly and other new features. However, it is doubtful that macros or relocatable code will ever be supported.

EDAS is not organized in overlays. Normally it's all there in memory at the same time. However, if you're cramped for space, EDAS' X (extend) command lets the text buffer overwrite the assembler. This in effect achieves an overlay system, since the assembler can be reloaded and run after the source has been moved to disk. But you will rarely need to bother with that procedure because EDAS solves the long source file problem in another way.

Via the Get statement, EDAS assembles directly from disk. One assembly may use any number of Get statements. Thus EDAS lets you break your source program into several modules, each of which is stored in a separate disk file. Since a file is assembled as a result of a Get isn't actually buffered in memory, a module can be larger than 48K.

If your in memory source code consists only of a series of Get statements and an End statement, most of your computer's memory is free to hold the symbol table. This solves the source size problem and permits editing

and assembling of large programs. It also provides the basis of a subroutine library management system.

In fact, there are no ORGs in any of the Get modules, they will be sequentially arranged in memory relative to the ORG in the text buffer. The EDAS assembler also accepts a View command, which causes it to display the requested disk file on the video screen. And to further enhance its disk, EDAS allows viewing disk directories (including file size allocations and free space) and killing files without leaving the editor.

Three other nice features of EDAS have a synergy resulting in a breakthrough in debugging ease. I am referring to the /IM assembly feature (lets you assemble a program directly into memory), the Branch command (lets you jump to any specified address), and EDAS' warm start capacity.

EDAS supports lowercase and offers two modes of operation. In one, both upper- and lowercase characters are maintained in the text buffer just as they were keyed in. In the other, certain lowercase characters are converted to uppercase as they are inserted, because assembler syntax requires instructions, mnemonics, and labels to all be uppercase. Even in this mode, EDAS does not convert material in quotes or comments, since lowercase in those areas would not generate assembly errors.

EDAS has several other improvements over other editor/assemblers. For example, labels may be up to 14 characters long.

The package's documentation is well written and complete; it

details the operation of the editor/assembler only and is not a tutorial for those seeking a tutorial in Z80 machine-language programming.

Finally, I have discovered a few minor bugs with the utility. They are cosmetic and don't interfere with the program's operation. For instance, when explaining the Write command, the documentation indicates that a comma should be used to separate the starting and ending line numbers. Actually, a colon must be used.

*(Mitsosys, 5904 Edgehill Drive, Alexandria, VA 22303; Version 4.0, \$100.)*

Paul Weiner  
November 1981

★ ★ ★

### **EDAS 4.0**

**Galactic Software  
Model II**

The EDAS 4.0 Editor/Assembler package for the Model II uses EDTASM commands as a subset of its own command set. An attractive feature of this software program is that you can assemble directly to either memory or disk. You can use this direct-to-memory option in conjunction with the Model II debugger, which gives you a powerful method of developing machine code software.

The editor function is excellent. You can edit in either the line or global mode, and each changed string will be displayed by the editor as it comes to it. Subroutines, tables, and even blocks of lines can be moved with ease from one place within the text to another, and the

editor will automatically renumber the text.

Although the EDAS 4.0 is high priced, this assembler/editor package does it all, and is a useful piece of software for the Model II.

*(Galactic Software, 11520 N. Port Washington, Mequon, WI 53092. EDAS 4.0 has been replaced by an expanded version, EDAS 5.0, which is available for \$199.)*

Jake Commander  
August 1980

★ ★ ★ ★ 1/2

### **EDTASM-Plus**

**Microsoft  
Model I**

Microsoft's EDTASM-Plus is one of those miracle programs that cures just about every hassle you might think of. If you are just starting to get into Assembly language and the inner workings of a TRS-80, EDTASM-Plus is just what you want.

EDTASM-Plus is an incredibly complete Assembly-language development system oriented to the cassette TRS-80 user, and takes into account the overall design of the TRS-80, the way it uses memory, and so on.

EDTASM-Plus has three important parts. The first, the editor, is much like the same feature in the original Editor/Assembler. Using numbered program lines, it lets you type in Assembly-language programs in Z80 mnemonics, or code words. The editor stores these lines for later conversion, or assembly, into the actual numeric machine-level commands from which the Z80 chip in the TRS-80 was de-

signed to work.

Editing features in EDTASM-Plus have been greatly expanded over those in the original Editor/Assembler. EDTASM-Plus lets you move any part of your Assembly-language program to another set of program lines. It also lets you edit a given series of program lines without having to reenter the Edit function for every line.

EDTASM-Plus also adds a couple of important features not found in the original Editor/Assembler. For example, it accepts macro definitions, and it also allows conditional assembly. You can establish conditions that must be met for a given part of the program to become part of your assembled machine-language program, and if the condition is not true, the assembly just skips that part of the program.

The second part of EDTASM-Plus, the assembler, reads your mnemonics and converts them to actual digital instruction for Z80. EDTASM-Plus provides a variety of error messages and warnings when your program is assembled, like the original Editor/Assembler.

EDTASM-Plus assembles your program directly into the TRS-80's memory so you don't have to record it on cassette to run and debug the program. It also gives you plenty of the interactive features that are inherent with a Basic interpreter, and eliminates a tiresome and often discouraging series of cassette loads and reloads just to test a program.

Z-Bug, the third part of EDTASM-Plus, is like a T-Bug whose IQ was beefed up by some miracle drug! Z-Bug is a

monitor, a program that lets you look into the TRS-80's memory locations and change them if needed. Use Z-Bug to run your assembled machine-language program, see what it does, make corrections and try again, set breakpoints, and jump back to EDTASM-Plus and the source program.

Z-Bug goes beyond T-Bug in that it displays program steps as mnemonics rather than simple byte-by-byte hex numbers. It's like a line-by-line disassembler, and because Z-Bug stays resident with your source program and EDTASM, you can reference locations to be examined with Z-Bug using your own symbols rather than specific hex memory locations.

With Z-Bug, you can have eight breakpoints in your program, and you don't have to fix a breakpoint after it's reached, as you do with T-Bug. Z-Bug also lets you step through your programs by single steps, which is a lot easier than trying to test a program on the fly at microsecond speeds. It's great for looking at the TRS-80's ROM sub-routines and the program routines in the reserved RAM areas, so you can learn something of how Basic does what it does.

The instruction handbook is as excellent as the software itself. Microsoft even included a notice of a couple of obscure bugs that would arise in tricky assemblies, and tells you how to use Z-Bug to make the corrections to EDTASM-Plus and punch yourself a corrected tape. EDTASM-Plus obviously represents a huge amount of work by its programmers and Microsoft.

*(Microsoft, 10700 Northup Way, Bellevue, WA 98004; now available from Tandy/Radio Shack for \$34.95.)*

Chris Gundlach  
October 1981

★ ★ ★ 1/2  
**INTASM 2.0**  
**Singular Systems**  
**Model I and III**

Part of the difficulty of learning to program in Assembly language is that such a large first step must be taken. A programming aspirant must first gain at least a mild familiarity with Assembly mnemonics from literature, then purchase, for a hefty price, an assembler and learn its operation and syntax.

Enter Singular Systems with INTASM, a mini-Assembly-language development system. INTASM is composed of two machine-language modules: ASM1—a Basic editor and Assembly interpreter that recognizes a limited subset of Z80 Assembly-language source statements—and ASM2, a mini-assembler that accepts the same source instructions as ASM1 and assembles them into memory for subsequent execution.

Three versions of this program—1K, 32K, and 48K—are supplied on tape (they can be dumped to disk) in Level II format. The programs are run by loading one of them into your machine and entering Level II or Disk Basic. Reserve memory so Basic doesn't stomp on the machine-code module stored in high memory.

INTASM has no new editing commands to learn—the same old Basic ones are used. There

are only 54 of the Z80's more than 700 mnemonic codes to learn, but you do have the option of executing other instructions if you want to enter their machine codes.

You are not limited to using only hexadecimal numbers; INTASM accepts base-10 constants. You can reference instructions by jumping to or calling a line number instead of a memory location. This makes up, in part, for labels not being supported. Once your program is written in ASM1, it can be run by simply entering the Basic command, Run.

ASM1's Step-On instruction executes the program one step at a time, letting you follow each instruction's effect on the registers. This not only helps in debugging, but it's also about the fastest and easiest way to gain familiarity with each Z80 instruction's function.

Once you have gained some experience programming with ASM1, you can get the feel of working with an assembler by using ASM2. You use the same mnemonics, but this time you are required to assemble your program before running it.

There are two sides to every coin. With all the advantages INTASM has, there are a number of drawbacks as well. It is definitely easy to use, but the very features that make it simple also limit its power. The most obvious of these features is the limited number of mnemonic codes it accepts; 54 out of over 700 is a very small subset.

There are many unsupported Z80 functions. For example, it recognizes LDIR, but you cannot use LD BC,NN or LD DE,NN to



load the BC or DE registers. Since LDIR's operation depends on the contents of these registers, and there is no single instruction to load either of them, LDIR loses a lot of its ordinary ease of operation. Although it is very convenient for debugging purposes to assemble to memory, INTASM does not have the ability to assemble to disk. This prevents you from using your Assembly-language routines independently from INTASM.

ASM2 limits the user to only 1,900 bytes of machine code and 80 bytes of stack space. Although there is quite a bit you can do within this limit, it is much smaller than that allowed by ordinary assemblers.

Since INTASM seems to be designed primarily as an aid for Basic programmers to learn to write their first programs in Assembly language, these limitations are not as severe as they might at first appear. Once you are familiar with the subset of mnemonics, and their effects on the registers and flags, the operation of the ASM2 assembler, and the use of break points to debug Assembly-language programs, you can set aside the program and purchase an assembler and an interpretive monitor/disassembler such as TASMOM or Macro-MON.

INTASM is an excellent programming aid that definitely helps beginners taking their first few steps toward becoming the Assembly-language superstars of tomorrow. For \$20, it is well worth its price. (*Singular Systems, 810 Stratford, Sidney, OH 45365; \$20.*)

Joel Benjamin  
September 1982

★ ★ ★ 1/2

## MZAL

### Computer Applications Unlimited Model I or III

The MZAL (Modular Z80 Assembly Language) editor/assembler is for disk systems only. The package includes a thick manual in a large three-ring binder and a diskful of programs, including a full-screen text editor, an assembler, a linking loader, several demonstration programs, and a small but useful library of assembler routines.

The manual doesn't teach Assembly-language programming, and it expects some sophistication on your part. Beginners can use it, but they won't appreciate the more advanced features. The manual includes a Z80 technical manual that covers the Z80 commands in detail. But, the manual has one glaring problem—no index. This is the only major error I found in the entire package.

MZAL is a modular system that promotes modular programming. You write programs using a full-screen text editor, assemble them using a separate assembler, and link them together using a separate linking loader. Your Assembly-language programs are limited only by a 30,000-byte symbol table, or on-line disk storage.

Release 2.0 provides several programming options: changing Assembly source programs from one assembler format to another, eight-character labels, recursive and symbolic substitutive macro constructs, conditional assembly, linking after assembly with linking loader, and specifying

ORG of object-code linking time.

The TXEDIT program is a full-screen editor that does not support macro key definitions and does not take lowercase input. However, TXEDIT lets you insert as many lines as you need before two lines.

With the various commands in TXEDIT, you can search for any string and optionally change it to any other string, you can renumber your program, or copy or move single lines or blocks of lines much easier than with the normal EDTASM or Basic line editor.

MZAL's multipass assembler ASMBLR takes the Assembly-language text file you created with TXEDIT and assembles it into executable Z80 object code or into a relocatable module for the linking loader. The MZAL assembler accepts the standard Z80 opcodes. It also supports a variety of operation codes, not standard for the Z80, that set assembler options and control how the assembler generates the code.

A linker is a program that takes CMD machine-language modules and links them together into a single program. It lets you relocate a program without having to reassemble it. These features are accomplished through an /RLD file. It contains all the needed information for the linker to process the machine-language program.

Without a linker, you must reassemble the entire program each time you want to test it. With a linker, you can keep /CMD and their associated /RLD files on disk, and link them into your programs without reassembling them.

The LEXCONV program is in-

cluded on Release 2 of MZAL. It lets you read disk files created by other assemblers and converts any of four file formats (MZAL, Apparat EDTASM, Macro-80, or unnumbered ASCII) to any other of the four.

LEXCONV is menu-driven and easy to use. It loads the entire text file into memory and then processes it. It doesn't let you swap disks but begins writing the output file immediately. This is a slight fault.

The TXEDIT program fits into 32K. The manual provides information for using the linker to modify it to reside in the top of 48K.

If you are a casual Assembly-language coder, do not buy this package. It is expensive, and, except for the text editor, its most powerful features will remain unused for a long while. However, if you write programs in which the Assembly-language source code exceeds memory, or other constraints demand that you write it in modular form, then MZAL is well worth the cost.

*(Computer Applications Unlimited, P.O. Box 214, Dept. ABM, Rye, NY 10580; \$149.)*

Bruce Powel Douglass  
October 1982

★ ★ ★  
**SDS80C Editor/  
Assembler/Monitor  
The Micro Works  
Color Computer**

Anyone interested in developing software for a 6809-based microcomputer system will be interested in The Micro Works' editor/assembler/monitor package. Its price includes the

software on a ROM pack cartridge, a completely documented 41-page manual, and an MC6809-MC6809E microprocessor reference card.

The editor is screen oriented. What appears on the display is what is in the text buffer. The editor constantly displays the amount of RAM left in the buffer, in decimal, at the top right corner of the screen. Since the program is located at hex address C000 in ROM, most of the RAM memory is available for use by the text buffer.

Functions supported include finding strings, moving and copying blocks of text, and reading and writing files to and from cassettes.

The assembler accepts source statements in standard 6809 Assembly language, and produces object code to tape or memory. It expects the source statements to be in memory in the format used by the editor. Source and object code can also be directed to the screen or the printer.

The manual does not go into great detail when it comes to 6809 Assembly-language programming. It assumes you are familiar with that language.

Some of its features and options are: support of 6800 instructions for cross assembly, local labels, conditional assembly, and pause/break/speed control of listings.

Under control of the monitor, you may move the cursor up, down, left, or right to display the contents of memory. Hex data may be entered or changed in this mode.

The manual contains some useful information including programming hints and techniques.

Appendix 2 contains information on the Basic ROM entry points, with some of the more useful subroutine calls listed. Appendix 3 contains information and programs on timing loops, and Appendix 4 contains information on interfacing a printer to the Color Computer.

The Software Development System SDS80C is quite a useful software package.

*(The Micro Works, P.O. Box 1110, Del Mar, CA 92014; Color Computer, \$89.95.)*

Howard Berenbon  
June/July 1982

★ ★ ★ 1/2  
**TRS-80 Disk  
Editor/Assembler  
Tandy/Radio Shack  
Model I**

The Disk Editor/Assembler package (catalog 26-2202) includes features that link together Assembly-language and Fortran programs. It also includes a library of Fortran subroutines that can be accessed from Assembly language.

The package has five components: an editor (EDIT), a macro assembler (M80), a cross-reference processor (CREF80), a linking loader (L80), and a Fortran library (FORLIB).

EDIT is a stand-alone program rather than a combination editor and assembler as is EDTASM. EDIT can be used on files created by it, Basic files in the ASCII format, and any other files in a suitable format. EDIT makes disk accesses from time to time during the editing session, because it doesn't keep the entire file being edited in memory

all at once. You can return to DOS by exiting and writing out the edited file, or by quitting and discarding changes (very useful if you've made a dreadful error).

EDIT has all the capabilities of EDTASM and the Basic editor, although some of the commands are different. It also has some new features and additions to the old ones. Line numbers can go up to 99,999, if desired. In addition, the file can be broken into pages, using special commands to insert and delete page marks. Line numbering can start over in each page, providing for many more lines or for a printed listing with page breaks at points other than the bottom of a page.

For insert and replace commands, a temporary line number increment can be specified. Ranges can be specified in terms of a number of lines as well as by beginning and ending line numbers. The break key is echoed on the screen as a \$. Editing within a line allows you to search for a series of characters as well as a single character. The Find command can locate and display more than one occurrence of the desired string with a single command. The Substitute command provides for changing a given string to another string for any specified range of lines.

The immediately obvious difference between EDIT and EDTASM is that EDIT is a stand-alone program. When you have created or changed a file, you must return to DOS and execute a separate program, M80, to perform an assembly. This is somewhat time-consuming.

There are a number of things about the editor that are irritating. First, the files con-

structed by the editor waste disk space. The line numbers are stored as 5-byte fields: if the upper limit for line numbering were set at 65535, the line number could fit in 2 bytes. I found that a file that was written under the old EDTASM memory-image format and used five granules of space took seven granules under the new format. That's 10 more sectors—a whole track!

Also, the file handling by EDIT is quite awkward. When you are making changes to an existing file, you must save the new version under a different file specification. This is done so that you have a back-up of the original file. It certainly contributes to using up the available disk space.

Third, in a lengthy editing session, you might want to save your revised version at different points so that a machine or power failure won't require you to repeat all your work. Each time you do this, a new file must be created.

The default file extension for the macro assembler is MAC. It would be nice if the editor used the same default, but it doesn't. Also, M80 accepts lines of up to 132 characters. EDIT will create and process 255 characters in a line. This could cause a problem, but you are unlikely to create a line longer than 132 characters.

The manual states that an insufficient-memory error can occur under certain conditions. I think that the editor should have some means of informing the user when memory starts to fill up. I also think there should be ways to invoke the editor and process more than one file, to append another file to the one you

are editing, and to read and process existing EDTASM source files.

The macro assembler is the portion of the Disk Editor/Assembler package that converts the source language module into relocatable form. The relocatable module can then be processed by the loader to create an executable program.

Commands to M80 consist of file names and switches. They are the object (relocatable) file, the listing file, and the source file. The switches are special instructions to the assembler. They tell it to produce a cross-reference file, list addresses in octal or hexadecimal (the default), and assemble 8080 or Z80 mnemonics (Z80 is the default). M80 supports the standard Z80 and 8080 mnemonics.

The most important feature is the ability of M80 to assemble a module in relocatable form, or without regard to its eventual memory location. This feature allows you to separate instructions and data. If you have a program consisting of three modules, and the middle one needs to have something added to it, the other two don't have to be reassembled. The program can be linked together again by the loader after the middle module has been reassembled, and the third module will follow the second one at the proper address. M80 also has several pseudo-operations that facilitate modular programming.

The macros are another big feature of the macro assembler. A macro is a model; it is a series of instructions that can be placed in a program at several different points by using the name of the macro as an opera-



tion code. The macro itself is written and placed in the program prior to the point at which it will first be used. M80 has three other functions similar to the macros. They are all statements that provide for the material between them and the ENDM to be repeated several times.

Three pseudo-operations are available for controlling the listing produced by the assembler, and M80 can produce a cross-reference listing. This program processes the cross-reference file and creates a listing file. The listing file has the cross-reference listing at the end, showing all symbols in alphabetical order, with all references to them.

M80 is a stand-alone program. This is a disadvantage, since when assembly errors are detected, you must exit from M80, execute EDIT to correct the errors, then execute M80 again to reassemble the program.

The assembler takes two passes to complete its work. Errors are displayed on both passes. The assembly is some-

what slower than the EDTASM assembly because of the disk accesses. The source program is read from disk, and both the listing and object files are written on disk. The manual could stand some improvement, and it is somewhat limiting to have macros defined only internally. Also, the assembler generates successive line feeds in some places.

The purpose of the Linking Loader (L80) is to construct an executable program from one or more relocatable object files created by the assembler. Commands consist of file names and switches. The file names used in a command tell the loader what files are to be loaded into memory. The switches are, in reality, special commands to the loader.

Some of the features of L80 include relocation, resolution of external references, separation of instructions and data, loading at specified addresses, restart, construction of a disk file, memory map, and execution after loading.

L80 also has several shortcomings. Immediate execution sometimes failed when I tried it. The display of data and program areas seems backwards in some ways; since CSEG is the assumed option in the source program, it seems that the loader should print the word program rather than data when loading the program. When you specify a starting address for data you must also specify one for the program if you are loading several modules, each of which has both program and data areas. You are not given the option of printing a memory map. Finally, there is no provision for accepting com-

mands from a source other than the keyboard.

There are some improvements I would like to see made to the package. These include reducing the amount of space required by source and listing files, changing the file handling in the editor so that new files don't necessarily have to be created each time changes are made, and revising the manual. I am happy with the package's features, especially the linking loader and the modular programming features of the assembler.

*(TRS-80 Disk Editor/Assembler is sold by Tandy/Radio Shack, Fort Worth, TX 76102; \$34.95.)*

Guerrri F. Stevens  
February 1981

---

## EDUCATION

---

★ ★ ★

### **Dilithium Tapes Dilithium Press Model I**

Dilithium Press has published many microcomputer books, and as far as I know, this is their first venture into the software field; they have made a good beginning. All five tapes are for the 16K Level II TRS-80, although most of them will run in 4K.

Tapc 1 is an applications tape. It contains six programs that assist you in various areas of your personal life. The programs are Biorhythm, Checkbook, Decide, Loan, Mileage, and Questionnaire.

Decide is the jewel in this col-

lection. It helps you reach a decision on a course of action or on the purchase of some item. The best part of the program is that it forces you to look more deeply into a decision before making it. Questionnaire would most interest teachers as it grades multiple-choice exams or questionnaires.

Tape 2 is an educational tape that includes programs for all ages. These include Arithmetic, Flashcard, Metric, Numbers, Tachist, and Vocab.

Tachist helps improve reading skills by flashing phrases on the screen. Every time you get a phrase correct, the next phrase is flashed at a quicker speed. There should be more practice at each level before the speed is increased. This package should help students with their schoolwork as well as get them interested in using the computer for something other than games.

Tape 3 consists of three games: Decode (Mastermind with a new face), Groan (a dice game), and Wari (a game of capture between you and the computer). In Wari, you begin with four stones in each of 12 squares. You move the stones around the board, attempting to catch the computer's pieces while it tries to capture yours. This is the best game in the package.

Mathematics is the theme of Tape 4. The programs have specialized functions. Curve fits a polynomial function to a set of data pairs. Diffeqn figures differential equations. Graph plots the behavior of a math function. Integrate finds the numerical integral of a function. Simeqn solves a set of simultaneous linear equations. Stats analyzes

a set of numerical data and displays several statistical parameters that describe it.

Tape 5 consists of graphics display and miscellaneous programs, and it is more for demonstration than practical application. The techniques demonstrated in these programs will be helpful when someone asks you, "But what can your computer do?" They will also help the budding Basic programmer to better understand the capabilities of Level II Basic.

The documentation is slender, but the tapes are self-prompting so you shouldn't have problems with them. Most of them are also described in a book from Dilithium Press called *32 Basic Programs for the TRS-80 Computer (Level II)* by Rugg and Feldman.

*(Dilithium Press, P.O. Box 92, Forest Grove, OR 97116.*

*Dilithium Tapes are now available only in 32 Basic Programs for the TRS-80 Computer, \$19.95.)*

Rod Hallen  
April 1980

★ ★  
**Disk Instruction Course  
Tandy/Radio Shack  
Model III**

The Disk Instruction Course is a step-by-step guide in the use of TRSDOS and Disk Basic. It comes with four disks, each with one lesson containing six to 11 parts. The format of the course is a series of short comments followed by a few questions. If the questions are answered correctly, you proceed.

Although the content of these

lessons is to the point, they are too simplistic. Neither by description nor through practical experience do they expand upon anything described in the original manual.

(Tandy/Radio Shack, Fort Worth, TX 76102; \$29.95.)

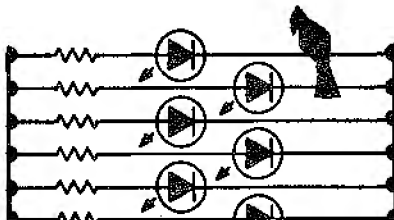
Robert L. Zeppa  
November 1980

★ ★ ★

### **Doctor and Fetch Omicron Software Model I**

Doctor and Fetch demonstrate the field of artificial intelligence. Their documentation is good, but loading is touchy. Doctor takes the part of a psychotherapist and asks you questions about your problems. The theory behind the program is good, but this implementation gets lost quickly if the program doesn't receive exactly what it is looking for. Fetch allows you to make declarative statements, which are stored, and then you can ask questions relating to the stored information.

Both programs are interesting and demonstrate some possible future computer applications. I do, however, feel that they are overpriced, and I think that most buyers will feel the same. Both programs on one tape for about \$7 to \$8 would be more realistic.



(*Doctor and Fetch*, once sold by Omicron Software, P.O. Box 2547, Sepulveda, CA 91343, are no longer available.)

Rod Hallen  
January 1980

★ ★ ★ 1/2

### **Eliza Tandy/Radio Shack Model I, II, and III**

Danger! This program can be hazardous to your mental health. This warning should be emblazoned across Eliza's instruction pamphlet.

Eliza is an interesting exercise in primitive artificial intelligence. The program simulates a Rogerian or a nondirective psychotherapist during an initial interview with a patient.

The program picks up on key words and sentence phrases. As the key words increase, the conversation becomes more personal until many feel as if they are communicating with another person instead of with a structured collection of bytes in a computer.

Several scientists, including Dr. Carl Sagan, have predicted that Eliza is the forerunner of an automated form of psychotherapy; however, Dr. Joseph Weizenbaum, who wrote the original Eliza, disputes that. In fact, he is disturbed at the tendency of people to humanize it.

The flip side of the program tape contains an Eliza that routes its output through the Radio Shack Speech Synthesizer, and I swear that the synthesizer develops a Viennese accent when Talking Eliza is running.

Eliza makes her home at hex



5000 to 7800 with the entry point at 77F2 right above the Level II reserved RAM, and where the DOS is stationed. Disk operators will have to relocate the program before use.  
*(Tandy/Radio Shack, Fort Worth, TX 76102; \$14.95.)*

John Warren  
 April 1981

★ ★ ★

**Fractional Sound  
 The Innovative Penguin  
 Model I and III**

Fractional Sound is a fractional drill program for the Model I and Model III with sound (by connecting the auxiliary plug to an amplifier). The program allows you to set parameters for: the type of operation (add, subtract, multiply, divide, or mixed types), maximum size of numerator and denominator, whether mixed fractions are allowed, and the number of students involved; it will also allow you to put in your own problems, if you want.

The program generates easy-to-read large numbers on the screen, and a random tone ticking sound while it's waiting, as well as a bugle charge when your answer is correct. If you get a problem wrong (it informs you), the program goes into its waiting mode. It will explain step-by-step how to do the problem as long as you press a key each time to continue. One nice feature is that a correct but unsimplified answer is not counted as wrong. If you combine the fractions correctly but neglect to simplify, you are told to simplify your answer and continue.

The only problem with this

program is that the format for putting in the answer is extremely difficult. You have to identify which of the eight positions each number of your answer goes into (two for the whole number, three for the denominator). If you and your kids can get used to this, then the rest is easy and enjoyable.  
*(The Innovative Penguin of Harvey, LA 70059, could not be reached for an update. Fractional Sound originally sold for \$14.95 cassette, \$18.95 disk.)*

Mary S. Gasiorowski  
 September 1981

★ ★ ★

**Heathkit Assembly Language  
 Programming Course  
 The Heath Company  
 Model I**

Heathkit offers an 8080-based training course (Model EC 1108) on Assembly programming for \$50. It makes no assumptions about previous knowledge and starts with the absolute basics of digital computer programming. It is well arranged and takes everything in nice, easy steps. With previous courses I lacked the opportunity to ask questions. The Heathkit course answered those questions before I had to ask them.

The course is divided into 10 concise lessons. Chapter 1 is an introduction to digital computers. After completing chapter 1 you are started on your way to building a working Assembly program. Each chapter introduces new instructions, explains how they work, and ex-

actly what they can do. They are then incorporated into the program you are building. You are led along the logic trail and learn the mathematics behind what is happening without being aware of it. Along with the lessons is a workbook that is indispensable.

The meat of each lesson is the course material. Any part of a lesson that introduces a new or exceptionally important concept is highlighted by its own subsection.

In many sections, you are asked to write the required program section without help. The lesson will then show how the author would write the program, but stresses that as long as your program works it is correct. An added benefit is an excellent Assembly number base conversion program and a monitor program that simulates CPU operation so you can see what is going on in the computer.

Also, each lesson has an appendix. This is usually a listing of the program the student is working on as it should appear up to that point. The appendices also contain subroutines that are of interest in the program that is under construction. When you get the course, you also receive a sealed final exam that is as comprehensive as the course. Once you finish the exam, you can mail it to Heathkit and, if you pass, receive education credits.

There are two major problems with the course. First of all, it is not Z80 Assembly, but 8080/8085 Assembly. The mnemonics are different and a compiler that handles 8080 is necessary. Also, since you are likely to be working on a computer other than a Heathkit, you will find some incompatibilities.

The course points out what they will be, but you must find the solutions. Examples include video output and keyboard scan. The answer is the same; simply use a ROM routine or write your own subroutine.

Heathkit certainly deserves credit for publishing this fine course, but the real credit goes to the author, Willard I. Nico. (*The Heath Co., Benton Harbor, MI 49022; \$49.95.*)

Richard C. McGarvey  
December 1981

★ ★ ★

### **The Individual Study Center TYC Software Model I**

Designed for children and adults, the Individual Study Center from TYC (Teach Yourself by Computer) requires a TRS-80 Level II with 16K. TYC states that the course encourages "users to teach themselves using the computer and to improve their skills in the subject areas of their choice."

Subject tapes include American History, Geography, Mathematics, Arithmetic, World History, Languages, and Spelling. The fields of study are broken down into different levels of aptitude.

The Individual Study Center includes four cassette tapes, two of which contain six graphic activity programs, the demonstration tape, a blank cassette with a maintenance program for making your own subject tapes, and a user's manual.

The demonstration tape in the package is a "random sampling" of questions included on other subject data tapes produced by

TYC Software, and the questions are also aimed at different skill levels.

TYC says each subject tape contains 80 questions and answers. However, the demonstration tape had only 40 questions and 40 answers. Maybe that adds up to 80 for TYC's documentation department, but it doesn't for me.

With this software, you need to purchase more than one data tape or be willing to make your own with the maintenance program. Even though there are six game programs, using the original equipment data tape for all of them is tedious, since the questions and answers are always the same.

The price of the Individual Study Center is steep, since only a random-sampling demonstration tape comes with the package. Overall, the graphics and variety of activity programs in the package make the games fun to play, provided the user has enough data tapes to sustain his or her interest.

*(TYC Software, Genesco, NY 14454; \$49.95 cassette, \$69.95 disk.)*

Emily A. Gibbs  
April 1980

★★★  
**Regression II**  
**Dynacomp**  
**Model I**

Dynacomp's Regression II is, as far as I know, the only non-linear regression program commercially available for any microcomputer. This program is one-dimensional; that is, it can handle one independent variable and one dependent variable.

The Dynacomp program uses the steepest-descent method, and, therefore, is reasonably stable in its convergence to the optimal parameters. It can take a long time to converge if you choose the wrong function, the data is noisy, or there are a large number of parameters to be fitted.

The documentation that accompanies the program is mediocre. It does mention the method and algorithm used with a certain amount of completeness, but it is only moderately useful. In the space used it would have been even more educational to delineate the routines used and their relationship to the algorithm.

The documentation mentions that there is one functional form that this program is not good at regressing, but not what it is. It turns out that polynomial regression is not well regressed. The examples are run with comments and cover everything that is in the program, which serves to reinforce your understanding of operation.

The program comes on both disk and tape. I bought the tape version and adapted it for sequential disk files. It runs nicely and even has the capability of editing the source file. Editing consists of changing, deleting or inserting data points. The program lists the data by page so that it will not scroll.

In conclusion, Dynacomp Regression II is a useful, powerful, non-linear regression program despite its less-than-optimal implementation. While correct in its mathematical usage, many small, minor adjustments need to be made to the program. Even so, it is the

only program of its kind currently available, and is well worth the asking price.

*(Dynacomp, Pittsford, NY 14534, could not be reached for an update.)*

Bruce Douglass  
September 1981

★ ★ ★

**REMASSEM-1**  
**An Introduction to TRS-80**  
**Assembly Language**  
**Programming**  
**Joseph E. Willis**  
**Remsoft Inc.**  
**Model I, II, or III**

I have just finished the REMASSEM-1 ALP course, and now feel that I am on the road to becoming an Assembly-language programmer. This course consists of eight cassette tapes, a text book, and a manual. Five of these tapes contain 10 40-minute lessons by Joe Willis. His lecture is leisurely, detailed and starts with descriptions of the basics—binary, octal, and hex number systems. His occasional humor is a welcome relief from what at times can be a difficult subject. As usual, a lot of important information is given verbally by the teacher, and is not on display, so taking notes is necessary.

The display cassettes are of very good quality. All the displays, except the example programs and the last lecture, are in large characters (32 per line). You can even direct the display to start over at section headings in a menu without reloading the tape. Also included is the third printing of Barden's *TRS-80 Assembly Language*

*Programming* book as the text.

The course manual describes how to load and run the display tapes, and has flowcharts and source Assembly-code listings for two programs. The back part of the manual contains 13 pages from the *Zilog Z80 CPU Technical Manual* listing the instructions in grouped form, handier than the alphabetic listing in the Radio Shack Editor/Assembler book.

To key in programs you will need an editor/assembler that uses the Zilog instructions, and T-Bug or DeBug (not supplied). The final summary is full of excellent design suggestions, such as forming a design before coding begins. Willis' list of common errors should be in front of you while you code and while you debug.

I benefited considerably from this course. It seems to be about the most painless way to learn Assembly-language programming. I think that for the beginner this is a good way to start, and it provides a good foundation.

*(An improved REMASSEM-III is sold by Remsoft Inc., 571 East St., Euclid, OH 44119; \$74.95 cassette, \$79.95 disk.)*

Jim King  
December 1981

★ ★ ★

**Space Waste Race**  
**Storybooks of the Future**  
**Model I or III**

Space Waste Race is an interactive computerized storybook. Based on the idea of sending our excess garbage into space to be rid of it, the program

is extremely interactive, and is a novel idea.

The program has good and bad points. The graphics capability of the TRS-80 is not as fine as an Apple or Color Computer.

The program allows for a good deal of manipulation. The complete story can be viewed without interruption, or individual pages can be repeated as many times as you like. You can practice moving the moons. By pressing the arrows or bracket keys, the face of the man in the moon appears to shake, nod, or look around.

The program suggests that the program is designed for nonreaders, but the nine-line story includes such vocabulary as "collecting," "jealous," and "imagine." In addition, the on-screen instructions are beyond the young reader. Perhaps the author intended an adult to read the story with a child.

After a cute title page with graphics and a musical theme, the program goes to the menu: Storybook; Look, Nod & Shake; Funtime; and End. Storybook and Funtime proceed to other menus. Storybook allows you to run through the story—the whole thing, individual pages, just the text, or just the title page. Look, Nod & Shake lets you play with moons, making them look in various directions.

All the games reward equally for correct and incorrect responses; the moon nods for correct answers, and shakes its head from side to side for incorrect ones. If there was more action for a correct answer, the child might be more inclined to find the right response.

Space Waste Race could be improved in many ways—the games could be more developed and better designed, a more age-appropriate story could be chosen, and the program could be friendlier by asking for your name when you start. However, the idea is an interesting one. Space Waste Race may be the Storybook of the future, but it will never replace parents reading to their children.

*(Storybooks of the Future, 527 41st Ave., San Francisco, CA 94121; \$19.95 16K cassette, \$24.95 32K disk.)*

Mary Gasiorowski  
May 1982

★ ★ ★ 1/2  
**Step-By-Step  
Program Design Inc.  
Model I**

Step-By-Step is a Basic program that teaches you how to program in Basic. It has outstanding documentation, and it begins with the assumption that the student knows only how to turn on the TRS-80.

The course is divided into 10 two-part lessons. From a simple Print "HI", through arrays and graphics, to complex programs, all Level II commands and statements are exercised.

The course is for anyone who wants to program effectively in Basic, and it's the kind of educational programming that personal computing needs more of. *(Program Design Inc., 11 Idar Court, Greenwich, CT 16830. Step-By-Step is no longer available.)*

Rod Hallen  
February 1980

## GAMES

★★★★

**Adventure  
Softwin Associates  
Model I**

Gordon Letwin of Softwin Associates has produced a rip-snorter version of Adventure that is almost indistinguishable from the original. A player explores a giant cave, finds treasures, and battles knife-wielding dwarves. The computer is directed with one- and two-word commands. Part of the fun is finding out what commands the computer accepts and when it accepts them. Adventure cannot be learned in a few minutes, played in half an hour, and forgotten in a week.

The game runs on a 32K, single-disk TRS-80. Almost every move causes a quick search, but the delay is insignificant—rarely over a second. The specially designed DOS seems to occupy only one sector, leaving the rest free for program and data. Because of the lengthy playing time, a save feature allows players to suspend the game for later resumption.

Adventure is attractively packaged and comes with a clearly written instruction book. This is an exceptionally entertaining game, and it should enjoy a steady increase in popularity.

*(Adventure is sold by Softwin Associates, a division of Microsoft, 10700 Northup Way, Bellevue, WA 98004; \$29.95.)*

John Warren  
June 1980

★★★★

**Air Flight Simulation  
Instant Software  
Model I**

Air Flight Simulation turns your monitor into the control panel of a small aircraft, and you must fly the plane completely by instruments. You can even decide what type of flight plan to follow.

Unlike many programs, this one requires a fair amount of thought. Avoid the temptation of running it first and reading the directions later. Read the directions first; you'll enjoy it much more.

The program monitors altitude, attitude, air speed (including overstressing the engine), and land miles traveled. The more I ran the program, the more I liked it. This is a program that should be run—not talked about.

*(Instant Software, Elm St., Peterborough, NH 03458; Model I and III, 16K, \$14.95.)*

Stephen F. Nowak  
April 1980

★★★★

**Alcatraz II  
Spectral Associates  
Color Computer**

Alcatraz II is a beautiful example of the improvements in the quality of the products for the Color Computer today. It is a prison-escape game, where by using a combination of skill, strategy, and a lot of luck, you might get out. As they say in the introduction, no one has ever escaped from Alcatraz II. The game features numerous wings

(or cell blocks). Your score increases with each one you make it through. Each wing, starting with wing number 2, becomes progressively harder than the one before it.

The three locked gates between you and your exit point from the wing are your least-serious problem; more serious are the laser cannons. Guards constantly wander down the halls opening gates as they go. When they come to an intersection, they make a random decision as to which way to go. If they find themselves in the same hall you are in, they will pursue you, opening (and leaving open) any gates in their way. If they catch you, the game is over. The guards are not too smart, but they outnumber you.

The fourth wing has no guards. It is a man-eating minotaur's den. A minotaur is bigger and faster than you are. Its lair has walls and force fields that only minotaurs can go through. When a minotaur crashes through a wall, he destroys it, providing a path you can use later.

This is one game you won't master in two or three hours. It is a very challenging, well-written game that won't cost you an arm and a leg. The game is a combination of medium- and high-resolution graphics. (Your character is a small block containing a stick figure as are the guards and minotaurs.) You move your character with either the arrow keys or joysticks. The color is great and the sound effects are super.

There are some irritating points you should be aware of. I picked up an I/O error on my first attempt to load the game

and had to load one of the three back-up copies. It was an irritation, but I was grateful the company sent more than one copy of the program. Also, if the program loads, then errors after typing run, type run again and it will probably be fine. If it errors again, turn your computer off and back on again, type PCLEAR 2, reload the program, type run, and it should go. This bug is not in the program but in Radio Shack's Extended Basic ROM version 1.0, and it can crash a program.

Another thing: when you make it through a wing, you are given a score that varies from game to game. I haven't been able to tell why it varies, or how to get the top score. It's no big deal. I just can't figure out why my wife gets 600 points for going through the first wing and I get only 450.

I highly recommend the game, and for the price it's a steal. (*Spectral Associates, 141 Harvard Ave., Tacoma, WA 98466; \$11.95 cassette, Extended Basic, 16K required.*)

Rich Petty  
August 1982

★★★★

**Alien Defense**  
**Soft Sector Marketing**  
**Model I and III**

Alien Defense is based on the arcade game Defender, and is just as interesting, challenging, and addictive.

The game features different types of aliens including: the lander is worth 100 points and easy to shoot down; the mutant, worth 150 points, bounces

around quickly requiring fast reflexes to destroy it; the cruiser (one of the most dangerous aliens) is worth 200 points and can follow your player at incredible speed. The 250-point bomber is simple to hit, while the pod is worth a full 1,000 points. The pod has the nasty feature of breaking into six smaller aliens known as swarmer, which are 150 points each but very difficult to hit. You can also rescue a man captured by a lander.

The Model I version uses letters and symbols to indicate your ship and the different aliens. But the Model III version uses the special character graphics capabilities, which are very appropriate and much more realistic. The man looks like a man rather than the letter "I." The other Model III graphics are excellent. The moving mountains below the spaceship are very well done and the laser fire from the ship is a smooth, straight, thin line.

Sound is also a big plus. Never before have I heard so much sound utilized, and the variety of sounds within the program make it more exciting. The sounds have a very human tone.

The controls are not easy to master. They include the numbers 1 through 3, the numbers 8 through 9, and the Enter key and space bar. I found it easiest to manage by using my left hand on the regular keyboard numbers (1-3) and my right hand on the keypad numbers (8-9). From this position I can access both Enter and the space bar.

I highly recommend Alien Defense to Model I and III owners. The fast action, exciting graphics, and sometimes

humorous sounds add up to a lot of fun.

*(Soft Sector Marketing, P.O. Box 340, Middlebelt, Garden City, MI 48135; \$15.95 16K cassette, \$19.95 32K disk. Alien Defense is now joystick compatible.)*

Tim Knight  
August 1982

★ ★ ★

### **Android NIM 80-NW Publishing Co. Model I**

NIM is an old game that presents the players with three piles of objects. Each player in turn can remove as many objects as he likes from any one pile. The player who removes the last object wins.

Android NIM is played by the standard rules, except that it uses androids (robots) as the objects. At the start of the game, three rows of androids are displayed along with an executioner for each row. When you indicate the row and number of objects to be removed, the executioner looks down the row, counts the androids, and shakes his head either yes or no to indicate if the move is possible.

If it's no, you have to change the number. If it's yes, the executioner looks down his row again, raises his phaser, and blasts the required number of androids into stardust.

Once you figure out the strategy, you can win the game every time. However, it is well worth the price just for the graphics. Not only do the executioners shake their heads, but the android's arms, heads, eyes, and ears are constantly in mo-



tion. This is one game program that you should have just for demonstration purposes, even if you're not interested in playing the game.  
*(80-NW Publishing Co., 3838 S. Warner St., Tacoma, WA 98409. Android NIM is no longer available.)*

Rod Hallen  
 February 1980

★ ★ ★ 1/2

**Ants**

★ ★

**New Starship Voyages**

★ ★ ★ 1/2

**Parsector 8**

★ ★ ★ ★

**Space Ace 21**

**Synergistic Solar Inc.**

**Model I and III**

No matter how good a game is, I am not likely to play it more than a couple of times if I have to struggle through a CLOAD whenever I want to relax. Synergistic Solar has finally answered my pleas: their new game package can be transferred to disk after a successful load.

The weakest offering of the lot, and the only one I cannot recommend, is another Trek game called New Starship Voyages. This is a fairly good program, but it has been done before, many times, many ways.

It is a testament to the programmer's skills that when I loaded Ants I ended up frittering away all of yesterday afternoon and two hours today on a "silly" game.

Why did I spend eight hours

playing Ants? Well, it is a visually entertaining game. There are eight distinct types of ants (two sides each having drones, workers, soldiers, and guards) in addition to the immobile queens. The mobile ants scurry about on the screen in a charming, and really quite apt, parody of the way real ants move. The sound effects are helpful, too.

Ants ranks high on the "mindless entertainment" scale, being a game that you will not have to take notes for, that you can expect to finish in 10 minutes, and that stays busy. But there is more to Ants than entertains the eye.

A serious deficiency in the game is that the TRS-80 graphics/letters combinations are really not up to displaying eight different types of ants; the screen is so confusing at first as to be meaningless to someone new to the game. Late in the battle, when the screen is crowded with ants, the display often lapses into incomprehensibility. Also, the game moves too fast.

For something entirely different, Space Ace 21 is an accurate, complex simulation of ship-to-ship space combat. It is definitely not an arcade game, nor is it simple. Without a computer to keep track of the minutiae, I expect that it would be impossible.

The first order of business in SA21 is ship design, which is a game in itself. Your spaceship is designed on a 3 by 7 grid. Each space can be filled by one of 12 different types of structure—engines, fuel tanks, sensors, missile pods, armor, the bridge, and so on. This lets you experiment with different ship designs, from multi-engine ships

with no armor but ferocious acceleration capability to slow, super-strong tortoises that can blast any ship they meet, or anything in between. Even if this game did not perform well in its other areas I would enjoy it for the ship-building alone.

The simulation is played in 10-second turns. Movement is strictly Newtonian—once your ship starts moving at a set speed and in a particular direction, it will continue until you do something to change its movement.

For a neophyte, just getting an enemy ship within disruptor range is an achievement to be proud of; for that matter, just not to run out of fuel before getting back to base is something of an accomplishment.

Space Acc 21 is a very good game and an excellent simulation. It is one of the best computerized war-games I have seen. I strongly recommend it.

Synergistic Solar's best game, however, is *Parsector 8*. It is a space-war game intended for two players (*Parsector 5* has a computer opponent).

The game comes with a sheet of instructions and a template for making a screen divider, which allows each player to see only his half of the screen. This secrecy is the reason *Parsector 5* and *8* are such good games. Your forces are hidden from enemy sight, as the opponent's actions are from yours; the action is mostly simultaneous, so the game has an overall feel of grand strategy.

*Parsector* is played in a flat universe of anywhere from two to eight units on a side. Each player controls a mothership, which launches bases to solidify ownership of an area and

cruzers and flyers to defeat enemy units, as well as launching weapon bolts of its own. The trick is to find and destroy the enemy's bases while protecting yours, never knowing where the enemy is initially located or what *Parsector* is under his control.

*Space Ace 21* is actually a much better design and a much more realistic game, but it is not an easy game to play. You have to do a lot of hard thinking to win at it. *Parsector*, while not nearly as good a simulation, is a better game because the players can sit back and relax.

(*Synergistic Solar Inc.*, P.O. Box 560595, Miami, FL 33156. The games come two to a pack and are priced at \$14.95 and \$19.95, cassette.)

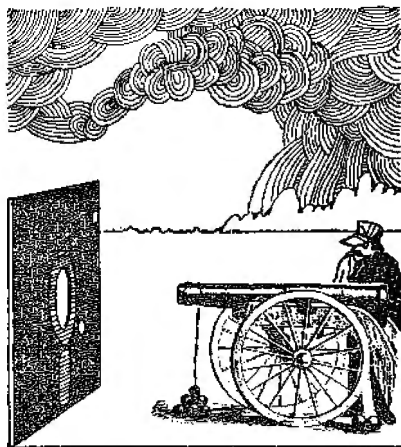
W.D. Ives  
April 1982

★ ★  
**Arcade 80**  
**Astro Mines**  
**Falling Bricks**  
**Star Run**  
**Datasoft Inc.**  
**Model I**

*Arcade 80* is a collection of three game programs.

*Astro Mines* captured my interest most. You have a fixed amount of fuel and time (about 1½ minutes) to blow up as many mines as possible. This feat requires quick reflexes, a sense of strategic planning, and a high frustration tolerance level. It is difficult enough to tax even a practiced hand.

*Falling Bricks* is similar to the popular arcade game *Invaders*, except it is your irrevocable destiny to be beamed by a brick.



The most you can hope to succeed in is to top your previous score. Although there is some element of strategy necessary, your predominant activity is to relentlessly and frantically pound the space bar, which launches your missiles.

Star Run does not work well or it doesn't work at all—it's hard to tell the difference. The graphics consist of moving letter H's for stars and a screen that whites out when you have been hit. The perpendicular sights at the center of the screen did not seem to move in response to the controls nor did the movement of the stars seem to change. However, even if the controls could be made to work, this game would not capture my interest for more than a few milliseconds.

Astro Mines is challenging and interesting, Falling Bricks is challenging, and Star Run is hopeless. A speed option, a running high-score display and more imaginative graphics would improve the first two. Children from ages five to twelve would probably enjoy playing

two of these three programs, but if you have high demands when it comes to computer games, think twice before purchasing Arcade 80.

*(Datasoft Inc., 19519 Business Center Drive, Northridge, CA 91324. Arcade 80 is no longer available.)*

Joel Benjamin  
February 1982

★★★★

### **Armored Patrol Adventure International Model I and III**

If playing with little dump trucks were as much fun as Wayne Westmoreland's and Terry Gilman's Armored Patrol, I might be tempted to head for the front yard. Armored Patrol's command-chair perspective of the action is unique. Your mission is to destroy enemy tanks. What separates this from the ordinary World War II-type games are plasma bursts instead of conventional shells, robots in lieu of human troops, and missions taking place on an alien landscape.

The idea is to maneuver your tank throughout the area in search of enemy tanks and robots. Some of the tanks are out in the open; a good number of them are hiding behind the blockhouses where they are least conspicuous and vulnerable. Frequently, an energy-zapping robot rolls into position brandishing twin ray guns. When you are hit, it depletes your energy reserves by one unit (you begin with 20). Once your energy level falls to zero, you lose one of your tanks. (You have four tanks per game. An extra

tank is awarded at 20,000 points.)

What impressed me the most about Armored Patrol is the amount of action going on off screen. The battle area is finite—your viewing area is limited to one narrow area—but throughout the entire battlefield, tanks are lumbering to and fro between blockhouses and across the open desert. Robots pop up now and then, dispensing their own deadly form of havoc. All this creates a you-are-there feel to the game.

If the degree of action is Armored Patrol's best feature, then the 3-D-style graphics are a close second. By using the left/right controls, you are able to do a 360-degree sweep of the entire area. As your tank moves forward or backward, enemy tanks, blockhouses and robots dimensionally swing into and out of view. Distant objects appear first as specks on the horizon and gradually grow in size as you move toward them.

Armored Patrol's biggest failing is the sound routines. I think a tank battle should be accompanied by low, rumbling sounds—not the blips and whines evident here. A second (and frequently irritating) problem is that one can be shot by an enemy tank within a split-second of the game's beginning. The final and most minor suggestion for improving the program would be making the blockhouse vulnerable to your fire. It would be nice if you could demolish one (over the course of several shots) with your plasma cannon and expose any tanks lurking behind.

Armored Patrol is a solid buy for the price. Its graphics are

surprisingly realistic, and draw you into the game. It took me quite a few playings before I realized that there was a method to this interesting madness—strategy and thought are definitely in order to rack up a respectable score. Armored Patrol is an enjoyable program and should be of interest to anyone who enjoys fine arcadia. (*Adventure International*, Box 3435, Longwood, FL 32750; \$24.95.)

Silas Pike  
August 1982

### **Astro Mines**

(See Page 86)

### ★ ★ ★ ★ **Astro-Scope Sex-O-Scope AGS Software Model I and III**

Professional astrologers are using microcomputers at an increasing rate to perform the calculations needed to cast horoscopes. Casting horoscopes by hand can take hours and hand-held electronic calculators simplify the process only a little. The use of microcomputers helps provide the information fast and can also provide greater quantities of it.

Astro-Graphics Services Software has released two programs that add a new dimension to microcomputer astrology. The programs are called Astro-Scope and Sex-O-Scope. They both plot and display excellent natal event charts plus all the planetary aspects. More importantly, they also provide 1500 plus word descriptions of the person whose

horoscope has been cast. This is a major advancement in astrological software.

The Sex-O-Scope program gives a reading of the subject's sexual proclivities and attitudes, and is based on texts written by authors and lecturers in the field of astrology.

These programs can be used by anyone, which makes them unique. I've run other astrology programs that produced sky maps and aspect charts, but I didn't know how to interpret the information. The new offerings from AGS Software, however, can be great fun, even for someone who has no knowledge of astrology.

Once you enter the basic information—date, time, and location of a person's birth—the computer will calculate and then display horoscope data and character descriptions. You can cast horoscopes for yourself, friends, family members, or famous people.

The text files that accompany the main program are quite large, so you'll need at least 32K and two disk drives to run these programs. The documentation booklets are some of the best I've seen, providing latitude and longitude figures for at least one city in every state of the U.S., blank charts that you can photocopy for your own use, and data on famous people that you can use to run their charts.

With these programs you can change your image from technocrat to mystic!  
(AGS Software, Dept. E, Box 28, Orleans, MA 02653; \$30 each, disk system.)

Michael Vose  
September 1981

★ ★ ★ ★ 1/2

### **Asylum Med Systems Software Model I or III**

Another triumphant tease from Med Systems Software, *Asylum* is all any *Deathmaze* or *Labyrinth* fan could ask for and more. *Asylum* is similar to earlier games, and yet unique: the game board is a maze of halls and doors, but it doesn't seem to double upon itself, as did *Labyrinth*. While the halls are more straightforward, the situations you find yourself in are not.

*Asylum* is a huge building full of locked rooms with keys just out of reach, grenades that don't explode—if you handle them right—nearly bare rooms, crazed inmates, equally crazed guards, and doctors you shouldn't trust. It is a maze of riddles and mysteries that could drive you insane.

The graphics in *Asylum* differ from those in the two earlier games. *Asylum* provides a visual relief, with graphic depictions of beds, chairs, desks, doors with bars, inmates, guards and so on.

*Asylum* commands are also different in that they are more sophisticated; there are far more commands than in the earlier two games. There are more things that you can do, and while you aren't directly interacting with the characters you meet in the maze, they can affect your situations directly.

Any game player who likes thinking games and isn't totally hooked on the graphicless adventures should like *Asylum*; it is an adventure with a twist. But be sure to bring along a

sharp imagination and a steady patience.

*(Med Systems Software, P.O. Box 3558, Chapel Hill, NC 27514; \$19.95.)*

Debra Marshall  
May 1982

★ ★ ★ 1/2

### **Attack Force Alpha Products Model I and III**

Attack Force is a machine-language version of the popular video arcade game that transforms your TRS-80 screen into a maze of hostile alien ramships and flagships. As you start each game, a convoy of eight ramships, cruising at flank speed toward your ship, appears at the top of the maze. It's up to you, firing missiles, to maneuver and outflank the ramships before they ram you.

To make the game more challenging, sidelined flagships will use their laser bolt to transform a ramship into a flagship or, to add to your frustration, into a mirror image of your ship. Firing at or colliding with the mirror image destroys your ship.

A word of caution when playing this game—you must be aware that your ship can only fire one missile at a time. Successive missiles do not fire until the preceding missile obliterates its intended target or crashes into oblivion off the screen. This can be a vital factor in later stages of the game. The point values of enemy ships increase as each wave of invaders is annihilated.

This game produces noises

that make battling aliens a delight to the ears. A little hardware rigging is required, however. Sound comes with every fired missile and each exploding alien ship, as well as with the start of every game and each addition of 1,000 and 10,000 points.

Attack Force allows one or two players on the field of battle. The disk version will store the ten highest scores.

After countless hours of battle, my calloused fingers totaled my best score at 45,560 points. Think you can beat that? Try Attack Force and you will be pleasantly surprised.  
*(Alpha Products Co., 79-04 Jamaica Ave., Woodhaven, NY 11421. Attack Force has been discontinued.)*

George Kwascha  
January 1982

★ ★ ★

### **B-1 Bomber**

★ ★ ★

### **North Atlantic Convoy Raider**

★ ★ ★

### **Nukewar Avalon Hill Model I**

B-1 Bomber, Nukewar, and North Atlantic Convoy are among the better new games released in 1980 by Avalon Hill. These new computer games are boxed in book size. Each contains precise documentation, a vacuumformed tape holder, and an Avalon Hill catalog.

B-1 Nuclear Bomber simulates a manned bomber strike into Russia. This is a navigation simulation with a nuclear

climax. There are reciprocating MIGs to consider, as well as surface-to-air missiles and many air-combat variables. There is more than ample entertainment in B-1 Bomber, enough to keep you coming back for more. Win or lose, a game summary is delivered in the end, so you have a chance to compare scores and strategies from different game times.

Nukewar is a bizarre simulation of a real-life problem that faces world leaders every day: how to provide strategic defense for a country when faced with enough destructive power to level the planet. During the Cold War, you are peacefully engaged in the building of new bases and, of course, spying on your neighbors. Any country may then declare war at any time. Negotiations for peace can ensue when the computer calls you on the hotline, wanting to make a deal.

North Atlantic Convoy Raider puts you at the helm of the mighty Bismark, pitting your battleship against the British. Fog, nightfall, and pesky British warships all conspire to bring your Bismark to her destiny of destruction. It takes a while to win, but it's worth the effort.

The best thing about the Avalon Hill games are their prices. In this day of \$49.95 boxed computer games, Avalon Hill has priced their products nominally.

*(B-1 Nuclear Bomber, Nukewar, and North Atlantic Convoy Raider are sold by Avalon Hill, Dept. C-10, 4517 Harford Road, Baltimore, MD 21214; \$16 each on cassette, \$21 each on disk.)*

★ ★ ★

**Bounceoids**  
**Cornsoft Group Inc.**  
**Model I or III**

The scenario for Bounceoids is a familiar one. You move your ship, located in the middle of the screen, by a joystick or keyboard control. Your mission is to destroy the large, floating space boulders surrounding your ship before they destroy you. This game, however, is not another version of asteroids.

Unlike the asteroid-type game, you can't penetrate the boundaries of the screen. Boulders bounce off the walls of the screen and ricochet at 90-degree angles. Boulders form groups of four or more and move in unison. The groups can become very large at higher skill levels and form a straight line that crosses the entire screen.

If these groups of boulders aren't enough to throw you there are several other features that will. One is a snake-like creature that periodically appears from the side of the screen to confront you. In the higher levels, this multi-segmented creature becomes so long it is almost impossible to destroy. Another deterrent is the appearance of alien artillery posts that fire at you. When playing in the lower skill levels they will wait a long time before firing at your ship, but as you move up in skill, they fire very rapidly. Your epitaph will be written quickly if you're not fast on the fire button.

Only the highest score is maintained on the system, and there are ten levels of playing skill for one to two players. Each player may start at a different level of play. The game is fast and

Bob Liddil  
 June 1981

should provide the ardent arcader with enough action to provide a challenge and sore fingertips.

*(Cornssoft Group Inc., 6008 N. Keystone Ave., Indianapolis, IN 46220; \$15.95 cassette, \$19.95 disk.)*

Mark E. Renne  
December 1982

★ ★ ★

### **Cave Hunter**

★ ★ ★

### **Color Berserk** **Mark Data Products** **Color Computer**

Neither Color Berserk nor Cave Hunter break any barriers in their use of the Color Computer's sound capability, nor do they startle you with graphic displays. But they are entertaining. After all, there's more to games than graphics and sound. There's action.

Cave Hunter's playing screen is a blue square maze (representing cave passages). You, the hunter, a flat four-legged green crab-like creature, are positioned at the top of the maze over an opening. At the bottom are four gold treasures. Your mission is to, one at a time, retrieve the treasures and safely deposit them outside the entrance.

Of course there's competition, not for the gold but for right of passage through the maze. Your three equally crab-like competitors work together to corner and snuff you out. When they do, your treasure is left at the scene of the snuff. Try to get your treasure as close to the top as possible so you won't have to go too far to retrieve it.

If you move your crab through one of four red boxes, called power deposits, in the maze, you become Super Crab, turn from green to red, and get to stomp the bad guys. That's what I did at first, but there's no real advantage in it. Before you know it you're green again and the bug-gers are swarming all over you. It's a better idea to use your red armor to safely carry the treasures to your lair.

In Color Berserk, the part you play is that of a human; a refreshing touch. You walk from room to room, each configured differently and with electrified walls, zapping hostile hunchbacks. They fire lasers at you; you return fire by pointing your joystick in their direction and pressing the fire button. The complication here is that pointing the joystick at your target also propels you toward it. The humpbacks dissolve slowly, so if you're too close you have to do some hasty backtracking while one goes "phiffit."

Evil Orville, a floating smily face, follows you like a demented balloon throughout the game. Just once I'd like an opportunity to blast that manic face with a laser, but Orville is impervious to attack.

The graphics in Color Berserk are better than those in Cave Hunter. Although the graphics are not special, both games do have nicely done opening displays. The sound is not very exciting, either. Both games require joysticks; one for single play and two for doubles. They are not play-once-and-shelve games. High scores take practice, and low scores are not tolerable.

*(Mark Data Products, 23802*



*Barquilla, Mission Viejo, CA  
92691; each cassette \$24.95,  
\$29.95 disk.)*

**Kerry Leichtman**  
September 1982

★ ★ ★ 1/2

**Color Pac Attack**  
**Computerware**  
**Color Computer**

Color Pac Attack is the Color Computer's version of the popular arcade game.

After a musical fanfare at the start of the game, the Pac person you control with a joystick tries to consume proton pellets around a maze while trying to avoid being caught by three muggers. The three muggers (Huey, Dewey, and Louie) start the game in a box in the center of the screen and chase the Pac person around the maze. If Pac Person gets to a proton energizer (a large pellet), it gives you the power to attack, but only for a few seconds. During this short time, the muggers turn blue and white and flee from the Pac person; but with your incredible energy you can travel faster to catch them. Catching the muggers gives you 200, 400, or 600 points and sends them back to the box in the center of the screen. You are chased around the maze as you consume the proton pellets, energizers, and muggers until you're caught three times. If you clear the screen of pellets and energizers, you're rewarded with another full screen and you start again.

If you just play for points, you need to know about the apple. The apple sometimes appears on the screen below the muggers'

box, but only for a few seconds. If you consume the apple, you get a 300 point bonus. Current score and the high score during any one game are displayed at the top of the screen. The game has three skill levels: easy, hard, and tough. The higher levels of skill speed up the pace of the characters and also limit the number of proton energizers around the edge of the screen. The easy level starts with 10 energizers; the hard and tough levels have eight and four.

Pac Attack is similar to the arcade game. The pellets are larger and fewer in number but the maze is more complex, which makes this game equally exciting and challenging. The color and sound are dynamic and add to the pressure as the game progresses. If you're looking for a game to provide hours of challenges and rewards without bullets, missiles, and bombs, then I recommend Color Pac Attack.

*(Computerware, Box 668, Encinitas, CA 92024; 16K, \$24.95 cassette, \$29.95 disk.)*

Max Treece  
August 1982

★ ★ ★ ★

**Games Trilogy:**  
**Color Space Invaders**  
**Meteoroids**  
**Space War**  
**Spectral Associates**  
**Color Computer**

The Color Computer is capable of high resolution graphics and these games prove it. You need a TV with good resolution to play. All three are in machine language and require 16K. Meteor-

oids and Space War require joysticks.

Color Space Invaders is the same old game that you never tire of playing, except that the invaders wave their arms and legs, and blink at you as they drop their bombs. This game can be played with joysticks, but it's easier to use the arrow keys and space bar.

A nice feature of the display is the lack of a perceivable border around the edge of the field, and the entire screen flashes when you take a hit. The game has 16 skill levels.

Meteoroids is an Asteroid-type game with action and speed. It includes most of the features of the popular arcade game, including alien spaceships that fire back. Your ship is supposed to be controlled by the joysticks, but in this and the Space War game, the ship is very sensitive and difficult to control, and the high resolution makes it hard to tell which way you are aimed.

Space War is similar to Meteoroids, but the main object is to destroy a death star that has only one weak spot. When you destroy it, you are in for a treat, but don't be too close when it explodes, or you will go with it.

The black hole's gravitation is what makes this game fun. You can simulate the effect on the Voyager spacecraft as it catapults past Jupiter, and even get into an unstable orbit for short periods.

These games have sound and are a better bargain than most software for the Color Computer. (*Spectral Associates, 141 Harvard Ave., Tacoma, WA 98466. Color Space Invaders and Space War are available separately for \$21.95 cassette and \$25.95*

*disk. They are also available in a games trilogy with Ghost Gobbler for \$59.95 cassette and \$63.95 disk. Meteoroids is now being marketed by Tandy/Radio Shack under the name of Microbes.)*

David G. Bartlett  
March 1982

★ ★ ★

### **Combat Adventure International Model I and III**

Combat, a next-generation computer game, requires two machines to play it. It is a real-time war game within a 64 by 64 grid divided into eight sectors. The machines keep track of eight tanks, three decoys, one base and a host of armaments belonging to each player. The machines do this, however, without revealing the whereabouts of the other side until you move one of your tanks within viewing range.

Your battle display shows the area around any one of your tanks, your base, and one of your three decoys on the right side of the screen. The left side of the screen contains either a command help list or a display of any one of the eight sectors of the playing field.

Since the game is designed to be played by two physically separated people, there is a command to transmit messages to the other player, thereby allowing nasty comments to be exchanged. The game requires two complete computer systems with RS-232 and modem or equivalent.

The program, which each player must have, prompts for baud rate (from 110-9600) and

establishes communication with the other machine without communications software packages. A save feature allows both players to interrupt the session and resume the game later. A practice mode is available to help you learn commands used in the game. In it, you deploy your forces and inflict casualties upon yourself; no opposition is provided.

I have found two disadvantages with the game: there is no warm-restart capability and response time is slow at 300 baud.  
(*Adventure International, P.O. Box 3435, Longwood, FL 32750; \$49.95 cassette or disk, 32K Model I required.*)

John W. Warne  
January 1982

★ ★ 1/2

**Computer Acquire**  
**Avalon Hill**  
**Model I and III**

Have you ever imagined yourself a financial wizard, the head of a vast hotel chain involved in power struggles and stock mergers? Do you like exciting fast-paced games? Then keep looking—this game is not for you.

Computer Acquire is the computer version of the board game Acquire. It is a game of high finance—a game of strategy and luck that up to six people can play. The object is to build and expand hotel chains. At the same time, you can buy stock in any chain and try to get a majority. Merging chains will give someone a profit, with a bonus going to the majority stock holder.

The game ends when all the chains are merged or are too

large to merge. The hotels and stock are then cashed in and the person with the most money wins.

Unfortunately, the game does not take advantage of various features of the microcomputer. The lack of graphics and sound is noticeable and the screen is difficult to read. This is one game that is more fun when played in its original form.

(*Computer Acquire, Avalon Hill, Dept. C-10, 4517 Harford Road, Baltimore, MD 21214, cassette \$20.*)

Mary Gasiorowski  
November 1981

★ ★ ★ ★

**Computer Major League: The Game of Professional Baseball**  
**Avalon Hill**  
**Model I or III**

Walk into any place where aficionados of the Grand Game congregate and ask what's the best major-league simulation game on the market. You will get a chorus of replies, but if you are looking for harmony, you will find more of it on a street corner in Philly.

Now into the fray enters the prestigious game house, Avalon Hill, with not only a board version of baseball, but one for the microcomputer.

Avalon Hill's game is not for the arcade set. It is a text game founded on the baseball fan's unflagging fascination with statistics.

You enter the home and visiting teams and you are ready to set up your lineup. All 26 major league teams and their

25-man rosters from 1980 are included.

By keying in V or H you can call up the roster of the visiting or home team. By each fielder is his number, positions he plays, batting average, slugging percentage, and overall running speed. At this point, it would be helpful to know a player's ability to sacrifice, hit and run, but these ratings (along with home run, walk, on-base running, and strikeout ratings) can't be accessed until the game is under way.

Although the documentation notes that the game incorporates a player's fielding ability into the fabric of play, you don't have access to that information—definitely a drawback in the late innings when you are hanging onto a one-run lead and want your best defensive players on the field.

The game keeps a running scorecard you can access anytime to see what a player or pitcher has done to that point in the contest. Major League also contains a print routine so you can print a box score of the game.

Major League takes the drudgery out of baseball simulation. Some fans might find it exciting to roll n-sided dice and pore through cards and charts, but as for us, we would rather have the micro do it for us. After all, the cerebral part of the game is making decisions and reacting to field situations—not acquiring eye strain by reading charts.

*(Avalon Hill, Dept. C-10, 4517 Harford Road, Baltimore, MD 21214; \$25 cassette, \$30 disk.)*

John P. Mello, Jr.  
May 1982

### ★ 1/2

#### **Conflict 2500 Avalon Hill Model I and III**

Conflict 2500 is a radical departure from first generation one-ship Star Trek games. The first part of the game sets the scene. You may choose one to 10 hyperfighters for the good guys, one to 20 planet pulverizers for the bad guys, and up to 10 planets and 10 bases from which you get power. The object is to chase the bad guys attacking the planets and either fire at them or leave a mine for them to run into.

Play begins with a galactic report showing the universe as a 9 by 13 grid indicating your ships, planets, and bases. Attacks on planets or ships are reported to you in verbal messages. The grid is then redrawn, you move to one of your ships, and you get a 2000 by 1600-megameter tactical view. Each sector is 1000 square megameters. Planet pulverizers are not shown on the grid, but you may fire once at them. Any angle from 0-360 degrees is permissible. You can then set a mine for the enemy to run into or you may take off. The planet pulverizers will return fire on you or your planet. To move, you must indicate heading and velocity. You have to change direction and velocity every move or your ship will miss its target.

The game is well-documented and the instructions are clear and concise; however, playing the game is very dull. There is no real time. Graphics are slowly drawn set-reset graphics and redrawn each time you change ships. If you play the game one-

on-one, the computer's ship is not drawn and you must wait for it to appear; or you can search 117 sectors looking for it. If you use multiple ship scenarios the play becomes intolerably slow as individual ship information must be fed in and all previous information neutralized. A summary at the end of the game indicates the outcome based on fleets, bases, and so on.

All in all this type of strategy game needs more excitement or should be left to pencil, paper, and the user's imagination. (*Avalon Hill, Dept. C-10, 4517 Harford Road, Baltimore, MD 21214; \$16 cassette, \$21 disk.*)

Mark E. Renne  
August 1982

★ ★ ★ 1/2

**Crush, Crumble, and Chomp  
Epyx  
Automated Simulations  
Model I or III**

I look for strategy in computer games, and I have been particularly pleased with the new Epyx entry, *Crush, Crumble, and Chomp*. In several ways, it is far ahead of other games I have tried.

This game is a major change of direction for Epyx, whose other games are attempts at putting fantasy-role games on a computer. This is a slightly tongue-in-cheek take-off on the grade-B monster movies in which you get to play a famous movie monster, such as *Godzilla* or *Mechismo*, and attack one of four major cities: New York, San Francisco, Washington, or Tokyo.

*Crush, Crumble, and Chomp* is a carefully designed game.

The graphics on the TRS-80 version are as good or better than anything I have seen before on the machine.

The documentation is also well done. The booklet is explicit and arranged for easy reference. It includes handy maps of the four cities, and commands for the various monsters are on separate, unbound cards for easy use.

The real victory of *Crush, Crumble, and Chomp*—the thing that makes it different from the competition—is that it is a true strategy game. You must make decisions from before the start of the game, and your success will in large part be determined by those decisions. You can play one game one way, and the next so completely different it may not seem like the same game at all.

Your decisions start when you set the parameters of the game. You decide which monster you want, and each has its own capabilities and weaknesses. Then, you pick the city you will play with. Finally, you determine how your performance will be scored by determining your primary mission—to destroy buildings, eat, kill combat units, or survive.

Epyx gives you some help in forming your strategies. The degree of difficulty is another issue that Epyx has considered carefully. Games can't be so easy that they are immediately conquerable, nor so hard that they discourage people from trying.

*Crush, Crumble, and Chomp* solves this problem and the question of giving the player enough variety to keep him interested game after game with

its variety of choices of monster versus city versus objective. Some of the choices are very difficult to succeed at.

This is not to say that *Crush, Crumble, and Chomp* is a perfect game. It runs very slowly. Movement is taken in turns with your monster moving first and then the other units—crowds, police cars, military units, and so on—moving one at a time. Each turn can seem to take forever.

I consider *Crush, Crumble, and Chomp* to be a must for the serious microcomputer gamer's collection, and I hope it is an indication of better things to come from Epyx as well as their competitors.

*(Automated Simulations, P.O. Box 4247, Mountain View, CA 94040; \$29.95.)*

Bert Latamore  
February 1982

★ ★ ★

### **The Datestones of Ryn Automated Simulations Model I and III**

Datestones, the most precious treasure of the Duchy, have been stolen by the brigand Rex and hidden in his mountain lair. You, as the player of *Datestones of Ryn* (a.k.a. Brian Hammerhand, mercenary at large), must retrieve and return the stones to the grateful Rynians.

As the game begins, you find yourself in a room at the cave's entrance. The game is timed; you have 20 minutes to roam about the cave, find and recover the datestones, and defeat or successfully flee from the cave's many horrors. At any point you may encounter and be chal-

lenged by a beastie: robbers, centipedes, wolves, spiders, skeletal bats, and slime. Some of the critters are stationary, others move about and pop up unexpectedly.

Scoring is based on the number of stones recovered and brought outside, the number of foes vanquished, how long you survive, and whether you escape from the cave at the end of the 20 minutes.

Computer response to fight or flight commands is sometimes too slow to avoid impending disaster. It's very frustrating to watch yourself being engulfed in slime while you keep punching the button that should get you out. Conversely, I have watched beasties die without my lifting a finger to aid in their demise.

The game is limited in scope and area, but it is a microquest rather than a full-fledged *Dunjonquest*, and you are fully warned of this distinction by *Automated Simulations*. It is best viewed as an introduction to their larger fantasy games. *(Automated Simulations, P.O. Box 4247, Mountain View, CA 94040; \$19.95 cassette and disk.)*

Debra Marshall  
April 1981

★ ★ ★ ★

### **Deathmaze 5000 Med Systems Software Model I and III**

I've had it! I'm sick of it! I hate it! . . . Well maybe I'll try *Deathmaze* just one more time.

This has got to be one of the most infuriating, aggravating, irritating, frustrating, angering, spellbinding games on the market.

I'm not going to describe it, except to say that it's different from any other adventure game I've seen so far. Just thinking about it makes my blood boil and my teeth chatter. One staffer here actually dreams of its corridors and doorways.

The game comes complete with practically no directions at all, but it does have a warning: "Be patient. You will not solve Deathmaze during the first week. Or the first month."

They are right! My boss is making dire predictions about my career when he's not making suggestions about solving the puzzle. I'm getting desperate. I can't make it past the first level. I've starved to death at least 50 times in the past month. Please, please, doesn't someone out there know where to find food and how to get out of the seventh room? Where's the door to the second level? Does anyone have a magic word to activate the junk I've been carrying around? Do I really want to put myself through five levels of this agony?

Ah, but such enticing, interesting agony!

Maybe I should try. . .  
(Med Systems Software, P.O. Box 3558, Chapel Hill, NC 27514; cassette \$14.95, disk \$17.95.)

Debra Marshall  
February 1981

★ ★ ★ ★

**Defense Command**  
**Big Five Software**  
**Model I and III**

Incredible graphics, sound, and voices add up to fun in Defense Command. The game is in machine language and loads

with the System command. The programmers made sure that if the program is run on a Model III, it takes advantage of special characters. The Model I and Model III versions are identical and they both check for the model type to use the applicable features of either machine.

The game is unlike any arcade game I have seen. The player is placed at the bottom of the screen, and is guarding highly important Krotnium fuel cells in the Kromforkrom galaxy. A whole slew of nasty aliens comes down and tries to steal the fuel cells from you.

You are transported to the area by a mother ship. Several other ships escort you down, including a flagship, which is worth a lot of points later on. After transporting down, you wait for the aliens to come, or you take a couple of shots at the aliens who escorted you. If you accidentally hit the mother ship, it sends down solar-energy crushers, which will inevitably smash you.

Once the aliens start coming down, you must dodge around trying to shoot them. There is also a dangerous slicer that will fly down, ignore you, and slice up your fuel canisters. You must kill the slicer before it gets to the bottom of the screen, or you lose.

You have another defense against the aliens: the anti-matter bombs. Any number key destroys every alien on the screen, but you only have four of these bombs. You receive an additional bomb with every 5,000 points, and an extra ship for every 10,000 points.

One of this game's most impressive features is the voice,

The voices are clear and loud and are far superior to the old Robot Attack game. They even change pitches (both squacky and low voices) and are present during different points of the game.

Defense Command is a superb game and an original idea. The sounds, graphics, and voices are excellent. Instead of being an imitation of some other arcade game, it is new and a great deal of fun.

*(Big Five Software, P.O. Box 9078-185, Van Nuys, CA 91409; \$19.95 disk, \$15.95 cassette.)*

Tim Knight  
August 1982

★ ★ ★ 1/2

### **The Dragons of Hong Kong**

★ ★ 1/2

### **Local Call for Death**

★ ★ ★ 1/2

### **On His Majesty's Ship Impetuous**

★ ★ 1/2

### **Two Heads of a Coin Adventure International Model III**

Interactive Fiction presents a person with the framework of a genre novel and lets him be a character in the book. (If you're a woman, forget it. The programmer's warning in *The Dragons of Hong Kong* applies to the entire series: "This story has a rather male perspective. We suggest you adopt a male outlook and a man's name.") As the plot thickens, you write your own dialog and make decisions influencing the outcome of the

story. In the first two offerings—*Local Call for Death* and *Two Heads of a Coin*—there is only one outcome: solving the mystery at hand. In *His Majesty's Ship Impetuous* and *The Dragons of Hong Kong* there are at least three possible endings.

*Local Call* takes place "a few years after the Great War." You are called on to assume the role of a reputed American detective in England rubbing shoulders with the crust of London society. Programmer R. Lafore offers some nice touches of high living here: You dine on Filet de Boeuf Strassbourgeois, drink 1913 Chateau D'olgt de Pied, and tool through the city on the Thames in a Hispano-Suiza, one of the finest automobiles ever made.

Although the plot of *Local Call* is engaging (an apparent suicide tickles your deductive instincts), in many cases characters fail to interact with you. On several runs of the novel, when financier Major Wormsley asked Alec Deepgrave (this reviewer's persona) his occupation, responses such as a brain surgeon, injection mold engineer, grave robber and gigolo triggered the same response from the major: "Oh, I see." When interaction becomes more complicated in *Local Call*, the give and take between characters can become nonsensical.

On the procedural level, the documentation for converting *Local Call* to Model III format was fine until it explained building the DO files. The docs left out a reminder to take the Model I version of the game out of drive 1 before you create the files; otherwise, you will get a CRC error.



In the second "novel" in the series, *Two Heads of a Coin*, James Conway, a middle-aged, well-groomed, prosperous businessman, enlists the help of a Sherlock Holmes-style detective to find his missing wife Georgina. However, *Two Heads* also proves to be irresponsible. Not only does it serve up the old reliable "I don't understand your meaning," but sometimes resembles the proverbial broken record.

Both *Local Call* and *Two Heads* have a drawback inherent in their genre. Once you solve them, there isn't much point in reading them again. But that's less of a drawback for a \$3 mystery than a piece of software.

On *His Majesty's Ship Impetuous* and *The Dragons of Hong Kong* are more ambitious and challenging than their predecessors.

In *HMS Impetuous*, you must not only pilot a ship-of-the-line through enemy-infested waters, but command a temperamental crew. Your troubles begin when a junior officer affronts your authority. How you handle this situation seems to determine the outcome of the novel because it has a profound effect on the behavior of your crew.

Lafore warns interactive novelists to "play it straight" to get the most fun out of the software, but outrageous behavior can yield some entertaining results in the face of the program's boilerplate ambience.

Although *Impetuous* has several endings and you influence them with your decisions, it seems you will always make it to the final chapter where your ship-of-the-line confronts an enemy armada of 40

ships, most of them superior in firepower to yours. You can't cleanse every sin you've committed to that point, but you will always salvage a measure of glory if you exhibit a stout heart.

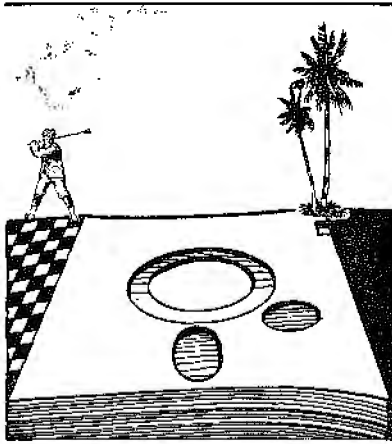
That isn't the case in *The Dragons of Hong Kong*. It is as Byzantine as a good adventure game full of traps and dead ends. If you're really lily-livered, you may even end up as an accountant and nothing of interest ever happens to you for the rest of your life. Unlike its predecessors, *Dragons* makes you feel you're interacting with something more than a thinly disguised adventure game—not only that you're interacting but also creating.

In *Dragons*, Lafore's prose walks the line between spoof and parody, giving the "novel" the flavor of the hardest boiled fiction:

"Not even a real estate agent would claim Big Al's Bar had a prestigious location. An iron mill crowds up on one side, and on the other is the kind of transient hotel guests tend to leave feet first.

"You cross a railroad siding and push open Big Al's door. The welterweight match on the tv is making a nice counterpoint with the jukebox blaring 'Born to be a Redneck,' and the smell of stale beer probably couldn't be cut with hand tools."

*Dragons* and its predecessors don't quite live up to the hype Adventure International advances in their cause. Even Lafore, quoted in *Omni*, admits that "The interactivity in these stories is now somewhat limited." But as *Dragons* indicates, the series is in evolution and getting better all the time.



*(Adventure International, Box 3435, Longwood, FL 32750; \$29.95 disk, 32K required.)*

John P. Mello, Jr.  
August 1982

★★★

**Draw5 and Stud5  
Wilson Software Division  
Model I and III**

The most unique thing about Draw5 is that the computer plays six hands against you—and the six hands all play against one another even after you have dropped out of the round. Each player that the computer simulates has a name and his own individual style of play. One player might consistently bluff while another will never bluff, and some of the players bluff at random. After a while each of the simulated players takes on its own individual personality—it's almost uncanny.

And if that isn't enough, after a few hands, the other "players" begin to figure out your style of play and alter their playing styles to counter your style.

Draw5 is played according to standard Gardena Card Club rules (a pair of Jacks or better to open, and so on) and the program doesn't allow cheating. If you can't open, it automatically passes you.

Stud5 is identical to Draw5 in concept except that it plays five-card stud poker. Stud5 is as well designed as Draw5, except that I enjoy playing draw poker more than I enjoy playing stud poker.

I find the simulation completely engrossing and a good buy. Draw5 has definitely sharpened my playing skills—anyone for a friendly little game?

*(Wilson Software Division, 539 Springhouse Lane, Camp Hill, PA 17011; cassette, Draw5 \$20, Stud5 \$15.)*

Lloyd Martin  
January 1982

★★★½

**Eliminator  
Adventure International  
Model I and III**

You are the pilot of the star-craft Eliminator. The controls of your ship are simple to learn, but difficult to master; to defeat the enemy, you must have split-second control of your ship.

Your goal is to keep the aliens from capturing your energizers and taking them to the top of the screen. The energizers are placed on gantry towers throughout the planet. If you destroy the first wave, another wave appears after you receive bonus points for remaining energizers. There is no end to the villains until you lose all your ships or all the energizers are captured.

The enemy comes in assorted

degrees of difficulty. Disruptoids are the most common and the easiest to destroy; they score 150 points. Disruptoids hover around the energizers waiting for a chance to kidnap them. The second type of alien is a drone; although they don't fire at you, their purpose is to plant aerial mines. If you hit a mine, your ship is destroyed. They score 250 points. The final enemy is a tracer dispersal unit, or TDU. They are harmless to your ship and score 1,000 points. After being touched, five tracers are released; each tracer is worth 150 points. These tracers follow you and each one packs a deadly force. With practice, you can score extra points by destroying a disruptoid carrying an energizer.

Screen information includes score, ships, and bombs remaining, and current record high score. At the top of the screen is a long-range scanner that covers the entire planet. It indicates the location of the aliens and if any energizers are being picked up. Sound is available through the cassette AUX plug.

To score points, manipulate the controls as if they were an extension of your hand. It's imperative to eliminate the first wave without using any disruptor charges. You need them more during the later waves. By saving the tracer dispersal units for last, you can eliminate the tracers with greater ease. Rapid fire is not advised: you usually end up destroying as many energizers as aliens.

The game instructions are clear and include loading instructions for all computers. Eliminator is fast, challenging, and saves enough quarters in a

month to pay for itself. Graphics and sound are superb, just as advertised. Eliminator is one of the first arcade games to come from Adventure International and certainly not the last. (*Adventure International, Box 3435, Longwood, FL 32750; \$19.95 tape 16K Model I or III, \$24.95 disk 32K Model I or III.*)

Mark E. Renne  
August 1982

★ ★ 1/2

**Everest Explorer  
Acorn Software Products Inc.  
Model I and III**

Everest Explorer starts you in Katmandu, the traditional kicking-off point for Everest expeditions, with an operating budget randomly selected from a low of \$80,000 to a high of \$275,000. You spend this money on climbers, Sherpa guides, tents, oxygen, food, and fuel.

The first challenge of the expedition is to find a route through the Khumbu Glacier to the site of camp 2. You must always be aware of weather conditions as well as the health status of your climbers.

Once you establish camp 2, the climbing instructions become more cumbersome. You must assign each climber a destination, and also decide what each is to carry with him. You must make sure that each camp has sufficient supplies for the expected climbers. This is complicated by the lack of a unified listing that could show how many climbers will be in camp the next night and what they will be bringing with them. To make up such a list, you will

have to go through three of the four data banks listed in the menu and take written notes.

If you enjoy dealing with the intricacies of logistics, you will love Everest Explorer. If it's adventure you're looking for, I suggest you save your money and head for the nearest mountain.

*(Acorn Software Products Inc.,  
634 North Carolina Ave. S.E.,  
Washington, DC 20003; \$19.95  
32K disk or 16K cassette.)*

Bert Latamore  
August 1981

### **Falling Bricks**

(See Page 86)

★ ★ ★ ★

### **Forbidden City Fantastic Software Model I and III**

"Arrgh, I'm dead!" my faithful amplifier spits at me in spite. "Play again?" asks another voice. I scream "NO" and hope this time the robots don't bump me off so quickly. That's right, Forbidden City is a talking adventure system. Not only does it have sound effects, but the game has three separate voices with gender and emotion.

The adventure is in machine language on a self-booting disk and requires 48K. Standard verb/noun commands are accepted and the split screen is used for display. The game has a vocabulary of over 140 words for input and a separate vocabulary for spoken words. Game saving is easy and up to 10 games can be saved at one time with comments for each.

What sets this game apart

from most adventure games is the voice. The voice is not meant to help or guide you through the adventure. It does, however, dispense critical information at times. The game can also be played with the voice off without missing any clues.

The voice of the city computer is female; she has an alluring voice and is usually looking for information such as passwords. The voice of your computer is pleasant. It excitedly says "I found something" if an exam item is fruitful. It also sounds a little sick after radiation poisoning; you know death is certain without treatment. The last voice belongs to the robots guarding the city. Once you're spotted by guards they chase you shouting "Stop Alien," or "Stop Intruder." It's only a matter of a few turns before they kill you. I have not figured out how to deactivate them yet.

The robots make the game challenging and difficult. The three voices are a definite plus; their words are very clear and their emotions are convincing. Adventures may never be the same.

*(Fantastic Software, P.O. Box  
27734, Las Vegas, NV 89126;  
\$39.95 disk, requires 48K.)*

Mark E. Renne  
August 1982

★ ★ ★ 1/2

### **Galactic Empire Broderbund Software Model I**

Galactic Empire is a 16K, Level II strategy game that makes you the commander of the planet Galactica's forces. Your mission is to conquer the

20 worlds of the Central Galactic System within 1000 years. Your forces are fighters, transports, and scouts, and your method is—brute force.

You move from system to system, conquering when you can. All your actions must be carefully planned if you expect to conquer the galaxy in a mere thousand years.

Although Galactic Empire is well designed and fun to play, there are some problems. This is a long game—one session can easily last from 4–6 hours.

The good points outweigh the bad. Galactic Empire is never dull. The continuous changes in layout and planetary characteristics present you with different problems each game. The attacks can be full of surprises; you can lose a high-probability game if you aren't watching carefully. You can also pull off a victory from what seems like a hopeless position if you have the courage to fight on.

*(Broderbund Software, 1938 Fourth St., San Rafael, CA 94901. Galactic Empire is no longer available.)*

Dan Cataldo  
August 1981

★ ★ ★

**Galaxy Invasion**  
**Big Five Software**

★ ★ ★

**Invaders Plus**  
**Level IV Products**

★ ★ ★

**Invasion from Outer Space**  
**The Software Exchange**  
**Model I and III**

Invaders Plus is a 2.9K machine-language program that

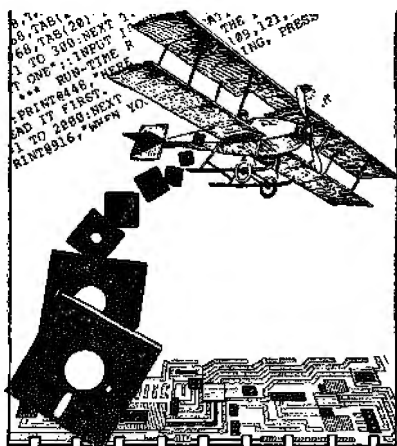
presents the player with 49 aliens of two types arranged in seven rows of seven aliens each. An additional alien continuously zips across the top of the screen. The animation of the aliens is effective—they seem to take little steps in their march across the screen, dropping missiles on the defending base. Five gas clouds protect the bases, at least until they are eroded away by the alien attack, or from below by the missiles fired from the base.

I have a few minor complaints. The base moves right and left so quickly that it is hard to control. It isn't possible to fire at the aliens while the base is moving. There is no ongoing display of how many bases remain.

Invasion from Outer Space is a 16K machine-language program. It presents you with 105 aliens of only one type, arranged in seven rows of 15 aliens each. The aliens aren't animated—they just move back and forth across the screen, dropping CHR\$(191)s on the defending base, which is protected by three gas clouds.

Defending missiles can't destroy descending alien missiles. Sound effects are appropriate to the action. The score is incremented for each alien destroyed, and there is no penalty for missed shots. The game continues until five bases have been destroyed.

My complaints are minor ones. The previous high score is not displayed. The CHR\$(191) used for missiles by both the aliens and the bases seems too big. And sometimes when you press the space bar it doesn't fire a missile; presumably the CPU is updating the display, and not monitoring the keyboard.



Galaxy Invasion is a 9.2K machine-language program that presents 40 aliens of four different types, arranged in four rows of 10 aliens each. An additional six aliens in two groups of three provide an upper fifth row.

Animation of the aliens is superb. Not only do they flap their wings as they march across the screen, but they attack the base by swooping down in singles and groups. They also drop missiles, represented by single graphics points, but their aim is poor.

One or two people can play alternately. For the single player, the game goes into a demonstration mode, with the computer controlling the aliens and the base. I have several complaints. The CPU is so busy with the display that there is often a delay between the space bar and the firing of missiles. That adds the frustrating element of luck to what should be a game of skill. The sound effects are the poorest of the three.

*(Invaders +, Level IV Products, 32429 Schoolcraft Road, Livonia, MI 48150; \$19.95 cas-*

*sette, \$24.95 disk. Invasion from Outer Space is sold by The Software Exchange, Milford, NH 03055, which could not be reached for an update. Galaxy Invasion, Big Five Software, P.O. Box 9078-185, Van Nuys, CA 91409; \$15.95 cassette, \$19.95 disk.)*

John T. Phillip  
August 1981

★★★  
**Gambiet**  
**Microtrend, U.S.A.**

★★★½  
**MyChess**  
**Programma International**

★★★  
**Sargon II**  
**Hayden Book Co.**

★★★  
**SFinks**  
**William A. Fink**  
**Model I**

A tournament among micro-computer chess programs—it sounds like a simple enough idea. The first hitch that I discovered is that there is much more to a computer chess tournament than determining which program wins the most games. Features such as speed, book moves, graphics, board set-up, and think-time display must also be considered.

MyChess probably plays the strongest overall game. It plays aggressively and quickly, making the most of its moves in well under three minutes. The main pros of this program are thinking on opponent's time, setting time limits and saving up to six in-progress games on a disk. The main cons are no think-time

display, poor graphics, and an inadequate board set-up option (pieces can be moved and removed, but not created).

SFinks plays the most conservative game of the four (games often stretch to 70 moves or more). It used more than the allowable three minutes per move while playing on the recommended tournament level. It seems to average about three minutes and 20 seconds per move at level 8. SFinks' standing may have been different on a lower level of play. The graphics and think-time display are very good, and SFinks' is the only program that gives an audible signal to indicate the completion of a move.

The fact that Gamblet 80 only won one game in the tournament is not indicative of the strength of the program. It plays very competent chess and all its games were hard-fought battles. Its features are definitely the most impressive. The think-time display shows the move it is currently considering, the best move it has found so far, and the number of moves left to evaluate. The chess clock, graphics, and minimum system requirement (16K Level II) are also pluses.

Sargon II, although it is over a year old, still has a few things over the newer programs. It can be played on a 16K Level II. It is tied with SFinks as the least expensive and it still has the easiest and most versatile set-up mode. Some of the new features such as printer option are lacking in good old Sargon II.

All four of these programs play excellent chess, considering their size and clock speed. Anyone from a beginner to a

fairly advanced player should be satisfied with any of these programs.

*(Sargon II, sold by Hayden Book Co., Rochelle Park, NJ 07662, is no longer available. MyChess is sold by Programma International, \$34.95, Canoga Park, CA 91315, which could not be reached for an update. Gambiet was originally sold by Microtrend, U.S.A., \$39.95, Loutsville, KY 40202, which could not be reached for an update. SFinks 3.0 for the Model I and III is sold by William A. Fink, Lighthouse Point, FL 33064, \$39.95. An SFinks Chess Tutor sells for \$19.95.)*

Randy Jenne  
December 1981

★ ★ ★ 1/2

**Gomoku**  
**Discovery Bay Software Co.**  
**Model I**

Gomoku, an ancient Japanese game, is a variation of tic-tac-toe. It is played on a 9-by-9 multigrad, and the object is to place five of your markers in a row, either horizontally, vertically, or diagonally.

In typical tic-tac-toe games, once you've discovered the pattern or key, the results are predictable. Such is not the case with Gomoku. The computer has a strategy, but I have yet to win a game.

The computer is slow in determining its moves, but that is normal for a Basic program. The graphic layout of the board and the area where you enter your moves is well done. The computer even makes remarks such as "Tsk! Tsk!," "Uh-huh," and "Chuckle" during play.

I rate a game program on how

well it holds my interest and whether it makes me want to play it again. Gomoku is the kind of game I could play for hours; in fact, I have to force myself to stop. I give it a high rating.

*(Discovery Bay Software Co., P.O. Box 464, Port Townsend, WA 98368; \$15.)*

Rod Hallen  
February 1980

★ ★

### **Haunted House Tandy/Radio Shack Model I**

Haunted House is short enough for a beginning adventure player, but needs the expertise of a professional; it is therefore not suitable for either.

The instructions are very brief and the vocabulary is also limited. After using the enormous vocabulary of Hi-Res Adventures and Scott Adams adventures, I was frustrated not only by the small vocabulary, but also by the omission of standard vocabulary such as Go, Use, and Throw.

On a positive note, the game uses a sense of humor in solutions and responses.

The object of the game is to get out of the haunted house alive. The game consists of two parts. In Part I you are on the first floor dodging levitating knives, looking for a secret passage, and getting past animated suits of armor and fires.

On the second floor, you find a magic sword and three ghosts, one to the west, one to the east, and one to the south. After destroying these three, you will

discover a ghost to the west of the western-most ghost. You quickly realize he is immune to attack. To escape, you must move around the room in certain directions to trigger one of the entrances to open.

Playing straight through, without any wasted commands, the program only takes six minutes to complete, and that includes loading time. It is suited to the beginner who will go on to bigger and better things.

In conclusion, with the game's only plus being humor, and its minuses being limited instruction, vocabulary, and short game time (mostly because of its limited memory), I advise you to save your money for 16K or larger adventures.

*(Tandy/Radio Shack, Fort Worth, TX 76102; Model I and III cassette; \$9.95.)*

David Williamson  
August 1981

★ ★ ★

### **Hellfire Warrior Automated Simulations Model I**

Hellfire Warrior is the second in the Dunjonquest series of full-length fantasy role-playing games produced by Automated Simulations (the first is the Temple of Apshai).

The object of the game is simple: Wander through the maze, collect all treasure on all levels, buy advanced magical aids, proceed to higher levels, and eventually rescue the warrior maid Brunhilde held deep in the dungeon. Exit the maze with Brunhilde, treasure, and body



intact and you win.

This game is perhaps unique in the fantasy game field because you have the option of creating your own character or bringing a favorite character from any other role-playing game with you to Hellfire to be your hero.

The game's graphics consist of single-dimensional top-view representation of corridors and rooms found in the cave you venture into. The character appears on the screen as a triangle created from graphics pixels, treasures are rectangles, and critters and nasties appear as squares of different sizes, which disintegrate into shooting rays when they are defeated.

Hellfire Warriors is available on disk (32K, TRSDOS) and cassette (16K, Level II). It is written in Basic, which makes for slow graphics drawing and reaction time. However, the game is in real time, so you can't let your attention wander.

The Book of Love is one of the most enjoyable things about this game. It is nicely illustrated, slickly presented, and makes for enjoyable reading, which is saying a lot for something that is software documentation. The directions and explanations are thorough and explicit. You should read the book completely before attempting to play the game.

Hellfire Warriors is not easy to win. The beasties change types on each level, different magic is required, and there is enough going on when the graphics aren't being drawn to keep your interest up. On the whole, any true game fanatic will find this game intriguing and entertaining. It is a welcome change from

adventure games without graphics.

(Automated Simulations, P.O. Box 4247, Mountain View, CA 94040; \$39.95.)

Debra Marshall  
October 1981

★ ★ ★ ★ 1/2

### Hypergate Centurion

★ ★ ★ ★

### Hypergate Patrol Synergistic Solar Inc. Model I or III

Synergistic has developed two very sophisticated space simulations that require only 16K and yet are stimulating to even the best space cadet.

In Hypergate Patrol, you are the captain of a very small patrol vessel wandering through the galaxy at the command of the Hypergate Centurion. In Hypergate Centurion, you control several Hypergate Patrol ships and a plasma weapon that spans the universe. The inhabitants of both games are identical, but the strategy is very different.

In Hypergate Patrol, nine quadrants and five galaxies make up your patrol universe. The galaxies are all joined in the center by a Hypergate. By entering the Hypergate at different angles, you can travel from one galaxy to another. Your performance is rated on a merit-demerit system. Killing the enemy rates five merits while killing an ally rates five demerits. You need a ratio of 7 to 1 to survive. The game is played in real time and you must complete your mission within a specified amount of cycles.

The video display is divided into seven sections, providing all the vital information. Your ship features a variety of weapons, and a call to the Hypergate Centurion is always possible if you find yourself in a tight spot.

Hypergate Centurion is a machine-language, multi-level, space simulation of the highest degree. You are buried deep within an asteroid and control 15 quadrants of space. You have a fleet of five seekers in the hanger bay, ready to be dispatched with any of five different orders.

As each object enters your jurisdiction area it is assigned a number. You can scan the area in four different ways—area, quadrant, tactical, or interrogation. This gives a variety of magnification and types of information depending on the scan.

Documentation for these two games consists of 45 pages of instructions in the form of an operations manual full of Centurion secrets. These are not arcade-type games, but simulations requiring skill, logic, and mental expertise. Even though they require only 16K, they are first class and offer a challenge that won't be mastered in one sitting.

*(Synergistic Solar Inc., Box 560595, Miami, FL 33156; \$39.95.)*

Mark E. Renne  
December 1982

★ ★ ★ ★

**The Institute  
Med Systems Software  
Model I or III**

The Institute is a unique entry into the market from the com-

pany that brought us 3-D adventures. It is not 3-D nor does it have any similarity with Asylum, Labyrinth, or Deathmaze. Graphics are used only for decoration or to signify dramatic changes in the plot. However, it's one of the finest adventure games I've ever played.

The disk version is self-booting on either Model I or III. There are 10 sections on which to save games in progress and a lengthy comment to describe where you left off. Saving and loading games is fast and easy.

At the top of the screen are visible items, if any; a description of the room or its contents is next. Your input is entered at the next level, followed by an area which displays the result of "talks," "listens," or special happenings. While "talk" and "listen" are commands that most adventures do not use, in this one they are probably the most important. That's right, you talk to characters (dwarves, counselors, midgets) and they talk back. If you don't keep your ears open, you'll never get anywhere.

Upon entering a room, you normally look about to receive more information. Sometimes a door or another exit shows up, sometimes not. To examine an item, you "look" at it and if you're lucky you will find something interesting. There are many exits that are not indicated in any manner at all; they are found by trying all directions from each room. Nobody said this game was easy.

The input routine is very critical; descriptions must match exactly or no action occurs. The clues within the game and actions required are very subtle.

Multiple "looks" are required for shelves with more than one item, also.

The game has five different levels. The first one is your starting point at The Institute. By use of a strange powder, you physically enter your dreams to complete your escape. Dreams include a giant forest on another planet, a voyage on the Titanic, an ancient temple, and a prehistoric forest. It's much like five adventures in one. Sometimes you must be killed to wake up or pass into the next dream, though if killed you keep whatever inventory you had and start over—nice feature!

This game is tough for even experienced players. I enjoyed the changes in syntax, screen design, and vocabulary. The "talk" and "listen" features add a whole new spectrum to play; the graphics are well placed and serve a very worthwhile purpose. For a 16K adventure, The Institute is one of the best. (*Med Systems Software, P.O. Box 3558, Chapel Hill, NC 27514; \$21.95 disk, \$19.95 cassette.*)

Mark E. Renne  
September 1982

★ ★ ★

**Interlude: The Ultimate Experience**  
**Syntonic Software Corp.**  
**Model I**

Silicon pundits have told us for a long time that computers will play a major role in our changing lives. We will see the advent of electronic mail, electronic money, even electronic crime. It seems to me that if the

devilish device is ever to be mainlined into our culture, it must service more basic needs than game playing and accounts payable. Something really basic, for example, like sex.

The time has come. The era of home computer sex is finally at hand, brought to you by Syntonic Software, a forward-looking subsidiary of Software Technology Inc. These people are not above pandering to your prurient instincts for a buck, or 19 bucks, actually.

Billed as the ultimate experience, the Interlude package is a curious piece of software for several reasons, not the least of which is the unmitigated gall the folks at Syntonic were able to muster to market this X-rated program in the first place.

In essence, Interlude is a well-structured interview designed to plumb the depths of your sexual preferences in 10 probing questions, after which it refers you to a book of "interludes" that accompanies the software.

The paper-bound, 96-page volume of interludes contains 106 elaborate sexual scenarios, written with acute attention to detail. Titles indicate content. For instance, "Wet Fun On A Hot Summer Night," "Good Vibrations," "Fellatio By Firelight," "Caveman," "My Way," "Satin Chains," "Macho Man." The list goes on.

If the computer-suggested interlude suits your fancy (or fantasy), you may opt to act out the scenario. In addition, certain interludes are stored in the computer and do not appear in the book. Getting one of these gems is apparently cause for self-congratulation. Your answers must have been outstanding!

This computerized extension of spin the bottle is not a revolutionary premise but is, one has to admit, an unusual application for the home computer and certainly not what the gang at Tandy had in mind.

The software algorithm of this machine-language program is straightforward. The interview format is logical, and all questions require a numerical answer, indicative of the degree of enthusiasm the respondent demonstrates. Some of the milder examples are:

How complicated do you want this interlude to be?

1. Back to basics.
2. Keep it fairly simple.
3. I'm not choosy.
4. Maybe a small production.
5. I'm ready for a really big show!

-or-

If you were a movie, which would you be?

1. Pillow Talk
2. Tom Jones
3. Gone with the Wind
4. Superman
5. A Streetcar Named Desire
6. Fellini's Satyricon

After answering several of these thought-provoking questions, the computer gets down to the nitty-gritty and quizzes you on specifics, hopefully picking up on your mores, folkways, and taboos. Your scalar answers are then processed, and an interlude indicative of your final tally is suggested.

The program is fairly interactive. Suzy, one of our foxiest secretaries here at *80 Micro*, displayed so little enthusiasm during one interview session that she was directed to interlude number 29: Stay home alone and curl up with a good book.

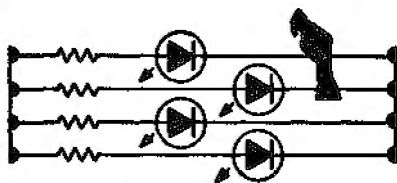
For all its daring, the Interlude package is fairly tame in its attitude toward sexual encounters. The maximum number of participants in any interlude is limited to two, so no free-swinging group gropes are possible. Furthermore, all interludes are strictly heterosexual. No gay or even bisexual interludes are allowed.

Syntonic's marketing has raised eyebrows in the industry, and their full-page living color ads have elicited both pro and con responses from readers. A few irate subscribers have taken offense at the quasi-naked lady approach Syntonic uses in their promotions and canceled. Generally, most readers seem curious or amused, rather than indignant. This may be just another indication of the decadence of the electronic age.

There you have it—Interlude: The Ultimate Experience. The first microcomputer-based adult computer program. Will it be a milestone in microcomputer software, or just another trendy diversion? Time will tell.

What's next? A microprocessor-controlled, RS-232-compatible love doll, perhaps. Welcome to the micro-millenia. (*Interlude: The Ultimate Experience* is sold by Syntonic Software Corp., Houston, TX 77052, cassette \$18.95, disk \$21.95.)

Chris Brown  
September 1980



★ ★ ★ ★

**Invaders From Space**

★ ★ ★ ★

**Pinball****Acorn Software Products Inc.  
Model I and III**

Pinball realistically simulates the feeling of being in the finest pinball palace. The action is so real that you find yourself rocking the keyboard. It's a good thing the program doesn't say tilt.

As anticipated, the ball is square instead of round. The space bar shoots the ball just like the plunger. The length of time you hold down the bar determines the speed at which the ball enters the playing area. Ball travel is extremely natural, and even accounts for the slightest gravitational effect. In fact, you can even catch the ball by the tip of the flipper to deflect it.

Sound effects are not fancy, but are more than adequate. The right side of the screen displays your score, bonus points awarded, and the best score of the games played during the session.

We've played all of the invader-type games, and *Invaders From Space*, without question, comes closest to duplicating the original arcade game on a TRS-80.

The format stays close to the arcade version, with the invader munchkins descending down the screen to attack your bases. You can determine the number of bases, game speed, the number of shots that you can have on the screen at any one time, the number of bombs fired back at you, and their accuracy. This makes it possible for

players of all abilities to play at a comparable level of competition.

To beat the invaders, you must score 655,350. If nothing else, think of all the quarters you will be saving while trying to reach this goal!

*(Acorn Software Products Inc.,  
634 North Carolina Ave.,  
Washington, DC 20003; \$14.95  
cassette, \$20.95 disk, each.)*

Dan Keen  
Dave Dischert  
April 1981

★ ★ ★ ★

**Invaders Plus****Level IV Products Inc.****Model I and III**

I have been eagerly awaiting a good TRS-80 imitation of *Space Invaders*, and *Invaders Plus* is it.

One feature of this game is that if you hook your AUX plug into an audio amplifier before starting, you will listen to a fine rendition of the theme song from *Star Wars*. This is amusing the first few times, but since it can't be bypassed, it soon become a nuisance.

There are nine levels of play, which graduate in difficulty. After selecting your level of play, the screen fills with the first wave of 49 invaders. Each of these marauding meanies is depicted complete with blinking eyes and animated movements.

The object of the game is to terminate the invaders before they send you to an early grave. Should you be successful clearing a wave of invaders, you are treated to a screenful of even deadlier demons. These attackers drop more bombs and move more quickly, making them harder to kill than the

preceding group.

Unlike Space Invaders, where the number of screens is only hindered by the player's skill, Invaders Plus offers you only four waves of persistent pests. After killing all four groups, you have the title of Hero bestowed upon you via the computer.

Invaders Plus is far superior to any other invader program I've seen. This is a truly magnificent programming effort by Larry Ashmun, and well worth the asking price.

*(Level IV Products Inc., 32429 Schoolcraft Road, Livonia, MI 48150; is no longer marketing Invaders Plus.)*

Owen Linzmayer  
April 1981

### **Invasion from Outer Space**

(See Page 105)

### **Invaders Plus**

(See Page 105)

★ ★ 1/2

**Kim's Game**  
**General Computer Co.**  
**Model I**

Put this program, a keg of beer, and 20 friends in your living room, and what have you got? The last party of yours that those people will ever come to.

Kim's Game, the "mind-expanding game of memory and recall," combines the excitement of math drills with the gaiety of Roget's Thesaurus to provide you and your guests with hours of unrelenting boredom.

In all fairness to General Computer, Kim's Game is far from shoddy. It is well-produced and well-packaged, and the manual includes the Basic source listing.

And it will improve your memory.

The name is taken from Rudyard Kipling's novel, Kim, whose title character plays a memory game in which he tries to remember objects on a tray. But there is a huge difference between remembering physical objects and remembering words, which is what you must do in this game. The former is visually exciting, but words on a computer screen are dry, static, flat, and lifeless.

Ultimately, Kim's Game becomes Kim's Endurance Test. An improved memory may have its advantages, but this program makes forgetfulness look like an attractive alternative.

*(General Computer Co., 4873 Langer Lane, Woodbridge, VA 22193; \$19.95 cassette, \$24.95 disk.)*

Eric Maloney  
May 1981

★ ★ ★ ★

**Labyrinth**  
**Med Systems Software**  
**Model I and III**

Labyrinth is Frank Corr's (and friends') second excursion into the dark, dim corridors of three-dimensional graphics—kind of.

There are more hallways, more boxes full of junk, more empty rooms, and a good selection of pits, as well. Labyrinth looks like Deathmaze, and the information sheet that comes with it is equally brief and bare of help or hints. But that's okay—finding commands that work is part of the mystery of this adventure. If you get stuck—really, truly engulfed—the Med Systems people

are very nice about giving hints.

I've walked for hours in circles before realizing what I was doing. Like *Deathmaze*, this game has to be mapped—if you can do it. The fog makes it difficult to see, or get through, certain parts of the maze. And this is the only place I've ever visited where you can jump into a pit and come up somewhere on the very same level you jumped from. Mapping this maze is like putting together parts of a jigsaw puzzle. Expect to spend many hours doing it.

The mapping has to be carefully done in sections. And prepare to suspend your normal perceptions. Up isn't necessarily up, it may be over, and down is usually a quick route to somewhere you wish you weren't. The labyrinth seems to overlap itself in at least one spot.

Be sure to bring along some extra torches and a picnic basket or two. If you don't, you're going to have to search for them, and they're very well hidden, as are several nasty surprises that you'll just happen upon on your way. You'll have your pockets full of junk by then; maybe some of it will come in handy, handy.

*(Med Systems Software, P.O. Box 3558, Chapel Hill, NC 27514; \$14.95 cassette, \$17.95 disk.)*

Debra Marshall  
August 1981

★ ★ ★ 1/2

**Laser Defense**  
**Med Systems Software**  
**Model I and III**

Laser Defense is Med Systems' first attempt at an arcade-type game and they score big. This is

not just another version of *Missile Command*; it's a totally original game.

Your mission is to keep missiles from destroying your cities while destroying Russian launching stations. This game is a combination of strategy and arcade fun. You begin with six cities and four hovering satellites, while your computer enemy has 12 silos and an unlimited number of particle-beam weapons traveling on MX tracks. These weapons appear at intervals to fire at your satellites. Two views or maps are available to you: one of the United States and a Europe/Soviet Union map. The U.S. map indicates cities and incoming ICBMs. To destroy a missile in flight, you position your sight with the arrow keys and fire with the space bar. If the nuclear blast is close enough, the incoming missile is destroyed and your city saved. When all the cities are destroyed, the game is over.

Your ultimate weapon against ICBMs is the eradicator. It is engaged by pressing E, and destroys any missiles in flight immediately. Should you happen to survive the first wave of silos, you will be awarded a bonus of 200 points for remaining cities and 10 points for any remaining energy units. Laser energy is refilled and another wave begins. As you go through each wave, you face an enemy that becomes more difficult to defeat. Missiles are launched faster, in greater numbers and particle weapons appear more often.

Fast reflexes, marvelous finger coordination, and a methodical plan of attack improve scores. Both maps feature full wrap-

around, and mastering this aspect of the game makes a big difference in the latter stages.

This is a machine-language game for one or two players, and the top five scores are saved on the disk version. Complete sound effects are featured, including an indication that a particle-beam weapon has appeared. This game has 10 degrees of difficulty. At the highest, it shows just how fast machine language can be.

The game is fun, works well, and presents a challenge even after several hundred plays. It's a super original arcade game designed for the TRS-80. (*Med Systems Software, P.O. Box 3558, Chapel Hill, NC 27514; \$15.95 cassette, \$18.95 disk.*)

Mark E. Renne  
August 1982

### **Local Call for Death**

(See Page 100)

★ ★ ★ 1/2

### **Maces & Magic Adventure International & Chameleon Software Model I and III**

The Maces & Magic series consists of three adventures: Balrog, Stone of Sisyphus, and Morton's Fork. The difference between this series and other adventure-type games on the market is their cross with role-playing games. Your character is unique in structure, and events occur differently depending on your own attributes.

The games may be played in any order or separately. You begin in the general store. Here

you will roll up a character and arm him for future battles against evil. Individual attributes are strength, IQ, luck, constitution, dexterity, and charisma. If you've played *Dungcons & Dragons* or *Runequest*, you'll recognize these characteristics; they have the same function here as they do in those games.

Now comes the time to choose a weapon. There are 80 different weapons. Each has a name and power, weight, cost, dexterity, and strength ratings. Weapons do different degrees of damage depending on their power. Armor is then chosen in the same way. The best costs more, weighs more, and protects more. There are more than 25 types of armor.

Commands are all one letter and include (G)et, (L)eave, (P)ack, (H)elp, (S)tatus, (W)ait, (I)nventory, and (\*)save game. Playing the game is much like playing most adventures, with a few twists. You find objects and try to manipulate them correctly. The rooms or locations are loaded one at a time off disk. This means the computer constantly accesses the disk if you're moving along quickly. Your options are all listed on the screen.

The game is fun and a nice blend of adventure and chance. My only complaint is that it's written in Basic; it's very slow. The game seems unbearably slow when saving a game as you must traverse different programs to accomplish this. It comes with excellent documentation and loading instructions. If you have a Model III and are using TRSDOS 1.3, patch your DOS to allow killing one file while another is open. Aside from the



speed, I would recommend these games without reservation to role-playing computerists. (*Adventure International & Chameleon Software, Box 3435, Longwood, FL 32750; \$29.95 each, disk.*)

Mark E. Renne  
August 1982

★★★★

**Master Reversi**  
**Instant Software Inc.**

★★★★½

**Odin**  
**Odesta Publishing**  
**Model I and III**

Master Reversi and Odin are two microcomputer versions of the popular board game Othello. The games, like Othello, are played on an 8-by-8 grid, similar to a chess board. Unlike chess, pieces are not moved after being set on the grid. Your object is to occupy territory by placing pieces on the board and trapping your opponent's pieces between them. Once trapped, an opponent's piece is flipped over, reversing its color.

Master Reversi requires a 32K Model I with one disk drive. It has nine skill levels, and an easy-to-understand manual. The game responds quickly up to the fourth or fifth level; but the response time increases for the higher levels, typical of heuristic, tree-searching algorithms. You can change the level of play at any time during the game. Master Reversi has a tournament clock and lets you play the computer or another person.

This game uses the arrow keys for positioning the flashing cursor. I prefer this to Odin's

method, which uses numbers to choose the square. There are several options included, the nicest of which is the output to the printer.

Master Reversi's thinking mode lets you see what Aldaron (the computer) is thinking. What you see ranges from the last board positions to the moves Aldaron is considering and how highly he values them. Aldaron also lets you set up possible board positions and analyze them. In fact, you can have five different board positions in memory at once. You can display or review the moves made during the game, either in standard Review notation or with the board itself.

Master Reversi's tournament mode is for serious players. In this mode, Aldaron thinks on your time, plays in accordance to tournament rules, and uses book openings. I prefer this mode. Master Reversi plays a good game. Although its response time is slightly slower than Odin's, it is better than most Othello games I have seen.

Odin also requires a 32K Model I with a disk drive. Its well-written manual includes a complete description of rules, history, and strategy. Odin automatically displays your possible legal moves when it is your turn and has a number of options, although not as many as Master Reversi.

The tutor mode displays all your moves, with Odin's opinion of their relative worth. This mode reflects Odin's opinion of the immediate situation without considering future moves, so it is of little help except to the novice. In the principal variation mode, Odin shows the move he thinks

is best for you, what his counter move will be, and so on, for the number of moves equal to the depth of the search. The tournament level works only if you have a 3x speedup modification. Level B, however, plays approximately within the 25-minutes-per-player limit used in tournaments.

Odin plays more competitively and beats me more often than Master Reversi. It doesn't, however, have the options found in Master Reversi, and sorely lacks the printer-output option. I prefer Master Reversi's method of choosing squares for moves to Odin's name-that-square approach. Both play a good, solid game and have ample skill levels to keep you busy for quite a while.

*(Instant Software Inc., Peterborough, NH 03458; \$29.95. Odesta Publishing, 930 Pitner, Evanston, IL 60202; \$34.95.)*

Bruce Powel Douglass  
November 1982

### **Meteoroids** (See Page 93)

★ ★ ★

### **Microcosm III Basics and Beyond Inc. Model I and III**

Microcosm III is a collection of 20 programs with sound. At least four of these programs alone (Pinball, Submarine, Knockout, and Battleship) are worth \$15.

Pinball, the most worthwhile program of the series, is a machine-code game with very fast graphics with a spinning pinwheel and assorted other goodies: various bleeps and

bloops as the ball bounces around.

Submarine lets you try to sink ships through a periscope view. It has very nice and quick graphics with ships that fight back and try to sink you with depth charges.

Knockout displays six rows of bricks on the left side of the screen. The object is to knock out all the bricks by hitting them with a bouncing ball. You rebound the ball by controlling a paddle that moves up and down the left side of the screen. It has fast graphics for a game written in Basic.

Battleship is a computer version of the popular paper and pencil game with a selectable size grid. The computer goes after a kill when two portions of your ship are found. This game can be played against another person. It is enjoyable to play but, unfortunately, if the computer wins, it doesn't show you where its ships were placed—very frustrating when you have been unable to find them during the game.

The other programs included are: Computer Composer, Long Division, Multiplication, Typing Tutor, Cliff Hanger, Crossword Puzzler, Key Sound, Computer Organ, Children's Hour, Message Marque, Gunfight, Seige, Instant Replay, Monster Chase, and Shootout.

Even if, as I suggest, a little less than half of these programs are worth \$10 or \$15, you can wind up ahead. For \$29.95 it is still a good buy.

*(Basics and Beyond Inc., Amawalk, NY 10501; currently \$29.95.)*

Carl A. Kollar  
January 1982

★ ★ ★

**Micro-opoly**  
**Level IV Products Inc.**  
**Model I**

Computer games usually eliminate the aggravating paper-shuffling and bookkeeping that takes the fun out of a game. With Micro-opoly you get the added bonus of an honest banker.

The single sheet of typeset instructions will not answer all your questions. Assuming you have an understanding of the Parker Brothers board game Monopoly, however, you can figure out what's happening.

New zoning ordinances were passed for this game. You do not need to own all the property in a color sequence to build on any square you own (just enter your answers through INKEY\$), and while the computer is limited to buying houses for only one property at a time, you can buy up to four houses or a hotel for as many squares as your bankroll will allow.

If you land on computer property, you'll have to pay rent. If you are forced into debt by any payments, the computer requires you to sell houses or hotels until you have a positive balance. You also have the option of continuing to sell property until you have sufficient cash on hand to feel safe.

Purists will decry some rule changes—automatic and immediate payments are required when you land in jail, there is no Get Out of Jail Free card, houses are sold back to the bank for full (not half) price, and so on—but for the most part Micro-opoly leads to an interesting game with a reasonable time frame.

And there's no arguing about who picks up the game and puts it away.

(Level IV Products Inc., 32429 Schoolcraft Road, Livonia, MI 48150; \$14.95 disk.)

Alan and Nick Grassel

October 1980

★ ★ ★

**Midway Campaign**  
**Avalon Hill**  
**Model I**

Midway Campaign is one of those games that on the surface seems laughably simple, but through playing, proves to be anything but. With all the graphics in use today, the lack of them in Midway Campaign is striking. Play is text-oriented with the only graphics being a 12-by-12 grid of dots. These portray a map of the Pacific Ocean around the island of Midway. The two American task forces and Midway are under the player's control. The computer controls the three Japanese naval groups.

At the beginning of the game, the American units are placed in their historic positions. Japanese forces are on the map but not shown. Even though the American forces are visible to the player, the computer does not know where they are. It, too, must perform searches to locate the enemy.

First, the player issues a Fleet command. There are four Fleet commands that display the map, the status of American aircraft carriers, change the heading of the task forces, and conduct aircraft operations. The computer remains in an interactive mode until an integer number is

entered. This represents the length of time (in hours) the player wishes to play.

Next, the computer takes over. It moves American forces in accordance to Fleet commands, decides upon and executes Japanese actions, conducts searches and combat (if any), and checks for the end of game. Unfortunately, this can be rather lengthy, and with no graphics involved, the player has no recourse except to sit and wait. This waiting, in my opinion, is the game's major flaw.

There is, however, enough in the way of realistic decision-making to keep the game interesting. Tactical decisions will have a great bearing on its outcome. A minor oversight can result in a sunk carrier. The computer plays a very good game as the Japanese commander and is not easy to beat.

Midway Campaign is written in Basic. It comes with four pages of rules, historical background, examples of play, and instructions for loading and running the program. I would recommend Midway Campaign to those new to conflict simulation. Advanced game strategists will probably tire of it quickly. (*Midway Campaign is sold by Avalon Hill, Dept. C-10, 4517 Harford Road, Baltimore, MD 21214; \$16 for cassette, \$21 for disk.*)

David Tintis  
June 1981

★ 1/2

**Mind Thrust**  
**Hayden Software Div.**  
**Model I**

Here at the office, the first reaction to Mind Thrust was,

"Too simple. What's the point?" Mind Thrust is simple, all right. Unfortunately, it is also simplistic.

Mind Thrust reminds you vaguely of Othello. You are given a 6 by 8 grid on which you and the computer each place five pieces. Then, taking turns placing pieces, you each try to form a chain from end to end, or eliminate each other's pieces by attacking.

You can attack any piece touching one of yours. But the computer can defend some of its pieces. If you have a choice of five pieces to attack, the computer can defend three. If you attack a piece it chooses not to defend, it loses the piece. If the computer defends successfully, you lose your piece, all those adjacent to it, and any left isolated. The computer can also attack.

The computer does not think strategically. Its opening moves are random; in some cases you can form a chain and win unchallenged. Also, the computer has no idea when to attack; it will plod along trying to form a chain when attack is its only hope of winning. Also, when you attack, your choice of which piece to go after is strictly a matter of guesswork. The computer uses no strategy when it defends.

Finally, you cannot develop any logical plan of defense. The computer randomly decides which of your pieces it will attack; you must randomly decide which ones to protect.

The game is not a bad idea and could be fairly entertaining if the computer were programmed with some more smarts. But as it is now, Mind Thrust falls somewhere between

being a game of chance and a game of strategy, providing the satisfactions of neither. It offers about one thrill every hour, if you can sit still that long. (Hayden Software Div., 50 Essex St., Rochelle Park, NJ 07662; \$16.95. Level II, 16K required.)

Eric Maloney  
August 1982

★ ★ ★ ★

**Mod II Games**  
**Small Business Systems**  
**Group**  
**Model II**

The fact that you own Tandy's most expensive business machine doesn't mean you shouldn't be able to play computer games every now and then. Let's face it—Mod II owners are people, too, and there is nothing like a good computer game to reestablish friendship with a dull business machine.

Small Business Systems Group has had an excellent package of Model II games available for some time. It includes Star Trek, Checkers, Concentration, Treasure Hunt, Banko, and Dog Star Adventure.

The Star Trek game is a superior example of the genre, demanding both strategic planning and tactical know-how. It is complicated enough to be absorbing, but not so demanding that it is boring.

Dog Star Adventure also rates as a top game. Even a jaded computer adventurer will find those dastardly twists and shocking surprises entertaining.

Banko is essentially a Black Jack game, with the computer as a dealer playing to win.

The remaining games on the disk I can live without. While they are well-written, I can't get excited about Checkers or Concentration under any circumstances. All the programs were well-documented, and contain no bugs or glitches.

This Lance Micklus package can be a real eye-opener for the business-minded Model II owner. (Small Business Systems Group, 6 Carlisle Road, Westford, MA 01886; \$75.)

Paul Grupp  
May 1982

**MyChess**

(See Page 106)

**New Starship Voyages**

(See Page 85)

**North Atlantic Convoy Raider**

(See Page 90)

**Nukewar**

(See Page 90)

**Odin**

(See Page 117)

**On His Majesty's**  
**Ship Impetuous**

(See Page 100)

★ ★ ★ 1/2

**Paddle Pinball**  
**Tandy/Radio Shack**  
**Model I and III**

Radio Shack scores high with this excellent game, the best feature of which is the option to change the playfield. In fact, you can create and save dozens of different layouts and load any of them once Paddle Pinball is

aboard. The game also features lively sound effects, which operate on games of your own design as well as the game provided. The program is in machine language and comes on cassette.

The play combines pinball-style action with a paddle that you guide across the bottom using the left/right arrows. You control the direction of the ball's rebound by bouncing it off different segments of the paddle. The shift key speeds paddle movement.

The object is to amass points, of course, but also, if more than one person plays, to score credits, which entitle you to an extra game. There is also a music block, which starts with the number 2 in it. The number augments each time you hit a corner of the box, and you are rewarded with a tune consisting of that many notes.

You can modify the playfield with the keyboard arrows. You can also create bonus dots, and you can save the field you design on cassette.

Paddle Pinball is a clever and creative program, well worth the modest price if you like arcade-type games.

*(Tandy/Radio Shack, Fort Worth, TX 76102; \$9.95.)*

Charles Gulick  
February 1982

**Parsector 8**  
(See Page 85)

★ ★ ★ 1/2

**Penetrator**  
**Melbourne House Software**  
**Model I and III**

Penetrator is loosely based on

the arcade game Defender. You are the sole survivor of a fighter squadron whose mission is to make it through four defense rings and blow up an illegal cache of neutron bombs. Along the way you face anti-aircraft missiles, radar installations, and paratroopers; it's enough to give Al Haig goosebumps.

This is a game of skill; your craft always moves forward with only momentary thrust and braking at your disposal. Your weapons are forward missiles and bombs, via the right arrow and space bar respectively. Timing is critical on the bomb drops; they are used primarily on the radar installations you must destroy to avoid detection, and are also your only means of destroying the neutron bombs.

The landscape is as treacherous as your enemies. Certain spots require precise braking and maneuvering. In many places there is no room for error, especially if you have to avoid enemy missiles as well.

But this is where the best feature of Penetrator comes in: you can make custom landscapes to suit your ability, removing difficult areas and adding or subtracting missiles and radar bases. This landscape editing takes a few minutes, but it is a unique way of providing different difficulty ratings in a computer game. Another good feature is the training mode. You can play continuously until you get the hang of the game; it's as close to immortality as you'll get.

The graphics, similar to Adventure International's Eliminator, are about as good as is possible on the TRS-80, and the sound is great. The documen-

tation is better than it probably has to be—a small booklet instead of a one- or two-page instruction sheet.

Although I enjoyed this game, I didn't like having the thrust and fire control on the same key (right arrow). This prevents firing while thrusting and occasionally results in an unwanted thrust. Although I didn't use a joystick, I don't see how using one would be an improvement: to fire, you jiggle the stick to the right and hold it continuously for thrust.

Penetrator is a very well done game program and worth the asking price.

*(Melbourne House Software Inc., Dept. CS, 347 Reedwood Drive, Nashville, TN 37217; \$24.95, disk or cassette.)*

Michael E. Nadeau  
September 1982

### **Pinball**

(See Page 113)

★ ★ ★ 1/2

### **Planet Miners Avalon Hill Model I and III**

Planet Miners is Avalon Hill's first effort at a microcomputer game outside the strategic war genre and is an excellent example of thorough programming. The game's scenario is a futuristic Gold Rush. You have five ships, their opponents, and a limited amount of mining claims to try for. To add realism to the game, the playing options include claim jumping, protesting claims, and sabotage.

The game is for zero to four

players, zero being the computer versus itself. It requires 16K of RAM and works on either a Level II Model I or a Model III.

I noticed two detachments from reality in the game. One is that there are 10 planets in the solar system (the addition being Ceres), and the second is that once you set a ship's destination and it takes off, you can't alter the destination until the ship arrives. This adds some spice in some situations.

When attempting sabotage or a claim jump, keep in mind that the Space Patrol might arrest you if you are caught in the act. If no Patrol ships are around and your opponent catches you, your ship crew will be detained and tortured by your would-be victim.

The scars still show, and I've become a wary miner indeed (I told you it is realistic). Planet Miners is an exceptional buy. *(Avalon Hill, Dept. C-10, 4517 Harford Road, Baltimore, MD 21214; \$16 cassette, \$21 disk.)*

Darren DeVigili  
October 1981

★ ★ ★ ★

### **Planetary Lander Instant Software Inc. Model I**

Planetary Lander is one of the best spaceship landing programs available. Via graphics, the game shows the position of your ship in relation to an altitude yardstick, the scale of which changes as your ship approaches the planet's surface. Next to the altitude display is a control readout that gives elapsed time, speed in

meters per second, thrust, and other indicators: these displays are updated at one second intervals. Although Planetary Lander is only available in Level I, Radio Shack's conversion program transforms it to Level II. Combined with Stellar Wars in a two-program package, Planetary Lander is an outstanding value. (*Planetary Lander, Instant Software Inc., Elm St., Peterborough, NH 03458, is no longer available.*)

John Warren  
September 1980

★  
**Pyramid  
Tandy/Radio Shack  
Model I**

Pyramid is Radio Shack's entry into the world of adventure games for the TRS-80. The scenario starts with the would-be explorer in the desert, standing in front of the tip of a pyramid. You proceed using the assumed and known vocabularies.

Unfortunately, the game suffers from the lack of a command word base, and has limited explanations of the instructions. You are on your own to figure out how to use it.

(*Tandy/Radio Shack, Fort Worth, TX 76102; Pyramid is now available for the Model II, III and Color Computer for \$14.95.*)

William O'Brien  
August 1980

★★★★  
**Quest  
Aardvark-80  
Color Computer**

Quest is an adventure game

written in Basic. Your goal is to travel through the lands of Alesia to Moorlock's citadel and, ultimately, defeat the forces of evil within. What makes Quest unique is that it's different every time you play it. It also makes good use of the Color Computer's graphics.

You start each adventure with 10 loyal companions. Your object is to gather men, weapons, armor, and food capable of defeating the forces at the citadel. You gain men and money by fighting or coming to terms with bands of ogres, wizards, pilgrims, and soldiers. You can also find money in caves, ruins, castles, and towers. All are randomly placed with each new game, and do not appear on the screen until you approach them while exploring Alesia. You can buy food, armor, and weapons at the marketplace in each city.

A realistic feature of the game is the weight system used. Each man can carry only 100 pounds. The computer keeps track of the total weight of your items (plate mail—50 pounds, broadsword—10 pounds, and so on) and compares it to the amount your men can carry. If the total is too high, you must drop items until the weight is acceptable. You can't ignore these warnings, since you will be unable to move until enough material has been left behind.

Once you have crossed the river and you've gathered what you consider to be enough men, weapons, and food, attack the citadel. The display for a fight lists ever-changing tallies of your men, weapons, experience level, luck, and survivors, against theirs. To stop the fight, hit the



space bar when you think your lucky number is higher than your opponent's, although this is just one factor in deciding the outcome. You can see how many survivors each party has. You can fight, run, or come to terms whenever a fight has been stopped.

Quest is a terrific game for beginning adventurers, because there are no sudden unreasonable ways to lose. For example, if you are losing men in a fight, you can always try running away. The game is playable in two to five hours, and since the game is different every time, you are not likely to lose interest in it the first time you defeat the forces of evil. (*Aardvark-80, 2352 S. Commerce, Walled Lake, MI 48088; \$14.95 cassette, 16K required.*)

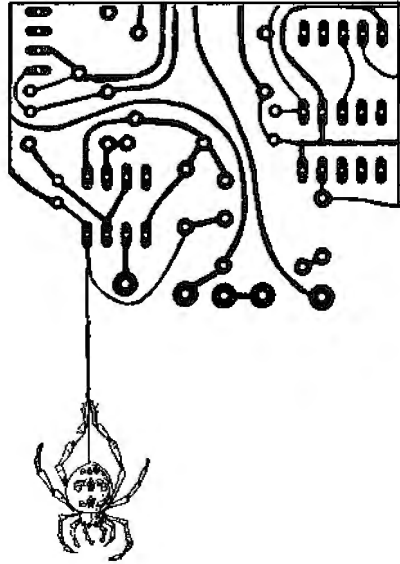
Beth Ann Norman  
December 1982

★ ★ ★ 1/2

**Rat's Revenge**  
**Med Systems Software**  
**Model I and III**

Rat's Revenge is interesting! Up to now every maze game I've played has led to tedium. But suddenly, I'm a rat, scurrying through a maze looking for swiss cheese. I can move one space at a time, run down straight hallways, and even consult a Hint Guru.

Everything I see is in rat's perspective. I don't get an overhead look at the maze until I find the cheese or quit. Oh yes, there is one other way to leave the maze—on little phosphorescent wings. However, dying is not enough; the game also



haunts me with dreaded cheese hallucinations.

Besides what you see as a rat, you receive assorted messages. The mazes are randomly generated and take from one to six minutes to complete.

Rat's Revenge, as with all Med Systems Software that I have seen, is well worth its price. With its well-drawn graphics, total rat perspective, variable skill levels, humorous touches, and more, it is one game you won't play once and shelve. (*Med Systems Software, P.O. Box 3558, Chapel Hill, NC 27514; \$14.95 cassette, \$17.95 disk.*)

Darren DeVigili  
November 1981

★ ★ ★ 1/2

**Robot Attack**  
**Big Five Software**  
**Model I and III**

Robot Attack is an arcade

game for one or two players that pits an Earth warrior against several Jidyian robots.

Your warrior roams the maze looking for robots. At first it seems too easy to destroy the robots without being shot at in return, but suddenly the robots counterattack and you're kept busy just trying to avoid their laser fire. A robot laser-blast can wipe out your warrior from anywhere on the screen as long as a clear path exists between your warrior and the attacking robot.

To keep you from becoming too familiar with the game layout, your warrior is in a different set of rooms (or maze) each time you play the game. Robot Attack also comes with sound and speaks to you during the game, using such words as "player one," "player two," "chicken," "lucky," "game over," and "great score." You're awarded an extra warrior after 5,000 points and another after 10,000 points.

*(Big Five Software, P.O. Box 9078-185, Van Nuys, CA 91409; \$15.95 cassette, \$19.95 disk.)*

Carl A. Kollar  
August 1982

**Sargon II**  
(See Page 106)

★ ★ ★ 1/2  
**Scarfman**  
**Cornsoft Group**  
**Model I and III**

Scarfman, like the arcade game Pac-Man, features several colored monsters running around a simple maze. You con-

trol another creature shaped like a large C. The object of the game is to eat the monsters. At certain times, the monsters change color and can eat you.

Each game begins with your character at the bottom center of the screen and the five monsters at the top center. Arrow keys control your character. Little square boxes dot your path, and as you pass each one you gain points. Plus signs occur around the maze and force all the monsters into the safe-to-eat stage. When you have disposed of blocks and plusses, the maze fills with them again and the level of difficulty increases. The monsters improve at finding you or hiding, and they revert to inedible monsters more quickly.

Some people say that this version is quicker than the arcade version. This black-and-white version is superior to the one written for the Color Computer.

Sound is an extra feature. A speaker attached to the cassette cord produces sound effects each time a square, plus, monster, or you are devoured. The sounds are simple and synchronous with the speed of the action.  
*(The Cornsoft Group, 6008 N. Keystone Ave., Indianapolis, IN 46220; Model I or III cassette \$15.95, disk \$19.95, Color Computer cassette \$19.95.)*

J.L. Latham  
March 1982

**Sex-O-Scope**  
(See Page 88)

**SFinks**  
(See Page 106)

★ ★ ★ ★  
**SFinks 3.0**  
**William Fink**  
**Model I and III**

Although I've played with a number of micro-chess programs, most of them play so poorly or inconsistently that I lost interest pretty fast. These frustrating experiences helped me to appreciate SFinks 3.0.

The first thing you notice about the game is the clear, unambiguous graphics portrayal of the pieces. SFinks has the least distortion of any TRS-80 chess graphics I've seen.

With a printer on line, moves are printed out as they are played. If you put your tape recorder on record and plug in an earphone, you'll hear some beeps every time it's your turn to move. This is nice if you want to read a book, write a poem, or design a cyclotron while you're waiting for SFinks to move.

After choosing your color and which of nine levels you'd like SFinks to play (Level 1 gives the program only six seconds per move, Level 7 a maximum of two minutes and 50 seconds; level 9 analyzes continuously until you tell it to make its move), you indicate your move in standard European algebraic notation. Columns are labeled with the letters A through H and rows with the digits 1-8. To move, indicate the square from which and to which you're moving. The edge squares have little digits and letters to help those who are used to the P-K4-style notation.

The left side of the screen displays the level of play, the last 12 moves, the move currently considered best by SFinks, and

the time being taken for the current move. You can force SFinks to immediately make the move it has so far determined to be the best by pressing the @ key.

If you press S when it's your turn, SFinks recommends a move. To make that move, press enter. The move is made but, unfortunately, the display remains unchanged until SFinks makes its own move. This occurs any time your move has been anticipated by SFinks. It's a bit annoying having to stare at a board that has not been updated.

SFinks has a special edit mode that lets you take a move back, change levels or colors, or set up any position by entering the square and the piece that will go there. You can start from the current position or from a clear board if you wish.

SFinks plays the strongest game I've encountered on a micro. I would estimate its rating at 1,700 in levels 5 through 7. One reason for this strength beside the sophisticated move-choosing algorithm is that it thinks on the opponent's move.

Although SFinks is a remarkably good micro-chess program, there are a number of features I wish it had. I would like to be able to decide to print out a game after it had been played rather than having to print out each move as it is played. The display should be updated every time a move is made whether or not it has been anticipated. It would be nice if all the moves could be stored in memory so the course of the game could be retraced and analyzed. A save-to-disk option would enable a game to be continued or analyzed at a future date.

Regardless of my wish to see SFinks do more, what it does now is quite remarkable. I highly recommend it as a good chess opponent and a learning tool to improve your game.

*(William Fink, Suite 24B, 1105 N. Main St., Gainesville, FL 32601; \$39.95, 32K is required.)*

Joel Benjamin  
September 1982

### **Space Ace 21**

(See Page 85)

### **Space War**

(See Page 93)

★★★★

### **Star Blaster**

#### **The Micro Works Color Computer**

Star Blaster is a real-time version of the arcade game Asteroids written in 6809 machine code, and the action is as fast and furious as the original version. The game is played with joysticks or a predefined set of keyboard characters that move the mother ship and fire missiles. The game consists of a missile-firing spaceship located somewhere in an asteroid-infested quadrant deep in the uncharted depths of space. The idea is to shoot to pieces all asteroids in sight before they smash into you and destroy your ship. The ship can fire missiles and move around to dodge the debris.

Micro Works' version starts with a generous supply of five ships. Pushing left or right on the joystick (or pressing the appropriate keys) rotates the ship through 360 degrees, whereas a

forward push gives the ship a kick of power that jolts it in the direction it's facing. This takes a little skill, since the ship gains momentum and continues until it decelerates under some undiscovered law of physics. This is where Star Blaster has its own custom trick. By pulling back on the stick, a bubble-like shield surrounds your ship and prevents any asteroids from coming into contact.

Various sizes of flying saucers randomly enter your quadrant and take pot-shots at you. It's just one of those facts of space-life, and naturally you have to defend yourself by counterattacking or dodging the missiles raining down upon your ship. Star Blaster awards points according to the difficulty of the targets you hit. Every time you earn 10,000 points, you are rewarded with a beep and an extra ship.

The program uses only two colors, black and light green. This is the equivalent video mode in Basic as PMODE 4 and SCREEN 1,1. It's not the prettiest of the Color Computer's displays, but the software's designer has opted for maximum resolution.

The documentation is a card outlining the basic rules and scoring system. Sound is also an integral part of the game, but is only a minor part of the overall package when compared with the quality of the display.

I recommend Star Blaster to any arcadophile. It's a definite must for the game library.  
*(The Micro Works, P.O. Box 1110, Del Mar, CA 92014; \$39.95 cartridge.)*

Jake Commander  
August 1982

★ ★ ★ 1/2

### **Starfighter Adventure International Models I and III**

Starfighter contains a 32-page *New-Pilot Introduction Manual*, published by the Solar Galactic Authority, Periodical Office/Landbase Central, and two cassettes.

The Trainee Simulator tape allows a novice pilot to explore the operating parameters of the SC-78503 Starfighter craft and to learn about the universe without the hazards of actual combat. The simulation stops if the pilot makes an error, and the correct procedure is indicated.

Make no mistake about the Main Mission tape, though; it is the real thing. The pilot departs Landbase Central as a paramilitary mercenary combatant, after being given an SC-78503 Starfighter craft free of charge by the Solar Galactic Authority. As in the past, there is no free lunch, since the pilot is now responsible for proper maintenance, fueling, and recharging of the SC-78503.

The theories of Hyperspace, Hypercharge, and Stellar Pumping are classified and cannot be related here, but are thoroughly covered in the *Introduction Manual*. Briefly, the pilot uses Hypercharge to get into Hyperspace to travel from now to elsewhere to elsewhere.

The pilot must seek out, identify, and destroy all craft hostile to the SGA. Identification is achieved by careful observation of visual profile or characteristic craft movement and evasive tactics. Performance data of all known craft is detailed in the *Introduction Manual*.

The SC-78503 Starfighter provides the standard real-time visual screen, a combat computer with target ranging and direction displays, targeting grid, and target lock controls. Included on the control panels are a velocity indicator, speed and direction controls, and a digital-beacon message readout. A unique feature of the targeting grid gives the pilot rear, as well as forward views of space, quite helpful for detecting hostile craft approaching from the rear.

Once a pilot reaches the rank of Star Lord, a secret code word appears on the control panel. Sending the code word to Adventure International will bring a special gift by return mail.

Starfighter is a challenge and doesn't get boring. This simulation seems realistic, and the real-time action does not leave much time for even a sip of coffee. (*Adventure International, Box 3435, Longwood, FL 32750; \$24.95 16K cassette, \$29.95 32K disk.*)

R.J. Brown  
February 1982

### **Star Run** (See Page 86)

### ★ ★ ★ ★ **Star Warrior Automated Simulations Model I and III**

Star Warrior transforms you into one of the Furies, paid agents of justice and revenge hired by the oppressed inhabitants of the planet Fornax to free them from the government of the Stellar Union.

In the first game scenario, you and a fellow Fury have been

dropped onto Fornax. Your mission is to move north to create a diversion while your partner moves south to kill the Governor. You want to be seen and shot at (Furies being either exceedingly brave or incredibly stupid). You select your combat suit, the number of minutes the scenario will last, and one of five levels of difficulty.

The second scenario reverses the roles. You must find the Governor, kill him, and escape as quickly as possible. The Governor can move around; his forces will shoot you on sight. There is no time limit to this scenario, but you lose points if you take more than 20 minutes.

Your monitor shows one square at a time of a 7 by 9 grid with four types of terrain (plains, swamps, mountains, and forests). The maps for the two scenarios are completely different. Each of the Furies' powered armor suits has different characteristics: flight speed, armor strength, invisibility, size of blaster, and so on. Disk version users can build a custom suit; cassette users can do almost as well by making appropriate adjustments to the data lines of the program.

The enemy has many different types of weapons—robot tanks, nitron guns, maulers, flitters, and infantry. They are all deadly. The game uses a varied assortment of chirps, whines, and other interesting noises; if you choose not to use sound, the game proceeds much too quickly. The handbook is well-designed and easy to understand.

The price seems a bit steep. If you can afford it, *Star Warrior* is a very fine program.

*(Automated Simulations, P.O.*

*Box 4247, Mountain View, CA 94040; \$39.95 32K disk, or 16K cassette).*

Dan Cataldo

April 1982

★ ★ ★ ★

### **Super Micro Pro Football Micro Pro Systems Model I and III**

The ultimate computer football game would need 48K, fill a disk with program modules, load each at certain times, have an opponent with four skill levels, and have graphic cartoons to break up dull times now and then. It would also be filled with cute comments about stadium fans and tv cameras and it might even play a darn good game of football. The Super Micro Pro Football is all this and more.

Instructions are complete and easy to understand. The game uses over 100 random number generators so you can expect the unexpected. The computer reviews your strategy and nails you to the wall if you use only a few plays. The screen shows a standard football field and indicates ball location with a pair of parentheses. The screen also shows score, direction of travel, time remaining, quarter, play timer, down, and ball location.

You may play one of four different teams. After you enter your team name and skill level, the computer creates a file with this information and remembers it from then on. There are 18 offensive and 8 defensive plays. The correct defense will not always stop the offense, just like real life, but the chances are good. Gains may be 99 yards on kick-off returns, interceptions, or punt returns. You have 10

seconds to enter a defensive play and 30 seconds to enter an offensive play. If you wait too long on defense the computer uses a general purpose defense. On offense you will either fumble the ball or receive a delay-of-game penalty if you wait too long.

The game also has graphic cartoons for the coin toss, field goals, kick-offs, injuries, scoreboard, and even a message from the opposing quarterback. There is a graphic half-time show and a stalker. (Sorry, no graphics.)

The game narrative is fun and always full of surprises. You never know when a player may get crunched or when someone might throw Howard Cosell on the field. It is fun to throw the bomb on the first down just to confuse the other guy (the computer). The game plays extremely well and is error-free. If you have a 48K machine and enjoy football, you will enjoy this game.

*(Micro Pro Systems, Route #2, Box 533, Cumming, GA 30130; \$19.95 32K disk, \$22.95 48K disk.)*

Mark E. Renne  
August 1982

★ ★ ★ 1/2

**Super Nova**  
**Big Five Software**  
**Model I and III**

Super Nova is a machine-language program with some of the fastest graphics I've seen so far. Asteroids glide across the screen in all directions, and aliens shoot at your spaceship as they go by. When you hit a meteor it splits into fragments, and you've got to beware of those, too. When you are hit by a meteor or by alien fire, the screen flashes, and

your ship is reduced to rubble.

One or two people can play this thoroughly enjoyable game. Action is fast, and the game is challenging.

My only criticism is that it is hard to tell in which direction your ship is facing. I've been able to tell the front end only by firing a missile. With all the work that went into creating this program, a little more work could have been expended toward making the direction of the ship more easily recognizable.

But after becoming familiar with this interesting game, I'd still buy it.

*(Big Five Software, P.O. Box 9078-185, Van Nuys, CA 91409; \$15.95 cassette, \$19.95 disk.)*

Carl A. Kollar  
August 1981

★ ★ ★ ★

**Swamp Wars**  
**Instant Software Inc.**  
**Model I and III**

Swamp Wars reenacts a hostile encounter between a tramp space racer called the Stellar Spaniel and the indigenous population of an unnamed planet in the Bragthos sector.

The local aborigines—known to the less informed as the Slizards and Muck Monsters—are lazing around the lean-tos one afternoon when a fellow who is out washing his car happens to see the Spaniel land on one of the planet's nine islands. He naturally calls in a report to the Mutual Aid, and soon just about everyone has heard the news on their scanners.

As it turns out, the Spaniel is



only stopping for repairs. But the townies have had previous troubles with bounty hunters and mercenaries, and know that the rest of the solar system has as much use for them as a bucket of cold farts on a rainy day. So they mobilize their SWAT squad to go out there and deal with the situation.

But life ain't no chair of bowlies. The ship's captain, with the improbable name of Legion J. Muldoon, is a crafty SOB, and soon has four 'droids out combing the islands for transporters left behind by a bunch of missionaries. He hopes to use the transporters to repair his ship, and book.

The Slizard SWAT team isn't bright. In fact, there's a warrant on the agenda for the next town meeting to raise the entrance standards. They come charging up the island hell-bent for bongoburgers, and Muldoon cleans the beach with their faces. He even picks off a few Muck Monsters, who shoot faster and with greater accuracy, before sending his 'droids to another island.

Naturally, the locals are upset. This sort of thing costs a bundle

in ritual bereavement ceremonies. So they send out another squad, with instructions to stay low and kick ass.

These guys do a little better, backing one of the 'droids into a public restroom and turning him into swamp gas. But the 'droids bob and weave, transport from one end of the island to the other, and generally make themselves difficult targets. The result is the same, and the Honor Guard deep-sixes 18 more shoe boxes.

These first two confrontations set a pattern that continues from island to island. The Slizards are beginning to feel like they are riding mopeds through the Lincoln Tunnel. The Muck Monsters start to pack their station wagons and make hotel reservations in Miami Beach.

But somewhere around island five, the action picks up. The Slizards kick out the jams and get serious, wasting two more 'droids. Muldoon is tired of pushing all these buttons, and the remaining 'droid is complaining about heartburn. He is about ready to toss a mud ball at the mother ship when a Muck Monster sneaks up from behind and grinds him into silicon dust.

That's it, of course, for Muldoon, who drinks six bottles of Scope and dies of minty breath.

The Slizards and Muck Monsters dismantle the ship and use the parts to repair their tractors. Muldoon's boots and the six empty Scope bottles are sent to the Museum of Modern History, where they are put in the basement and forgotten. The story makes the wire services, but the public loses interest when the Pope arrives for his first tour. (*Swamp Wars* is available from



*Instant Software, Peterborough,  
NH 03458; \$14.95.)*

Eric Maloney  
August 1981

★ ★ ★ ★

**T80-FS1 Flight Simulator**  
**Sublogic Communications**  
**Corp.**  
**Model I and III**

The T80-FS1 package is a reworked version of a program originally designed for the Apple II, improving the characteristics of the game/simulation. Before you can think about playing the included war game (British Ace), you must first learn how to fly the aircraft.

The pilot finds the plane sitting on the approach end of runway 27 (that is, facing due west or 270 degrees), awaiting departure. The pilot has a view looking out through the forward cockpit. There are two views possible: One is out the front of the cockpit, and the other is from on top of the aircraft looking down.

This aircraft is not meant to be an F-15 Eagle. Rather, it is a Sopwith Camel with some of the more sophisticated equipment of the present day.

The flight controls consist of a number of keys on the TRS-80. You must hold a key down more than an instant to assure that the computer has read your request. Failure to do so can result in turn rates, for example, that can lead to a dead man's spiral.

Control of the aircraft on the ground is different from that in the air. On the ground, you steer the plane like a car. In the air the kinematic effects of lift, gravity, drag, and momentum are brought into play.

There are instruments for you to learn, as well as three indicators for the game: score, bombs, and ammo. The bombs are released by the X key and the machine-gun ammo by the space bar.

The program, to its credit, is both a game and a simulation. But the game is a real-time game. By hitting the W key, the war is on. There is no turning back.

The more you experiment with the program, the more skilled you become in flying the simulation. Happy landings. (*Sublogic Communications Corp., 713 Edgebrook Drive, Champaign, IL 61820; \$25 on cassette, \$33.50 on disk.*)

John Lindsay  
August 1981

★ ★ ★ 1/2

**Tanktics**  
**Avalon Hill**  
**Model I and III**

Avalon Hill's historical simulation games require hours or even days studying battle manuals and maps and working out intricate strategies. Tanktics is better than any previous simulation. It uses a mapboard, markers, and a battle manual, but does not take days to play.

The board has 768 numbered hexagons superimposed over a countryside map. The object is to defeat the computer's tanks, using one of five different scenarios and several combinations of tanks and antitank guns; although you have up to eight tanks, the computer gets twice as many.

The computer plans its moves, referees yours, determines combat results, and tells you when

your tanks have been sighted by the enemy. You never know when the enemy will appear or where they are after an attack. The cassette version loads in two parts. It can be run from disk with a utility such as Twodisk from the Alternate Source.

Avalon Hill has succeeded in keeping the best part of their board games and letting the computer do the tedious part for you.

*(Avalon Hill, Dept. C-10, 4517 Harford Road, Baltimore, MD 21214; \$24 cassette, 16K or 32K.)*

David G. Bartlett  
April 1982

### ★★★ **Temple of Apschai Automated Simulations Model I or III**

Like its brothers in the noted Dunjonquest series, Temple of Apschai is a computerized offshoot of fantasy war-gaming. Dungeon and Dragon players will feel at home as they bargain with the Innkeeper for supplies and set forth into the four-level abandoned temple.

The game consists of two Basic programs, Innkeeper and Dunjonmaster, and four data bases that represent levels within the temple. The cassette documentation contains details on transferring the data bases to disk.

Innkeeper designs a character and equips him or her for the upcoming battles. All characters have six attributes: intelligence, intuition, ego, strength, constitution, and dexterity. The values assigned to each govern a player's fighting effectiveness.

When the adventurer leaves the inn and enters the temple, Innkeeper loads Dunjonmaster, which reads the appropriate data base and displays the first segment on the screen. Even when the programs and data have been loaded onto disk, this is a lengthy process.

Dunjonmaster works in real time; the monsters do not wait for the player's response before striking. Fortunately, player's can select the monsters' response speed.

The graphics are unexciting but acceptable. All the monsters are identical—sort of a graphic cross—while the character is a graphic "V" with the point indicating the direction of travel. The initial drawing of a room when the character enters is painfully slow. Thirty to 40 seconds might not seem like much, but when you are on the track of fame and fortune, it is an eternity. A machine-language routine called through a USR routine might be more effective.

At a time when most documentation appears to be written by an illiterate chimp, the Temple of Apschai manual is a welcome change. It is almost as much fun to read as the game is to play.

Aside from detailed playing instructions, the manual contains two short stories that give background for the adventure and describe the treasures and monsters that abound within the walls.

A major drawback to Temple of Apschai is that the cassette-based program cannot remember a character. Players are required to write down the characteristics, weapons, armor, and treasure so they can

describe their character to the innkeeper. The most effective technique seems to be to have a partner keep track of treasures and magic healing potions as well as read the descriptions and make a much-needed map.

Temple of Apshai brings the excitement of role-playing games to the world of home computers. Buckle on your sword, don your armor, and type Run. (*Automated Simulations, P.O. Box 4247, Mountain View, CA 94040; \$39.95 cassette and disk.*)

John Warren  
February 1982

★ ★ 1/2

**Tower of Orlandor**  
**Compu-things**  
**Model I and III**

Tower of Orlandor, a new graphics-based adventure program, is an invisible maze graphically depicted from an overhead view.

The game begins in an empty square, with the player represented by a blinking graphic block. Using the arrow keys, you blunder around in the dark until you hit a wall of the maze, which lights up, showing its location. Some walls have traps that, if activated, deduct life points from the player.

The tower has 10 levels. On each level is a hidden trap door that, once found, lets you pass down to the next floor until you reach the ground floor and (presumably) safety.

Much is left to the imagination. Missing is the verbal interaction of traditional adventure, the frustrating three-dimensional maze images of the

Med-Systems series, or even the room shapes of Gauntlet of Death. There is the feeling of being lost in a dungeon (and there are treasures and potions to obtain), but author Dave Huntress would have us extrapolate every detail of the game except that which appears sparingly on the screen from time to time to tell us we've hit something or found something.

At \$14.95, the program is aimed at the younger crowd, who need a game simple enough to play but complex enough to hold their attention. For the sophisticated Adventure/Deathmaze/Trapmaze player, this program will fall short of his expectations.

(*Compu-things, Suite #2, 270 Broadway, Revere, MA 02151; \$14.95 cassette, \$59.95 disk.*)

Aaron Silverstein  
August 1982

★ ★ ★ ★  
**TRS-Opera**

★ ★ ★ 1/2  
**Word Challenge**  
**Acorn Software**  
**Model I and III**

This TRS-80 voice program includes five songs, all classics that are quite familiar. Only one of the songs appears to be full length, probably due to the memory restrictions involved in presenting a fair selection while keeping the program within the 16K margin.

By using very fast alternating tones, amazingly full single voice sounds have been produced. What appears to be two notes is really one note quickly followed by another. The tone quality is good, for a TRS-80.

In my opinion, TRS-Opera is one of the most ambitious musical programs written for the '80.

Word Challenge is a cross between the conventional game of hangman and the television game show, Wheel of Fortune. It uses sound and graphic combinations to capture the interest of the user.

The game can be played by one or two players. In the one-player mode, the computer supplies the phrases to be guessed from its store of about 25. These phrases are good, but few. In the two-player mode, you can create your own phrases, but you can't save them. This is unfortunate because you could exchange data with a friend or use the program for educational purposes.

There are minor problems in two delay loops; the screen clears before you have a chance to read it properly. However, the sound quality in this program is just as good as that in TRS-Opera. Many of the sounds are akin to those used in most arcade games, and are done meticulously.

Word Challenge is a good program, particularly for those of you with school-age children. (Acorn Software Products Inc., 634 North Carolina Ave. S.E., Washington, DC 20003; these products are no longer available.)

Darren DeVigili  
August 1981

**Two Heads of a Coin**  
(See Page 100)

### ★ ★ **Video Checkers** **Compuquote** **Model I**

Video Checkers, according to the vendor, plays checkers according to the international rules. The graphics of the playing surface are well done. I like the fact that the board is displayed continuously. A message on the right side of the screen indicates whose turn it is and also prints each move as it is made.

One bug shows up occasionally. After the computer makes its move, it sometimes leaves a letter in the space where you enter your move. This results in an illegal move message one or more times before the letter is removed. The program is also very slow—it takes 40 seconds to determine a move.

Despite the good graphics, the program doesn't play an intelligent game. Video Checkers invariably makes stupid mistakes, and I've never lost to it. One time, it had two pieces ready to move in to be kinged; instead it moved in front of me and set up a double jump. (Compuquote, Canoga Park, CA 91305, could not be reached for an update.)

Rod Hallen  
February 1980

### ★ ★ ★ **The Voyage of the Valkyrie** **Advanced Operating Systems** **Model I and III**

Graphics and sound can make or break a computer game; both are done very well in The Voyage of the Valkyrie.

The game itself is no innovation. You are the pilot of the ship Valkyrie; your mission is to capture the island of Fugloy. The island has 10 castles, each guarded by a different number and type of bird. Your first task is to map the island. Roughly two hours of trial and error gives you the map, but be prepared to die a few times in the process.

You must shoot the birds to capture the castles. The elite Fugloy Airforce can appear anywhere on the island. These birds shoot back and some are more powerful and quicker than others. Each shot consumes valuable energy points. You must also keep an eye on your shield level.

To shoot the birds you must aim crosshairs that appear on the screen when you put on your shields. How well you aim these crosshairs bears directly on your success at the game.

There are 10 levels of play, with each successive level putting more birds into the fight. Once you have mastered the art of shooting the birds, however, the game only becomes longer with each increase in difficulty level.

The graphics are undoubtedly the best feature of Valkyrie. The birds move unpredictably (at first) with their wings flapping and shots flashing. The castles are all different and are equally impressive. Few games can boast TRS-80 graphics as good as these.

When you first begin to play, it is difficult to shoot the birds. Eventually patterns emerge and the game becomes much easier to play. Once you map the game and develop your shooting technique, the game ceases to be

a challenge.

The game remains entertaining because of the graphics and an unusual use of sound (selections from Richard Wagner's operas). The game begins with the march from "Tannhauser," and "The Ride of the Valkyrie" (my favorite) plays each time you capture a castle.

The Prelude from Act III of "Lohengrin" signals that you have conquered the island. There are the usual blips and blaps, as well, but they are uninspiring.

I like this game, but I can't recommend it to anyone who wants a challenge everytime it is played. It is quickly mastered, and can become tedious. The graphics and sound redeem what would otherwise be a mediocre game.

*(Advanced Operating Systems, 450 St. John Road, Michigan City, IN 46360; \$34.95 cassette, \$39.95 disk.)*

Michael E. Nadeau  
May 1982

★ ★ ★

**Win21**

**Discovery Bay Software Co.  
Model I**

Win21 is a blackjack program that can be played as a game pitting you (and up to six of your friends) against the dealer. But there is more to it than that. It is intended to be used as a tutorial in winning at a real blackjack table.

Included with Win21 is a 200-page paperback book by Edward O. Thorp, Ph.D., entitled *Beat the Dealer*. This book presents a theory and system for regularly winning at blackjack.

Use Win21 along with the book to improve your card skills. You can decide whether you will make your own moves, follow a computer-suggested course of action, or let the computer play for you. You can also choose to play with from one to four decks. With more than one deck, the shuffling time is quite long—one of the drawbacks of complex programs written in Basic.

The graphics are not dramatic, but they are effective. An outlined square is provided in front of each player for his bet, and then the cards are dealt face down or face up as required.

It's claimed that the theory is sound and that you can win if you follow it. However, if I set the program to play automatically and let it run for a long time, I always lose.

Win21 is an interesting program, and anyone who plays blackjack for money might benefit from it. In any case, it's fun to play.

*(Discovery Bay Software Co., P.O. Box 464, Port Townsend, WA 98368; \$29.)*

Rod Hallen  
February 1980

★ ★ ★ 1/2

### **Winged Samurai Discovery Games Model I or III**

You are a Japanese Squadron Commander defending a New Guinea naval base in the spring of 1942. The Allied bombers are still 28 miles from their target when they are spotted by 16 imperial A6M2 fighters (the dreaded Zero).

In this game, you can choose one of 13 different fighters, each with its own flight character-

istics, for your mission. You will be facing 15 types of Allied planes.

The action proceeds as you type in commands to your squadron. There are 25 two-character commands. After you enter a command, the sideview is updated with information and you can make other command decisions. Your goal is to destroy all the bombers before they reach your base.

The game is easy to learn, but difficult to play because of all the possible aircraft combinations. Each game is different. In *Winged Samurai*, as in real aerial combat, chance plays a major role. This is simulated by including a random element into all encounters. Thus, for no apparent reason, your fortunes may turn sour rapidly.

*(Discovery Games, 936 W. Highway 36, St. Paul, MN 55113; \$19.95 cassette.)*

Art Little  
August 1981

★ ★ ★ 1/2

### **The Wizard Programs Unlimited Model I**

I first played *The Wizard* at a New York Computer Show. Doubtful that this program was different from games I have played and shelved in the past, I reluctantly selected a subject. Suddenly, all hell broke loose; the TRS-80 started to make strange noises, the monitor began to flash, and I found myself in a battle against time to answer the questions correctly. I was there for some time trying to prove I was smarter. It was a draw.

The Wizard is reminiscent of coin arcade games that ask you a series of questions on various categories in exchange for points and a free game. You can play The Wizard alone, against the clock, or against another player. When two people play, you are not only pitted against the clock, you also play against your opponent, who can steal your questions or force you to answer questions he rejects.

The Wizard is more than a game; it is also educational. The real power behind The Wizard is the utility that comes with it allowing you to program in your own questions and answers on any subject you want. This utility is easy to use.

The Wizard would complement any program library and is well worth the asking price. (*Programs Unlimited*, 125 S. Service Road, Jericho, NY 11753; \$19.95 on disk.)

Joe Simon  
May 1981

### **Word Challenge** (See Page 135)

★ ★ ★ 1/2

#### **Wordo** **Micro-Fantastic Programming** **Model I and III**

A cross between Scrabble, Jumble, and Hang Man, Wordo provides more than ample opportunity to test your knowledge of five-letter words.

Wordo can be played by one or two persons. The program provides a list of over 1,000 five-letter words, from among which it randomly chooses one word for each play. You enter a five-

letter guess word. The computer adds this word to your list and indicates how many of its letters are also in the game word. However, it does not indicate which letter in your word is contained in the five-letter-word category.

After each guess, you can enter or eliminate letters known to be, or not to be, in the game word on the "scratch pad" at the top of the screen. You are limited to 30 guesses, after which the word is displayed on the screen.

The scoring method, determined by the number of players, is based on either accumulated points or on the average number of words entered per game. The score is carried over and modified from game to game.

Wordo is not a cinch. You'll be surprised at how many five-letter words you don't know, or at least you don't commonly use. To win the game, your challenge is to find several words with letters that are not in the game word; this is not always as easy as it sounds.

In desperation, you may cheat, as I did. The computer accepts nonsense words that are in the form of true words. It rejects nonsense words that do not follow an English vowel-consonant pattern, and words of less than five letters.

Wordo is not geared for children. There is no way to direct the computer to select words appropriate for a child. As a matter of fact, there is no way to direct it to select words appropriate for adults who have low frustration thresholds, either.

Those of you with patience will find this game interesting and challenging. Have fun!

*(Micro-Fantastic Programming, New York, NY 10001 could not be reached for an update. Words originally sold for \$14.95.)*

Debra Marshall  
April 1981

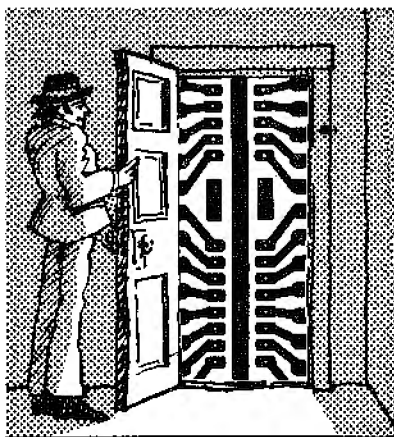
★ ★ ★ ★ 1/2

**Zork**  
**Personal Software Inc.**  
**Model I**

Zork is a complicated and sophisticated underground kingdom whose treasures lie both above and below ground. One of its greatest treasures is its ability to react to fairly complex commands. You are not limited to a vague, general activity because the program recognizes only two-word commands.

Zork is a joy to play. There are no graphics; the screen carries a description of your surroundings and the directions of travel that are available to you. Your number of turns and points acquired are also on the screen at all times.

It is for you to decide where to go, and how to get there. Ex-



plore, but remember that directions change; just because you went north to enter the chasm doesn't mean you can leave it by travelling south. And, please, watch out for the pickpocket.

Zork contains many areas, and consequently should be mapped. Because of its complexity, mapping will take time, and a thorough exploration of each site will take longer. A save procedure marks your place so you can avoid having to start from the beginning each time you play.

A 15-page instruction book accompanies the disk. The book gives background, directions and explanations of the game, and also describes in great detail loading and saving on both one- and two-disk systems. The directions for initializing the TRSDOS disk are also included.

Mysteries abound in Zork. I'm not sure that it can be solved—there is so much to know that the possibilities seem staggering. Since the program allows you to do pretty much what you want, it leaves open marvelous opportunities for those of us who love to break into things to see how, and if, they work. Above all, be ready to ask questions and to giggle. The program has answers, and also a sense of humor.

The booklet describes Zork as Part I; I truly hope that means we can expect a second part sometime soon.

*(Personal Software Inc., of Sunnyvale, CA, has changed its name to VisiCorp. Zork I is now available from Radio Shack for \$39.95. Zork II and Zork III are distributed by Infocom, 55 Wheeler St., Cambridge, MA 02138. They may be purchased*



through the Zork User's Group,  
P.O. Box 20923, Milwaukee, WI  
53220 for \$39.95. Model I  
and III.)

Debra Marshall  
August 1981

Worth, TX 76102; \$9.95.)

Dan Keen  
Dave Dischert  
April 1981

---

## GRAPHICS

---

★ ★ ★

**Dancing Demon  
Tandy/Radio Shack  
Model II and III**

Dancing Demon features two animated-graphics, demonstration programs for the TRS-80. With this program you enter individual notes to compose a song, then type in dancing instructions and choreograph a tap dancer to the music.

Letters correspond to musical notes. Each note is followed by a click to simulate the dancer's tap shoes. The machine writes the timing, while you select the tempo. Speed selection ranges from 1-255.

Once your song has been typed in, you can create a dance routine by entering various letters and numerals that represent different steps and their counts. The dancer can be made to move left or right, jump, stomp left or right, squat, stand, or even spin.

We enjoy the big production these programs put on. The curtain rolls up and a figure dances to the music, taking a bow or two at the end of his performance. So sit back and let your computer entertain you!  
*(Tandy/Radio Shack, Fort*

★ ★ ★ 1/2

**ElectraSketch  
Macrotronics Inc.  
Model I and III**

I love my ElectraSketch. It's a creative addition to my TRS-80 trappings, and it's inexpensive.

It contains six files, and when transferred to disk they enable me to create graphics, store pictures on disk, retrieve pictures from disk, animate graphics, vary animation speed, obtain printouts, draw line vectors, fill in backgrounds, and intersperse alphanumeric with graphics.

When creating your own displays, you do have to keep track of the remaining RAM, making sure that your Basic program fits into a reserved spot.

The package comes with clear, point-by-point instructions to lead you through the 17-step loading process.

I recommend ElectraSketch as an inexpensive, practical approach to computer graphics. *(Macrotronics Inc., Turlock, CA 95380. This product has been discontinued.)*

Joseph H. Cowen  
January 1981

★ ★ ★

**ENHBAS  
Cornsoft Group  
Model I, II, and III**

ENHBAS is a graphics package similar to Level III's, and more. It comes in both tape and disk formats. Disk ENHBAS comes with versions for 32K and

48K machines on a transfer disk with no operating system.

The disk version of ENHBAS uses high memory, while the tape version uses low memory. Thus, only one tape version is needed for all memory sizes from 16K-48K.

During use of ENHBAS, clear acts as a control key, and most standard control functions are supported. If a lowercase modification is present, clear in conjunction with the numbered keys provides special characters such as brackets.

Several features of ENHBAS make use of sound, provided a suitable amplifier is connected to the cassette output jack. A click is generated whenever a key is pressed, errors return a short two-tone beep, break or control C has a high-pitched tone, and control G plays the Winchester Chimes.

Thirty-five shorthand entries are provided. By pressing a combination of clear/shift, and a letter or number simultaneously, a Basic key word is generated. While saving keystrokes, this is a bit cumbersome.

For some reason, the H key is undefined and I was unable to find it documented. Choice of implemented key words could also be better. Many keys are dedicated to special ENHBAS key words and some lesser-used Disk Basic functions (MKD). Some common key words are omitted, for instance, GOTO.

Sorts, one of the most powerful features of ENHBAS, are handled through numeric and string arrays singly dimensioned. Key designates the array to be sorted; Tag designates arrays carried along with the Keyed array.

Other features of the utility include several new branching commands, graphics commands, and some new constants. Exec allows execution of a string expression as if it were a program line, and ZSTEP allows pausing between execution of every program statement. Hitting any key during a pause causes the next statement to be executed.

Several new error messages are included. The tape version of ENHBAS, in addition to the new features, brings some of the features of Disk Basic to tape users. Included are MIDS, INSTR, and Line Input. Also &H, &O, and &B allow the use of hex, octal, or binary constants in numerical expressions.

All in all, ENHBAS appears to be an excellent utility for serious programmers who need more programming power than the available Disk Basic versions can provide. I would definitely recommend it to anyone who is nostalgic for the fast graphics enjoyed back in the tape days with Level II Basic.

*(ENHBAS, currently being revised, is not available at this time. Cornsoft Group, 6008 N. Keystone Ave., Indianapolis, IN 46220. The package was originally priced at \$59.95 Model I and III, \$99.95 64K Model II.)*

Ronald H. Bobo  
January 1982

★★★★  
**G.E.A.P. 1.0**  
**(Graphics Editor**  
**and Programmer)**  
**J.F. Consulting**  
**Models I and III**

G.E.A.P. (Graphics Editor and

Programmer) is a Basic graphics program that allows the programmer to manipulate all or part of the screen.

G.E.A.P. has four modes. Regular mode is signified by a flashing dot cursor. The cursor can be moved in any direction with the arrow keys in any combination. If the shift key is held as the cursor is moved, a line is left in the trail of the cursor. To erase the line, you rewrite the line without the shift key. The regular mode is activated by turning the power on, and can be reached from all other modes (except the print mode) by hitting the period key.

The second mode is the print mode. Its main function is to put text on the screen, and it is entered by hitting the asterisk. To exit, you must first enter the keypad mode by hitting the clear key. Once in keypad mode every other mode is readily available by hitting the appropriate cursor key.

The keypad mode is available at the touch of the < key or by the clear key, if you are in the print mode. To exit the keypad mode, hit the cursor key of the mode you want.

The fourth mode, the designated mode, is entered by hitting the hyphen key. A flashing hyphen is used as the cursor, and it will not erase as it moves. This mode is the guts of the G.E.A.P. program. While in this mode, you can select any area of the screen, reverse it, move it, tilt or rotate it, multiply the views of it, magnify it, fill it in, shrink or expand it, and save or cancel it. You can draw half of a screen, duplicate the half and merge it into an entire screen.

There are nine options

available while in all modes except the print mode. These options are entered by hitting the number nine. One of these options, Dump Program, lets you save your creation or reload another for re-editing. The Dump option is compatible with disk, cassette, or Exatron's Stringy Floppy.

You are also given the option of saving the screen in compressed or noncompressed code. You can save a designated figure in a Basic string. You can even specify the line numbers for screen storage so you can merge it with a program more easily.

G.E.A.P. can incorporate input statements with user-specified or default variables. Graphic animation is possible by updating only the moving part of the screen. You can insert delay loops, or even create your own modules and expand on G.E.A.P. if it doesn't have the feature that you need.

*(G.E.A.P. 2.1 is available from RCM Computers, 221 Hirshfield Drive, Williamsville, NY 14221; disk \$49.99.)*

Richard C. McGarvey  
March 1982

★ ★ ★ ★ 1/2  
**G.E.A.P. 1.3 and 2.0**  
**J.F. Consulting**  
**Model I or III**

G.E.A.P. (Graphics Editor and Programmer) is designed to permit the use of intricate video and printer graphics and text combinations so nonprogrammers can produce professional-looking output. It is to graphics what a word processor is to text. This review is based on versions 1.3 and 2.0.

G.E.A.P. is a utility program that allows easy control of the cursor, drawing graphic displays anywhere on the CRT, moving sections of the display around the screen, the integration of text into the screen, and programming input statements into the screen display.

It can also magnify, rotate, tilt, and multiply the views of any part of the display; duplicate part of the screen so only half a picture need be drawn; shrink, expand, and break apart portions of the screen; and reverse the display, changing all or part of it from light on dark to dark on light.

Although it requires a brief familiarization period, G.E.A.P. can be used effectively by non-programmers the first time out. It also has options so the experienced programmer can write expansion modules to meet his needs.

The program's documentation is very good. Its step-by-step style starts with the basics and slowly guides you into the more advanced G.E.A.P. features.

G.E.A.P. has four main operating modes. The first is called the regular mode. It is active when you turn the computer on, and, among other things, it accesses the menu. The second mode is the print mode. Its main function is placing text on the screen.

The keypad mode, the third mode, draws graphics characters. The final mode, the designate mode, is the heart of the G.E.A.P. program. With this mode, you can select any area of the screen; designate it in various ways; and then reverse it, move it, tilt or rotate it, multiply the views of it, magnify

it, fill it in, shrink or expand it, and save or cancel it. You can draw half of a screen, duplicate the half, and merge it into an entire screen.

Expansion modules are optional with version 1.3 and standard with version 2.0. They are special function programs that overlay part of G.E.A.P. memory without disturbing the program. They allow the use of the keyboard to type letters that are medium-sized, large-sized, and cubed.

Additional modules let you use the Epson MX-80 printer to reproduce the screen graphics on the printer and also allow the integration of G.E.A.P. and Newsprint for producing some unique text and graphic effects.

Due to the use of expansion modules, an increase in memory size is not necessary to expand the G.E.A.P. program. Version 1.3 is compatible with cassettes or disks; however, if you have a 48K disk system, you should get version 2.0. This version has all the features of version 1.3 plus several more that give you the full advantage of the disk system.

Version 2.0 is compatible with LDOS, DOSPLUS, TRSDOS, and NEWDOS80 1.0 and 2.0. I assume G.E.A.P. will also work with other common DOS types.

G.E.A.P. may appear to be a game creator or a toy rather than a workable tool. That's not the case. G.E.A.P. is limited in business applications only by the user's imagination.

G.E.A.P. is functional, fun, and worth the money. It is the most powerful utility of its kind. *(G.E.A.P. 2.1 is available from RCM Computers, 221 Hirshfield Drive, Williamsville, NY 14221;*

version 2.1, \$49.95.)

Richard C. McGarvey  
October 1982

★ ★ ★ ★

**Powerdraw  
Powersoft  
Model I or III**

Powerdraw is one of the new generation of screen graphics utilities for the TRS-80. The program lets you create graphics screens and either print them or use them in your programs.

It's easy to use and you can save your creations in any of six different formats on disk or tape. The documentation is excellent; it explains all commands in great detail.

The program is in machine language and uses either joystick or cursor controls. It has two different modes: text and graphics; all commands can be used in either mode. The graphics mode lets you draw on the screen using the arrow keys. This mode has several special functions, including mirror-image top-bottom and right-left, reverse, and flip sides.

Move is a powerful command that can make animation seem effortless. By entering the Move command and pressing a direction key, you can change the location of the entire screen in any direction. Text is entered by simply typing at the locations desired and can be screen-edited. Screens can be sent to the printer using any of three screen-printer drives within the program.

Powerdraw makes designing program covers easy as well as creating cartoons to reward

children using educational software. There are few drawing assignments that this program could not handle easily.

The only thing missing from the program is a way to convert the screen into a string-pack format. If you try to edit the existing code and change Prints to AS=; the line will no longer execute. This is a minor bug.

The overall performance of Powerdraw is great; I recommend it to anyone interested in a screen editor.

*(Powersoft, 11500 Stemmons Expressway, Suite 125, Dallas, TX 75229; \$39.95.)*

Mark E. Renne  
October 1982

★ ★ ★ 1/2

**SECS  
(Screen Edit Control System)**

★ ★ ★ 1/2

**Sigmon  
Datasoft Inc.  
Color Computer**

The Screen Edit Control System, SECS, is one of the most useful tools available for color graphics. SECS is two programs: one in machine language and the other in what I would call Color Graphics Basic.

With SECS you can produce any 8-by-10 dot-matrix graphics character you desire. A group of 64 characters can be created and used at any one time. The characters and programs generated can be saved to tape.

An Editor program allows you to make corrections, additions or deletions at any location on the screen.

Sigmon is a 6809 machine-code monitor program that gives

you 18 powerful commands including Dump, List, Disassemble, Assemble, Find, Move, Step, Break, Set, Go, Write, Read, PLOAD, PRT, NOPRT, Modc, Exit, and Speed. Sigmon comes with a 14-page instruction booklet and a source listing. The program is written in code that can be relocated. It is 6K long, and located at the top of memory and can be protected by Basic's Clear command.

The monitor is easy to use, and the instruction manual is well-written and informative. Even though it is a large program, it is a good one and contains the capabilities a serious programmer needs. (*SECS and Sigmon, by Datasoft Inc., 19519 Business Center Drive, Northridge, CA 91324, have been discontinued.*)

Douglas R. Cook  
March 1982

---

## HOME APPLICATIONS

---

★ ★ ★  
**Deluxe Personal Finance**  
**Small Business Systems**  
**Group**  
**Model I, II, and III**

The Deluxe Personal Finance is an extensive budget program to be used in an individual household. The system is in the form of a mini-general ledger and includes an accounts-payable program.

With the system, you can record 900 check transactions

annually, allow nine budget categories per check, provide summaries of income versus spending by month, assign expenses to 28 categories, and generate end-of-year reports for tax purposes, among others. The program runs on two disk drives and needs 32K of memory.

The program maintains a good set of records, but only if they are utilized. Often, there is a tendency to use a budget program awhile and then stop.

However, if you do use Deluxe Personal Finance, all data you need at income-tax time will be right at your fingertips. (*Small Business Systems Group, 6 Carlisle Road, Westford, MA 01886; Model I \$35, \$50 compiled, Model III \$50 compiled, Model II \$75.*)

Reese Fowler  
May 1981

★ ★ ★  
**H-O-R-K-S**  
**Edu-Ware East**  
**Model I and III**

H-O-R-K-S (Home Office Record Keeping System) is a bookkeeping program for those with complicated financial lives. It doesn't balance the checkbook or reconcile the bank statement, but it organizes financial records so you can find out where the money goes as well as where it comes from. The minimum system required is a Model I or III with 32K and one disk drive; the Model I holds about 800 records, the Model III about 2,000.

Before using the program, you name up to 33 credit and 33 debit accounts. Credit accounts might be salary, interest, or

dividends; expenses include all the categories for itemized deductions on an income tax return, as well as standard household costs. Names can be changed if necessary; any dollar amounts in an account will remain after a name change.

After the program is initialized with account names, transactions can be entered. Each account name and its code number is displayed on the screen; you enter the transaction date, item, amount, tax, and account code number. A transaction might correspond to a check you've written, or a cash expense, or income. After the entry is complete, you're allowed to correct any errors. Later, a file can be corrected using the audit function; the audit feature records all changes made in a file, storing the old and new information.

Several printouts or screen displays are available, such as summaries by account, transactions by month or for entire file, and profit/loss statements. The program tallies the tax on any item separately.

This is a useful program, but there are some aggravations, which are apparent when you use the audit file. The more changes you record, the fewer expenses/credits will fit on the disk, so it's important to keep audit changes to a minimum. Unfortunately, it's easy to add audit changes unintentionally from inadequate error trapping in the program.

For example, the date must be entered as dual digits separated by a comma (12, 31, 81). If you forget this format and type 12/31/81, the program scrolls the screen and prints two question marks, looking for the rest

of the date. If you compound the mistake by typing the whole date, you get an "extra ignored" message and the date will be garbage. Fixing this error at the end of the transaction leaves out the two top lines, which show the account names and code numbers. The only way to correct a date error is to delete the transaction and retype it. Unfortunately, this is learned by trial and error, and each trial is recorded by the audit function.

Select printouts one at a time; you cannot ask for printouts of all expense accounts and then walk away. Paper must be adjusted manually between each printout. There is no sort, so entries are printed in the order they were typed.

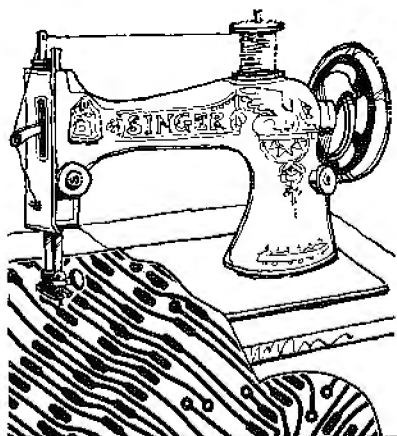
In a business environment, this program would cause too many errors and is too awkward to use. It is marketed for the home office, and as such, is a good buy. For the money, the program is very useful, and could help organize a complicated personal or sideline business/hobby budget. (*Edu-Ware East, P.O. Box 336, Maynard, MA 01754; \$29.95 disk.*)

Wynne Keller  
September 1982

★ ★ ★

### **Personal Finance Tandy/Radio Shack Color Computer**

The Personal Finance Pack helps you keep track of 26 personal-budget categories. The program loads instantly when you turn on the computer after inserting the ROM Pack in the slot.



The screen displays a listing of 26 budget categories common to the average person. The list can be personalized and remains in effect every time you begin a new session. You can change categories at any time.

The program accommodates up to three separate bank accounts and balances. Your expenses can be recorded as they occur or at a convenient regularly scheduled time. Cash purchases can also be recorded in your budget.

The program has one-key commands to move up or down the screen to list the budget categories, to change from one menu to the next, replace check numbers, advance to the next check number, and to cancel checks and deposits.

After each session, the data can be stored on cassette tape. You can store up to 1,918 checks on a Color Computer with 16K of memory. Three copies are automatically made on the same tape to reduce chances of loading errors.

This program is easy to use and well documented. Its one drawback is that you can't print

your budget. Nevertheless, it's a good buy when you consider that setting up your budget will help you organize your priorities and understand your spending habits.

(Tandy/Radio Shack, Fort Worth, TX 76102; cat. #26-3101, \$39.95 Program Pack.)

Carlos Calle  
October 1982

★ ★ ★

### **Small Home-Business Software Blechman Enterprises Model I and III**

These programs, useful in the home or in small businesses, are designed for the Model I with 16K and an 80-column printer. They do not require disk drives and are compatible with the Model III following a few line changes.

Speed Letter, a mini word processor, is simplicity itself. Handling both upper- and lowercase, it features 10 single-letter commands and can pack 150 lines of text into 16K of memory.

The program operates on lines of text rather than on individual words or characters, but within this single constraint, allows text reviews, line correction, replacement, insertion, and deletion. You save text to tape or stringy-floppy. Margins and page lengths are defined just prior to printing. Printed output is line selectable, but vertical spacing can only be achieved by adding blank lines to text. The programs appear to be crashproof.

The 12-Column Ledger maintains and prints financial records for income tax and other pur-



poses. All column headings are displayed throughout data entry. This software performs all necessary arithmetic and produces a clear printout.

The Three-Across Mailing List accepts up to 220 names and addresses (in 16K), sorts data according to your preference, and prints addresses in three-column format on standard mailing labels. Each label contains four lines of 24 characters.

The Auto Dialer uses a \$3 telephone interface relay to dial the number of any of 500 names held in memory.

Each of the four programs are available separately, but the combined price is an exceptional buy.

*(Blechman Enterprises, 7217 Bernardine, Suite M, Canago Park, CA 91307; \$49.95.)*

David Smith  
November 1981

ing you 11,264 bytes of usable RAM when in the Level I mode. It is also compatible with most printers and includes driver software. The documentation is excellent.

Although it is a useful tape, Basic-1P does have limitations. Since it was designed as a resident interpreter in high RAM to act like a Level I Basic ROM, it does not accommodate most Level I machine-language programs. Also, Level I programs exceeding 11,264 bytes do not fit into the memory available when Basic-1P is resident. The entry point in RAM is not controlled by the user, but is programmed with the machine language.

*(Basic-1P is sold by Small Systems Software, P.O. Box 366, Newbury Park, CA 91320, which could not be reached for update information.)*

Fred Blechman  
May 1980

---

## LANGUAGES

---

★ ★ ★  
**Basic-1P**  
**Small Systems Software**  
**Model I**

This program provides full TRS-80 Level I Basic capability for Level II 16K (and up), adds two Level I commands, LLIST and LPRINT, and contains two additional commands that Level II Basic doesn't have, LPRINTON and LPRINTOFF. Basic-1P is a machine-language tape that loads into the top 4K of RAM with a System command, allow-

★ ★ ★  
**CBasic**  
**FMG Corporation**  
**Model I**

The CP/M disk operating system by Digital Research has been adapted to many systems. I chose the one by Don French's FMG Corporation. The system has the CP/M disk and CBasic.

CBasic is a compiled Basic, not interpreted, and this has its advantages. First, you use the CP/M editor to write the program that lets you create a Basic source-code file and edits it as necessary. The only line numbers you need are labels for GOTO, GOSUB, and so on. Since there aren't any line numbers, you almost have to indent con-

tinued lines to quickly tell if they are continued lines or a new line.

The statement keywords are mostly the same as those in TRS-80 Disk Basic with a few exceptions. For example, you have to DIM every array; there is no default. However, you can REDIM an existing array without getting an error.

CBasic has only integer and real numbers. Real numbers have 14 digits of precision. The first 31 alphanumeric characters of a variable are significant, permitting long variable names with the first two characters the same as another. CBasic has no graphics and is fussy about the difference between integers and real numbers. You must be consistent.

This Basic has no file statements, and I like the CBasic files better. There are fewer statements to remember, and files can have fixed or dynamic record lengths.

The variable record lengths save a lot of programming, and they are the great advantage of this program. They save the complicated stepping through the 255-byte TRS-80 Disk Basic record to find the record you want. You can't believe how often this eliminates program bugs until you try it.

The second major advantage of CBasic is its compiler, which catches all the syntax errors for you. You can correct errors before you run the program.

One disadvantage of the program is that CBasic data takes up more room, but the ease of accessing data from the variable file lengths far outweighs this. Changing a CBasic program requires that you edit the source code (call the editor and load the

file into memory), recompile the program, and run it again, usually several disk changes in a long program. This, too, is a disadvantage.

A versatile system and fairly easy to learn, CP/M DOS includes many more options than TRSDOS. Two of the manuals contained information on how to customize CP/M for your system, but I found them heavy going.

The CP/M DOS is a powerful system, and once you learn all the features, it rivals a large computer's operating system. The beauty of the system is its effectiveness at the beginner's level, while challenging the experienced user.

The CP/M editor has several advantages. With it, you can find a given group of alphanumeric characters anywhere in the program on any line with a single command. You can search for every occurrence of a specific group of alphanumeric characters and replace it with another group.

The CP/M assembler is an 8080. I generally have trouble with assemblers, but this one works well and is easy to use. It has the usual options, pseudo-ops, and rules for syntax. The DDT is a debugger with many useful features. It replaces the TRSDOS program DISKDUMP, placing a named file into memory and reading it out in blocks, hex, and ASCII.

An idiosyncrasy of the compiler is that it only prints out lines up to 80 characters long in its listing. It sees and correctly compiles the remainder of the line but doesn't list it. This is a nuisance when the compiler gives you a syntax error in a part of the line it doesn't display. The compiler runs slowly, but it

prints out all errors it finds when the compilation takes place.

It's easier and faster to create and debug a program using a Basic interpreter, but a compiled program runs faster and the initial weeding out of syntax errors saves later debugging problems.

Disk routines are more easily written in CBasic. I like the system but hate the time it takes to go back to the editor to edit a program, recompile it, and run it again after you make a change. I also miss being able to access the variables during program debugging by simply breaking the program and printing the variables. But that is the essential difference between a compiler and an interpreter.

*(FMG Corporation, P.O. Box 16020, Fort Worth, TX 76133; \$150.)*

Ken Knecht  
April 1980

★ ★ ★ ★ ★

**CCForth**  
**Frank Hogg Laboratory Inc.**  
**Color Computer**

The CCForth package is an example of the software beginning to appear for Color Computer users. Forth is a stack-oriented interpreter and one of the lower-level languages around. Low level means that certain functions (such as floating-point numbers) aren't available as standard features in the language. This isn't as bad as it sounds and is outweighed by the speed advantages offered.

Stack-oriented means that the language is structured around a last-in, first-out (LIFO) stack as used in machine-code programs.

Numbers are computed by pushing, pulling, and otherwise bullying the numbers in this list. For example, there are commands that allow numbers to be duplicated, rotated, swapped, and other desirable functions.

The fact that it's interpreted gives you the same advantages as languages such as Basic. Code can be entered, edited, tested, and debugged without converting it to machine code and handing it over to the processor. Thus, if your code contains a bug, it won't be wiped out by a crash (although in Forth, if you try hard, you can do it). If you don't want to stick with the interpreter, you can drop down into machine code. For this purpose, CCForth contains a 6809 assembler enabling you to link machine code with the interpreted CCForth code.

The highlight of the Forth language is that you can define your own words and can enter them into the Forth dictionary and save them if required. Subsequent use of the new word causes it to be interpreted as if it were a standard feature.

It doesn't stop there though; if you want to define another word, you can use the word you just defined as part of the new definition. In effect, you can create your own language and in doing so, you raise the language from lower level to higher level. This gives you great power and flexibility. If you don't like something in Forth you can just rewrite it!

CCForth comes on disk with a user's manual. The disk contains the interpreter and some goodies to experiment with. These include games, music generators, an assembler, source

listings for two editors, and various samples for a total of nearly 90 screens.

Accomplished users of Forth find themselves on familiar ground as CCForth was written around the Forth-79 standard published in late 1980. The only unfamiliar sight can be the use of screens 32 characters wide instead of the usual 64. This was the only sensible course in view of the Color Computer's screen limitations. Screens are the method Forth uses to store material on disk.

The user's manual is a gem. My small gripe is that although this manual is the right size (8½ × 11 inches), it comes as a bound book with the pages glued at the spine. It's impossible to insert updates.

As you become more familiar with the language you can use the glossary at the back of the book instead of searching through the examples. In fact, the whole manual becomes more serious towards the latter half, a touch that makes the book functional as both a tutor and a reference.

Frank Hogg Laboratory Inc. appears to have the best intentions as regards user support. A second version was rushed to early purchasers of the package when it was realized some preproduction releases of the software had found their way onto the market. This bodes well for future support. I heartily recommend this package. It will win a lot of support for the Forth language, and justly so.

*(Frank Hogg Laboratory Inc.,  
130 Midtown Plaza, Syracuse,  
NY 13210; \$99.95.)*

Jake Commander  
November 1982

★★★

### **Color Pascal Computerware Color Computer**

With this compact compiler, you can explore the concepts of structured programming for which Pascal, the most successful new language of the '70s, is so well known. This is an integer Pascal of limited capacity, the price you pay for a 16K machine. Still, enough of the flavor of the language comes through to let you explore some of its novel control structures and data types.

The compiler comes on one side of a short cassette with three demonstration programs on the other side. You also receive brief user's manuals from Dynasoft of Nova Scotia (developers of the compiler) and Computerware (its modifiers for Color Computer use). The booklets form a guide to the syntax of this particular dialect and to Color Computer memory usage; they are not Pascal references, and the user will not get the full benefit of this package without obtaining one of the many complete works now available.

In the 16K version, Color Pascal uses the Power Pack's Monitor program to perform I/O operations. The compiler, which loads with the monitor's L command, consists of two programs: the Supervisor and the Editor. The former is the more general control program; you use Editor for creating and modifying the source code of a Pascal program.

Entering the E command gets you into Editor; to enter source code, you must get into the in-

sert mode, which you do by entering "I" followed by a decimal number for the number of lines. You can use an arbitrary number, like 9999, to reserve an arbitrary amount of workspace. You then enter your source-code statements, indenting with the space bar as appropriate for the various levels within the program. A major departure from Basic is the absence of line numbers; the Pascal Editor uses an internal line pointer to indicate which line of code you are working on at any time. You can move this pointer throughout the workspace by Up, Down, Top, and Bottom commands, and use it for editing, adding, or deleting lines of code.

After entering source code, press Break to get back to Editor's command mode. The Dynasoft manual refers to the use of control-C for this job, while the Computerware manual does not describe the procedure at all. The editor's Q command returns you to Supervisor. Save your source code first—the compilation process destroys the code in memory—and you are ready to compile.

If errors are detected during compilation, the compiler prints a one- or two-digit error code and upward-pointing caret. Depending on the nature of the error, however, the caret may actually point to the line following the one in which the mistake occurred. Experience in interpreting the cryptic messages (with the aid of the Dynasoft manual) is necessary.

Since the display scrolls continuously during compilation, it is impossible to keep track of error messages in any but the shortest programs. The value of

a printed listing immediately becomes clear. The program documentation, though, is of little help. The easy way out is the Power Pack Monitor's Echo command which duplicates everything on the screen on a printer connected to the RS-232 port; invoke the Monitor by hitting Reset, and reenter the Pascal compiler by the Monitor command "J 0700."

One of the most enjoyable things about Basic is the relative ease with which you can format program outputs. Color Pascal is less convenient in this respect. For the new user, the manuals are useless.

Color Pascal uses two output statements, Write and WRITELN, which generate a return and line feed. Thus, using WRITELN in one of Pascal's several types of loops generates a new output line for every pass through the loop, just like a Print statement in Basic. Use Write in the same situation, however, and successive outputs will be printed on the same line, one after another. A new line only starts when you reach the end of your display's or printer's line length.

Color Pascal assumes that every integer expression is six spaces wide, unless each integer variable or expression in an output statement is followed with a colon and specification of the minimum field width to be used. The starting point is always at the left margin, unless you incorporate into the output statement's argument a character variable defined in the program to have the value 32, which is the ASCII code for the space operation, and repeat the variable in the output list as

many times as there are spaces required.

Similarly, you can insert blank lines into an output list by using WRITELN without an argument, or Write with a character variable argument which you have assigned the value 13 (ASCII for line feed with carriage return). A character variable assigned the value 8 generates a backspace on the video display, although dumb printers like my Line Printer VII do not recognize it. These are the only video control codes the Color Computer recognizes; however, you can use the same technique to include printer control codes.

What about input? It turns out that Color Pascal's Read statement, READ(X), handles numbers in a straightforward manner. Just type in your number and hit Enter, as usual. Character variables are read into an array one character at a time.

Color Pascal is nothing more than an introduction into the world of structured programming. The 16K version represents a real accomplishment on the part of its author(s), but requires an investment in time and effort if you are going to use it for complicated programs. (Computerware, P.O. Box 668, Encinitas, CA 92024. Color Pascal is currently only available for 32K Color Computers; \$49.95 cassette, \$50 disk.)

Scott L. Norman  
September 1982

★ ★ ★

**Colorforth**  
**Armadillo International**  
**Software**  
**Color Computer**

★ ★ ★ ★ 1/2

**Starting Forth**  
**Leo Brodie**  
**Prentice-Hall**

Colorforth is an implementation for the Color Computer's 6809 microprocessor of FIG-Forth, the standard promoted by the Forth Interest Group. The documentation is a 12-page Armadillo Software pamphlet and a detailed FIG glossary. The pamphlet contains details on changing the memory maps for disk and cassette systems, as well as a few special features of Colorforth. The FIG publication includes the manual for the source-code text editor.

### **Starting Forth**

There's no way a stranger to Forth could learn the language from this material. Armadillo Software suggests you purchase Leo Brodie's book. There are tutorial reviews, problems with solutions, and carefully marked sections for novices and relative experts.

The pages of *Starting Forth* are inhabited by marvelous cartoon characters who illustrate the functions of Forth words and features. None of this, however, hides the fact that Brodie has produced a serious book. It is best read at the keyboard where you can test the examples interactively, since certain sections are geared to Forth Inc. products and discuss features not relevant to Colorforth. In most respects, *Starting Forth* is excellent in both style and content, and I recommend it.

### **Colorforth**

The Colorforth software comes on cassette, with versions for

16K systems with or without disk drives. There are complete instructions for making cassette back-ups and for modifying and storing disk and cassette versions for 32K systems.

There is also a discussion of the simulated disk screens used by cassette-based systems. Forth uses virtual storage, shuffling 1,024-character screens between disk and RAM as needed. Colorforth reserves RAM for the purpose of simulating such storage units: four of them in 16K, 12 in 32K.

By using the starting and finishing addresses specified in the documentation, you can store the simulated screen's contents on tape so your newly defined text isn't lost at the end of a terminal session. You can also define new words without using a screen, although you can't use the text editor to correct errors. Words that compile correctly become part of Colorforth's dictionary, at least for the duration of the session. The Colorforth pamphlet also gives the addresses used in saving a dictionary that has been extended.

### **Text Screen Operations**

You can define a whole series of colored backgrounds if desired. The best way is to use the text editor by entering a whole group of definitions on a screen. Remember, words defined this way don't have to be sequential statements in any one program. When using a disk, prepare the system for screen input by entering "empty-buffers," followed by "disk." The command Editor invokes the editor; screen -1 can then be loaded with blanks and selected for editing by 1 CLEAR.

A text screen consists of 16 lines of a maximum of 64 characters each. The lines are numbered 0-15, and you enter text on line -n by prefixing your material with nP.

The FIG documentation discusses the text editor. It performs most functions you expect from an editor, although some operations aren't as convenient. For example, it takes two commands to change one text string into another: X oldtext, C newtext.

### **Dealing with Numbers**

Colorforth initializes in decimal, just as Basic does. It does have a built-in hexadecimal capability, however, and it's easy to define the system you want and have Colorforth operate in any other base.

You can switch between decimal and hexadecimal with the words Decimal and Hex. These are toggled commands, so exercise caution if you change bases in the middle of a routine.

Colorforth is set up to do fixed-point arithmetic only, using signed 16-bit numbers (32767 to -32768). This is a general property of Forth, even when used on 8-bit microprocessors. Some double-length operations are available, and it's possible to write floating-point routines, but this is one reason Colorforth wouldn't be the language of choice for scientific number crunching.

To enter a double-length number, you punctuate it with a period; you can enter 100000 as 100.000, for example. Colorforth also has formatting commands similar to Print Using for outputting large numbers.

Although the range of

numbers usable for calculation is limited, some operators do interpret 16 bits as an assigned number (0-65536). As you'll see, this is what you need to address memory-mapped I/O in the Color Computer.

### **Comparison and Control Structures**

Colorforth has a fairly complete set of comparison operators, which are used to control program branching within If. . . Then. . . Else statements. Because of RPN, the syntax of these commands takes some time getting used to, although the ideas are familiar enough from Basic. Here are Color Basic and Colorforth program fragments that accept a number from the keyboard, test it to see if it's positive or negative, and print the results:

#### **Color Basic**

```
100 INPUT N
110 IF N>0 THEN PRINT
"POSITIVE" ELSE PRINT
NEGATIVE
```

#### **Colorforth**

```
:TEST O>IF, "POSITIVE" ELSE,
"NEGATIVE" THEN:
(Used by entering N TEST)
```

Notice that in Colorforth, the words If and Then span the possible branches the program might take, but not the decision-making portion. That comes before the If, in the form required by RPN; for example, (N>0) becomes (NO>), where, in my example, N is put on the stack from the keyboard. This example also introduces dot-quote, the word used to print literals. Neither example considers the possibility that N equals zero.

In the Basic example, control passes to the statement following line 110 after a message has been printed. Similarly, Colorforth passes to whatever follows Then. There is a difference, of course, in that the Basic If. . . Then could have altered program flow by using statement numbers instead of the Print commands. Because Colorforth is a structured language without line numbers, there are no other words within the If. . . Then to execute actions other than printing, however. After they are executed, control eventually returns to the word following Then.

Colorforth comparison expressions can be quite lengthy, as they are in Basic. Simple algebraic comparisons can be logically combined, using AND, OR, or XOR operators.

Color Basic has only one kind of repetitive structure: the definite For. . . Next loop where the index's lower and upper bounds are specified. Colorforth has a similar construct, called a DO loop, plus a pair of indefinite forms, the Begin. . . Until and Begin. . . While. . . Repeat loops. The DO loop can't be executed interactively; it must be compiled into a word, although you can specify one or both limits from the keyboard when the word is executed.

### **Constants, Variables, and Arrays**

Although Colorforth uses the stack for temporary storage, it also lets you name constants and variables for future reference. The naming phrases are: Variable (name) and (value) Constant (name). These phrases are not colon definitions. It is not



necessary to specify the functions of Variable and Constant, they're predefined words, and their machine code takes care of that.

Storing data in an array or retrieving it is more complicated than in Basic, because Colorforth doesn't support subscripted variables. You specify the first storage location by using the array name, and then add an offset to get to the desired location.

Colorforth can also handle multidimensional arrays. You must decide on a convenient method for converting multiple subscripts to single address offsets. Brodie's book gives a nice example for two-dimensional arrays, which wind up being stored column-wise. All the elements of the first column, starting with the uppermost, are stored, then all the elements of the second column, and so on. This is actually the way many high-level languages treat such data structures.

### **Something Completely Different**

Here are a couple examples of the way you can use Colorforth with the Color Computer's I/O structure.

Anyone who has used the Color Computer's joystick knows about memory address 65280, which is PEEKed to test the status of the trigger buttons. You can give this address a name, such as Port: 65280 CONSTANT PORT. Colorforth interprets this properly. The word Looksee prints the contents of this location: LOOKSEE 2 TIMER PORT C@:.

Entering Looksee from the keyboard causes the system to wait two seconds and then

prints the contents—255 or 127 if you press neither joystick trigger, 254 or 126 for the right trigger, and so on. Since a single memory location can only contain an 8-bit number, the proper fetch command is C@. It is easy to incorporate Looksee into an If. . . Then statement, causing a program branch, rather than having it print and stop.

While Port is a constant whose value is put on the stack when used in another word, there's nothing wrong with this value representing the address of something else. Remember, if Port is defined as a variable, its address is put on the stack, not the address of the trigger-button sensor.

The final example uses this sort of labeling, and is less trivial: a sound generator. This program generates a square wave at approximately 500 Hz and routes it to the Color Computer's composite video output. If you're interested in this sort of thing, you need detailed reference material. I recommend the Radio Shack reference or service manuals, or the Facts book from Spectral Associates.

### **Summary**

Colorforth may seem complex when compared with Color Basic, but it's closer to Assembly language in the level of detail required. Remember, with Colorforth you get a higher degree of control over machine operations than Basic offers, coupled with high speed. I haven't covered text manipulation, but the words for sophisticated text I/O are present in the starting vocabulary. This is where I should encourage you to get the software and the book and find out for yourself.

Just one more thing: You can apply the bit twiddling used in the definition of Soundoff to program the SAM and VDG circuits for graphics.

*(Armadillo International Software, Austin, TX 78712; \$49.95. Prentice-Hall, Englewood Cliffs, NJ 07632; Softcover, 348 pp., \$15.95.)*

Scott L. Norman  
December 1982

★ ★

### **Compiler Basic Tandy/Radio Shack Model I or III**

This software is not a Basic-language compiler. It is Compiler Basic, an entirely new type of Basic that is designed to be compiled by this program.

Radio Shack says that this compiler is business oriented and intended for use as a tool for the development of new software. They are serious about this and unless your goal is new business software, look elsewhere.

Compiler Basic is composed of several programs that work together in the development and operation of a compiled program. The first part is RSBasic, the Basic language that the Compiler recognizes. It is similar, but not identical, to Level II Basic. RSBasic does not support many Level II instructions. RSBasic also has many unique commands.

The difference between Basics is only a minor inconvenience. Before buying it, ask yourself if you will use the compiler enough to warrant learning the new Basic. RSBasic is not com-

patible with Level II Basic; you can't compile Level II Basic programs without making substantial changes.

The second part is the run-time program, which you must have in the computer before you can run any compiled program.

The third part of Compiler Basic is the compiler, a function of RSBasic. The compiler is invisible to the programmer. With the run-time, DOS, and RSBasic programs in memory, a 32K machine has about 1,500 bytes of programming room left. A 48K machine would have a little over 16K of programming room.

The fourth part of Compiler Basic is the TRSDOS operating system, version 2.3B. It is not compatible with any other DOS. Radio Shack supplies an upgrade module that will permanently change any previous TRSDOS to version 2.3B.

If you use a different operating system, forget it. If you have implemented speed modifications on your computer you must disable them to get the slow TRSDOS to operate.

The package contains additional modules such as BEdit, an editor module for editing the Basic source code. These modules are mostly controlled by RSBasic and are fully covered in the documentation.

Expecting that Compiler Basic would work more quickly than Disk Basic, I compiled a short program to test the speed. While this simple program was not a good test of the compiler, it did give interesting results. The compiled program took 24.5 seconds to run. The same program under Disk Basic took 12.6 seconds.

If speed is your goal, stay

away from this compiler. Speed is not the goal of Radio Shack's Compiler Basic, and it does not appear to be a secondary benefit.

Operation is fairly simple. With features such as the Auto command, you can enter your Basic program from the keyboard. If you have a Basic program already on disk, you can load and finish it in RSBasic. RSBasic loads Level II Basic programs if they are on a 2.3B disk, but it will not compile them.

One drawback is that the so-called compiled version you save on disk is not really compiled. The compiler in RSBasic compiles the program into intermediate code. Each time you want to run your program you must run it with the run-time program that compiles the intermediate code into final code. This leads to some problems when you try to sell your programs.

I do not rate this program highly for flexibility. First, it recognizes only the exact syntax of RSBasic. If you have spent a great deal of time writing Level II programs, expect to spend the same amount of time to make them work in RSBasic.

Second, you are restricted to TRSDOS 2.3B. No other DOS works. You cannot move the finished program to a different DOS. Third, you must have the run-time disk to run the finished product. This limits the salability of the product you create by establishing a minimum price of \$20. Your profit is then tacked on top of that.

If Radio Shack has intended to create a complicated way of making Basic programs secure from theft, they have succeeded. With Compiler Basic you can

write a Basic program that cannot be listed. The original source code will be almost totally indecipherable.

Is the program valuable to the average programmer? To answer this, consider its cost, flexibility (discussed above), security (a person purchasing your program can't alter it but can make limitless copies), use (designed for developing business-oriented software), and size (a 48K machine can compile only a 17K program).

Radio Shack's Compiler Basic is a poor investment. To be fair, the program seems to function well and it appears to be properly documented. If you need this type of program, then Compiler Basic is a good risk. It does exactly what it claims.

*(Tandy/Radio Shack, Fort Worth, TX 76102: \$149.)*

Richard C. McGarvey  
February 1982

★ ★ ★ ★ ★

### **Extended Color Basic Microsoft, Tandy/Radio Shack Color Computer**

Extended Color Basic for the Color Computer is a powerful language rivaling Disk Basic in its repertoire while lacking some features of Level II.

All arithmetic functions are carried out in nine significant figures and it is amazingly fast, especially in graphics functions. You can draw a rectangle in a few tenths of a second and fill it with a different color in two seconds or less.

Extended Color Basic offers these new instructions to 8K Color Basic: ATAN (same as Level II); CIRCLE (for high-

resolution graphics); CLOAD M (loads a machine-language program at 1500 baud); COLOR (foreground and background); COS (same as Level II); CSAVEM (writes a machine-language program to tape); DEF FN (same as Disk Basic); DEF USR (same as Disk Basic); DEL (same as Delete in Level II); DLOAD M (loads machine-language programs at 300 or 1200 baud); DRAW (graphics instruction); EDIT (Level II, but with a few important differences); EXP (same as Level II); Exponentiation (same as Level II); FIX (same as Level II); GET (puts a section of a high-res screen into an array); HEX\$ (converts decimal to hex); INSTR (returns the starting point of a substring within another string); LET; LINE; LINE INPUT (same as Disk Basic); and LOG (same as Level II).

The instructions also include: MID\$ (same as Disk Basic); PAINT (fills an area of the screen); PCLEAR (clears high-res graphics memory); PCLS X (sets the screen background color); PCOPY (save graphics pages); PEEK (same as Level II); PLAY (sound); PMODE (select resolution); POKE (same as Level II); POS(X) (returns the current cursor position); PPOINT (high-res POINT); PRESET (resets a point to the background color); PRINT USING (same as Level II); PSET (high-res SET); PUT (places an array set up by GET on the screen); RENUM; SCREEN (retain contents of screen when switching modes); STRING\$ (same as Level II); STR\$ (same as Level II); SQR (same as Level II); TAB (same as Level II); TIMER; TRON-TROFF (same as Level II); and VARPTR (same as Level II).

The well-written instruction manual includes plenty of exercises and examples. There are 21 complete graphics and sound programs, as well as a memory map with entry points for a number of useful ROM routines.

When using the higher resolution graphics it is not possible to put characters on the screen. They must be drawn with Draw instructions. In high resolution, the pages of screen memory are retained and can be recalled or erased by PCLS.

For its price, the Color Computer with 16K and Extended Color Basic is a bargain, especially if you use it for games, graphics or educational purposes.

The main differences between Extended Color Basic and Level II or Disk Basic are in the memory map and the way Basic tokens are used. Whereas Level II and Disk Basic use only one number to store an instruction, Extended Color Basic uses up to three. Line numbers and pointers are stored the same as Level II except the order is MSB, LSB for the 6809E.

Some Level II instructions I miss are On Error GOTO, ERL/2 + 1, Edit, DEFINT, DEFSNG, DEFDBL and the serial printer output is slower than parallel.

Extended Color Basic does have upper- and lowercase capability (for the printer only) and can access Videotex with inexpensive software.

The 32-character by 16-line screen is a disadvantage, but is easy to read. For some, the joysticks are a plus. The cassette system has not given me a poor or difficult load in over 300 tries and is three times as fast as Level II and six times as fast as Level I.

*(Extended Color Basic Upgrades are available from your local Radio Shack store for \$99. A minimum of 16K is required.)*

Franklyn D. Miller  
June/July 1982

★ ★ ★ ★

**Extended muMATH  
Microsoft  
Model I and III**

muMATH is a symbolic mathematics package for microcomputers. An abbreviated version of this package has been available for some time, but is much weaker than the CP/M version. Microsoft finally released the extended version and it is worth the wait!

muMATH is written in muSIMP, and RLISP-like cortex parser for an internal Lisp structure. It performs exact arithmetic (to 611 digits) and does not normally attempt to reduce fractions— $3/7$  remains  $3/7$ , not some floating-point approximation like 0.4285714. Hence, muMATH arithmetic is exact. Extended muMATH can print answers in decimal notation to any number of digits you like.

muMATH allows mathematical manipulation of symbols, like X, without assigning them a value (or binding). The use of unbound variables gives muMATH its great power. You can assign non-numeric values to a variable. You can bind A to  $(X + 2*Y)/Z-Q$ . You can evaluate A by temporarily assigning X, Y, Z, and Q numerical values, using the EVAL or EVSUB functions.

The extended muMATH package has several enhancements to the original abbreviated TRS-80 version including:

- allowing decimal output of numerical results in addition to the standard rational arithmetic output;
- an enhanced integral calculus package for more sophisticated integration;
- fast loads and saves of muSIMP environments to and from disk;
- a Limit package to evaluate limits;
- a Sigma package to evaluate sums and products (including infinite sums and products when combined with the Limit package);
- a complete Matrix package including matrix addition, transposition, inversion, and division;
- an equation-solving package for solving various types of equations (including simultaneous equations; when combined with the Matrix package);
- a Trace package used for debugging muSIMP and muMATH functions;
- EDIT80, a line and character-oriented editor for creation of your own function files for muMATH;
- interactive lessons on disk for learning to use muMATH;
- an improved manual; and
- all the files (with some updates) that came with the abbreviated TRS-80 version.

Other enhancements include MEMORY(X,Y), which is similar to POKE X,Y. TRS-80 graphics functions are supported including CLS to clear the screen; Cursor to position the cursor; and Point, Set, and Reset to test, turn on, or turn off any graphics pixel on the video screen. TRS-80 ROM routines can be called either by assigning the jump vectors correctly, or with the PUTD command. You should use this capability with care

however; the ROM has many exits to Disk Basic, and if one is taken, it's off to Neverneverland.

Overall, I am quite impressed with the expanded version. It is powerful and can handle more sophisticated integration problems, compute limits, finite or even infinite sums and products, and contains very powerful matrix manipulation abilities.

The package lacks a function to save ASCII files from the muMATH environment. This would let you debug functions within muMATH and then save them directly on disk. Further, it would be nice to be able to save the functions in the distilled D-code onto disk for quicker loading. Currently, single functions and entire packages must be loaded and are incrementally compiled during the load. The exception is entire muSIMP environments that can be saved and loaded.

A complete tutorial on muSIMP in the manual would be helpful as none are available anywhere on muSIMP or its cousin RLISP. And better distinction could have been

made in the differences in usage between Function and Subroutine. The former is call by value and the latter is call by name.

Even with these reservations, the expanded version of muMATH is one of the few programs that, by itself, warrants purchasing a computer.

Who should buy this package? Parents and educators wanting to teach mathematics to students' students who want to learn more about mathematics in an interactive environment; professionals who work with mathematics on a daily basis, and would like to save time performing time-consuming mathematical manipulations; and the curious, who want to know more about the topics of artificial intelligence, computer algebra, structured programming, or symbolic mathematics. (Microsoft, 10700 Northup Way, Bellevue, WA 98004; \$250.)

Bruce Powel Douglass  
November 1982

★★★★  
**GRBasic 3.0**  
**Med Systems Software**  
**Model I and III**

GRBasic integrates with any operating system to provide enhanced graphics and sound capabilities for your Model I or III. It can be a useful tool for the development of games and even some plotting programs. With it you can view graphics in a variety of perspectives and create useful line, circle and arbitrary shape-drawing commands, as well as sound-producing commands.

With GRBasic you can draw shapes from a shape table or en-



code a shape as a set of numbers and POKE it into a block of memory (a 127-byte block is reserved for this).

The commands fall into four categories: line drawing—LDRAW(R) and LDRAW(S); circle drawing—CIRCLE(R) and CIRCLE(S); shape drawing—SDRAW(R), SDRAW(S), Size, and Turn; and sound generation—Audio.

When you execute GRBasic from your operating system, it automatically loads in Basic and begins Basic for you. The commands are integrated into the interpreter's parser, so you only need to type in the command, with the appropriate syntax and parameters, to execute it. If you have improper parameters or have incorrect syntax, you are informed through the normal Basic error routines.

Besides the GRBasic programs, you also get MIS-SILE/BAS, a demo program, and SEDIT/BAS, a shape-table editor, written in Basic.

I received a disk copy, which includes a version for each memory size. It comes on a non-system disk, so TRSDOS users may send a TRSDOS disk and they will place GRBasic on it.

The manual is well-written and reproduced. It's about 20 pages long and has a very useful table of contents.

As good as GRBasic is, I would like to see the following improvements: Change the Turn command to allow rotations by other than 45-degree increments; allow the user to input three-dimensional objects and have GRBasic display the two-dimensional isoclines of the figure (projection of the figure onto a plane); and change the

Audio command so you can play musical scales, both major and minor.

GRBasic is a quality product backed by a company that provides excellent service and support.

*(Med Systems Software, P.O. Box 3558, Chapel Hill, NC 27514; cassette \$19.95, disk \$24.95.)*

Bruce Powel Douglass  
June/July 1982

★ ★ ★ 1/2

### **ICL (Interactive Control Language)**

**XYZT Computer Dimensions Inc.**

#### **Model I and III**

ICL is a simple job control language for TRS-80s with NEWDOS 2.1 or NEWDOS80 and 32K or 48K of RAM.

ICL is an interpreted language that controls various DOS commands and functions and allows you to program any order of their execution. The simplest example of a use for ICL is running batches of programs that require no intricate program intervention or supervision. Who wants to sit around and load programs and watch them execute? ICL frees you from supervising your computer.

The five divisions of ICL commands are I/O, I/O support, ICL interpreter, execution, and control interception.

ICL is semi-structured in that it supports procedures, labels, and line numbers, as well as variables. Before a given ICL line is executed, all variables are parsed and their values are substituted into the expression.

You can display text

messages, read data from the keyboard, clear the keyboard queue, cancel a running program, specify a number of options, output text to the printer, place text in the keyboard queue, or suppress display of text. The options mentioned include: disable common video display driver, toggle the display of the current ICL line being interpreted, turn on/off the keyboard queue, allow for single-byte keyboard requests, copy display output to the keyboard queue, enable/disable non-alphanumeric characters to be present in data taken from the queue. ICL also returns an error code, displaying the execution state of the last procedure.

The 65-page ICL manual includes a useable table of contents, an alphabetical reference section of the commands and pseudo-ops, examples to aid learning the language, and an Advanced Techniques section for developing skill in more interesting uses of ICL.

ICL is sufficiently powerful to automate many of the more tedious procedures that we all have to perform on our computers. It is a good product and may be just what you are waiting for.

*(XYZT Computer Dimensions Inc., 2 Penn Plaza, Suite 1500, New York, NY 10121; disk \$59.)*

Bruce Powell Douglass  
June/July 1982

★ ★ ★ ½  
**Level III**  
**Microsoft**  
**Model I**

Level III Basic is not just another Basic, but an actual enhancement to the versatile

Level II Basic. It is attractively packaged and accompanied by an excellent instruction manual as well as a convenient fold-up card containing all statements, commands, and functions of both Levels II and III. Level III statements are printed in red on the card.

As soon as Level III is loaded, there is more reliable tape loading and no more keyboard bounce. There are many one-letter abbreviations and some even include enter. Other features include LSET entries that are user-changeable, line renumbering, spelled-out error messages, one-operation date and time setting, and conversion of hexadecimal numbers.

Level III graphics capabilities are considerably advanced over those of Level II. Two graphics modes are used in Level III. In character mode, the screen is divided into a grid composed of 64 character positions across by 16 down. In graphics mode a finer grid, measuring 128 across by 48 down, is used. Three new Level III graphic statements can be used in either mode, and the graphics themselves are fast.

Level III Basic is a worthwhile acquisition for any serious TRS-80 user, especially one who would like the power of Disk Basic without the added expense of a disk drive. With all the extras not contained in Disk Basic, it becomes desirable even for a disk owner.

*(An updated version of Level III Basic is sold by Microsoft, 10700 Northrup Way, Bellevue, WA 98004. It is now available for the Model I and III, disk \$49.95 or cassette \$29.95.)*

Ronald H. Bobo  
May 1980



★★★★

**Lisp**  
**Supersoft Inc.**

★★★★

**UOLISP**  
**Far West Systems and**  
**Software**  
**Model I or III**

Lisp stands for list processor. Lisp is the language of choice in artificial-intelligence programming, because of its flexibility, power, and ability to work with symbolic expressions. The two Lisps reviewed here are good implementations of the language; they do, however, take different approaches.

Supersoft's Lisp is not a "tiny" Lisp, but a complete version allowing full recursion. It has a PROG construct to allow using sequential types of programs written in Lisp. Lisp runs in 16K with cassette systems and supports single-precision, floating-point arithmetic and functions. It uses the Basic ROM routines to do this, which accounts for considerable reduction in the size of the Lisp interpreter.

The package provides a few abilities beyond normal Lisp packages. Basic transcendental functions, Explode and Implode functions, PEEK and POKE, LPRINT, and an Editor function are all included in Supersoft's Lisp package.

This Lisp is excellent for beginners and advanced programmers. It is easy to use, and many TRS-80 functions and floating-point routines are supported, adding to its flexibility.

UOLISP is not quite as complete an implementation as Supersoft's Lisp, but it has other

attractions. The extras that come in the UOLISP package include a Lisp interpreter, a Lisp compiler, a Meta language writing system, an RLisp compiler, a Trace function, and a small Lisp editor. UOLISP doesn't support floating-point numbers, Explode, Implode, or Expand functions.

The Meta Translator Writing System sits on top of the Lisp interpreter. The intended use for Meta in the UOLISP package is for language-design experimentation. Meta supports BNF-like syntax, recursive-descent parsing schemes, lexical primitives, pattern-directed code generation, and syntax-error message generation. Examples are given, and a calculator-like language is implemented before your very eyes. If you've ever wanted a simple way to translate Fortran into Basic or Pascal into PL/C, this is the proper environment.

Another feature of UOLISP is that it contains an RLisp parser. RLisp is a cortex syntax for Lisp. It provides Lisp with a more understandable syntax. It contains a parser that reads in RLisp syntax and parses the expressions into Lisp format, and then passes it over to Lisp to evaluate. It is easy to move back and forth between RLisp and Lisp.

Although UOLISP lacks floating point and a few Lisp functions, it is fairly complete and easy to use, once you figure out how. If you have experience in Lisp and like to experiment, UOLISP is a great choice.

*(Lisp is sold by Supersoft Inc., P.O. Box 1628, Champaign, IL 61820; \$100 disk, \$75 cassette. UOLISP is marketed by Far West Systems and Software,*

P.O. Box 3301, Eugene, OR  
97403; \$109.)

Bruce Powel Douglass  
December 1982

★ ★ ★ ★  
**MMSFORTH**  
**Miller Microcomputing**  
**Services**  
**Model I or III**

Forth is the latest language for the TRS-80 from Miller Microcomputing Services (MMS). It is a hybrid language, combining the clarity of a high-level language like Basic, with the speed of low-level language programming in Assembly.

Unlike Basic and most other high-level languages that separate the programmer from machine code, Forth lets the programmer maneuver back and forth between high-level easy-to-use programming techniques, and direct assembly-language programming. Thus you can interchange programs between different processors using Forth, with only minor alterations to specific routines.

Forth uses a technique known as Indirect Threading Code. This means that Forth is a stack-oriented programming language. It is composed of a number of Assembly-language modules, or primitives, each with a specific task or purpose. These primitives are each named and can be selectively called up or combined into words. It is this technique that gives Forth its tremendous power.

Unlike other programming languages, the ability to define new words, or primitives, in the Forth dictionary is always

available to you through its catalog function. And there is no run-time penalty for these new words. For example, CLS is a Forth word made up of two primitives—home cursor and clear to end of page. So whenever you want to clear the screen, you enter the word CLS.

Another example is List. List is used to put a page of memory on the video. Now suppose that you want to clear the screen each time before you put the new page on the video. You can enter CLS and then 10 LIST (to list page ten of memory), or you can define a new word that uses these two words in its own definition, calling it CLIST, like this:  
:CLIST CLS LIST.

From then on, every time you enter 10 CLIST, you get a clear screen and the new page 10 listed. And CLS and List are still available for use separately.

If you want to save added new words for future use, it can easily be done using the command DWTSECS (include the proper parameters, of course).

New primitives can be defined in Forth using the built-in Forth Editor Assembler, and added to its catalog, if your present version of Forth doesn't have the proper primitives or predefined words to do the task you have in mind.

There are three stacks used in Forth. The most commonly used stack is the parameter stack, which starts at the high end of memory and works its way down. This particular stack's function is for user inputs and outputs, and is also used as a scratch pad by the various words and primitives. The second stack is the return stack, which also starts at high

memory and grows toward low memory. It is used primarily by Forth as a loop counter. The last stack is a hardware stack used by the Z80 processor and is not normally available to you in programs.

Because of these stacks, permanent memory locations don't have to be assigned to temporary variables. This cuts down tremendously the amount of overhead memory required to run Forth.

One very nice mathematical feature of Forth is its ability to switch from one number base to another without any run-time penalties. All the Forths support the standard bases of HEX, OCTAL, and DECIMAL.

### In Summary

Forth is a very versatile language, combining the best sides of both high-level and low-level languages, excluding many of their disadvantages. It extends and contracts easily. It lets you interface equipment to software with comparatively little work. Most MMSFORTH programs can be switched from one processor to another with only minor changes. Programming with MMSFORTH is structured, with control passing from the most general of commands downwards to the most specific, and many programs may be executed just by entering their name.

In view of these facts, many programmers will find Forth to be just what they need to develop software for new hardware devices compatible with the TRS-80.

There is one last note. Unlike the other Z80 versions of Forth, MMS supports only the 8080 Assembler commands. For those

who don't want to write a new assembler for the Z80 commands (in Forth of course), MMS is now selling an add-on utility that supplies the Z80 Assembler Editor and full floating-point math routines.

*(Miller Microcomputing Services, 61 Lake Shore Road, Natick, MA 01760; The MMSFORTH version 2.0 requires 32K and one disk drive, \$129.95. MicroForth Primer \$17.50.)*

Terry Kepner  
November 1980

★★★★

### **MMSFORTH 2.0** **Miller Microcomputer** **Services** **Model I and III**

I was intrigued for some time by ads for MMSFORTH. I read with interest claims that MMSFORTH was 10 to 20 times faster than Basic, that it was stack-oriented, that you could add your own commands, that it had a great editor, and a variety of utilities, and that MMS would provide professional support. It seemed almost too good to be true, so I bought the language. Now my Model I has a new personality.

MMSFORTH is available for the Models I and III on disk or cassette. The Model I cassette version requires only 16K to run; the Model III requires 32K. Both disk systems need only one drive.

The System, Program, and Utility disks arrive in a three-ring binder with a 126-page instruction manual. The manual includes chapters on Forth operations, editing commands,

input/output to disk and printer, data declarations, text handling, conditionals, branches, and loops. Three programs, ranging from easy to difficult, provide detailed study notes. This is supplemented with instructions for using MMSFORTH system utilities and a system index. The documentation is complete and well-written, but it is not a textbook on Forth programming and requires careful reading for the first-time user.

The optional Utilities disk includes a Cross-Reference utility (XREF), Floating Point Math, and a Z80 Assembler.

For the rest of the documentation you must sign two license agreement forms promising not to sell or give away MMSFORTH. Programs written for sale require the end user to have the MMSFORTH system or be under a separate licensing agreement. Personal use and back-ups are not restricted.

### **System Utilities**

The Model I System disk comes ready to run on a 16K system, but you can configure the disk to your particular hardware.

Customize permits you to specify the directory block, the lowest unprotected block, memory size, printer margin, the number of block buffers, disk startup speed, and number of disk drives. For each drive you may specify single or double density (Model III), number of tracks, and track access speed. Finally, there is an Auto command to perform a sequence of commands on booting the system.

You can select the maximum number of block buffers (editor-

like functions while inputting from the keyboard), a lowercase driver, and a special printer driver for the MX-80.

Other utilities on the System disk allow copying a range of blocks to another location, finding any word in a range of blocks (with options to print and edit matches), and translating source code from version 1.9 to the present 2.0 version of MMSFORTH.

As delivered, MMSFORTH is limited to character (8-bit) and single-precision (16-bit) integer arithmetic. What this lacks in handiness and dynamic range is compensated for by speed. You can add the following from the System disk (depending in memory): double-precision integers, arrays, strings, random numbers, graphics, screenprint, cassette functions, clock, and a few other utilities. Loading them all takes an additional 8K when compiled onto a System disk using Customize.

The optional Utilities disk has single and double precision floating point math derived largely from ROM routines to conserve memory. You can select radians or degrees modes for trig functions, rectangular to polar coordinate conversions, complex numbers, imaginary numbers, and a superfast program to solve quadratic equations. A flexible cross-referencing program will print all references to words you choose in the range of blocks specified. A complete Z80 Assembler rounds out the Utilities disk.

### **Blocks, Editor, and Assembler**

The basic storage unit of the

Forth language is a block, made up of 1,024 characters (1K). Because a block fits perfectly on the screen, writing, editing and listing programs is a breeze.

Information is transferred in one-block units from disk to block buffers in RAM. Access to disk blocks is almost as easy as accessing RAM (either directly or under program control). With four drives you have the equivalent of more than 250K of virtual memory and more than 500K on the Model III. Files are created within blocks by calculating the offset from the first byte of the block to where a particular record is written within the block.

The editor is easy to use; I wish I had its range of commands in a word processor. In edit mode you can: insert or delete characters or lines; lock into insert mode; truncate to end of line; copy lines; treat the block as a continuous page of 1,024 characters; and use the arrow keys to position the cursor anywhere on the screen.

The editor can also be used on non-Forth disks as a Superzap-type program, with direct visual access to any byte on a disk.

It is easy to include 8080 assembler code in a program using the MMSFORTH word Code to start the sequence and either Next, PSH, or PSH2 to return (passing nothing, HL, or DE and HL to the user stack). The 8080 assembler is incomplete, but MMSFORTH words like CMOVE (which moves a series of bytes from one memory location to another) render some assembler functions unnecessary. Control structures and word order for the assembler follow Forth conventions.

## Data Types and Functions

You can handle data in MMSFORTH by character, single, double, and triple precision integers, single and double precision floating point, and complex numbers.

Quite unlike Basic, which is limited to the 8-bit PEEK and POKE commands, MMSFORTH lets you store and read from 8-bit characters to 64-bit complex numbers as units.

Strings are a maximum of 255 bytes long (the first byte is the length). String functions are a superset of Disk Basic string functions. Unlike Microsoft Basic, there are no string pointers, making it easier to directly access and change strings in memory.

Constants, variables, and arrays can be defined for any data type, including strings. A constant's name evokes its value; a variable's name evokes its memory address.

You can write complex programs in Forth without one variable or constant. Forth uses a stack (a set of memory locations) to hold values and pass them from word to word. Passing numbers from one word to another is as simple as leaving the results of one calculation on the stack for the next word to pick up.

To use a last-in-first-out (LIFO) stack with its semi-automatic management of numbers, you cannot use ordinary algebraic notation, with its parentheses and equals signs. Instead, you use a more efficient notation called Reverse Polish Notation (RPN).

## Conclusion

MMSFORTH is a complete ver-

sion of the Forth language (a superset of the Forth 79 Standard) and makes the TRS-80 a very powerful tool for developing programs. You can build a powerful set of subroutines, and build on your previous work with an ease unknown to Basic. (*Miller Microcomputer Services, 61 Lakeshore Road, Natick, MA 01760; \$89.95 cassette, \$129.95 disk.*)

Nicholas Spies  
June/July 1982



### **Model II Compiler Basic Tandy/Radio Shack Model II**

To use Radio Shack's Model II Compiler Basic, I knew I would have to modify my existing programs. Radio Shack had indicated that the new language differs from interpretive Basic, and suggested it is primarily suitable for new-application development.

To achieve higher performance, I expected that the compiler language would require some additional specification. But, I was utterly unprepared for what I encountered: gratuitous changes for the sake of change, callous disregard for human factors, and woefully disappointing performance.

In the compiler Basic, programs are written and debugged using the Development System. This is a complete environment that closely resembles interpretive Basic—you can write, save, execute, and modify programs without returning to TRSDOS. A stand-alone editor, BEdit, allows more powerful editing operations, and a stand-

alone run-time package executes previously compiled programs.

The editing commands in the Development System are annoyingly different from the familiar Edit command of interpretive Basic. In fact, there is no Edit command; its function is performed by Change, which is better suited to global changes than to intra-line corrections. The BEdit program uses another scheme. This makes it incredibly difficult to switch from the Development editor to BEdit.

If you have existing programs in interpretive Basic, forget trying to use them as a starting point. The file format used for storing source programs by Compiler Basic is different from either of the formats used by interpretive Basic.

The compiler supports only two types of numeric data, integer and real. Integers occupy 2 bytes and perform as in interpretive Basic. Reals, on the other hand, are stored in floating packed decimal and occupy 8 bytes each.

Strings have maximum and current lengths. Unlike interpretive Basic, strings always require enough storage to hold their maximum length. This avoids the interpreter's string-compaction delays, but it might prevent programs with large string arrays from fitting into memory.

Some of the differences seem pointless. Their primary effect is to frustrate users who are accustomed to interpretive Basic. Among these are: Load will load only an object file, not a source file—you must use Old for source files.

In debugging a program, you cannot use Print <var> to see its

value: you must display <var> or DI <var>. The end statement marks the end of compilation, not the end of execution. And the INSTR function has been relabeled POS and accepts only the two-argument form.

These few examples indicate the nature of the changes. If the compiler were to be used in a vacuum, the choice of syntax wouldn't matter. But every Model II is already supplied with interpretive Basic, and most potential users of Compiler Basic are probably already using it. These changes make life needlessly difficult.

The file I/O are also markedly changed but with justification. Compiler Basic provides a rich variety of I/O options, including fixed or variable-length records; sequential, direct, or indexed access; and stream, formatted, or binary formats. However, most of the interesting combinations seem to waste an inordinate amount of space.

Perhaps the most welcome feature of Compiler Basic is that it allows true subprograms, which can be called by name and passed arguments. Except where declared as Common or as formal parameters, the variables in the subprogram have nothing to do with those in the calling program.

A close reading of the manual reveals that Compiler Basic is not a true compiler. It does not convert source programs into machine code, but into an intermediate pseudocode, which must be interpreted at run time. The pseudocode is more compact than true object code would be.

Subprograms can be compiled either separately or in conjunc-

tion with the main program. If they are compiled separately, they can be loaded together before beginning execution, but the result of the execution can't be saved. The pseudocode cannot be linked with true object modules.

The inability to combine independently compiled modules into a single executable whole makes the concept of separate compilation virtually useless.

The program chaining facility is a mild improvement over what's available in interpretive Basic. It allows you to save the contents of certain variables using Common, where they can be retrieved by programs later in the chain.

Compiler Basic doesn't let you combine modules that were compiled separately, and you can't compile a big collection of modules at once. But, chaining is available.

I wrote a small program to compare the speed of interpretive and Compiler Basic. I was amazed by the relative performance of the two systems: Interpretive Basic took 13 seconds; Compiler Basic took 22 seconds (plus 14 seconds compilation).

Where I expected at least a threefold improvement in performance, I received a degradation of about 70 percent. Either Microsoft does things fully well, or the authors of Compiler Basic are doing something terribly wrong!

I have never bought a program with such high expectations and wound up feeling so totally ripped off. It is inconceivable to me that a compiled program—or even a partially compiled one—can perform so poorly.

I am equally appalled by the

unnecessary incompatibilities between this and the standard version of Basic for the Model II. It shows an utter lack of concern for the people who will inevitably use both.

If the compiler is intended for developing applications programs, the authors should recognize that program linkage is a necessity, and that disk space might be a precious resource.

After my first experience with the compiler, I put it on the shelf and dismissed it as useless. Several months later, I brought it back for another try, hoping I had missed something crucial. It appears that I hadn't.

*(Tandy/Radio Shack, Fort Worth, TX 76102; \$199.)*

Larry Clark  
February 1982

★ ★ ★

### **Pascal I and II Computer Information Exchange Inc. Model I**

Although People's Pascal I and People's Pascal II are subsets of full Pascal, there's enough to be interesting, fun, and perhaps even useful.

Depending on line length, the editor/compiler for Pascal I can handle 50 to 200 lines of Pascal source text at a time. Larger files can be created, edited, and compiled by saving portions of the large file on tape as the text buffer is filled. The compiler is set up for use with one cassette recorder, but it can be easily modified to work with two. All actions required of the user are prompted. Compiler options include output to line printer and

compilation without generation of an object file.

Once the compiled P-code is on tape, it can be either translated into Z80 code or executed via the interpreter, which, like the editor/compiler and the translator, is written in Basic. Commands are provided for controlling and analyzing program execution. During conversion, the P-codes and corresponding Z80 codes will be optionally listed on the video monitor and/or the printer. At the end of translation, the start and end addresses of the Z80 code and the Z80/P-code ratio are displayed. One nice feature of the translator is that conversion can be customized for speed or for size.

Several run-time routines, such as keyboard scanning, video output, and 16-bit arithmetic routines, have been included by the authors of People's Pascal I. They are written in Assembly language for easy modification, and you can load them from tape via the Level II System command.

People's Pascal II consists of three sections: the Monitor, which includes the run-time routines and P-code interpreter; the editor, which is used to enter and modify Tiny Pascal source code; and the compiler, which creates P-code from source. The three are loaded and used as one program via the Level II System command, making Pascal II quicker and easier to use.

Tiny Pascal's greatest limitation is that it handles only two data types: integers and one-dimensional integer arrays. However, these constraints can be overcome. Valid Tiny Pascal statements include: assignment of variables and constants; defi-



nition of procedures and functions; conditionals; repetitives; Read; Write; compound statements of any of the above. Not provided are label definition, GOTO, READLN, and WRITELN. Procedure and function parameters are passed only by value. There is no record of file-handling capability. Additional features access memory, use machine-language code, and allow hexadecimal constants, "else" branches on Case statements, logical operations and printing of character strings.

Pascal offers greater versatility than Pascal II when creating source code, but it requires the loading of at least three fairly long programs from cassette and runs slowly. Because the PEEKs and POKEs conflict with DOS Basic, it can't be put on disk as Pascal II can. Each system has particular functions and differs in its final products, but each one is definitely worth trying. Although Tiny Pascal probably won't displace Basic as your 80's native language, it might satisfy your appetite for something new and prepare you for that Big Pascal yet to come.

*(Pascal I and II, once sold by Computer Information Exchange Inc., San Luis Rey, CA 92068, are no longer available.)*

Fred Monsour  
May 1980

★★★★

## **Pascal-80 New Classics Software Model I and III**

Pascal-80 is not "just another compiler" but an honest attempt to provide TRS-80 users with a meaningful implementation of

this language. It is a monitor system that provides an interface to the disk-file structure, a text editor, the Pascal compiler, and the Pascal interpreter. The system comes on a TDOS system disk in single-density, 35-track format for Model I users.

Users are not limited to this operating system; they can use Pascal-80 with any of the second-generation operating systems, such as NEWDOS80.

The System is easy to operate. Pascal-80 functions are available from the monitor, making it easy to save and load files, compile, execute, and edit source text.

The compiler/editor system uses an in-memory text buffer with 32K of memory space that holds about 1,100 lines of code. This can be increased with the use of text-compression techniques.

The in-memory compile feature accounts for the compiler's incredible speed. When timed on a Model I running at standard speed (1.77 MHz), I got compile rates averaging 1,000 lines per minute when listing to the video screen and 1,500 lines per minute using the NOLIST option.

The system has some nice features available from the monitor: The W command lets you write the compiled P-code to disk in a named file, and the X command lets you execute this compiled code directly from the disk. This allows you to compile lengthy programs once in the final version and then run them directly without compilation delays.

Pascal-80 has a full-screen editor that you can invoke from the monitor. Text displays on



pointed to by the integer value of expression.

A compile-time function is recognized to let you include source code from a disk file in the program currently being compiled. This is a powerful addition to the original Pascal-80.

Both integer and real values are displayed using the same format description. A field width of zero produces a display in the default format for the displayed expression type. Similarly, a field width of -1 produces the display of the expression in scientific notation.

Pascal-80 provides an additional source-code procedure (GOTOXY) on disk to let you position the video system cursor to any screen location. The call GOTOXY specifies the cursor's location.

The Pascal-80 master disk contains the compiler system and several other files. These files contain the Pascal source code for graphics, random-number, and GOTOXY linkages, plus several demonstration Pascal programs. Each program is designed to illustrate a feature of the Pascal language.

In addition to the demonstration programs, there are two utilities to convert Pascal-80 text files to ASCII character files (ASCII/CMD) and to convert from character format files to text format (TEXT/CMD). The ASCII files can be used with almost any word processor.

The two remaining files (AUTHOR/SRC and AUTH-CODE/CMD) let you write application programs in Pascal, compile them and use them as a CMD file that executes directly from the disk-operating system command level. New Classics

Software grants a license to original registered owners of Pascal-80 to distribute programs compiled by Pascal-80, provided you meet the requirements specified in the user's manual.

Pascal-80 is the first real attempt to provide a useful implementation of TRS-80 Pascal for a reasonable price. You should be using a Z80 compiler such as Fortran if speed is your major objective. But if ease in programming using this well-structured language is your objective, then Pascal-80 is perfect. (*New Classics Software, 239 Fox Hill Road, Denville, NJ 07834; \$99, disk. Pascal-80 requires 48K and one disk drive.*)

J.B. Harrell, III  
December 1982

★ ★ ★

**PL/I-80**

**Digital Research**

**Model I and III with CP/M**

Digital Research, the people who brought you CP/M, has compressed the huge, main-frame-oriented PL/I monster to fit into the world of the 8080.

The package contains two disks, one contains the PL/I compiler and supporting software, the other contains a generous collection of illustrative programs. Also included are four volumes of documentation. Three of these manuals are for PL/I-80; the fourth describes the loader program, LINK-80. The PL/I manuals include a 108-page Language Manual, a 180-page Applications Guide, and a 30-page Command Summary pamphlet.

The language has a few extensions to the standard Subset G,

and a few omissions. The most important extension is a badly-needed unformatted I/O mode for reading and writing variable-length ASCII records. PL/I-80 accepts source programs written with lowercase letters. I don't know why this trivial detail should make such a difference, but it does. I think it does more for program readability than indenting; once you have seen one lowercase listing, you will want to convert your whole library.

The documentation is conscientious, but not entirely successful. The language features are scattered between the Language Manual and the Applications Guide, and it is not entirely clear how they decide which things would go where. The manuals do not have an index.

These manuals are not tutorials, however; a separate leaflet provides an annotated bibliography of recommended PL/I textbooks. The best introduction to PL/I-80 is the set of illustrative programs (60 of them) on the second disk.

There are two licensing agreements, one noncommercial and one commercial. The non-commercial license gives you the right to do anything but sell executable PL/I programs. Once you start selling those, you are a software vendor and are required to sign the commercial licensing agreement, which includes provision for royalty payments to Digital Research. To sweeten the pot somewhat, Digital Research has announced an ambitious Independent Software Vendor (ISV) support program. Potential vendors should consider carefully the tradeoff between the royalty payments and the equivalent cash value of

the ISV program.

### Omissions

My major reservations concern the language itself. PL/I-80 omits the capability of dynamically dimensioning arrays by putting them inside a `Begin . . . End` block. Hence one of the main advantages of PL/I's block-structure capability is wiped out. Blocks are still useful for controlling the scope of a variable (i.e., the portions of a program in which its identity is known), but I feel this is a minor benefit.

The other omission is similar: Array dimensions in subroutines must be determined at compile time. This is a disastrous decision, because it means no subroutine intended to process arrays or character strings can ever be truly general-purpose.

An obscure feature of PL/I lets you get around this problem, within limits. PL/I has a data type known as a pointer. If you pass pointers to your subroutines, then, with a certain amount of ingenuity and care, you can con PL/I-80 into letting you handle arrays and strings of arbitrary size. I've tried it; it works; but it's tricky and you end up with programs unlike any PL/I programs normally encountered in the outside world.

I have a few other, less important, grumbles. Comments are delimited in PL/I by the symbols `/*` and `*/`. In PL/C, a popular teaching version of PL/I, comments are also delimited by a carriage return. This means that if you forge or misprint a `*/` the end of the line will terminate your comment anyway. In full PL/I, a missing `*/` can turn the rest of your program into one big comment. This kind of easy error

is not the mark of a well-designed language: Digital Research should have followed PL/C's policy, standard or no standard.

My own feelings about PL/I-80 are mixed. Like PL/I itself, it seems an uncomfortable mixture of unmatched capabilities and maddening drawbacks. It seems unrealistic to do any kind of serious business programming in any language but PL/I. The availability of PL/I on microprocessors is thus an incredible boon.

Many potential users are surely going to be put off by the royalty requirements, which I predict will prevent widespread adoption of the language. But the inability to use adjustable dimensions in subroutines and blocks is going to cramp the style of many experienced PL/I programmers, and this may prove a more serious impediment than the licensing problem. For myself, these restrictions make the language virtually useless. As long as they remain, I think PL/I will fare as poorly in the micro world as it has among mainframes—which is a shame, because we need a language like this.

*(Digital Research, P.O. Box 579, Pacific Grove, CA 93950; \$500.)*

Thomas W. Parsons  
June/July 1982

★ ★ ★  
**Tiny Pascal**  
**Supersoft Inc.**  
**Model I and III**

I'd been interested in Pascal for some time, but all the compilers I'd seen required a disk

drive or two and at least 32K of memory. Supersoft's Tiny Pascal needs only 16K and a cassette-equipped machine.

The manual is not a primer on Pascal, so I bought one of the references recommended and began learning this new language. The syntax is simple, and before long I had written my first program. When I decided to list the program on my printer, I was disappointed: there was no provision for printing lists, or outputting to a line printer under program control.

The system consists of four major parts: monitor, editor, compiler, and run time interpreter. The monitor is in charge of overall system control: from it, you can enter the editor to create, modify, or compile source code and run the compiled program. Provisions are made for saving and loading source code or P-code on tape.

The editor is adequate but doesn't compare with the Level II Basic editor. All or part of a program can be listed, lines inserted or deleted, and additional characters appended to the end of a line. Correcting an error within a line requires retyping it completely.

The compiler generates an intermediate code called P-code, which is executed by an interpreter at run time.

This leaves the programmer with about 4.5K of memory for source and P-code. Normally the source code is entered and the P-code immediately follows it in memory when compiled, but for large programs there are some options available. The programmer may choose to have the P-code replace the source code

as it is compiled, allowing a full 4.5K of source code, or overwrite the compiler and editor. For systems with at least 32K RAM, there is another version of the program on the tape which supports much larger programs.

Tiny Pascal only supports integer variables. Lack of real (floating point) and character (string) variables limits its usefulness. All major control structures of Pascal are supported, including Begin . . . End, Repeat . . . Until, While . . . Do, If . . . Then . . . Else, For . . . Do, Case, Procedures, and Functions. Statements provide Read, Write, and integer arithmetic in decimal or hexadecimal, including one-dimensional arrays. Additional intrinsic functions include memory access (the equivalent of PEEK and POKE), machine-language calls, I/O port access, absolute value, square, INKEY, graphics control similar to Basic, and block memory moves. Game programmers will be sorry to hear that there is no random number generator.

While Level II Basic is still more practical for any application, Tiny Pascal offers a "shoestring" approach to learning this structured language. (*Supersoft Inc., P.O. Box 1628, Champaign, IL 61820. Tiny Pascal comes on cassette for the Model I and III for \$20; A disk version for the Model I only is available for \$85.*)

Curtis H. Kyle  
July 1981

**UOLISP**  
(See Page 165)

★ ★ ★ ★ ★

### **XBasic Snappware Model II**

Snappware's advertisement promised "the best of Apparat's features in implementing NEWDOS Basic." After one phone call, one plastic cash number and three days, my disk arrived. To my amazement the software worked as good or better than the ad claimed.

The first of XBasic's modules is designed to reduce the number of keystrokes that are required during programming or software modification. There are six single keystroke commands; the left arrow lists the first line of a program, the right arrow lists the last line of a program, the up arrow lists the previous line, the down arrow lists the next line, the period lists the current line, and the comma edits the current line. There are also 10 abbreviations for commands.

Listing and editing your program becomes a piece of cake if you need only type the left arrow, then type the right arrow while holding down the Repeat key. This causes the program to list until you release the Repeat key. Then type a comma and you are in the current line. That is much faster than typing LLIST (enter) then (Break) and edit (line number).

The most valuable routine is XREF, Snapp's cross reference facility. This utility is similar to Apparat's in that you may list references to the screen and printer and list references to line numbers or integer constants and variables. But XREF places an asterisk in front of any line number not an Integer constant.

It also places an asterisk before line numbers during the variable listing if the variable is modified in that line.

The enhanced renumbering facility corrects some problems with Radio Shack's Renumber and allows you to specify an upper limit on a block of lines that are to be renumbered. It also allows you to scan the text for renumbering errors such as undefined line numbers or other errors in the text, after which the text numbering remains unchanged. XRENUM also permits a block of text to be duplicated in another area of the program without destroying the original text or its renumbering.

Beyond NEWDOS utilities for the Model I, Snapp offers the Model II user XDUMP, a dynamic variable-print facility. This facility prints the current value or all variables on the screen or printer.

XBasic also includes the string/key word cross-reference facility called XFind. This facility displays the line numbers where key words would appear. This is indispensable for converting from one Basic to another. XFind displays all line numbers that reference any string or portion of a string which you select. This is very handy for reducing space required for quoted strings that are used more than once.

Finally, XBasic offers XCompress which reduces your program size to an absolute minimum. This is accomplished in several phases, most of which may be optionally selected. Some of these phases are: removal of remarks, removal of irrelevant blanks, removal of irrelevant tab characters, removal of extraneous colons, removal of

GOTO following a Then, merging of multiple statements into single lines, and several others.

The previously discussed utilities are well worth the price, but Snapp Inc. chooses to add at no charge their DOSFIX, a set of repairs to the Model II disk operating system. These include both the official Fort Worth and Snapp originals.

The first 16 remove unnecessary files from the operating system, and junk like SYSL/SYS which only prints the logo on the screen during the power-up sequence. Another patch moves the Break key to CTRL 6. The last DOS fix has stopped my cussing upon power up or reset when the Forms command initializes the printer and the computer issues two line feeds, destroying my top-of-form alignment.

Thanks to Snapp XBasic and their DOSFIX, my Model II no longer seems to fight my every move.

*(Snappware Inc., 3719 Mantell, Cincinnati, OH 45236; \$200 Model II, \$125 Model III.)*

Rick Lederman  
September 1981

---

## MONITORS

---

★ ★ ★ ★

**Boss**  
**Soft Sector Marketing Inc.**  
**Model I and III**

Boss, a utility program for the TRS-80, has several features useful in writing and debugging Basic programs. They are: im-

proved trace capability, single stepping through programs, reviewing variables, staking programs, setting breakpoints, and complete relocation of the utility itself.

The program comes on cassette along with a separate lowercase driver (if you have the modification) and manual. It may be loaded into memory, relocated and saved on tape or disk.

Generally, when using Boss, you don't need to worry about it interfering with other machine-language programs in memory. When it loads, it will prompt you for the lowest address you wish to protect, and relocate itself beneath that address. It also gives you the proper response to the memory size question.

If you have ever tried to use the Level II Trace function, you know that it is hard to follow and destroys the display. Boss displays the Trace only in four rows in the right-hand corner. A right arrow moves to the row containing the executing line number. If the line is multi-statement, the number is displayed only once during the execution of that line.

Boss is one of the most powerful tools I can use. It allows me to single step through the routines in the program and review just what the variables are doing. Single step functions can be used in conjunction with Trace functions: POKE 16667,5 lets you insert as many breakpoints in the program as you like.

You can include most of the Boss commands as POKE statements, so that your program will run normally until it reaches these commands, whereupon the Boss functions called for (for

example, single stepping) will begin. Thus you needn't worry about doing everything in Command mode.

You can stack Basic programs in memory and PUSH and POP them around as you like, limited, of course, by available RAM. Memory size is adjusted automatically, so that programs won't get eaten by Basic variables and stack usage. If line sequence is correct, you can append programs, the current program with either the last or next-to-last saved program—not particularly useful with a disk system and the ability to merge programs, but handy for Level II users. DOS users should be aware that reboots are an inconvenience here, as BASIC\* will not recover the utility.

In conclusion, this utility is a powerful debugging tool, and very useful for anyone who would like to spend some time away from program debugging. (*Soft Sector Marketing Inc., P.O. Box 340, Garden City, MI 48135; \$29.95.*)

Bruce Powel Douglass  
July 1981

★ ★ ★ ★

### **C-Bug The Micro Works Color Computer**

The Micro Works is offering two outstanding utility programs for the TRS-80 Color Computer. They allow access to the machine at the Assembly level.

The monitor is a powerful 6809 monitor and is available on cassette. The two versions are virtually identical, except that the ROM version mounts into a Radio Shack ROM pack, allowing instant access. The docu-



mentation is complete, and includes loading, running, and operating instructions. Also included is a documented Assembly listing of the C-Bug monitor.

There are 19 commands that include the following: Memory examine and change, convert hex to decimal, Upload/Download to send and receive data from the communication link, and an Auto mode that allows you to use your computer as an intelligent terminal connected to a host system. These commands are completely documented in the manual, with examples of their use.

The disassembler is a program that will allow you to look into the Basic ROM, and is available on cassette requiring 16K of RAM to run. The program is well documented, and includes the following features: cross-referencing of variables and labels, output code that can be re-assembled, output to an 80- or 32-column printer, and a data table area specification which defaults to the table boundaries in the interpreter ROM, and more.

Both programs are highly recommended.

*(The Micro Works, P.O. Box 1110, Del Mar, CA 92014; Monitor Tape \$29.95, Monitor ROM \$39.95, Disassembler \$49.95.)*

Howard Berenbon  
September 1981

★ ★ ★ 1/2  
**Cocobug Debugging Monitor**  
**Allen Gelder Software**  
**Color Computer**

Typical of software by Allen Gelder, Cocobug is oriented

towards machine-code programmers. This is an approach I find very appealing, because Assembly programs can easily obscure the real binary operations which the microprocessor is undertaking.

Cocobug consists of two parts, a Basic menu/prompting program, and a machine-code subroutine for display of memory and execution and breakpointing of the user's program. After a very brief CLOAD and CLOADM the program is run. A cursor appears at the top of the screen waiting for a user directive.

Memory display options are M and N, followed by a hexadecimal address. The M command displays the contents of 112 bytes, 14 lines of eight rows of hex bytes. Fifty-six bytes in hex and ASCII are displayed under the N command. The small number of bytes displayable on the screen is a function of the Color Computer's 32-character lines. However, the screen may be scrolled up or down 8 bytes, or moved a single byte ahead or back.

Most interesting are the programming models included in Cocobug. All the registers are displayed, as well as the top of the two stacks and the condition code register. Under Cocobug's R command, two programming models are displayed: The first is the entry condition (before execution of the user's code), and the second is the exit condition (after a breakpoint). This display is disabled with the T-command.

Rumors of monitor programs as good or better than Cocobug are already being heard, but software from Allen Gelder is

unique. Like Accel, Cocobug is provided with complete instructions on where it resides in memory, what RAM patch points it uses, and how to make backup copies. Furthermore, Cocobug is fully supported by the author, and is the first in a group of modules that will include an assembler and other tools. Cocobug alone, however, is a capable and attractive program at a very reasonable price. (*Allen Gelder Software, Box 11721 Main Post Office, San Francisco, CA 94101; \$19.95.*)

Dennis Bathory Kitsz  
September 1981

★ ★ ★ ★

### **Macro-Mon—The Shadow Advanced Operating Systems Model I or III**

An absolute must for every Assembly-language programmer, beginner or professional, is a good program monitor. A monitor combining unique commands and ease of use is an exceptional programming aid.

Macro-Mon, while using commands similar to most monitors, adds commands allowing single instruction, interpretive operation, and complete, dynamic register display to video and printer. While executing one instruction, it displays the disassembly of the next instruction to be printed. The disassembly can be printed as the program is interpreted, providing a running printout of the registers along with the disassembled instructions.

I received the 16K Model I version and, although I have a 48K disk system, the monitor operated immediately. The first

thing I did was relocate Macro-Mon (it is completely relocatable) to upper memory, preventing any conflict with the disk operating system. My next step was to use Macro-Mon to save itself to disk. (Macro-Mon contains both disk and tape routines built in.) The monitor comes on a minimum-version DOSPLUS disk; I tested it with TRSDOS and NEWDOS80 versions 1.0 and 2.0 and found no problems at all.

Macro-Mon comes in one of the most professional-looking packages I have seen. The manual is divided into sections allowing for easy learning and includes a sample session that traces Level II Basic and works with either tape or disk.

The documentation gives only the information needed to use Macro-Mon and does not tutor the beginner in Assembly language or the terms used. A novice will need a source of basic information on Assembly to effectively use Macro-Mon. This point is not a drawback; the last thing a beginner needs is a long-winded, obscure manual that attempts to do more than explain the proper operation of the monitor.

One of Macro-Mon's outstanding features is its video display. The display has single-keystroke selection, for hex or ASCII representation, in each of two modes. The first is the register display mode which displays all registers and updates them as the monitor is used to trace or debug a program. The bottom four lines show 64 bytes of memory that can be selected by the user. These lines can be scrolled forward or backward by use of the arrow keys without affect-

ing the upper part of the display. The upper right of the screen shows the start and end hex addresses of Macro-Mon. This is helpful in avoiding monitor/object program clash; it updates automatically when the monitor is relocated.

The most powerful feature in LDOS is the System (SYSGEN) command. This command causes all your system configuration, drivers, filters, routing, linking, and so on, as well as Spooler, Verify, and Clock, to be written to a file called CONFIG/SYS on drive O. Each time the system is booted, it loads the user's configuration from this file. All system changes are made first in memory and do not become permanent until you execute a SYSGEN. This lets you experiment with different configurations without making a special System disk.

The LCOMM utility is an advanced communications package for use with the RS-232. Because it is so flexible (you can even use it to communicate directly with your printer, though not very well), it is difficult to learn; sit down with the documentation for at least half an hour and save yourself a lot of on-line frustration. The patch utility allows changes to the system or to other files.

The repair utility brings disks created on certain other systems up to LDOS standards. It updates TRSDOS's data address mark to an F8H instead of the FAH, making single-density disks directly usable on the Model III with LDOS. In fact, if your Model I is double density, you can switch disks back and forth between models without

even knowing what density the disk is! It also performs additional correction on the disk to ensure reliable operation. The HITAPE utility permits the use of high-speed cassette I/O on a Model III.

Although I have extolled the excellence of Macro-Mon, I should point out a couple of shortcomings. I prefer a disassembler that prints the ASCII codes for all values encountered during the disassembly; Macro-Mon does print out ASCII equivalents in some cases, but not for every hex code. Also, while Macro-Mon's symbolic output can be directed to disk, certain qualifications must be met first. The DOS used is what makes the difference. Macro-Mon does not have the facility for a symbolic dump of object code to tape.

For the programmer just starting out, or for the programmer looking for a better monitor, you can't beat Macro-Mon. Two minor conveniences are sacrificed, but in return you get much more valuable features that can't be found elsewhere. (*Advanced Operating Systems, 450 St. John Road, Suite 792, Michigan City, IN 46360; \$54.95 Model I cassette, \$59.95 Model I disk, \$69.95 Model III disk.*)

Richard C. McGarvey  
September 1982

★ ★ ★

**MON-2**  
**Hubert S. Howe**  
**Model I**

MON-2 is an Assembly-language monitor with a built-in Z80 disassembler that satisfies most of an Assembly-language

programmer's needs. It has good documentation, works well, and is easy to understand and use.

A memory block of any size can be displayed in either ASCII or hex; locations can be changed, and programs executed. Find Byte and Find Word features permit memory searches for a match.

MON-2 is an effective disassembler, but if it included an assembler that wrote object tapes that could be read with the System command, it would be perfect.

*(Hubert S. Howe, 14 Lexington Road, New City, NY 10956. MON-2 is now MON-3 on cassette, \$39.95; MON-4 and 5 on disk, \$49.95 and \$59.95 respectively.)*

Rod Hallen  
February 1980

★ ★ ★ ★

### **Power Pack and Color Diagnostics, Editor and Assembler Computerware Color Computer**

Although Color Computer owners can be perfectly happy programming in Basic, eventually the time will come to experiment with machine language. To do so, however, you need a monitor, editor, and assembler.

The Computerware system includes both hardware and software elements for the 16K version. Owners of 32K machines or disk systems need different versions. (More about this later.)

The heart of the system is its hardware component: a plug-in cartridge called the Power Pack,

containing a monitor program in 2K of ROM plus another 6K of static RAM (2716s and 2114s, respectively).

The cartridge fits into the Color Computer's expansion ROM port. Since the machine scans this port immediately when turned on, it "wakes up" in the monitor instead of in Basic.

The Power Pack features commands that return you to Basic immediately so it can be left plugged in all the time. Both ROM and RAM in the pack occupy locations out of the range normally addressed by Basic: ROM, SC000-SC7FF (49152-51199 decimal); RAM, SD000-SE7FF (53248-59391 decimal). RAM is available to Basic programs via POKE and PEEK commands and can also be used to hold protected machine-language programs if the proper loading address or offset is given. That is exactly what is done with the Color Editor.

Like all monitors, this tool lets you examine and change memory. This, augmented with additional commands, allows the Color Monitor to: have direct access to the 6809's registers, save and load binary cassette files, use the RS-232 port to communicate with another terminal or with a printer in an echo mode, and debug software by setting breakpoints for program execution.

Color Monitor commands consist of a single alphabetic character, followed by one or more hex arguments. With these commands you can enter your own hand-assembled machine-language programs or published programs. In the latter case, you would use the bytes in the second column of a standard

Assembly-language program listing. This is not the way to enter lengthy programs, though. For those, use the Editor/Assembler.

The manual offers a few goodies, such as a table of indirect addresses through which various Monitor routines can be accessed. Presumably, these could even be called from Basic via the DEFUSR and USR(N) commands.

The Color Diagnostics program is provided as a binary cassette file, loading and auto-executing through the Monitor's L command. It is stored in low Power Pack RAM; you can reenter it from the Monitor by entering J D000, as long as the pertinent memory has not been written over.

The diagnostics are organized around a set of nested menus. You needn't perform the tests in any specific order, and there are plenty of quit options allowing you to exit whenever you please.

The major tests are for: power pack, RAM, RS-232, joysticks, Basic ROMs, tape I/O, and sound.

The Color Editor provides powerful commands for managing text: Assembly-language source code, Basic, or English. The program searches for and changes strings of characters in one or many lines of text, and moves lines of text within a file, meaning it serves as an elementary word processor, as well. You cannot, however, run programs directly from the Editor.

The Editor's command suite comprises a total of 36 commands grouped into: line display and movement; line modification and replacement; pattern modification and replacement (for editing strings within a line);

and cassette commands. There are also some miscellaneous commands, such as those used to clear the text area, renumber lines of text, turn off the line numbers for printing, and exit to Basic or the Color Monitor.

All cassette files are in ASCII form, and the Editor chains files read in succession—no need for the POKEs and PEEKs used to chain standard tokenized Basic files.

The final component of the Computerware Tool Kit is a resident two-pass assembler for the conversion of 6809 Assembly-language programs into machine language. It loads with the Monitor's L command, and occupies about 8K of RAM—specifically, from \$0600 to \$2600. Fully 139 Motorola-defined mnemonics are supported, as are several directives and pseudo-ops.

The manual, 18 pages plus appendices, presents complete syntax listings for both directives and pseudo-ops.

The necessity of making a tape of the machine code before running (the inability to assemble directly to memory) is the result of a conscious choice by the program's authors stemming from their desire to permit the Assembly-language programmer the maximum freedom in locating a program in memory.

This utility forces you to deal with the "tyranny of Modes." You must get used to flipping back and forth between, say, the Monitor and the input and command modes of the editor. This can be especially daunting to the inexperienced user. It shouldn't dampen anyone's enthusiasm for these programs, though.

In the same vein, the manuals

can be cumbersome. You'll have to get used to juggling several computer manuals on your knee while working out a problem.

There are cassette versions of the Monitor, Editor and Assembler that run in 32K machines without the Power Pack. They cost \$29.95 each, including the Monitor source listing (\$15 separately). In fact, you can also buy a combined 32K Editor/Assembler package for \$49.95. This makes the Computerware products much more competitive. The Diagnostics package isn't available in a 32K version.

Computerware allows owners of 16K Power Pack versions to return their Editor/Assembler cassette and receive an upgrade to run in 32K for a modest fee. In the 32K Power Pack configuration, you can have the Editor and Assembler resident simultaneously.  
*(Computerware, P.O. Box 668, Encinitas, CA 92024; \$159.95, 16K required.)*

Scott L. Norman  
June/July 1982

★ ★ ★ ★  
**RSM-2 Monitor**  
**Small Systems Software**  
**Model I**

The Small Systems Software RSM-2/2D has more than the usual monitor functions of inspect and change, cassette read/write, jump and execute, breakpoints, and the like. RSM-2 is designed to operate with the Level II, 16K version of the TRS-80.

You can do I/O operations directly to and from disk sectors,

if your system is so equipped. RSM-2D is the same monitor distributed on disk.

RSM-2 loads into high memory and occupies 4,863 bytes of RAM. The disk version loads into high RAM of the 16K, 32K, or 48K disk configurations and can be used with a printer. RSM-2 supports the Small System Hardware TRS-232 serial data interface and any printer interfaced through the Centronics parallel port on Radio Shack's Expansion Interface.

One handy feature is the right arrow key that initiates simultaneous CRT/printer output. In fact, pressing the key before entering a monitor command causes that command to be printed as well. A nice bonus for documentation!

Besides the printer command, the RSM-2 monitor has 31 commands. They fall into the general groupings of control, tape, disk, and memory commands. I discovered user access to four routines within RSM-2: the CRT processing routine, the keyboard routine, the routine that checks for break key depression, and the printer driver.

Another unique feature of RSM-2 is its ability to halt cassette I/O operations with the break key, an improvement over the Level II Basic scheme of having to press the reset button to restart cassette I/O.

Most of the control commands are self-explanatory and are typical of those in other monitors; however, some deserve a closer look. For example, the Port Input and Port Output commands are directly accessible during monitor operation, particularly during the program-

ming and debugging of interface applications.

The User command gives you a minimum of 128 bytes in which to locate self-designed patches that might complement RSM-2 functions. The Binary Arithmetic command lets you type in two numbers, A and B, and compute A-B, A\*B, and B/A in both hex and decimal notation. This is handy for determining required memory offsets.

Many of the tape and disk commands are also self-explanatory. The Read System Tape command, however, has one minor drawback. It is sensitive to the noise generated at the I/O crossover points of data on the tape. One solution is to create a back-up copy of the system tape on a good-quality audio or digital cassette. This can be done with the RSM-2.

The Read Date Tape command is the unsung hero of the RSM-2 monitor. Unsung because it is not described in the documentation, and a hero because it allows reliable machine-language program tape copying. It ignores any input format processing and reads any cassette as raw data.

The Symbolic Dump, a disassembler program using Zilog mnemonics, cannot display ASCII characters with the mnemonic listing. Having to refer to a separate ASCII dump for analysis of machine-language programs is inconvenient. Another useful feature that is lacking is the ability to create an editor/assembler-compatible source file (ASCII text) from the disassembler process.

The RSM-2/2D is a versatile, easy-to-use, and reliable programming tool that belongs in

the software collection of any serious TRS-80 user. It might not be the ultimate monitor, but it offers exceptional value for the price.

*(Small Systems Software, P.O. Box 366, Newbury Park, CA 91320. RSM-2 is no longer available.)*

Bruce W. Churchill  
April 1980

★ ★ ★ ★

**SuperStep  
Allen Gelder Software  
Model I and III**

SuperStep transforms the lowly Radio Shack T-Bug into a heavyweight. It offers single-stepping, a two-speed trace mode, and a versatile disassembler that can run in conjunction with the single-step or trace modes. A processor model displays all registers and flags, both for the current instruction and for the previous one. An intelligent RAM window shows the RAM locations and contents affected by memory accesses.

SuperStep has commands that backspace in memory, clear the screen, and display memory contents in rows of 16—in hex or in ASCII. It gives direct access to the registers and to the flags. There is also a relative-jump pointer that shows where you are jumping to, and a high-speed Save/Load command included in the package.

The program's string editor allows inserting or deleting bytes in a sequence of code. The bytes on either side of the change move aside or close up, as necessary. Useful for editing ASCII strings or tables, this

feature also allows creating or deleting space within the program it modifies; it no longer needs to jump elsewhere to do its work.

I rate SuperStep as the best single program I have in my modest collection. If the program is useful as a debugger, it is invaluable as an aid to learning Assembly-language programming, and the Z80 instruction set in particular.

I do have a couple of minor bones to pick with the program's author. The documentation needs to be revised and reorganized. It could be greatly expanded, with many more examples. But this is hardly a fatal shortcoming since you learn to use this program by using it and experimenting. The only real reservation that I have about the program is that it needs T-Bug to run. SuperStep is a steal at its price, but you must also pay \$15 for T-Bug. Perhaps Gelder will rewrite the program as a stand-alone monitor.

*(Allen Gelder Software, Box 11721 Main Post Office, San Francisco, CA 94101. Superstep has been replaced by Stretch Superstep, \$39.95.)*

Mendel Cooper  
September 1981

★ ★ ★ ★

### **Tasmon The Alternate Source Model I or III**

Tasmon is one of the new breed of TRS-80 machine-language monitor programs. A good monitor lets you stop a machine-language program, examine memory locations,

change them and continue running the machine-language program. Tasmon provides these features and many more.

I received Tasmon on an easy loading cassette for the Model I that included all the disk commands; tape is merely the medium for distribution. Tasmon is perfectly usable if you don't have a disk, although you can't use the disk commands. Note that none of Tasmon's disk commands are implemented in the Model III version, at least not on Tasmon Version 2.12. The Model III version does support both high and low-speed cassette I/O, although it is done somewhat awkwardly via the monitor's Modify Memory command. Tasmon occupies about 8K bytes; it is for this reason that 32K bytes of RAM are recommended.

Tasmon comes a lot closer to my ideal than any other monitor program I have used. In 95 percent of the situations you face in machine-language program debugging, Tasmon makes all the features you need readily available. It does the job of at least four separate programs I previously used.

This is not to say that it is perfect, obviously. In fact a function I wanted was missing—a memory block comparison command.

Fortunately, Tasmon allows you to add new commands, although the documentation on how to do so is a bit skimpy. To his credit, the author of Tasmon has made commented source listings of the program available to its purchasers for \$15.

In addition to the commands to load a tape or disk machine-language file, Tasmon includes a



command to view a file. You can also write a machine-language file to disk or tape. These capabilities allow Tasmon to be used as a general backup and transfer facility. You can use it to move machine-language programs from disk to tape, providing a less expensive means of keeping backup copies than using additional disks.

The area where Tasmon shines is in its debugging commands. About the only thing lacking is the symbolic debugging capability found on some mini and mainframe computers, which allows you to refer to memory locations using the symbols from your assembly source.

The most novel feature is the ability to trace or single-step through a program in ROM. Tasmon also lets you specify whether the Z80 RST instructions should be executed in full or stepped through.

Breakpoint setting is extremely flexible. You can have up to nine breakpoints active at any one time. You just set the breakpoints once and forget them until you want to clear them. An extremely nice feature lets you specify the number of times (up to 256) that a breakpoint should be executed before it actually takes effect.

One more super feature is that Tasmon lets you specify a keep-screen buffer in memory. This 1,024-byte area of your own choosing gets the current contents of the screen before Tasmon writes its display out while single-stepping or tracing. One thing I would like to see added would be the ability to specify a memory buffer whose contents would be continuously

updated on the screen during single-stepping and tracing.

In addition to those already mentioned, some of the more notable utility functions include the ability to get an ASCII or hex dump of an area of memory, disassemble memory and display the output in Z80 mnemonics, search memory for a key up to four bytes long, modify memory in hex or ASCII, disassemble to the printer, change the contents of any register, perform hex arithmetic (but no decimal conversions, unfortunately), go to a user routine, move a block of memory, set a block of memory to some value and dump the screen contents to a printer.

The documentation supplied with Tasmon is very complete and professionally done. However, the documentation does not pretend to be a tutorial on Assembly language by any means. This is not to say that the beginning Assembly-language programmer wouldn't find Tasmon useful; it certainly is a lot easier to use than T-Bug.

A couple of Tasmon's features allow you to direct output to the printer. One prints the current screen contents, and the other directs a disassembly to the printer. Rather than use the standard ROM Call at 003B to put a byte to the printer, Tasmon goes straight to the parallel printer driver in the ROM at 058D. The problem with this approach is that if your printer requires a special driver, as many do, it will not be called.

I have found Tasmon to be an indispensable aid to debugging, moving machine-language files around, and just generally exploring Z80 code. Its flaws are

## SOFTWARE

---

minor, and more in the wish-list category than anything else.

*(The Alternate Source, 704 N. Pennsylvania Ave., Lansing, MI 48906; cassette or disk, 16K required, 32K recommended; \$29.95.)*

Rowland Archer  
January 1982

### ★ ★ ★ 1/2 Ultra-Mon Interpro Model I and III

Calling Ultra-Mon "Son of Debug" would not be missing the mark. Like Debug, it can execute when an error or breakpoint is encountered and also when a set boundary is overstepped. In addition, it shows the content of each Z80 register pair, the next 16 bytes at the relative address in ASCII and Hex, displays a full page of memory, executes single steps or calls, and lets you modify memory or registers.

Unlike Debug, it shows a disassembly of the next instruction to be executed, disassembles a block of memory, or does a dynamic trace at the rate of eight steps per second. All the data from a disassembly or a trace goes to the printer automatically, if one is available.

Ultra-Mon relocates itself, relocates a block of memory (changing all internal addresses), comes on tape (but can be put on disk), and runs in standard 16K, 32K and 48K Level II machines. What more could you ask? *(Interpro, P.O. Box 4211, Manchester, NH 03108; \$24.95 on disk and cassette.)*

Dennis Thurlow  
April 1981

### ★ ★ ★ ★ XBUG Computer Applications Unlimited Model I and III

XBUG is an excellent monitor and software debugger with efficient power, clear documentation, and low cost. XBUG has 25 commands and command variations. These can be categorized into memory and register manipulation functions and software development functions.

XBUG displays a page of memory by entering the command byte and starting address for the memory page. The memory display format for XBUG is 16 lines, 16 bytes to a line, in a split-screen format with the hexadecimal display on the left and the ASCII representation on the right.

Searching through memory for bit patterns up to 16 bytes long is also easy. XBUG searches all memory starting at the current page and displays either a page of memory on a find or a no-find message. Memory searching can be resumed by two keystrokes since XBUG retains the current search pattern until you enter a new pattern.

It modifies memory by either setting a block to a constant, or by writing hexadecimal data into memory byte by byte.

XBUG relocates to any address in RAM. Therefore, it can be used with virtually any system program regardless of its location in memory. The only requirement is that you have at least 2,560 bytes of RAM for XBUG either before or after the system program.

Another handy debugging technique is the use of program

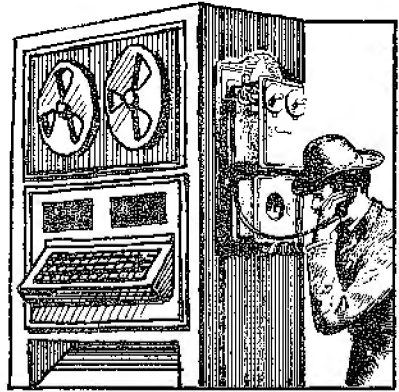
breakpoints to stop a program at a user-specified place. With XBUG you can set up to 10 program breakpoints and remove them either selectively or globally. When a program hits a breakpoint, the last register or memory display is updated and displayed on the screen.

During debug, XBUG's display and keyboard monitoring capabilities can be disabled allowing test program access to the keyboard for input and to the screen for output.

XBUG can route a copy of the screen display through the parallel printer port, or with the Model III through a serial port.

XBUG's simple manual describes all the commands complete with examples. I am impressed with the quality of this product; the manual, software and examples are excellent. XBUG is a good investment. (*Computer Applications Unlimited, P.O. Box 214, Dept. ABM, Rye, NY 10508; \$19.95.*)

Alan Burnes  
March 1982



bytes in memory and display them (while pronouncing the hex names).

The third program allows you to encode speech using hardware that Saville explains how to make. The routine is easy to write, essentially doing what ROM normally does when it reads tapes, except that here you shorten the timing loops.

The voice is extremely raspy, but usually intelligible. The verbal output is limited to the numbers and letters in the hex number system. The letters are output as phonetics for clarity.

Instructions are provided for hardware modification requiring a simple A/D converter. One method requires you to modify the keyboard itself. For the other you build an external device to connect to the expansion interface port or to the keyboard expansion port. Instructions also show you how to modify the tape recorder for use as a microphone for inputting speech.

Although the package is not expensive, I cannot recommend it. The tone quality is too low, and the A/D encoding takes up too much memory, using 512

## MUSIC/VOICE SYNTHESIS

★ 1/2

**Cheaptalk**  
**Alan Saville**  
**Model I**

Cheaptalk comes with three programs. The first is a machine-language routine to output sounds through the cassette port. The second is a Basic driver routine to read

bytes per second of speech. A 10-word vocabulary would require a full 5K of storage. Cheap-talk is a good idea, but it needs a better encoding technique to allow for better resolution of voice tones.

*(Alan Saville, San Diego, CA 92101; \$19.95.)*

Bruce Powel Douglass  
May 1982

★ ★ ★

### **Keynote**

**Hayden Book Co Inc.**

★ ★ ★

### **Music Master**

**Instant Software Inc.**

**Model I**

The Music Master software package consists of a Level II cassette tape containing four programs: Micro Organ, Kaleidopy, Keymania, and Composer. All will run in 4K of memory and require that the computer's cassette output cord be attached to an external amplifier.

The first program, Micro Organ, is a machine-language program that turns the top two rows of keys into an organ keyboard on which one note or voice can be played. You can choose between three tone colors, piano, organ, and harpsicord, but the names do not accurately represent the sounds.

Another drawback of this program is that the music is staccato sounding because you must release the first note before the second note can be played. Also, the only way to save a musical piece is to record it with a microphone and tape recorder.

The second program, Kalei-

dopy, generates random musical notes and produces a graphic design on the screen as it plays the notes. It is entertaining for a few minutes, but then each piece starts to sound the same.

In the third program, Keymania, random notes are played and the players must replay the notes in the order that they were heard. The game can be challenging and fun.

Composer is the final program and the least valuable. Like Kaleidopy, this program generates random musical notes, but plays them in user-selectable keys and tempos. It's boring.

Music Master is a fair package, ideal for demonstrations and for the person who is interested only in occasional experimentation with computer music. Its price is very reasonable, and its quality and documentation are good.

### **Keynote**

Keynote is a Level II machine-language/Basic program that requires a minimum of 16K RAM. The program permits the user to enter a musical piece using a special mini-language in which symbols are used to label a note's pitch and duration. The resulting piece can be edited and saved on cassette.

The program's synthesizer features a five-octave pitch range, three possible tempos, and the ability to repeat certain sections of the piece. Keynote also features extensive error-trapping routines that result in smooth program execution. After you enter a song, you can choose from a multi-function options list to edit and review the piece, play the piece, or save the piece on tape.

This package has some

negative points, however. The execution time of the Basic program is slow, especially in the edit and review mode. The tape saves and loads also get quite long. Execution time can make or break a program's effectiveness for frustration-free use, and Keynote is just too slow. The program also supports only one voice at a time, limiting the piece to melody—not enough for serious music programming.

Keynote is, however, an excellent program for the newcomer to computer music who would like to try it without committing himself to a large expense. The quality of Hayden's software is outstanding, and their documentation is good. For occasional use and for demonstrations, Keynote rates as a good buy.

*(Keynote is no longer available. Hayden Book Co. Inc., Rochelle Park, NJ 07622. Music Master, Instant Software Inc., Peterborough, NH 03458; Model I and III, \$14.95.)*

Jim Heid  
May 1981

★ ★

**Micro Music**  
**Tandy/ Radio Shack**  
**Model I**

Micro Music gives you musical notes including sharps, flats, and naturals over five octaves. You control the tempo with rests, tremolos, triplets, and staccato notes. There are also a number of voice-modifying commands that change timbre.

The documentation is adequate, but, as usual, it leaves out some surprises, which prove that Murphy will never be forced to the unemployment office

while computers exist. If your musical selections act peculiar or even weird, look out for inadvertent spaces after opening parentheses and, the worst offenders of all, an accidentally inserted closing parenthesis or misuse of the up and down arrows.

Micro Music does not provide unlimited space to write your song, but exactly 16 full lines of characters before you start writing over the top line. This is only destructive if you are busy transcribing a Bach fugue and forget to watch the screen!

With careful keyboard work, some study, and the ability to read music, you can enjoy this extension of your TRS-80 and Tandy's never-ending fight to educate, instruct, and now amuse. Besides, the sample listing in the documentation is a real tour de force presentation of "The Flight of the Bumblebee." If you are old enough, it will bring back memories of "The Green Hornet."

*(Tandy/Radio Shack, Fort Worth, TX 76102; Micro Music is no longer available from Radio Shack stores.)*

Allan S. Joffe  
October 1980

★ ★ 1/2

**Music Composer/Editor**  
**PFDC Software**  
**Model I**

Music Composer/Editor composes and plays music through a radio placed next to the TRS-80 keyboard unit. The documentation is very good, but the loading is touchy. Entering music into memory requires octave, note length, and note. If you can read music at all, you will be able to

enter notes quickly. Daisy, along with a random 25-note piece, is provided as a part of the program. Many other features, such as listing, tape saving and loading, editing and playing, are provided.

The music that results from this program is quite good; it is obvious that a great deal of thought went into its creation. However, there are some drawbacks. Only one note can be played at a time. Whenever a song is not playing, great amounts of computer noise emerge from the radio. All in all, this is a nice novelty demonstration program; I'll let you decide whether it is worth the price. (*Music Composer/Editor, once sold by PFDC Software, 784 Goucher St., Gretna, LA 70053, is no longer available.*)

Rod Hallen  
January 1980

**Music Master**  
(See Page 192)

★ ★ ★ ★

**Orchestra-80**  
**Software Affair**  
**Model I and III**

Sometimes a product has such good quality and value that it soon becomes the standard in its class. Orchestra-80 is this kind of program. This hardware/software combination for computer music generation is versatile, easy to use, and reasonably priced.

The hardware half of Orchestra-80 consists of a 1½-by-2-inch printed circuit board that attaches to the TRS-80's keyboard edge connector or to the expansion interface's bus extension. It is then connected to an external

amplifier, using a standard RCA-type cord.

The Orchestra-80 software, a 16K machine-language program, contains a digital synthesizer that features a six-octave range, four simultaneous voices, and a music language compiler that will accept music written in any key, any time signature, and any note value from whole to 64th notes.

The software also contains a text editor that provides a 14-line text display area and a two-line status display area; this is one of the program's best features. Equally impressive is the file-management system, which provides for orderly storage and retrieval of user files on tape or disk. Finally, the initialization routine lets you alter the tone colors of one or more of the four registers to achieve special effects or special dynamics.

Orchestra-80 is extremely well documented. The manual gives full instructions on setup and use of the system, including an important section for nonmusicians on reading sheet music.

Software Affair's Orchestra-80 represents a best buy in computer music systems. Its price is less than half that of comparable systems, its documentation is thorough, and its operation is bug-free and enjoyable. (*Orchestra-80 is no longer available; it has been replaced by Orchestra-85 and Orchestra-90. Software Affair, 858 Rubis Drive, Sunnyvale, CA 94087.*)

Jim Heid  
May 1981

**Orchestra-85 and -90**  
(See Page 310)

## TERMINAL PACKAGES/ COMMUNICATION SOFTWARE

**Automated Communications Exchange**  
(See Page 301)

★ ★ 1/2  
**Cassette Comm**  
**Tandy/Radio Shack**  
**Model I**

After getting the Modem I from Radio Shack, I needed some way to interface it with the computer. Cassette Comm, a software package written specifically for the Model I Level II computer, suited my purposes. It is a bit more limited than the RS-232C Interface Board, but for my needs, it does a good job.

I got the Cassette Comm package so I could connect with my company's computer and work at home and to call the local community bulletin board service (CBBS). In CBBS's menu, the user can choose to change both the baud rate and the duplex mode of the CBBS. While experimenting with this, I unearthed some Cassette Comm limitations. Usually the baud rate can be changed and the Modem I allows for any rate from 0-300. The RS-232C Board advertises a range from 50-19, 200. The Cassette Comm, however, is permanently set at 300. Thus, if the computer you want to reach has a baud rate of 150, you are out of luck.

The Cassette Comm is set in the simplex mode, which means that your modem will continue

to receive messages unless you hit a key on the board. If you do this while receiving a message, your sending signal will interrupt the receiving signal and what you see on the screen will not make any sense. By selecting the half-duplex mode on CBBS, the echo disappeared, and what I typed did not appear on the screen.

Another limitation with Cassette Comm deals with the ans/off/originate switch. Usually your computer can be at either end of the telephone line. You can call another computer (in which case you would be the originator), or another computer might call you (and you would be the answering computer). The Cassette Comm package, however, only allows you to originate the call.

Despite its limitations, Cassette Comm has what I need. (*Tandy/Radio Shack, Fort Worth, TX 76102; \$9.95, Model I, Level II.*)

Virginia Dible  
January 1982

★ ★ ★ 1/2  
**Colorterm**  
**Martin Consulting**  
**Color Computer**

Colorterm converts a 16 or 32K Color Computer into an intelligent terminal. It possesses an extremely important property: It lets you get "on the air" almost immediately by ignoring its advanced features. When used in this fashion, Colorterm is almost invisible, with no extensive set of commands to get in the way. Colorterm's advanced features can be learned at a later

time. Be assured, this is a painless package to work with at almost any level of sophistication.

Colorterm occupies less than 4,900 bytes of RAM, beginning at \$1C00. It requires a high-resolution (6144 byte) video screen for the software-defined character set, one of its most distinctive features. The display memory employed starts at \$400, so the Radio Shack disk system cannot be used in conjunction with the present version of the program.

That high-density character set featured in Colorterm's advertising gives you a true upper and lowercase display, 21 lines by either 51 or 64 characters (keyboard selectable). Lowercase characters have descenders. Colorterm uses the same characters for both line formats; spaces are just reduced for the higher-density display.

The normal Colorterm display is black on a green background. This can be reversed from the keyboard, although the legibility suffers a bit. This feature is useful for highlighting particular lines of text, such as operator instructions.

The Colorterm character set can be patched into other programs to dress up their output. The manual contains the details on calling several I/O subroutines from other Assembly-language programs.

Colorterm comes configured for the most common ASCII data format: 300 baud, 7 data bits per word, even parity, and 1 stop bit. This is the format employed by CompuServe and Telnet, and I presume that it suffices for The Source and other major services. It is supported by the IBM TSO

(time-sharing option) installations.

Should your requirements be different, any of these parameters can be changed by loading (but not executing) Colorterm and then performing specific POKEs from Basic. The manual gives the relevant addresses, as well as the data for saving the modified program to tape. There is also a keyboard command for switching between full and half-duplex transmission: the default is half duplex.

Experimenters will quickly discover that there is a misprint in the Colorterm manual. The definitions of the shift, up-arrow and shift, down-arrow combinations are reversed. These give an underline and a left bracket, respectively. The underline is an independent character; it cannot actually be positioned under anything previously typed.

When you're ready to use Colorterm, the right arrow gives you access to 18 predefined functions, each accessed by a single letter. In addition, you can define up to seven personalized functions using the arrow plus the letters T through Z; these functions generally take the form of ASCII control codes POKEd into the appropriate locations.

Colorterm has two buffers reserved for frequently used macros, each of which may be up to 128 characters long. Once you have customized Colorterm in this fashion you have to store it on tape and be sure to use the appropriate version in future work.

It is sometimes useful to preserve a window of data on the screen while the rest of the display scrolls. Colorterm allows



you to do this at the top of the screen. Highlighting the preserved message with reverse video can be effective, too.

Colorterm allows you to compose a file of useful size off line and then upload it in one shot. This process calls for close attention to loading offsets and addresses in general, a bit clumsy for my taste. You should be aware of the existence of this capability, though, because it is also employed in transmitting data from the buffer to a printer through the Color Computer's RS-232 port.

One more interest is the ability to scroll through a mass of data down-loaded from the host to the Color Computer. You might want to receive data and go off line to examine it at your leisure, for example. Colorterm handles this easily. Unfortunately, there is no provision for scrolling backwards.

I have found Colorterm easy to use and quite flexible for my day-to-day requirements. It is moderately priced, and the text densities are high enough to allow doing some serious work. (*Martin Consulting, 94 Macalester Bay, Winnipeg, Manitoba, R3T 2X5 Canada; \$34.95.*)

Scott L. Norman  
November 1982

★ ★ ★ ★

**The CompuServe Software Package**  
**Tandy/Radio Shack**  
**Color Computer**

CompuServe is an information service available through Radio Shack. A software package converting your TRS-80 Color Com-

puter to a terminal is available for \$29.95. It includes the TRS-80 Videotex Software, one free hour of access time on CompuServe, an operator's manual, and an identification number and password.

You need a modem or acoustic coupler connected on the back of your Color Computer to go on-line with CompuServe. Once you are logged-on to the service, you have several options and features available to you. Some of these services include: Micronet Personal Computing, MicroQuote, Newspapers, Electronic Mail, Home Information, Education, Special Services, Finance, and Communications.

Current CompuServe user rates are \$5 per hour connect time between 6 p.m. and 5 a.m. Monday through Friday (at 300 baud), all day Saturday, Sunday, and during any legal holiday. During prime time the cost is \$22.50 per hour. The fee includes 128K bytes of free disk storage when using the Micronet computing service.

MicroQuote, a stock market information service, is a recent addition to CompuServe. It allows the stock investor to get data on more than 32,000 stocks. MicroQuote is updated daily; historical prices, volumes, and dividends are available back to December 31, 1973. It costs you a minimum of \$1 each time you access MicroQuote. You are charged 5, 10, or 15 cents for each daily, weekly, or monthly data set, respectively. The data set includes the date, volume, high/ask, low/bid, and the closing price of the stock.

CompuServe news service is accessed through the main menu. There are several papers

to choose from; you also have access to the AP News Wire Service. Drawbacks to this feature is that it is time consuming to access the news when you can read the same information in your local newspaper for about 20 cents. The news is not up-to-the-minute, and you can't keyword search the news for desired subjects of interest.

Electronic Mail allows you to send a message to another subscriber. You must know the receiver's ID number. When the person is logged-on to CompuServe the system announces that a message is waiting. At present, this system is not an alternative to the U.S. mail system.

*(The TRS-80 Color Computer CompuServe package is priced at \$29.95. CompuServe packages are also available for the Model I, II, and III for the same price. For additional information contact CompuServe, P.O. Box 20212, 5000 Arlington Centre Blvd., Columbus, OH 43220; (800) 848-8990 or Tandy/Radio Shack. As of Dec. 1, 1982 CompuServe subscribers numbered 33,000 and the prices quoted above were accurate.)*

Howard Berenbon  
January 1982

★ ★ ★ ★ ★  
**Dialog Information Retrieval Service**  
**Lockheed Corporation**  
**Model I, II, and III**

The Dialog Information Retrieval Service is a commercial data base accessed by a modem. Its sole purpose is to provide specific information on almost any subject to

businesses, individuals, and families.

Having over 200 data bases covering some 55 million records, the service provides the computer user with an easy means of searching for and retrieving information. Bases currently on line include listings from medicine to music, legal resources to geology, chemicals to complete listings of grant and foundation sources. History, U.S. patents, most current periodicals, biographies, agriculture and energy are all subjects open to research.

Dialog is a giant clearinghouse for data bases from both private and governmental areas. Additional bases are always being added; one of the most recent is the Career Placement Registry, a file containing the names of graduates from over 1,400 colleges and universities. The scholastic records of graduates can be searched and their resumes ordered so that unique and qualified individuals can be found almost effortlessly.

What makes Dialog most attractive is the wide range of search techniques and aids available. You may search for specific information using certain key words found in the reference's title, subject index or abstract. Most data bases also provide additional indexes of searchable fields such as author, date of publication, language, journal name or classification code. Each data base provides a complete list of particular search fields.

To find specific references, Boolean operators can be used. If you are looking for records dealing with solar energy, for instance, you may search under

the descriptor "Solar" or under "Energy." In addition, the "and," "or," and "not" operators make search expressions such as "children and television and violence" possible.

Through the Search and Save feature, you may formulate a search pattern that is stored on the system and automatically performed as data-base updates are made.

References selected may be viewed on the CRT, sent to your printer in several formats, or printed on Dialog's own printers and mailed to you. If you desire complete documents, a Dial-order service places orders with a number of private retrieval services.

Upon signing on to the system and requesting the data base to be searched, Dialog estimates sign-on cost. In fact, Each time a new data base is selected the system presents a cost analysis, making it easy for information suppliers to keep track for billing purposes.

Unlike CompuServe, which charges a flat hourly rate, the on-line charges vary with each data base selected on Dialog. Rates range from \$15 per hour for several practice data bases (called Ontap) to \$300 per hour. The average rate ranges from \$35-\$70. In addition, a per-record charge for printouts made off line and mailed to you varies per data base but ranges from 10 to 30 cents with a few exceptions. All but 19 of more than 200 bases make no charge for records printed on line.

One final additional charge is a \$6-per-hour connect charge, made by either Telenet or Tymnet. Of course, you can dial Dialog direct if you live nearby.

Dialog's excellent, and unique, customer service softens the blow of long searches. Each month active customers receive a free newsletter explaining Dialog changes, new data bases and search tips. Usually included are announcements of free time provided on at least one data base each month. Dialog also operates a series of training seminars in principal cities all over the world. For the serious researcher these seminars are well worth the price since learning how to plan search strategies can save hundreds of dollars in connect time.

*(Dialog Information Retrieval Services Inc., 3460 Hillside Ave., Palo Alto, CA 94304. The average Dialog access rate is now \$25-\$75.)*

Alan Neibauer  
June/July 1982

★ ★ ★ ★

### **Omniterm Lindbergh Systems Model I and III**

David Lindbergh's Omniterm is a Model I smart-terminal package (the disk will run on the Model III after processing with the TRSDOS Convert utility) that boots up in a dumb-terminal mode. On the Model I it lists the RS-232's sense switch settings and configures the UART accordingly, activating the lowercase driver if the computer has a lowercase mod. Since the Model III RS-232 has no sense switches, Model III Omniterm uses a standard setting of 300 baud, 7 data bits, 1 stop bit, and even parity.

Pressing the @ key twice places the computer into the

command mode, showing a command menu and status display. The letter P toggles the printer on and off. If the printer is not fast enough to keep up, Omniterm can buffer up to 2,048 characters. R, screen reformatting, lets you set your screen width. This is handy when communicating with, for instance, Apple bulletin boards, which are set up for 40 characters across. In conjunction with reformatting, a carriage-return suppression function enhances the ability to print neat, even lines; line-feed suppression lets you communicate with a computer that sends only a carriage return at the end of a line rather than the standard carriage return and line feed.

The D command provides software control of half or full duplex. Default is full. E toggles the echo function used when you must supply an echo to another computer or terminal; default is off. G controls a CR/LF grouping function normally used when the TRS-80 is the host computer for a Teletype or video terminal. I toggles input to the buffer when downloading from a remote computer; O controls output from it. Omniterm provides for setting a delay between characters if sending to a slow machine, and has the ability to provide prompted output.

Other commands are one-time functions, such as returning to DOS, a cold start to reset all default values, or creating or changing an auto sign-on of up to 63 characters.

One of Omniterm's most powerful features is that it allows translation of a byte to or from a device to any other byte. Typing T from the command mode lets

you examine and modify 7-byte translation tables, one for each possible device and one for each direction, to or from a device. This is useful for code conversions, or when customizing Omniterm for use with additional hardware. For instance, if you have a printer built for use with IBM equipment, using EBCDIC code instead of ASCII, the needed conversion can be done while in Omniterm—no external program is needed. While other terminal programs can support code conversions, Omniterm is the only one I have seen where the operation is self-contained.

Typing B while in the command mode lets you scroll back the display by holding down the space bar. Omniterm holds in memory the last 2,048 bytes received at all times. If something goes by too quickly, just scroll back and read it again.

Z sets the real-time clock to zero, enabling you to keep track of time on a computer system with time charges, or when paying for a long-distance call. The Clock function must be activated from DOS before loading Omniterm so the clock displays on the screen.

F loads a file into the I/O buffer so it can be sent out from the TRS-80. S saves a received file to disk. In case of a disk error while saving, Omniterm closes the file so text already written is retrievable. The contents of the buffer are intact, so you can try again.

There are five additional utilities on the disk. Two convert binary files to ASCII hex files and vice versa; two convert any type of file to a special error-detecting, bit-packed format with checksums at the end of each line, allowing 50 percent

more data in the same amount of space, and vice versa. The last is a line-oriented text editor, useful for getting rid of garbage picked up when a buffer is opened too soon or closed too late. It can also create files to be transmitted by Omniterm.

Five file-setting table files are supplied, and can be modified for a user's requirement. One reverses the keyboard from typewriter style to standard TRS-80 format, shift for lower-case, and the others are for using the Models I and III with The Source and CompuServe or Micronet.

The documentation is excellent. It contains 76 pages of explicit directions as well as several appendices full of information such as code conversion tables and a glossary. All registered owners are notified of changes and improvements.

For the user in search of a top-flight terminal program at a reasonable price, Omniterm fills the bill admirably.

*(Lindbergh Systems, 41 Fairhill Road, Holden, MA 01520; \$95, 32K, disk system.)*

Ronald H. Bobo  
April 1982

★ ★ ★  
**Smart Terminal Program**  
**Howe Software**  
**Model I, II, and III**

The Smart Terminal Program (STERM) is designed for a 16K, 32K, or 48K tape or disk-based Model I or III, or the Model II with CP/M. STERM, with an RS-232 and modem, lets you access another computer over the telephone lines and transfer files to or from the host system.

STERM's communications parameters can be set to conform to the requirements of the host computer. The baud rate, full or half duplex, the number of bits per word, the number of stop bits, even, odd, or no parity, and a line feed after carriage return option are all software selectable.

The software stores whatever appears on the screen in a memory buffer and dump the buffer on command to a disk file, a tape file, or a printer. Disk and tape files can be read by both Scripsit and Electric Pencil. In addition, everything on the screen can be simultaneously routed to the printer, with or without saving it in memory.

You can type text or programs directly into the memory buffer for later transmission to the host computer one line at a time, upon receipt of a one to five-character, user-defined prompt string, or continuously.

Files to be transmitted may also be prepared with Scripsit or Electric Pencil, saved to tape or disk, and then read into the memory buffer by the terminal program. Disk users can read and store Basic programs in disk files. The host computer can automatically send and receive data from the TRS-80's memory buffer if the host is capable of sending Standard Device Control Characters.

Other special commands include a true break code and an exit to DOS without rebooting. You can also display all available commands on the screen and redefine the control keys to transmit any ASCII character. Thus, symbols not on the TRS-80 keyboard can be sent to time-sharing systems.

STERM does have faults. Only alphabetic control keys can be redefined to send arbitrary ASCII values. And special characters or control codes cannot be typed into memory for later automatic transmission.

Unless you have a disk-based system, Basic programs and data cannot be read into memory for transmission to the host system. Another shortcoming is that if you regularly require a specific communications protocol other than the default values or special control key definitions, you must set these values every time you load the program. You can not save the parameters to disk or tape. Also bothersome is that you can only scan data stored in memory one line at a time.

Additional minor complaints include the lack of absolute cursor positioning and no provision allowing the system clock to keep track of connect time. The Model III version only comes on a 500-baud cassette, making tape reading and writing unnecessarily time-consuming.

The detailed documentation is fairly clear. Only the discussion of changing baud rates and redefining control keys is confusing.

In summary, STERM provides many features not found elsewhere for tape-based computers. Disk users may prefer the significant advantages of somewhat more expensive programs such as ST80-III or Ominterm.

*(Howe Software, 14 Lexington Road, New City, NY 10956; Model I and III, \$69 cassette, \$74.95 disk, Model II \$79.95.)*

Richard K. Wallace  
June/July 1982

★ ★ ★

### **Smart80d** **The Microperipheral Corp.** **Model I**

★ ★ ★ 1/2

### **ST80-III** **Small Business Systems** **Group** **Model I, II, or III**

Smart80d and ST80-III are smart-terminal programs that, in conjunction with a modem, allow a TRS-80 (with a disk drive) to access another computer by telephone or radio. They are the best two programs I have seen on the market. Both are easy to use, well-documented, and readily adaptable to the various communication requirements of the system to be accessed.

A terminal program gives the TRS-80 the ability to communicate with another computer or terminal. A smart terminal program also allows the transfer of data or programs both to and from the other terminal or computer.

You can set the communication parameters of both programs to conform to the requirements of the host computer. Full- or half-duplex transmission is software selectable with both programs. You can select the number of bits per word, the number of stop bits, and even, odd, or no parity can be selected with Smart80d but only with the RS-232 version of ST80-III. In addition ST80-III lets you both send and receive a line feed after a carriage return and to send a predetermined number of nulls after a carriage return.

Both Smart80d and ST80-III

can store what is seen on the screen to a buffer in memory and then, on command, dump the buffer to a file on disk. The buffer can be opened and closed automatically by both programs.

Smart80d and ST80-III can send text or programs to another terminal or computer. Smart80d also can automatically open and close the receiving terminal's buffer if it is equipped for auto receive. Both programs have three other functions in common: the ability to route everything that goes to the screen to a printer, the ability to temporarily exit the program to execute a DOS command, and the ability to transmit a true break code.

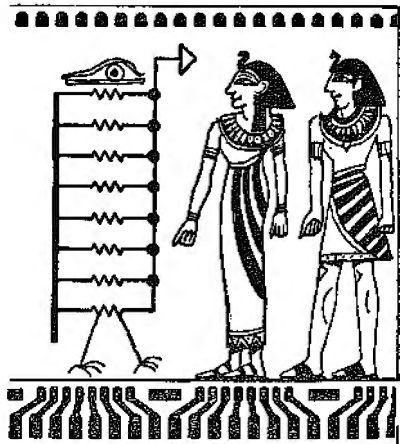
Smart80d has two buffers in which a log-on or any other message can be stored. ST80-III loads its log-on buffer from a table that is stored on disk and can transmit the message on command. ST80-III can display all of its commands on the screen and can be programmed so any key can send any code. All incoming codes can be changed to another single code. ST80-III has two features that are unique to terminal programs that I have seen: It allows use of the system clock, and it will allow an amateur radio operator to monitor a frequency for his call sign or any character string for which ST80-III is programmed.

Both Smart80d and ST80-III have similar utility programs on their disk (although Smart80d has better screen prompts). The first creates a message for uploading to a bulletin board. The other takes a Basic program that has been down-loaded and lets you delete extraneous carriage

returns and lines that are not needed in the Basic program. Both programs also have a utility list of all possible 255 codes that can be sent or received in decimal and their definitions. The Smart80d version also gives the codes' hex equivalent and its use in the TRS-80.

ST80-III's other utilities are: A program that creates a checksum for any file; a program to generate a table that contains the log-on message generator and can redefine control codes to eliminate possible conflicts between systems; and a program to convert binary files to ASCII or vice versa. These last two programs can also scramble and unscramble a file for a security transmission. There are also two programs to send and receive machine-code programs in Intel paper tape format.

The documentation provided with both programs is excellent. The Smart80d manual goes into more detail on the actual use of the software, and the ST80-III manual gives better examples of how it accesses various bulletin boards, The Source, and Compu-



Serve. The documentation for Smart80d also comes on the disk in a file that can be printed on a printer or listed from DOS.

Smart80d and ST80-III do an excellent job of making the TRS-80 a smart terminal. ST80-III has a few more features than Smart80d, but Smart80d is much more cost-effective than ST80-III.  
*(Smart80d, Model I \$29.95, Smart3d Model III \$49.95, The Microperipheral Corp., 2643 151 Place NE, Redmond, WA 98052. ST80-III, Small Business Systems Group, 6 Cartisle Road, Westford, MA 01886; Model I and III \$150, Model II \$250.)*

Bill Everett  
September 1981

★ ★ ★ ★

### **TDS/DFT**

#### **Big Systems Software Model I and III**

If you have a modem for your computer, you should not be without TDS/DFT (Tape Downloading System and Direct File Transfer).

These two programs are machine-language smart terminal utilities. TDS is similar to many smart-terminal programs on the market, while DFT can transfer machine-language programs without conversion (if the host computer also uses DFT).

In addition to uploading and downloading, with these programs you can access Basic directly, use all ASCII control codes, including upper and lowercase, transmit at full or half duplex, transmit at 10 different speeds, and define 10 control

keys. The Model III version includes a high/low cassette rate option.

One of the supplemental programs, SYSCVT, changes a machine-language program to an encoded Basic program. You can then transmit the encoded program to a host computer, but it must also have the TDS package to decode the data.

The second complementary program is AUTOL. You type a log-in code, and the program bypasses the manual BBS log-on procedures (first name, last name, city, and state).

The well-documented manuals explained the programs' use clearly. My only problem occurred when I first loaded TDS and attempted to call up the system. I had absolutely no results. I soon discovered that with a Model III and Lynx modem, I must enter the command POKE 16912,56 before using the program.

*(Big Systems Software, 27574 Via Rosalie, Mount Clemens, MI 48043; Model I and III, \$29.95 and \$19.95 respectively, \$40 for both.)*

Tim Knight  
June/July 1982

★ ★ ★

### **Uniterm**

#### **B. T. Enterprises Model I and III**

Uniterm is short for Universal Terminal Program. It works on both the Models I and III, and can be used with NEWDOS +, TRSDOS, DOSPLUS, LDOS, and NEWDOS80 versions 1 and 2. Your system must have at least one drive and 32K RAM, as well





as some type of serial interface, and a Lynx, Chatterbox, or Microconnection modem.

The following commands are available: Auto buffer open/close; load/save binary file; close buffer; display/print buffer; exit to DOS; half/full duplex; define initialization parameters; load ASCII file to buffer; change modem parameters; open and zero buffer; send buffer in prompt mode; send buffer with auto open/close buffer codes; save buffer in ASCII format; transmit buffer; set screen to desired width; type to buffer; and display command list.

The manual includes a glossary of terms used in smart terminal programs and instructions on how to build a special table if you so desire. If at any time there are updates for Uniterm, a section explains how to download those updates. A technical information section explains overlays and locations that Uniterm uses and for what.

A list of public access systems in use around the country is also provided. It includes what type of system, phone number, ac-

cess hours, baud rates, and special interest codes.

I enjoy using Uniterm—it conforms to the way I want to do things.

(B. T. Enterprises, 171 Hawkins Road, Centereach, NY 11720; \$79.95.)

Sal Navarro  
June/July 1982

---

## UTILITIES

---

★ ★ ★ 1/2  
**Asptch 3.2**  
**Byte Miser Software**  
**Model I or III**

Byte Miser Software has developed a patch to Editor/Assembler that assembles programs directly to memory. It is particularly useful for writing machine-language subroutines to be called from Basic.

Asptch 3.2 loads on top of Editor/Assembler. The tape replaces over 300 different areas, while stealing less than 400 bytes from the text buffer (just like Basic). This high memory is used for the machine-language program. Asptch already has keyboard debounce. You can also use this keyboard debounce routine from Basic.

The Asptch memory-size command (A) redefines protected memory. This is not a destructive command. It leaves the text buffer intact, unlike Basic. This is useful when you find that your program uses more memory than expected.

You can test your machine-language program. The Execute

command jumps to the starting address of your program just like Run in Basic. Asptch can regain control in several ways. A return instruction without a corresponding call will return to Asptch.

The machine-language program can jump to an Asptch or Basic reentry point. The reset button will put you in Basic. Once in Basic, you can use the System command to reenter Asptch. As long as your program does not disturb Asptch or its text buffer, you can regain control without losing a byte.

A unique and powerful feature of Asptch is the memory-size command. This lets you enter Basic with Asptch, your Assembly-language source code, and your machine-language object code all protected in high memory. Using this feature, you can develop machine-language subroutines for use with Basic.

The Convert Display, and Modify commands put Asptch in a monitor mode. You can display memory addresses and their contents in hex, decimal, and the printable character codes (ASCII). Using the up and down arrows, you can scroll through memory. Subcommands can modify locations and enter the Basic calculator mode.

Don't confuse this monitor with T-Bug, Z-Bug, or anything else. It has a provision for breakpoints or looking at registers' contents. This monitor is not designed for heavy debugging. The author assumes you would rather modify your source code and reassemble than play with your object code.

Asptch adds a few more features to Editor/Assembler. After each source dump, it pro-

vides a verify option. This gives you the opportunity to rewind the tape and let the computer compare the file to memory just like CLOAD? in Basic. Asptch will not work with Editor/Assembler version 1.1.

Asptch's major competition (on the Model I) is Microsoft's Editor/Assembler Plus. Asptch's advantages are lower cost (assuming you already own Editor/Assembler 1.2), source tape verify, Basic reentry with Asptch, an Assembly-language program saved in protected memory, and a larger text buffer.

*(Byte Miser Software, 720 W. Haven Blvd., Rocky Mount, NC 27801, could not be reached for an update.)*

Dan Zuckerman  
February 1982

★ ★ ★

### **AUTOK and QEDIT Discovery Bay Software Model I**

AUTOK provides an automatic keyboard repeat feature, so it replaces the repeat key—it causes any key that is pressed for more than one half second to repeat itself about eight times per second until it is released.

QEDIT is a screen editor for Basic program statements, and it is the TRS-80 Basic text editor I've been looking for. To edit a line, bring it to the screen with List, press the clear key to get into the QEDIT mode, and use the four arrow keys to position the cursor where you want to make changes. QEDIT also lets you move lines around within a program without having to re-type them.

You can't use QEDIT on a line that is more than 64 characters long or while the normal Level II text editor is in use, but these are minor disadvantages. (Discovery Bay Software, P.O. Box 464, Port Townsend, WA 98368. AUTOK and QEDIT are now Omnit-Key; \$23.)

Rod Hallen  
February 1980

★ ★ ★ ★  
**B17**  
**ABS Suppliers**  
**Model I**

Unlike many utilities I've examined, B17 is everything it's cracked up to be. It's both fast and easy to use.

ABS Suppliers claims that B17 will allow you to save, verify, and load Basic or machine-language programs almost four times faster than an ordinary cassette recorder. Data arrays (string or numeric) can be saved and loaded up to 165 times faster! All of this magic with no aggravation—nearly for free, for Level II 16, 32, or 48K.

There is no way to verify a data save but, using good quality tape, I have had only one bad save in 500.

Saving 60–75 percent of the time you spend loading programs makes B17 well worth the money. Its data handling speed makes it a steal. The introductory price is \$22, but by the time you read this, it may be \$50 and well worth it.

One note: If you have the cassette mod from Radio Shack, you must add one switch in order to bypass the mod. B17 won't work with any added audio processing devices.

(ABS Suppliers, P.O. Box 8297, Ann Arbor, MI 48107. B17 is available for \$14.95 plus postage.)

Stewart E. Fason  
March 1981

★ ★ ★ 1/2  
**BasicPro**  
**Softworx Inc.**  
**Model I and III**

BasicPro is a cassette-based, machine-language utility program that runs on a 16K Model I or III. Its capabilities include the following useful programming and debugging aids: renumber or copy statements, rename variables or line references, produce a cross-reference list of variables and line references, locate variables or line references, compress a program, recover a program, and merge two Basic cassette programs.

The package is very easy to use. Its documentation could serve as a model of clarity and logical organization. BasicPro is loaded into memory with the use of the System command; then the Basic program you are working on is loaded or entered.

The commands all have a simple and logical syntax. Each of them consists of a slash and one capital letter followed, when necessary, by the appropriate parameters. If you issue a command that requires memory beyond the capacity of the computer, an appropriate error message appears. A complete list of BasicPro commands and their functions appears in response to an invalid command.

Every base seems to have been covered by the program designers. I have found it a

pleasure to work with. Congratulations to Softworx for placing on the market such a useful and easy-to-use utility program at a reasonable price.

(Softworx Inc., P.O. Box 9080, Seattle, WA 98109; \$24.95.)

Joel Benjamin  
January 1982

★ ★ ★

### **Bootstrap Practical Applications Model I**

Bootstrap's advertisement sounded so fantastic that I had to try it, and I wasn't disappointed. This utility creates a machine-language program (BOOT/CMD) on the DOS disk that loads and runs your Basic programs.

The program arrived with a short but adequate instruction sheet. I had considerable difficulty loading the cassette, and resorted to using a volume setting about one quarter lower than normal. There were four dumps on the cassette, two for TRSDOS 2.2 or 2.3 and two for NEWDOS. BOOT/CMD loads at ABOOH 43786 decimal, so if you have any other programs in this area, they will interfere with it.

After running Bootstrap and loading your Basic programs, pressing enter twice lets you exit. Bootstrap then creates the BOOT/CMD program and clears the screen. By using CMD'S to get into DOS, type AUTO BOOT (enter) and your task is finished. Inserting your disk and pushing the reset button automatically loads the DOS, the verify command, your machine language programs, and Basic.

You can load and run as many machine-language programs as

you want, but you can only run one Basic program. All further commands will be ignored. Bootstrap can also be used to merge two or more programs and run them without interruption.

The instructions suggest you use the DIR command to list the disk directory, but if you are using more than one drive, the directory for the first flashes on and off the screen too quickly to read. The answer is to use the Free command which lists the disk name, formatting date and password for all drives in use.

You can fall into some traps playing around with Bootstrap, such as setting up BOOT/CMD so that it executes a continuous loop. If you can't get out with the break key, shut off the computer and remove the disk. Turn the computer on again, insert another DOS disk, and load it with the reset button. You can then reinsert the original disk and use kill BOOT/CMD to cancel the endless loop.

(Practical Applications, San Carlos, CA 94070, could not be reached for a product update. Bootstrap was originally sold for the Model I with 32K or 48K with cassette or disk.)

James Ranney  
July 1980

★ ★ ★

### **Compressor 1.1 Robert M. Chambers Model I and III**

There are many programming techniques for saving memory, but the most obvious ones eliminate unnecessary blank spaces and REM statements. This can be done manually, editing each program line, or you can use Compressor 1.1.

Compressor is a Level II machine-language program on cassette that removes all non-functional spaces and REM statements in a Basic program. REM line numbers are retained, since some GOTOs or GOSUBs within the program may jump to those line numbers. There are no combined lines, consequently no line numbers, GOTOs, or GOSUBs are changed.

Compressor doesn't remove spaces between quotes because the screen and printed text would be affected.

How much memory do you gain with Compressor 1.1? The five programs I wrote saved from 7 to 17 percent of original program length. The sixth one I tried was packed with individual line editing, yet Compressor squeezed out another 100 bytes somewhere, and the program ran perfectly.

While there are other packing programs available, Compressor 1.1 is fast, easy to use, and inexpensive. The documentation for Compressor 1.1 is easy to follow and includes information on how to retain the program in high memory for repeated use. Basic programming hints are also provided to save memory and gain high-speed execution of your programs.

*(Compressor 1.1 is sold by Robert M. Chambers, Napean, Ontario, Canada, who could not be reached for an update.)*

Fred Blechman  
January 1981

★ ★ ★ ★  
**Cross-Reference Utility**  
**Tandy/Radio Shack**  
**Model I or III**

While writing a new program

or improving an old one, it's easy to lose track of which variables go where and to forget how to get to a certain line. Fortunately, there is a good solution available for this problem. Cross-Reference Utility generates lists of references for the following important items: line numbers, variables, and reserved words.

The machine-language program is sold on cassette and comes with a well-written instruction manual, which includes a step-by-step procedure for operating the program. The manual also includes a section with concise directions for transferring the program from cassette to disk using the TAPEDISK utility.

The shortcomings in the documentation are very slight, but they are glaring when compared with the high quality of the program itself. Tandy needs to catch up on quality in the software department, and this program is a giant leap in the right direction.

*(Tandy/Radio Shack, Fort Worth, TX 76102; \$14.95.)*

Stephen F. Nowak  
May 1981

★ ★ ★ 1/2  
**Discat**  
**Myatt & Smith**  
**Model I**

When a little voice in your disk storage box whispers "One of these days we've really got to get organized," don't get upset—help is on the way.

The Discat Disk Catalog Index is a menu-driven machine-language program for the Model I, with a Model III version to be released soon. It will load an in-

dex file of your programs in about 20 seconds, and then tell you where to find any one of some 17,000 programs (around 7,200 for a 32K system), displaying the disk number and indicating which side the program is on. A disk number query displays the complete listing for either side of any disk in your index, and the program automatically keeps track of free space available.

Discat permits not one, but nine different index files, each of which can hold 800 program locations. As each disk directory is read, you can add to the current index any program on display, so you need not reorganize your entire disk inventory to put all your utility programs, for example, into a separate index.

While nonstandard DOS disks such as Pascal, Forth, and CP/M must not be inserted into the update drive, Discat allows manual entry of these disks' numbers and programs into whichever index you choose.

Probably the most useful routine in Discat is the search program. A search by program name does not require that the entire name and extension be entered; the name alone, the first few letters, or just the first letter will call a display of desired programs in alphabetical order.

A full editing menu allows deletion of a specific program or an entire diskful of program names. All submenus return the user to the main menu, where the current index name and its total number of disks and programs are always displayed. The current index in memory can be saved to disk at any time.

The program satisfies just about every need I could envi-

sion for a disk file organizer. It is easy to use, fast, adequately documented, and capable of handling far more information than most TRS-80 users will probably need. When I discovered how many disks I could recover just by eliminating duplication and recapturing unused space, the savings nearly equaled the price of the program.

*(Myatt & Smith, Tustin, CA 92680. This company could not be reached for updated information. Discat originally sold for \$50 and was available for the Model I with 32 or 48K, expansion interface, and one disk drive.)*

Robert C. Daigh  
July 1981

★ ★ ★ ★

### **Disk Directory Mumford Micro Systems Model I**

The Mumford Micro Systems Disk Directory is a very efficient record keeper. The Mumford sorts data alphabetically either by file or by disk, holds 280 files in a 32K system or 850 in a 48K system, and either displays the files on screen or sends them to your printer.

Now for the big surprise: The Mumford reads the information off the disk directory. All you do is insert your disk, give it an identifying name or code, and hit enter, and it loads the disk system, the file names, and then tells you how many free granules are left.

How do they do it? The program (it's in Basic) is heavy in PEEKs and POKEs, with a USR(0) and several lines of

numeric data statements. Mumford is able to POKE a subroutine that will call the DOS DIR program while retaining control in their own program. The program reads the screen through PEEK statements and stores the information in an array.

When you load all your disks in this file, you can sort, save, modify, or list your files 10 different ways. This program also has side benefits. The sort routine is very fast. There is a nice subroutine that buzzes the expansion interface cassette relay when the sort is completed. Mumford's relay selects between cassette 1 and cassette 2 in the interface.

*(Disk Directory is sold by Mumford Micro Systems, Box 435, Summerland, CA 93067; \$29.95.)*

Richard K. Riley  
January 1980

★ ★ ★ ★

**The Disk Doctor  
Superior Graphic Software  
Color Computer**

The Disk Doctor is a Basic program that assists in the reconstruction of a damaged Color Computer disk file, including machine-language programs.

You don't necessarily have to have a problem with one of your disks to use the Disk Doctor: It is also useful for good disks. If you locate the beginning of a binary file and read the first sector, you are given the start, end, and execute addresses. I recommend using this program on good binary files so you can find and store this information should you ever need to reconstruct.

Disk Doctor's written documentation is minimal, but the eight pages are packed with information on reconstructing a crashed disk and preventative care.

As a bonus, the author has included a crashed disk with your program disk. If you successfully reconstruct the crashed disk, you get a back-up of the Disk Doctor and a Packer program. Packer, a binary routine, removes excess spaces from any Basic program to make the program run faster. On the crashed disk, data and graphics files are included to demonstrate how they can be reconstructed. It takes very little time to learn the reconstruction process, if you do the assigned homework. The homework is an overview of the disk allocation information in the Color Computer's disk operating manual.

The Disk Doctor teaches you disk anatomy as well as preventative medical information, allowing your disk files to lead long and fulfilling lives. I feel safer when creating important files and programs having the Doctor "on call."

*(Supertor Graphic Software, 406 Little Mountain Road, Waynesville, NC 28786; \$49.95.)*

John Steiner  
December 1982

★ ★ ★

**Disk Drive Timer (DDT)  
Disco-Tech Microcomputer  
Products  
Model I, II, and III**

The Disk Drive Timer allows you to check if the disk drives are running at correct speed. It

is really a graphics-display tachometer. This can be important since one may not be able to format or back up disks if the motor speed is out of range. Incorrect motor speed can also cause data to be lost when loading or saving programs and files.

When DDT is executed, it automatically loads Basic and runs. The menu selection consists of three basic options. The first option analyzes all the disk drives in your system that are ready. It displays on the video screen the motor speed in rpm and the percentage error from the required 300 rpm speed.

The second option is essentially the same as the first, except that you specify the drive to be analyzed.

The last option is the feature that makes DDT so useful as a diagnostic and maintenance-software tool. Using this option, a linear-horizontal scale is displayed near the bottom of the screen with a graphics block indicating the motor speed of the specified disk drive, which is updated approximately once a second. Unfortunately, Disco-Tech doesn't include instructions on how to take your TRS-80 disk drive apart and adjust the speed if it is too low or high.

Each disk contains three copies of the program so that if one copy of the program somehow becomes damaged you have a replacement. If you want to save some money as well as time, the DDT is a worthwhile investment.

*(Disco-Tech Microcomputer Products, P.O. Box 1659, Santa Rosa, CA 95402; \$29.95 on disk.)*

Howard M. Berlin  
April 1981

### ★ ★ ★ Disk\*Mod Misosys Model I

When I acquired a disk system, this utility helped make my Radio Shack cassette Editor/Assembler more useful.

Once I loaded the tape, using the program name DSKMOD, I saved the program to disk using TAPEDISK. The program runs from disk much more smoothly than from tape and permits easy input and output of data (assembly text or object code) to either tape or disk. Disk\*Mod prompts when filespecs are needed. Anyone familiar with EDTASM will find it easy to use.

Disk\*Mod also displays a disk directory showing the memory usage of each disk file and tells how much memory the current program is using, as well as how much text buffer is left. When you exit the program, you can go to DOS automatically or specify a destination.

Most cassette users will find the sequence used for debugging with Disk\*Mod quite different from what they're used to, but it is very effective and almost painless.

I found only one problem with the adapted program: it doesn't handle some assembly text errors well. When I had a statement that wasn't in correct format, that line and the next several lines printed erratically and illegibly, but the error wasn't pointed out. If the print during assembly starts to look strange, look for errors and you can fix the program and the print at the same time.

*(Disk\*Mod is sold by Misosys, 5904 Edgehill Road, Alexandria,*



VA 22303; *This product is no longer available.*)

Buzz Gorsky  
December 1980

★ ★ ★ 1/2

**DISKNAME/BAS**

★ ★ ★ 1/2

**FLOPYCAT/BAS**  
**Marvin Plunkett**  
**Model I or III**

This disk library program from Marvin Plunkett has been pegged with the rather inelegant moniker, FLOPYCAT/BAS, but the inelegance stops there. Everything else about this program is indeed elegant.

First, it is fast. Using machine-language subroutines, relocatable for sorting and other high-level tasks, this Basic program works as fast as any high-quality word processor. Second, FLOPYCAT/BAS is easy to use. The screen presentation is first rate—all prompts are well laid out and easy on the eye.

Most importantly, FLOPYCAT/BAS is versatile. It requires a minimum of 32K and it lets you store 429 disk file names. With 48K, you store 1,084 file names. You can use manual input or let the program read the files automatically. Search routines help you find a specific file, and you can sort your disk library by disk name or file name.

The program has extensive error-trapping abilities. I enjoyed being warned, as I was trying to exit to Basic, that the file had been changed but not written to disk.

I give FLOPYCAT/BAS the

edge over other disk directories on the market because of the author's attention to detail, the overall polish of the screen displays, and the user prompts. The documentation exhibits these same qualities and never bogs you down with unnecessary details.

For \$10 more, you can purchase a companion program, equally elegant, called DISKNAME/BAS, which lets you change the names of your disks. Using these programs, your disks will be much easier to keep track of, and they will be instantly organized.

*(Marvin Plunkett, a Microcomputer System Consultant, Roseburg, OR 97470, markets these programs. FLOPYCAT/BAS sells for \$30, and DISKNAME/BAS sells for \$15. As a package, they cost \$40. The Model III version is DOSPLUS-compatible only.)*

G. Michael Vose  
May 1981

★ ★ ★ 1/2

**DSMBLR**  
**Misosys**  
**Model I or III**

The best thing about DSMBLR is its wide variety of commands and flexibility. Even the tape version I reviewed came with three programs to fit different memory sizes.

The commands are broken into three groups: Control, Output, and Special. The Control commands include B (Basic), which returns you to the Basic Ready prompt; C (Clear), which completely clears the symbol table of the disassembler; E (Equate), which toggles the equate statements on and off—an

unusual feature in a disassembler; S (System), which loads a machine-language tape, finds the start and end addresses, and readies the program for disassembly; and T (Test), which merely loads the start and end addresses into memory.

The Output commands are even more thorough, all outputting memory address, contents of memory, line number, symbolic table, disassembled instruction, and character output to the screen tape, or a printer. The printer output titles and formats pages very nicely! For special commands, Clear is the logical interrupt for any prompt; Break interrupts command request entries; and Shift @, as in Basic, stops a continuous scroll (listing).

The only shortcoming I see in this program is that the data elements and ASCII strings are transformed into Z80 instructions. Data will be interpreted as something else, making modification necessary. This is not serious and will not present a problem to most programmers.

The program and documentation were made specifically for those with a knowledge of machine language. If you have this knowledge, are on a tight budget, and have not been able to find a good disassembler, DSMBLR is the program for you. It is top quality for the Model I or III tape or disk user, and worth much more than the modest price.

*(Misosys, 5904 Edgehill Drive, Alexandria, VA 22303. DSMBLR, version 2, is available on cassette transferable to disk for \$20.)*

Tim Knight  
September 1982

★ ★ ★ ★

### **Edit Southern Software/Allen Gelder Software Model I and III**

I used to wish that the easy editing capabilities in Scripsit were possible with my Basic programs. Things such as cursor control, scrolling, line functions, deletions and insertions, replicate functions, and global find and replace were so neat to have, but were unavailable on the Model III.

Well, my troubles are over. Edit, published originally by Southern Software in England, makes all those things possible.

The cursor is controlled via the arrow keys, hitting break once enters you to the special command mode, while hitting it again returns control to Basic. The enter and clear keys also have special functions, and @ is the control key. There are also special keys available in the edit mode. Both the keypad and alphabetic keys are very important. Pressing the @ and another number completes functions such as cursor motions, deleting, inserting, and so on. Pressing the @ and a letter performs line commands.

One of the features I have found only in Edit is the replicate function. This makes a new line identical to the one the cursor is placed on. Edit also features block functions and global search and find.

The thorough documentation will teach you how to use this utility quite effectively. I recommend its purchase.

*(Allen Gelder Software, Box 11721 Main Post Office, San*

Francisco, CA 94101; \$40.)

Tim Knight  
June/July 1982

★ ★ 1/2

**ESP-1**  
**Small Systems Software**  
**Model I**

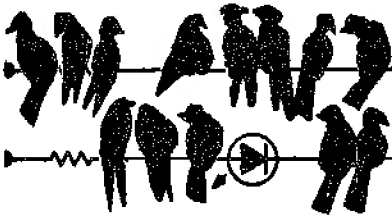
ESP-1 is an old program that has been rewritten for the TRS-80. It is an editor, assembler, monitor, and debugger. Assembly-language source files are generated with ESP-1 using the editor, and each line has a line number just like a Basic program.

The assembler portion of ESP-1 is a full-blooded assembler with all the standard features. The system monitor and debugger has over 20 commands for controlling your TRS-80.

One drawback to ESP-1 is that the assembler recognizes Intel 8080 mnemonics instead of Zilog Z80 mnemonics. This is great for the 8080 programmer moving written programs to the TRS-80, but it restricts the person who wants to learn to use Z80 code.

(Small Systems Software, P.O. Box 366, Newbury Park, CA 91320. ESP-1 is no longer available.)

Rod Hallen  
February 1980



★ ★ ★

**Faster 1.2**  
**Model I**

★ ★ ★

**RPM**  
**Model I and III**

★ ★ ★

**XTEND40**  
**ProSoft**  
**Model I and III**

Faster is advertised as a software speed-up for Basic programs. Its approach is, however, unique. Faster is not a compiler; it is a program analyzer. It monitors the execution of your programs and keeps track of how they access their variables. It points out which lines you should pay attention to, and the variables you should define first.

Faster's disk version comes without an operating system per se; its special disk copies its programs onto one of your disks by itself, saving one-disk-drive owners the hassle of trying to load a non-system disk onto systems disks or buying another copy of TRSDOS.

The manual is short (11 pages) but attractively printed and very well done. The instructions on how to install the programs on your own disks were incorrect, presumably since the self-copying disk is a new idea, but that disk came with its own instructions.

The manual discusses how to activate Faster to monitor Basic program execution, and how to use the output to modify Basic programs so they will run faster, as well as some common problems that may occur (none of them happened to me) and some other suggestions for speeding up Basic. Faster also works with

hybrid Basic/machine-language programs.

Basic creates a variable list that it must search each time a variable is referenced. If your most often-used variables come at the end of the table, Basic must search longer for them. After your program has run long enough for Faster to get an accurate picture of how it operates (you do not have to run it to completion), the analyzer lists variable names, types, and number of times referenced, and the manual shows you how to set up your variable table to maximize speed. Most often, adding one or two lines, such as a DIM statement to allocate variable space, will speed up your program by 20 to 50 percent.

Unless you always write very short programs, Faster offers an inexpensive way to speed up your Basic programs. If you cannot afford a compiler, and would like a significant increase in the run-time speed, then buy Faster.

### **XTEND40**

If you have upgraded your system to 40-track drives and you have many 35-track disks, how do you avoid wasting those five extra tracks and 10 extra granules of space? You can get a 40-track formatted disk and copy all your files over, one at a time—a lengthy process—or you can use XTEND40.

In 15 seconds, XTEND40 formats those unused five tracks, verifies them, and updates the GAT sector. If they are already formatted, you receive an error message and are asked if you wish to do it anyway. This enables you to repeat the procedure if a sector gets locked out during the formatting process.

The manual consists of a single printed page, but the program is so simple to use, nothing more is required. XTEND40 will not query you for another attempt if a track is locked out, but since the whole process takes only 15 seconds, this is a minor inconvenience. If you have more than 20 disks to format for 40 tracks, you can run through them in no time with XTEND40.

### **RPM**

RPM is a useful, well-written disk-drive timing program. It measures (and gives easy instructions on how to adjust) the speed of any drive in your system.

A hybrid Basic/machine-language program, RPM gives not only the current speed and deviation from normal (300 rpm), but shows a continuously updated average and a graphics display of this variation. You may change to another drive simply by pressing the number of that drive. The speed variation graph and the averages are reset automatically, as is the number-of-observations counter.

The manual is a terse four pages, but is complete and describes fairly well how to adjust the speed. The continuous display lets you easily adjust drive speed within very narrow limits, once you find the drive's trimpot or potentiometer (in the lower left corner of my Perlec's main circuit board, not the upper corner as the RPM manual says).

The program is easy to use and provides a good diagnostic tool for your drives. If you have drive problems I recommend try-

ing RPM before paying for repairs.  
*(ProSoft, Dept. C, Box 560, N. Hollywood, CA 91603; Faster, Model I, \$29.95; XTEND-40 is no longer available; RPM, Model I and III, \$24.95.)*

Bruce Powel Douglass  
 April 1982

★ ★ ★

**FED File Editor for LDOS  
 Galactic Software Inc.  
 Model I and III**

FED is a file editor designed to run only under the LDOS operating system. It is a unique concept in zap programs in that it operates on the file level only, rather than the track/sector level. It is a great improvement over the usual zap program.

There are two display modes: One is a standard 256-byte mode with the ASCII display on the left instead of the usual right, and the other is an extended 128-byte mode with the ASCII across the top of the screen, the hex in the center, and plenty of space for other information. The T key toggles between these two display modes. The latter is particularly nice for modifying ASCII files.

The usual file modifications are available plus much more. You can modify in hexadecimal or ASCII, and can find what the character under the cursor is in binary.

There are three different file search commands which include a rarely seen search command for a load address. Other command include printing an entire file to the line printer; sending the current edit buffer to the

printer; and for the machine-language programmer, a command that jumps from load block to load block within the file.

For \$40 you receive a disk-file editor that has capabilities you'll wonder how you got along without. I use mine almost daily and keep a copy on my working disk system so that it is always available to me. If you're an LDOS user and have a need to modify files at the disk-sector level, you should have this program in your library.  
*(Galactic Software Inc., 11520 N. Port Washington, Mequon, WI 53092; \$40 disk.)*

Charles P. Knight  
 December 1982

**FLOPYCAT/BAS  
 (See Page 213)**

★ ★

**FORTRANslator  
 The Management  
 Model I**

If you are not a capable Fortran programmer or at least a very dedicated learner, skip this article. However, if you are a dedicated learner with a good supply of Fortran manuals and a good printer, you might be interested.

The FORTRANslator utility, despite what you might expect, does not really translate Basic to Fortran. In fact, the resulting translation is totally useless to a programmer without a good Fortran vocabulary.

The program consists of one disk with the BATRAN/CMD program and one page of documentation. I always expect problems

with new programs, and I was not disappointed this time. The one-page letter included all the instructions and also contained a list of what the translator did and did not do.

The instructions indicate what kinds of problems might be encountered. However, they give no suggestions as to how to solve them. Of course, if you don't know Fortran already, you have no chance in hell of solving them. There are also problems with the printer output. The resulting translation requires extensive editing and a line printer is a must.

The printout procedure has some flaws. The first part of the printout is a cross reference of Basic calls that you have to alter. It is helpful, but when there is no output, the heading is printed and a whole page is wasted.

The second page is a printout of the arrays used, broken down by GOTO array and subroutine arrays. This is necessary since Fortran does not use subroutines as Basic does, and they must all be moved to the end of the Fortran program. Again, when there is a short list or no list, the page is headed and wasted.

The third part of the printout is the Fortran program. It is neat and clean and there is plenty of room in which to write editing notes. There are no problems with the printout, but the content is another story.

I've been picky so far, but it gets worse. A DO loop is the Fortran version of a Basic For . . . Next loop and is used in the same way. But, the translator's DO loop is not correct. It must be edited to correct format.

All the translator does is change the word "FOR" to "DO"

and the word "next" to "continue." If the original For . . . Next loop has a subroutine branch nested in the loop, it will be lost and the logical flow of the program must be adjusted or the Fortran program will not run correctly.

Another important consideration is compatibility with the Fortran compiler. The translator produces text in Fortran form. It inserts C lines (REM in Basic), and it prints in an easy-to-read form with room for manual editing on the page. The disk output is the same text that is on the printer—it will not work in the TRS-80 (Microsoft) Fortran compiler.

There are many more problems. Text lines are cut off and write lines become doubled with text format lines. Even if you could get a working program out of the translator, you should consider one more warning. The translator does not approach efficient use of Fortran. Microsoft Fortran is about 60 times faster than Disk Basic. The benefit of faster Fortran is lost if you do not use it efficiently.

The translator does have some value. It aids in reducing typing time, and it provides a Fortran format for the Basic program that is a skeleton starting point. It aids in following the logic of a Basic program that you are rewriting.

One final word—I have only pointed out the major problems that occurred in the translation of short Basic programs. The problems are extensive and would be insurmountable to a programmer without Fortran knowledge. I shudder to think what problems might occur in the translation of an extensive

Basic program. But, if you write the original Basic program carefully, you might eliminate some of the translation problems.

*(The Management, P.O. Box T, Aledo, TX 76008, could not be reached for an update.)*

Richard McGarvey  
May 1981

★ ★ ★ ★  
**Generalized Subroutine Facility**  
**Racet Computes**  
**Model I, II, and III**

Generalized Subroutine Facility (GSF) provides utility subroutines that can be called from Basic. Loading is fair and the documentation is excellent. The subroutines are loaded into protected high memory and are available to the calling programs at all times. Each routine is numbered and is called with the Level II USR statement. The manual carefully explains each subroutine with a sample listing of a Basic program using that routine. All the example Basic programs are also recorded on the GSF tape with a menu to provide selection. I wish that all software gave the buyer this kind of demonstration.

I only found two minor areas of complaint: The tape label says that protected memory should start at 29950, and the manual says 30000. It works great at 30000. Also, no error-checking is done as far as addresses are concerned.

GSF comes in 16K, 32K, and 48K versions. If you buy the 16K version and later increase the size of your system, you can get a 32K or 48K version for \$5. I personally feel that all three ver-

sions should be included on the original tape. In any case, I definitely recommend GSF as a good buy for anyone who is seriously writing his own programs.

*(GSF is sold by Racet Computes, 702 Palmdale, Orange, CA 92665. Model I, \$25; Model III, \$30; Model II, \$50.)*

Rod Hallen  
January 1980

★ ★ ★  
**Infinite Basic**

★ ★ ★  
**Remodel and Proload**  
**Racet Computes**  
**Model I**

Remodel and Proload, available in three versions (16K, 32K, and 48K), load into the top of memory with the System command and use approximately 3K. Also included are two well-written manuals. The program comes on a high-quality cassette, loads without hitch, and can be easily transferred to disk or stringy floppy.

Remodel (REnumber MOVE DELETE) gives complete freedom in renumbering any portion of a Basic program (or the entire program) and in moving or deleting sections of Basic code (either a line or a complete subroutine) at will. About the only restriction on this freedom is that Remodel and Proload will not execute a command to overwrite an existing line.

The command format is simple and straightforward, and it returns to the command input state after executing each command. Exit and reentry are also uncomplicated.

Remodel and Proload encourage structured programming. The program is divided into two major parts: a control section consisting of a loop that calls in sequence the relevant subroutines via GOSUB statements, and a list of subroutines that make up the body of the program. Proload creates a library of useful general and special purpose routines that can then be appended or merged into your application program as needed. Remodel and Proload have increased my productivity at least threefold.

Infinite Basic comes in two versions (cassette and disk) on a high-quality cassette, and includes two well-written, complete manuals. Infinite Basic has a system module as well as several application modules.

With so many functions available not all of them are loaded into RAM at once. You must first select the functions for a particular application program, and, with the systems module, create a machine-language load module containing these desired functions. This load module can reside at almost any memory location and is saved separately on cassette, disk, or wafer. When reloaded, the load module provides the desired functions.

The Infinite Basic functions run considerably faster and are more compact than similar Level II Basic subroutines. Moreover, the machine language load modules can be stored in Basic statements. Thus, the modified load module can be selectively loaded and saved by Proload as well as becoming a permanent part of the user's application program.

The Matrix I/O group is extremely efficient because it operates on entire blocks or arrays. This speeds up data input and output at least twofold. The Infinite Basic Matrix package also contains a generalized subroutine call and return function.

The Infinite Basic string package uses a number of functions to compress and decompress strings and data from 8 bits per character to 4, 5, 6, or 7 bits, and pack and unpack character strings by special encoding of repeated characters. Appropriate use of these functions lets you store more data in memory than you could otherwise. String translation functions, as well as others, are supported. Finally, the sorting functions of Infinite Basic are extremely powerful, flexible, and fast (30 times faster than the fastest sorts in Basic).

In a nutshell, Infinite Basic confers upon your Level II ROMs a considerable degree of extensibility.  
*(Racet Computes, 702 Palm-dale, Orange, CA 92665; Remodel and Proload, Model I and III \$35; Infinite Basic, Model I \$50, Model III \$60.)*

Maurice M. Small  
September 1981

★ ★ ★ ★

**KEEPIT 2.0**

**The Alternate Source (TAS)  
Model I**

For those unable to justify the added expense of moving up to disk operation, there is now available a very underrated program that could make serious



applications on the Model I a whole lot easier.

KEEPIT is a utility program that packs a lot of power into less than 1K of machine code. One of its features is the inclusion of the KBEEPPIX routine which provides keyboard debounce, automatic character repeat (after a short delay) when a key is held depressed, and an audible beep at the cassette output port each time a keystroke is entered.

The following feature is the real workhorse of the program because it allows Basic programs to be saved in the middle of a run with all its variables intact. Here's how it works: At whatever point you want to save your program, you press Break. Set up the cassette to record the program, then type the command: SAVE/RUN "PROGRM".

What happens is: The Basic program is reloaded, along with all variables, systems pointers, and the KEEPIT program itself; even the video display is restored just as it was. Only free space (memory not used during execution of the program) will not be affected. You simply continue your Basic program right where you left off!

Such a feature can be used to debug new programs by saving a program at various points throughout a run, thus allowing you to go back and reconstruct what was happening in the logic flow of the program just before the crash.

You may be wondering if you can execute the SAVE/RUN command from within the program. It is possible, but it will be the last statement executed in the program, and you must use caution. Another feature of KEEPIT

is the machine code monitor. Typing: \*OPEN "NNNN" where NNNN is replaced by an address in hexadecimal, displays 16 bytes of memory beginning with the specified address. The bytes are displayed in hex with their associated ASCII characters displayed on the next line.

The final command, \*NEW, restores a Basic program that has been wiped out accidentally by typing NEW.

KEEPIT is supplied on cassette with both the SYSTEM (object) program and the editor-assembler format source code. It is compatible with other special-command routines, such as the Exatron Stringy-Floppy routines. If you don't have a disk system, KEEPIT may prove to be one of the most useful utilities you own.

*(The Alternate Source [TAS], 704 N. Pennsylvania Ave., Lansing, MI 48906; \$9.95.)*

Jack Decker  
December 1980

★ ★ ★

### **Level I in Level II RAM Apparat Inc. Model I**

This program makes your Level II Basic system act as though it is a Level I Basic machine. It is written in machine language and occupies about 4.3K bytes of memory space at the upper end of your Level II 16K-byte memory bank, leaving almost 12K bytes of memory for use by any Level I program you want to run.

Level I in Level II RAM comes without any supporting documentation. However, all you need to know is written in seven

words and symbols on a label attached to the cassette's shipping box.

After you load the program, your system will operate in Level I Basic, so you don't have access to the convenient editing functions that Level II Basic provides. The program lets you load successive Level I Basic programs into your computer without having to be reloaded.

I have uncovered two minor problems that I attribute to my having loaded the program with my recorder's volume set too low. One of my programs contained a line that used AS and BS separated by a colon. The BS string was continually shortened even though it consisted of less than the allowed 16 characters. I moved BS to a separate program line and had no more problems.

Another problem involved my use of the ON N GOTO statement. Since the problem didn't appear anywhere else in that

program or other programs, I assume the problem was self-induced.

I have heard that Level I in Level II RAM will not accept programs written in machine language, but I was unable to test for this.

I'm pleased to have the capability that Apparat's program provides. The price was right, and the program's operation has been better than satisfactory.

*(Apparat Inc., 4401 S. Tamarac Parkway, Denver, CO 80237. Level I in Level II is no longer available.)*

Sherman R. Wantz  
March 1980

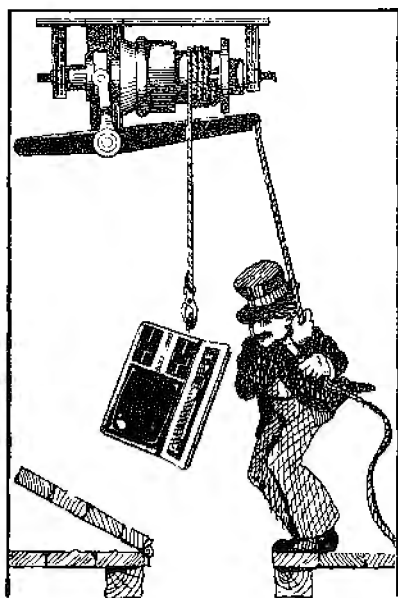
★★★★

### **Line Renumbering Software Associates Model I**

Line Renumbering renumbers Basic program lines and allows the loading of more than one program at a time (merging). It loads fairly well and the documentation is excellent. The tape contains 16K, 32K, and 48K versions, which will all work with Level II or disk. This is a machine-language program that resides in high memory. It can be called from Basic at any time by using the System command. This is the type of applications software that the personal computer owner needs. It is a shame that this was left out of Level II Basic.

*(Line Renumbering, once sold by Software Associates, P.O. Box 2248, Springfield, VA 22152, is no longer available.)*

Rod Hallen  
February 1980



★ ★ ★

### Master Control Soft Sector Marketing Color Computer

Master Control is a tape cassette program for redefining Color Computer key functions. The cassette is accompanied by an instruction leaflet that is brief but sufficient. The purchaser attaches an overlay to the computer's keyboard surface.

Master Control is written in machine language and is relocatable. You invoke it with the Exec command. To use any of the new key definitions, press the down arrow, which serves as the special shift key; control is not toggled.

Some of the functions of the redefined keys require Extended Color Basic. You can combine them with individual letters to cut down the number of key-strokes for other commands. A few keys do multiple jobs, making use of the regular shift key.

Space constraints make some of the notations on the keyboard overlay rather cryptic. This ceases to be a problem after a short time.

Master Control operation is not as cumbersome as it sounds. It takes a little while to get used to the new command positions, but in return you save considerable time and effort in program entry. You can avoid an awful lot of repetitive typing.

I would make minor changes, though. Master Control inconsistently includes opening parentheses with some commands. Similarly, it does not seem especially worthwhile to have included Print in Master Control's repertoire when the Color Basics already have a two-stroke ab-

breivation for it.

(Soft Sector Marketing, P.O. Box 340, Garden City, MI 48135; Version 2.0 \$19.95 cassette, \$24.95 disk.)

Scott L. Norman  
March 1982

★ ★ ★ 1/2

### Master Directory 1.1 Micro-Systems Software Model I and III

MASDIR 1.1, from Micro-Systems Software, helped organize all of my disks and files. Now finding a particular disk is hassle-free.

This program requires one or more disk drives. When loaded, the data query flashes on the screen, and the date entered is appended to all directory files entered in this session. This date is also used in the heading of the printout, if requested.

MASDIR tallies the number of disks and the total number of files entered in the upper-right corner of the screen. Once entered, the directory entries can be recalled to the screen or printer in one of the following formats: an individual disk directory, by assigned disk name or number, a master listing of all disks in numerical order, or alphabetical order, and a listing of all files with a common extension.

Although at first it seems that these folks thought of everything, the program does have a few faults. There is no provision for anything other than the standard Radio Shack parallel-port printer. Also, you cannot modify the internal print format in case you want other

than 66 lines per page and some directory files written by NEWDOS 80 can't be read or listed by MASDIR. In addition, MASDIR won't read directories formatted for CP/M.

All things considered, this program is one of the most useful in my entire library. At every computer session, the first and last program I use is MASDIR 1.1. First, I use it to find the file that I want to work on, and when I am finished, I delete and reenter the disks that I have updated during the session.

*(Micro-Systems Software, 4301-18 Oak Circle, Boca Raton, FL 33431. Master Directory 1.1 is no longer being marketed.)*

James B. Penny  
April 1981

★ ★ ★ ★  
**Maxprint**  
**Peggytronics**  
**Model I or III**

Designed for use with Models I and III, 32K or 48K with one disk, Maxprint is a printer-driver utility consisting of over 3K of Z80 Assembly language. It offers support of all MX-80 print styles, total control from text, partial control from menu, proportional justification, underlines, subscripts, superscripts, horizontal centering, and line spacing (1/72-inch increments).

Two features of the package are very impressive. First, it comes with a manual that contains approximately 40 pages of usable instruction material with most of the special print commands listed on the back cover. Second, the source code for both the Scripsit conversion as well as

Maxprint itself is printed in the instruction manual.

Although Maxprint was created primarily for use with Scripsit, it can be used with other word-processing programs; you need only advise Peggytronics prior to purchase what your system is.

Maxprint is a powerful, easy-to-use utility. My biggest problem came when trying to understand how to combine it with my DOS; the manual could have been more helpful if it had included a step-by-step procedure.

The only thing that disappointed me involved the hardware. Certain cables (mine) make it necessary to break into one of the pins and add a SPST switch to allow Maxprint to call for its own line feeds.

However, if you own an Epson MX-80, use it for word processing, and want to get everything out of the printer that it was designed for, look into Maxprint. *(Peggytronics, 381 First St., Suite 5147, Los Altos, CA 94022; \$27.95.)*

E. M. Collins  
October 1982

★ ★ ★ 1/2  
**Newtalk**  
**Star-Kits**  
**Color Computer**

Newtalk is an ingenious addition to the toolkit of anyone doing machine- or Assembly-language programming on a 6800 or 6809 system, including the Color Computer. Peter Stark's Newtalk is a memory-examination utility with audio readout. It is not a full-featured monitor but a helpful assistant

for such chores as comparing the actual contents of memory with printed listings.

Newtalk is written in relocatable machine code, and the brief manual contains complete details for putting the program anywhere in memory. It goes much further, in fact, and includes a complete Assembly listing—a commendable gesture at a time when some vendors are reluctant to supply data needed for making legitimate back-ups.

Newtalk requires just over 6,500 bytes. The speech samples are not synthesized from scratch, but are derived from actual voice recordings by sampling and storing the zero-crossings of the audio waveforms. The resulting intelligibility is quite good, although a price is paid in the amount of memory required. Newtalk can use this technique because of its limited vocabulary.

The male voice is quite intelligible, although some characters are rather buzzy; the fact that you know the limits of its vocabulary undoubtedly helps. The quality is considerably higher than that of Spectral Associates' Compuvoice synthesizer, for example, although not as good as that from the Votrax chip used in Type 'N' Talk and some other hardware synthesizers. By changing the contents of a particular delay counter location, you can alter the pitch of Newtalk's voice.

Newtalk is something of a novelty and should not replace a real monitor. By itself, it doesn't have a facility for altering the contents of a memory location. I have found it useful, however, to

help follow machine code entered from the keyboard.

Newtalk has proven to be of real value in reading out code to me while I follow the listing. It is much faster and more pleasant than visually scanning from printed page to video screen. If you think this sounds like a convenient way to work, and if you can spare the 6,500 bytes, then pick up a copy.

(*Star-Kits, P.O. Box 209, Mt. Kisco, NY 10549; \$20.*)

Scott L. Norman  
October 1982

★ ★ ★

#### **OBJREL**

**Hubert S. Howe**  
**Model I**

OBJREL is an object-code relocater that moves a block of object code from one memory location to another. It changes memory references within a program as long as they are easily discernible as addresses and fall within the boundaries of the program being moved. Tables of addresses are not changed. OBJREL is a useful program for the Assembly-language programmer.

(*Hubert S. Howe, 14 Lexington Road, New City, NY 10956. OBJREL is now part of MON-3, 4, and 5.*)

Rod Hallen  
February 1980

★ ★ ★ 1/2

#### **Packer 1.5**

**Cottage Software**  
**Model I and III**

Packer 1.5 is a machine-

## SOFTWARE

---

language program that enhances Model I or Model III Basic with or without disks. It is especially helpful to non-disk owners since it provides some commands not found in other nondisk systems.

This utility's commands include: Short, which packs the resident Basic program by removing unnecessary words, spaces, and remark statements; Unpack, which breaks the program into single-statement lines, places spaces between all data statements, and also between almost every keyword in Basic (resulting in a very neat-looking program, though somewhat less efficient, and consuming a great deal more memory); Pack, which condenses the program as much as possible, making it highly memory-efficient; Renum, which renumbers your program; and Move, which moves any block of program lines to any place.

Anyone frustrated with the limitations of Basic should consider Packer for his software library. It is easy to use, well documented, and for the money, the best utility I have bought for my TRS-80.

*(Cottage Software, 614 N. Harding, Wichita, KS 67208; Model I and III, two cassettes \$29.95.)*

Tim Knight  
June/July 1982

commands to Basic. Pressing shift and break brings up the PRO prompt and allows the user to delete, move a block, pack a program into less space, or append from tape.

The renumbering routine lets the user pick where the renumbering should start, what the line should be, what the increment should be, and at what old line number you should stop. It does these well. The rest of the utility is, unfortunately, flawed.

Pack is supposed to remove all spaces not in a string, delete all remarks, and if a reference is made to a deleted line, update the reference. The problem occurs when two or more lines are in sequence. Only the first is deleted.

Move inserts a block of Basic text, designated by starting and ending line numbers, into another location, also designated by line numbers. It deletes the moved text and renumbers it in its new location. It will not renumber the program to make room for the lines to be inserted. According to the documentation, if there isn't room, an error message is generated.

I hope Rational can repair the shortcomings of this package, as it promises to be extremely useful.

*(Rational Software, Pasadena, CA, could not be reached for an update.)*

★★ 1/2  
**Programmer**  
**Rational Software**  
**Model I**

Dennis Thurlow  
January 1981

Programmer is a machine-language utility that fits into the top 1.4K of memory and adds

**QEDIT**  
(See Page 206)

★ ★ ★ 1/2

**GSD Utility Disk #1—LDOS  
5.0.2****Quality Software  
Distributors  
Model I**

The utilities in this collection range from the trivial to the how-did-I-get-along-without-it. Many were written by some of the heavies in the home computing field: Kim Watt, Earle Robinson, and Tim Marr.

DIRLOWER/FIX gives the Model I upper- and lowercase instead of all caps. Two additional patches, STATUS/FIX and KSMSTAT/FIX, modify the device command to show what special functions, such as double density and the keystroke multiply filter, are active; a fourth utility, the stand-alone STAT/CMD, does about the same thing but includes a report on what disk drives are ready. (There is a mistake in the documentation; the proper password for the /FIX programs is GSLTD, not .RRW3.)

RESCUE/CMD permits resurrecting killed files that have not been overwritten, giving a nifty little directory of the killed files for Y/N selection for saving and indicating whether a file is partially overwritten. Unlike most of the programs on this disk, it works only in single density. For would-be speed typists there is DVORAK/CMD to remap the TRS-80 as a Dvorak keyboard (see *80 Micro*, December 1980).

Things go uphill rapidly from here. BINHEX/CMD puts binary data into hexadecimal format and back again. CHANGE/CMD is a drive-routing program. CLONE/CMD allows, as its name implies, copying files. You can

make single-drive copies on drives other than drive 0. Some disk swapping is required, but not as much as with the XFER command.

If you need to justify the cost of utility disks, DCAL/CMD shows exact and smoothed speeds for calibrating disk-drive operation. Two filter programs could be useful for printing: UP-CASE/FLT converts all lower-case output (which some printers cannot handle) to upper-case, and ADDLF/FLT adds a line feed after a carriage return, again something that a few printers look for.

VDISK/CMD verifies the readability of a disk's sectors on a given drive, providing a quick check of disk quality and some information useful for fixing things up with LZAP/CMD.

If your disk arrives with everything glitched except LZAP, LPURGE, and LCOPY, you got more than your money's worth. Kim Watt's LCOPY and LPURGE work in single density; Frank Luff's LZAP works equally well in single or double.

LCOPY's main virtue is that it solves LDOS' slowness problem when doing multiple file copies. It is versatile, allowing prompted selection for copying system files, visible files, or all files; setting step speed on the destination disk; prompting for disk mounts; and offering a help file if you get puzzled.

For cleaning up a disk, use LPURGE. It has a super advantage—everything is done in memory until a specific command is issued to rewrite the directory to reflect the purge. Until then, you can change your mind and all is forgiven (and saved).

LZAP does for LDOS much of what Superzap does for NEWDOS80, plus a few things that Superzap should do but does not. It allows direct entry to debug and return to LZAP. It finds and goes straight to the directory, no matter what cylinder (track) it is on. It finds the location of a file from its directory line, generates hash codes for files, removes passwords, searches an entire sector for a hex byte, and, like Superzap, allows corrections to be made, byte by byte. If its error-trapping mechanism is shut off (which also cancels some of the more useful automatic functions such as searches), LZAP can read some otherwise unreadable disks, including ones with zapped boot tracks.

*(Quality Software Distributors is now named Powersoft, 11500 Stemmons Expressway, Suite 125, Dallas, TX 75229. An updated version of The Toolbox for the Model I and III sells for \$69.95. A hard-disk version is also available. The Master Mechanic, comprising the Toolbox's most used utilities, is sold for \$39.95.)*

George Bond  
April 1982

★ ★ ★

### **Quill Driver: A Text-Formatting Program The Alternate Source Model I and III**

Quill Driver is not a word processor; it is a text formatter. This information and its ramifications are not explicitly laid out in its documentation, and this can lead you to believe Quill Driver is

actually a word processor.

I spent two frustrating hours reading through the manual, waiting to find some reference to input. When I reached the end, it occurred to me that the manual would make much more sense read back to front. The last chapter contains the operating instructions for the program.

The program, which is written in Basic, performs all the standard word-processing functions when printing, is amazingly versatile, and works perfectly with files prepared in the methods described in Appendices A and B of the manual.

It had some trouble keeping track of the .AL counters in EDTASM-prepared text. Normally, it lets you switch back and forth between two files of text opened simultaneously. This is great for typing form letters while adding a different heading or personalized comments on each.

Once you figure out how to use Quill Driver, you will be delighted at how well it works. But please, read the documentation backwards.  
*(The Alternate Source, 704 N. Pennsylvania Ave., Lansing, MI 48906; \$39.95 disk.)*

Dennis Thurlow  
May 1981

★ ★ ★

### **Remodel-Proload Racet Computes Model I or III**

Remodel-Proload can do several tasks for you. Besides renumbering program lines selectively, it searches the program and renumbers all line



references. It can also move sections of your program.

One disadvantage of Remodel is the space it puts before and after a changed line number. This is bothersome if you want a tightly-packed program for speedy loading and execution. To correct this, go through the program with the "nD" editing command and eliminate the extra spaces.

Proload lets you merge a subroutine with your main program. Dump the subroutine onto tape, load in your main program, and indicate where the new material must be read into your program. Tell the Proload section what you want and load the subroutine tape. If the space indicated is clear, the two programs will merge.

The instructions for Remodel don't explain that when you hit clear after loading Remodel, you must type the command letter by letter, not with the clear key. Using the key gives an OM error when loading the program tape. Otherwise, the manual is concise and well-written.

*(Racet Computes, 702 Palm-dale, Orange, CA 92655; \$35.)*

Charles Leedham  
March 1980

### **Remodel/Proload** (See Page 219)

### ★ ★ ★ 1/2 **Renumber** **Tandy/Radio Shack** **Model I**

Renumber is a machine-language tape, available for 4K, 16K, 32K, and 48K memories. It

is a simple aid for programmers who need to make changes in their line numbers.

The program asks you what line number you want to start with, what the new number of that line should be, and what interval you want for the remainder of the renumbered program. Renumber makes the changes quickly and also renumbers every reference to the old line numbers.

*(Tandy/Radio Shack, Fort Worth, TX 76102; \$9.95.)*

Charles Leedham  
March 1980

### **RPM** (See Page 215)

### ★ ★ ★ **Search and Sort** **Micro Pro Systems** **Model I and III**

Search and Sort is a set of machine-language modules that can easily be called from a Basic program. They perform several related tasks essential in any program handling data files.

This utility lets you search through sorted lists of strings for a specific string. If the target string is found, the program returns a value corresponding to the string's position. If it is not found, you have the option of inserting it in its proper place. You can also sort a string or numeric array in ascending or descending order.

The documentation briefly describes how routines are called from Basic through theUSR function and then refers you to the sample Basic program that is provided on tape or disk.

I do have a few reservations about Search and Sort's sample program. Since there is no listing of the program supplied, if you don't have a printer, you must decipher it from the screen. This can be a tedious chore because the code is not formatted to be easily read. Also, I don't feel it should be used as a substitute for amply-written documentation. It would have been helpful to have included a simple application or two.

Search and Sort is an excellent utility package which can be used very profitably by both advanced and, with a bit of struggling, beginning programmers. I recommend it.

*(Micro Pro Systems, Route 2, Box 533, Cumming, GA 30130, \$11.95 cassette, \$13.95 disk.)*

Joel Benjamin  
August 1982

★ ★ ★ 1/2

### **Snapp III Extended Built-in Functions**

★ ★ ★ ★

### **Snapp IV Auto Map Snappware Inc. Model II or III**

Snappware Inc.'s Extended Basic Enhancements solved some of the more frustrating problems with the Model II Basic Interpreter. But not all of them. Slow sorting routines sometimes took 30 minutes to process all the arrays in my programs. Enter Snapp III Extended built-in functions. Among others, they include an incredible sort routine that uses no disk or high memory space. (When installed in a Model III, it requires a few

hundred bytes of memory and disk space.)

The sorting speed is impressive. An array that Basic fills in 52 seconds, Snapp III sorts in 36 seconds.

This utility's functions include the following: XTIMS calculates the difference between two time strings; ETIMSS calculates the difference between two time strings; DATES requires a number of string functions and gives you the date. None of the Snapp functions require a DEF FN statement.

VIDEOS allows you to read information directly from the video display with the syntax FN VIDEOS (row, column, number of characters to be read).

I use the IDS function to be sure that our software users insert the proper disks into each drive. For example, PRINT FN IDS(2) displays the name of the disk in drive 2.

The FN Files command returns the number of buffers allocated for disk I/O. Snapp III also provides expanded error messages. If you entered Basic with only two files reserved for disk I/O and then attempted to open a file to buffer these, you would get the following error statement: Bad File Number (52) ?BN Error. Normally Model II Basic only displays ?BN Error. SCMD "LMSGOFF" returns the error messages to their normal state, while SCMD "LMSGON" displays error messages in Snapp's long form.

Snapp's FN MAX returns the largest value from your supplied list, and FN MIN returns the smallest value from your list. It converts all supplied values to double precision; it selects the largest or smallest, then it con-

verts to the numeric type that you request.

The functions `FNUCS` and `FNLC$` take a given string expression and convert each byte to upper- or lowercase.

`FN FMT$` arranges data into a string variable in the same manner as Basic's `Print Using`. If you enter `PRINT FN FMT$("GOOD-BYE----";10)`, the screen displays the following: `GOOD-BYE 10`. Any programmer who manipulates numbers and mixes them with strings for printing or storage will soon find himself saving a tremendous amount of computer time and memory space by using `FMT$`.

For those of you who can't stand programming without `PEEKing` or `POKEing` to the innards of `TRSDOS` or Basic, Snapp provides `FN PEEK (n)` and `FN POKE (n)`.

`OPEN "E"` allows you to open a sequential file and add to it. `SCMD "RUN"` lets you run a new program from Basic and allocate a different number of buffers for data going to and from the disk. `SCMD "CLEAR"` sets file buffer space, string space, and/or memory size, all while running a program. Another command, `SCMD "ERASE"`, can be used if you need more memory space while your program is running; it eliminates all arrays. `SCMD "ROW"` allows you to protect a portion of the screen from scrolling.

With `SCMD "VDOFF"` you can completely turn off the video display, which spares you from seeing the flickering and streaking during extensive updates. To turn the video back on, type `SCMD "VDON"`. `SCMD "CURSOR"` lets you change the blink

rate and the size of the cursor, or remove it altogether.

I thought that I had seen it all from Snapp until I received their Extended Basic mapping Support, Snapp IV. Consider writing a program that opens a sequential file and inputs up to 99 items, prints all 99 items on the screen with prompts, inputs changes to any or all of these items (or fields), and rewrites the updated file.

Reading or writing files is easy, but with 99 print statements, 99 prompts, and 99 input statements, I hope that you have a 64K Model II and plenty of time! With `Auto Map` you can do all of the above and more.

A program called `Generate/BAS` develops the screen display which will be used in your program. It also assigns each piece of data to a field with its own attributes. The attributes include the row and column of the display and a caption, which is the same as a prompt in an input statement. (You can display the caption and/or data in reverse video if desired.)

`Field length` defines the maximum number of characters to be displayed or accepted from the keyboard. You can also protect the field, which means that you cannot modify its data from the keyboard.

The next attribute is the variable name. This can be a simple variable, a subscripted variable, or it can contain an expression such as `J%(I + K + 5)`. When the variable is numeric rather than string, `Generate/BAS` asks if you would like to accept only positive numbers, and allows you to specify a limit to the number of decimal places.

When you're creating a screen, Generate/BAS displays a prompt line on the bottom of the video display giving you all the current options. These normally include <A>dding a new field, <I>nserting and <E>diting a field. <N>ext moves the cursor to the next field. <P>revious moves the cursor to the previous field. <D>elete discards a field. <S>ave saves the current screen to a file named earlier.

After you set the field's positions, the arrow keys will move any field to a new location. This makes designing a professional screen display very easy.

The screen and field attributes can be accessed from your program using SCMD "SEND". Any unfilled data areas contain a series of small graphic blocks showing where you can enter data. The cursor does not appear until SCMD "RECEIVE" is encountered, at which point it jumps to the beginning of the first or any selected field.

If you enter the last allowable character in a field, the cursor jumps to the first character in the next field.

The tab key moves the cursor to the next field while Escape is a "back tab." The up arrow acts as an express back tab, moving the cursor to the first character in the previous line, while the down arrow is an express tab to the first field in the next row.

While entering string data, use the arrow keys to position the cursor over a character that needs correcting. There is no need to retype the line.

Enter, F1 or F2 takes you out of the data-entry mode and ends SCMD "RECEIVE". The first time I used Auto Map I was amazed. I designed a screen that

input and printed latitude and longitude in degrees, minutes, and seconds; an identification string; a date and time; and a special fifteen-character serial number, all with no errors. Updating this information from my files is surprisingly easy with Snapp's cursor controls.

Try the software yourself. I'm sure you'll wonder how you ever got along without it. (Snappware Inc., 3719 Mantell, Cincinnati, OH 45236; Snapp III Model II version \$100, Model III version \$75, Snapp IV Auto Map Model II version \$100, Model III version \$75.)

Rick Lederman  
January 1982

★ ★ ★ ★

### **Snapp V Extended File Mapping Support Snappware Inc. Model I, II, and III**

Snappware Inc. uses the new verb SCMD (Snapp Command) to implement their version of the Get and Put statements that eliminate the need for LSET, RSET, MKSS, and CVI.

Rather than use Basic's conversion statements and Field, you need only give the proper SCMD and your variable names that appear in your program. The variable names may even appear in string form; Extended File Mapping Support takes care of the rest.

Snapp V gives you a new data type, the 1-byte integer that can have a value from 0-255. This helps you to pack data that won't exceed the value of 255. (Snappware Inc., 3719 Mantell, Cincinnati, OH 45236; \$29

*Model I and III LDOS, \$59 Model III TRSDOS, \$75 Model II, \$115 Hard Disk.)*

Rick Lederman  
May 1982

★ ★ ★ ★ ★  
**Sole**  
**Misosys**  
**Model I**

Sole is the name of a new software package from Misosys for use with LDOS, the Model I, and an appropriate double-density adapter. Sole eliminates the need for Model I double-density users under LDOS to use single-density disks throughout the entire booting process.

The LDOS ROM bootstrap routine requires a disk be formatted in single density. It doesn't under any circumstances read a double-density disk. The way around this is to format cylinder 0 in single density and the rest of the disk in double density.

The problem is not solved by simply reformatting the track. The code loaded from the boot sector must be capable of reading double density, so a whole new boot routine must be written. In addition, SYSO must be stored in only one extent. Sole takes care of the boot code, and the documentation allows you to take care of the other easily.

Once the disk is formatted in double density, a program called SOLE1 runs on it, allocating the entire boot cylinder to the file BOOT/SYS. This ensures that no other file will occupy any part of cylinder 0 since it will later be reformatted in single density. Then the single-density system

disk is backed up to the double-density disk. A program called SOLE2 is run on the resulting disk. The result is a double-density disk that can be booted on the TRS-80 Model I.

Now, this sounds like a lot of trouble, and it is. But it only needs to be done once. After you create the first double-density booting disk, you can file SOLE1 away and forget it. To back up this disk, format a disk in the usual manner for double density, run SOLE2 on it, then use the back-up program in the normal way. Your disk now has 70K more free space than before and on drive 0 where you need it most.

There is one bug in the Sole program system: You cannot back up a double-density System disk using only a single drive. If you have floppy disk drives you'll have to copy to another disk, and then copy it to the original's backside. This is a problem in the back-up utility rather than in Sole and presents no problem at all to owners of two or more drives. If you have a single-drive, double-density system, you can use LDOS in double density, but you'll have to borrow a drive to make back-ups of these system disks.

The documentation is short—only three pages—but meets the high standards of LDOS documentation. It presents its subject lucidly and discusses the program's technical aspects in an understandable manner.

Roy Soltoff is the author of Sole (he's also the author of LDOS) and he has certainly written a masterpiece. At this writing, Radio Shack's double-density adapter is not available

for testing. LDOS intends to support this adapter with a driver and there is no reason to believe Sole won't work correctly with that driver as well.

(Misosys, 5904 Edgemoor Drive, Alexandria, VA 22303; \$25.)

Charles Knight  
November 1982

★★★★

### **Southeastern Textan Southeastern Software Model I**

The Southeastern Textan provides editing features for writing Basic programs on a cassette-based Model I, and Southeastern Software has a winner. From the well-written manual to the flawless execution of each and every command, this has to be one of the best packages I've seen.

The editor is incredibly powerful. The program gives you 24 cursor-control commands that let you insert or delete characters, words, or lines of Basic, search for any character, replace any character, or load another program.

In addition to text-handling features, there is an abbreviated keyboard. Pressing shift and any letter types a complete Basic command or statement. Complete error messages are displayed on screen and thoroughly explained in the manual, and you can merge programs in the buffer. When you exit the editor, all text is fed to a compiler that returns it to Basic compression codes.

This is a dream utility for the Basic programmer.  
(Southeastern Software, 512 Conway Lane, Birmingham, AL,

35210, is no longer in business.)

Dennis Thurlow  
October 1981

★★★½

### **Trakcess Roxton Baker The Alternate Source Model I**

To quote from the Trakcess manual, "Congratulations! You have just purchased the most powerful TRS-80 disk-access utility yet written." The program's author, Roxton Baker, credits William Barden's *Disk Interfacing Guide for the TRS-80* as an inspiration for this program and recommends it to help you use Trakcess efficiently.

The nonexpert should also have a copy of the data sheet on the 1771 floppy-disk controller chip. You can find a copy in the Expansion Interface service manual or obtain one from Western Digital or National Semiconductor.

Trakcess requires a 48K TRS-80 Model I. The Copy and Duplicate commands require two disk drives. Printer output is available, but a printer is not necessary.

The program is in two parts on the disk—Trakcess, a Basic program, and Trakcess/CMD, a machine-language routine. Everything is menu-driven. Disk-related commands will not work until a drive is selected and activated.

Trakcess lets you do several things with your disk. You can read into a specified block of memory, or write from it. You can take from or put to disk a whole track (about 3,120 bytes) in one operation.

Build is a powerful command. It lets you tailor a track to your own specifications, with no restrictions other than overall length. C will search the current track, build a matching format track in memory, then write it out to a target disk. E is a scrolling editor utility. With it, you can edit memory or fill memory between specified addresses with a specified byte.

Trakcess is a good program; it works. Although it's not a replacement for Superzap or similar programs, you can use it for purposes other than copying. Besides that, it gives you a mini-education on the working of the disk controller and the various disk formats possible. (*The Alternate Source, 704 N. Pennsylvania Ave., Lansing, MI 48910; \$24.95.*)

Ronald Bobo  
October 1981

★ ★ ★

**TSHORT**  
**Web Associates**  
**Model I**

TSHORT is an object file that loads into low memory and occupies 562 bytes. It comes on a cassette with a program for Level II on one side and one for DOS on the other.

TSHORT lets you enter 32 Basic instruction words in a single uppercase keystroke. For example, pressing Q enters System, and pressing F enters LEFT\$. Pressing shift and the space bar puts you in the edit mode.

A Kustom key lets you store up to 64 characters for use in entering the program. You can use any key, except break, in this string. Thereafter, the entire

string can be recalled by typing shift K.

This is handy for a subroutine or data line that you have to call often, a USR call, constructing a graphics block using CHR\$, or entering a complex math statement that is used several times.

Some options are also available. For a fee, Web will alter the program so it can be relocated to any point in high memory you specify.

Complete instructions are given for DOS 2.1, 2.2, NEWDOS, and BASICR, and the package comes with the standard warranty. (*Web Associates, Monrovia, CA 91016, could not be reached for an update.*)

John Adams  
April 1980

★ ★ ★ 1/2

**The Utility Pack**  
**B.T. Enterprises**  
**Model III**

The Utility Pack spooler works with DOSPLUS, NEWDOS+, NEWDOS80 Versions 1 and 2, DBLDOS, and TRSDOS in single and double density, with all different combinations of line lengths, page lengths, and buffer sizes. You can also feed the paper one line at a time from the keyboard by pressing the J and K keys together. Pressing the JL keys executes one top-of-form; JMN aborts an operation and clears the printer buffer.

Older operating systems previously unable to link printer output with video display can now do so by executing in Basic a PRINT CHR\$(255). The spooler performs well with fast and slow printers.

You also get three other programs, Map, Cat, and Erase, on the same disk. They only work in single density. Map displays the disk's name, the date, and the number of free granules remaining, with a detailed graph of the disk usage (showing an X where the granule is used, a period for a free granule, and a hyphen for a locked-out granule). It will not show more than 40 tracks (only the first 40 tracks on an 80-track system). The directory must be at the standard location (track 17).

Cat, short for catalog, alphabetizes your directory before displaying it on the screen. This program has the same limitations as Map, except it reads the entire directory of an 80-track drive.

Erase gives you three different modes for killing files. All three are compatible with NEW-DOS80, only two are compatible with TRSDOS and NEWDOS +, and only one works with LDOS. Mode 1 works in double density.

Included in the package is the source code for each file, for people with knowledge of Assembly language who might want to alter them. All these programs will benefit most computerists, and are well documented with many examples. The Utility Pack requires 48K. The Model I disk supplies both Mod I and Mod III versions. All programs on the Model III disk work in double and not single density. (B.T. Enterprises, 171 Hawkins Road, Centereach, NY 11720. This utility is available for the Model I and III with 48K for \$49.95.)

Sal Navarro  
April 1982

### ★ ★ ★ ★ VARKEEP

#### Circle J. Software Ranch Model I and III

VARKEEP is an incredibly powerful extension of Basic that works with either cassette or disk. With it, it is possible to do many things without affecting data stored in arrays and variables. It also makes it possible to instantly zero or null an array without time and memory consuming For . . . Next loops. Even more amazing, it now makes it possible to do the unthinkable: string space allocation can be changed on the fly, with all stored data preserved without change. This is great help when a program presents a varying need for string space.

Besides all the goodies just described, VARKEEP provides one additional feature that is (by itself) worth the price of admission: true program chaining. Using cassette or disk, programs can be moved in or out of RAM without disturbing stored data. This multiplies the power of the computer by making possible elaborate systems of related short programs that use a huge data base stored in RAM! Every one of the features just mentioned above is useful and powerful. With all it will do, it's amazing that VARKEEP occupies little more than 700 bytes.

Once loaded into memory, VARKEEP becomes an integral part of Basic. Commands are invoked through the Name sector, which is already built into Level II ROM. Of course, this means that with VARKEEP in residence, it is impossible to use BASICR in TRSDOS 2.3 without



first saving the affected Basic program and reentering DOS. I believe most people will find this an acceptable trade-off because VARKEEP will be used more frequently than the renumbering feature of BASICR.

VARKEEP has saved me hours of frustration. The ability to edit program lines without demolishing data has greatly reduced program-development time, and markedly reduced the need for driver programs to load and test data. The ability to chain from program to program has permitted previously impossible program systems.

In my opinion, VARKEEP is essential for any serious TRS-80 programmer. In many applications, it easily doubles the power of any Level II or disk machine. (*The Circle J. Software Ranch, Carrollton, TX 75006; \$29.95 cassette and disk.*)

Tom Andrews  
April 1981

★★★

### **XBE**

#### **Computer Applications Unlimited Model I**

XBE makes editing Basic programs much simpler than any other program of its type. Its capabilities include selective renumbering of a line or block of lines, finding any string of characters, changing the string to any other string, block deleting lines, definable macro keys, paging the program, scanning the program, or jumping to any line directly.

The program comes on tape with a relocating loader to place

it at the top of memory with 16K, 32K, or 48K. It can be saved on tape or disk and occupies 5,120 bytes. You have a blinking cursor with reasonably complete ease of movement and two speeds to move it along a line.

Editing is simple, and includes insertion and deletion methods. Shift X extends the line by placing the cursor at the end of the current line and entering the insert mode. In the insert mode, shifted characters act as macro keys.

A mode called clear command mode is initiated by pressing the clear key. Its many features include deleting and inserting entire lines, returning you to Basic, finding a previously defined string, and marking a statement for a variety of purposes.

You enter the extended clear command mode with clear/space bar. You can change the increment value on lines, renumber the entire program, search for a line number, search for a string, change a character string, or redefine your macro keys. Enter executes the command.

I have used this program extensively since I received it in the mail, and I am still impressed. It is an excellent piece of well-written software that is long overdue.

(*Computer Applications Unlimited, P.O. Box 214, Dept. ABM, Rye, NY 10580; \$29.95.*)

Bruce Powel Douglass  
November 1981

**XTEND40**  
(See Page 215)

★ ★ ★

### Z80ZAP/CMD

**Org-Tex Industries  
Model I**

Z80ZAP/CMD makes lost file recovery, file patches, and file maintenance a joy; it calculates the hash code for any filespec instantly. The Z80ZAP/CMD disk comes with back-up on auto when it is booted up and also contains a write-protect tab.

The documentation is contained in four brief, but comprehensive pages. Most of the commands are either automatic or self-prompting, making Z80ZAP a breeze to use. Written in machine language, the run-time is almost instantaneous in all modes, including power up. Z80ZAP has several useful functions, including the hash index code, sector comparison, and a find command. The documentation includes easy instructions for removing passwords from all files and for recovering a killed file.

*(Z80ZAP/CMD was sold by Org-Tex Industries, Lewisville, TX, which could not be contacted.)*

Bill Vick  
September 1980

---

## WORD PROCESSORS

★ ★ ★

### C.C. Writer 1.0

**Transformation Technologies  
Color Computer**

C.C. Writer is one of the first word-processing programs to be

announced for the Color Computer. It is written in Basic with 23 pages of well-written documentation. It will run on a 16K machine, but since this allows you to write only about two pages of text before saving to cassette, 32K is recommended. With 32K of RAM, you will be able to handle eight to 12 pages before saving to tape.

Because the program is written in Basic, it can't compete with more sophisticated word-processing systems. C.C. Writer lacks almost all the features I depend on in the Model I Electric Pencil and Model II Scripsit.

C.C. Writer is sentence-oriented; you must press enter at the end of a sentence. The program also displays information on the screen at the end of each sentence, such as the number of paragraphs you have used, and the number of characters you have left. This conspires with the small screen size of the TRS-80C to let you see very little of what you have written at one time. It is rather frustrating if you want to scan what you have just written before you go any further.

Once you have entered a sentence, the only way to correct it is to go back to the menu, select one of the edit modes, find the sentence you want to change, and correct the mistake.

Entering text into C.C. Writer may be frustrating, but printing it once it is in the machine is considerably more satisfying. The author has thoughtfully provided all the important functions. You have complete control over margins, line spacing, paragraph indentation, and page length.

Most documents produced on

C.C. Writer will be completely indistinguishable from those printed on, perhaps, a TRS-80 Model II with Scripsit. (*Transformation Technologies, 194 Lockwood Lane, Bloomington, IL 60108. The new C.C. Writer, version 3.1, is available on disk for \$30. An upgraded cassette version [1.6] is also available.*)

Paul Grupp  
March 1982

★ ★ ★ 1/2  
**Chertext**  
**Apparat Inc.**

★ ★ ★ 1/2  
**Hexspell**  
**Hexagon Systems**

★ ★ ★ 1/2  
**Microproof**  
**Cornucopia Software**

★ ★ ★ 1/2  
**Miz'spell**  
**Programs Unlimited**

★ ★ ★ 1/2  
**Proofreader**  
**Aspen Software**  
**Model I and III**

Apparat's long-heralded text checker was the last of the lot to reach the market. Sold on two formatted data disks, it requires a minimum of 48K RAM and a two-drive, single-density or one-drive, double-density system. Chertext is largely interactive, and while the documentation does not explain every step of the program, it is quite adequate.

The Chertext package consists of an Assembly-language program and object file, a 65-gran

ASCII character dictionary, and a patch to Scripsit which permits the user to activate Chertext and process a document simply by entering the special command P,CHX.

During the suspect word review, the user can elect either to ignore a word not recognized by the program, add it to the dictionary, or mark it with a - in the source text file. You can list any portion of the dictionary to the screen, and manually add or delete words from the vocabulary.

Using Scripsit's hyphenation feature before Chertext processing affects the number of suspect words found, as the program ignores any word particle that includes the AD byte put in by Scripsit. During the test, Chertext refused to recognize "abbom-inations" until the pseudo-hyphen had been either removed or replaced by the equivalent 2D ASCII character.

Chertext does not allow you to see the context in which your error has occurred, nor to correct words while processing a document. Instead, you can mark suspect words by changing the last letter of each to a #, then do a global search to locate each bad word so it can be fixed.

Chertext combines the accuracy of a literal dictionary with considerable operating speed, particularly on long documents. Apparat offers all registered owners the opportunity to obtain free of charge whatever size dictionaries will suit their own hardware configurations—up to a 50,000-word version on a dual 80-track, double-density disk.

### **Hexspell**

During the test trial, Bernard

Hughes' Hexspell was the only one of the programs to prove 100 percent effective in detecting true errors. It has the same minimum system requirements as Chertext, but since its vocabulary is twice as large, Hexspell questioned only half as many correct words.

Hexspell is a Compiler Basic program consisting of a Microsoft run-time package, the program files, and a compiled code dictionary (55 grams). The dictionary is divided into rows of words not necessarily in alphabetical order; as words are called up during the text-checking process they are moved to the head of the row, and a new word added to a full list pushes an old one out the back. As a result, commonly used words accumulate at the front of the vocabulary where they are found fastest. There is an almost human quality to Hexspell, for it seems to pick up speed as it becomes more attuned to the user's own vocabulary.

Source text is scrolled up the screen at a fast reading speed, with each suspect word displayed in full context. This gives the user an opportunity to change or add it to the dictionary on the spot. The feature makes Hexspell's operating speed slower than most other checkers', but lets you run your own quality control on the text at the same time Hexspell is working, catching mistakes such as "principle" for "principal," which no proofreading program would recognize.

An enhanced version of Hexspell is planned. With added features such as an expandable dictionary, text scroll speed control, an automatic vocabulary

add function, and a programmable character set, it should be a formidable tool at proof-reading time.

### **Microproof**

Cornucopia Software's Microproof is in a class by itself. It alone will operate on a one-drive, single-density 32K system, and its documentation is at least three times its competitors' size.

The version tested had a main program and dummy file, three dictionaries (56 grams total), and programs that let you print and expand the dictionary. Standard Microproof sends the list of suspect words to the screen and also the printer, if turned on. A text-correction feature is available as an option; if you elect to see the context in which a suspect word occurs, Microproof displays only a portion of the source text, putting three question marks after the word.

If speed were the sole consideration, then Microproof author Phil Manfield would be king. The program was the fastest of the checkers when handling either long or short documents. Where accuracy is concerned, however, Microproof made an extremely low grade on the test trial, detecting only 53 percent of true errors in the sample document. The root word/prefix/suffix coding system that gives Microproof its speed seems to be the principal cause; each of the errors that went undetected was a compound word.

Unlike the other programs tested, Microproof does not offer the option to close files and exit to DOS if and when you desire, nor are you given a convenient means to delete unwanted words from the dictionary. The text-

correction module should be easier to use. Keys selected for the two principal edit functions require an initial shift to uppercase, and have not been IN-KEYed; the initial video review of suspect words is done entirely in lowercase.

Cornucopia Software has been very responsive to user feedback and has already made a number of improvements in their original version of Microproof. A 20,000-word literal dictionary will soon be available which should greatly improve the program's overall accuracy.

### **Miz'spell**

A streamlined package of only five components, Miz'spell comes on two TRSDOS disks and works with only one disk drive, though it requires 48K RAM. Like Hexspell, it is a Compiler Basic program with a Microsoft run-time package. However, it uses a different hashing routine to transform words into 3-byte values that can be stored in the dictionary. When checking a word, it first looks through a basic vocabulary of 1,700 words in memory and then goes to the dictionary file on disk, which in its virgin state is about 16,000 words.

Unhappily, Miz'spell lived up to its name and failed to identify one-third of the errors in the test document. Though it zipped along on short texts, it took more time on long texts than the others because it must leave room for its in-memory dictionary, and cannot load all of a large document at one time.

Miz'spell offers a number of features missing on other spelling checkers. There is an auto-learn command to automatically

add all the new words in a document to the dictionary, an excellent feature if you intend to create alternative dictionaries (though you must take special care to ensure that all the words are correct in any document used).

### **Proofreader**

Like Hexspell, Miz'spell offers the advantage of a video scroll of the full text being processed. At the end of the proofreading session, Miz'spell displays statistical information on the total number of words checked, the number added to the dictionary, and the percentage of words recognized by the program.

Aspen Software's Proofreader is another component package that you can upgrade with the addition of a correction feature called Proof-Edit. The complete program is available on four disks, which require a minimum of two drives and 32K. Documentation is quite good.

Offering the largest stock dictionary of programs tested (109 grams), the package comes with utilities for editing and adding to the basic word list. It can handle documents with as many as 1,600 different words, a limit seldom reached on a 48K system.

Proofreader creates a bad-word file with the extension /BWD, which Proof-Edit then reads. The program has its own routines for identifying source and output files, but you can easily use two switches to override the defaults.

The program and dictionary together are huge. Nothing less than an 80-track, double-density system will permit the entire program to reside on one disk. Proofreader does not scan for the



presence of Proof-Edit before exiting to DOS, nor have the edit functions been INKEYed. The only error that Proofreader failed to recognize in the test document was the word "diction-arys," since if the letter before a final "s" is not a vowel or another "s." Proofreader assumes the word is a regular plural.

Proofreader has all the advantages of the full video scroll and accurate in-context evaluation of suspect words that Hexspell and Miz'spell offer. But the hashing algorithm Aspen Software used significantly reduces the amount of time needed to process longer documents. This program is the only one that saves suspect words to a disk file for later reference.

### Conclusion

If operating speed and the ability to call the program from within a word processor are the principal considerations, then either Microproof or Chertext is the clear winner. If you demand accuracy combined with the convenience of a correction

feature, then the choice seems to be between Hexspell and Proofreader.

If you have only one disk drive and a single-density system, choose either Microproof or Miz'spell; but if that drive is a double-headed 80, then for the largest available literal dictionary choose Chertext. If you want value for your money, Miz'spell is a "best buy." If you write many text files in excess of 10 pages each and can afford it, consider using two of the programs—one for speed and convenience and another for accuracy. (Chertext, Apparat Inc., 4401 S. Tamarac Parkway, Denver, CO 80237; \$59.95.

Hexspell has been replaced by Hexspell 2, Hexagon Systems, P.O. Box 397, Statton A, Vancouver, BC V6C 2N2; \$99.

Microproof has been replaced by Electric Webster, Cornucopia Software, P.O. Box 5028, Walnut Creek, CA 94596; \$89.50 Models I and III, \$149.50 Model II.

Miz'spell, Programs Unlimited, 20A Jericho Turnpike, Jericho, NY 11753; \$59.95 disk.

Proofreader and Proof-Edit, Aspen Software, P.O. Box 339-M, Tijeras, NM 87059; \$80.)

Michael M. Finefrock  
April 1982

★ ★ ★ 1/2

**Color Scripsit**  
**Tandy/Radio Shack**  
**Color Computer**

At \$40 a copy, Color Scripsit is a good value. However, some compromises have been made.

The program allows maximum use of your memory

because of its ROM-pack location. It is accompanied by a good 40-page manual; its biggest failing is its lack of caution regarding the program's limitations.

The text-editing mode is probably what you'll use the most. It is accomplished by several facilities that make the job relatively painless; you can use the arrow keys to move the cursor one position at a time. You can easily insert text or move or delete characters, words, or blocks of text.

One operation can lead to disaster if not executed carefully. When inserting text, pressing both the shift and the left arrow inserts spaces until the next tab location. This works well unless there is no tab between the point of insertion and the end of the line.

In this case, if you press the shift and left arrow to insert spaces to the end of the line, the machine hangs up and the only way to recover is to hit the reset button and lose the document in memory. To avoid this, use the enter key to space out to the end of a line.

When saving on tape, don't misspell the name of the document. If you do, the processor looks for a nonexistent document and continues to do so even after the end of the tape is reached. The only way to regain control of the computer is to press the reset button.

The print function lets you direct the contents of memory to the printer or to print to tape. Printing to tape saves the text so it can be read by a Basic program.

You can change a page's format by altering seven standards, including text width, margin

size, hyphenation minimum, lines per page, first page number, line spacing, and capitalization.

Color Scripsit operates quickly on documents not exceeding 5K or 6K; on documents larger than this a noticeable delay exists between the keystroke and the action on the screen. For personal or limited business use, Color Scripsit is a good package. (Tandy/Radio Shack, Fort Worth, TX 76102; \$39.95.)

Stephen G. Stone, III  
October 1982

★ ★ ★ ★

**Electric Pencil 1.0**  
**Michael Shroyer Software**

★ ★ ★ ★

**Scripsit 1.0**  
**Tandy/Radio Shack**  
**Model I**

Electric Pencil and Scripsit are both sophisticated word processors for the TRS-80. Each program is written in Assembly language and can be used with either cassette or disk systems.

Either Pencil or Scripsit can be used as an uppercase system without any modification to your computer. To produce lowercase, a conversion is necessary. Luckily, for the thousands of owners with non-Radio Shack modifications, Scripsit will work with the normal Electric Pencil lowercase modification. However, anyone who has had Radio Shack install a lowercase kit is stuck with Scripsit. The Electric Pencil will not produce lowercase with this official modification.

Electric Pencil comes with an

operator's manual, written as a guide rather than a textbook; the average user can develop a good understanding of the system in an afternoon of reading and experimenting. The Scripsit disk is supplied with sample text files, two program versions, and a three-cassette audio instruction course. The manual is not usable on its own, and the taped instruction course requires about six hours of listening time.

The Electric Pencil has a menu of print parameters that you set before printing. Unlike Scripsit, the parameters are not stored with the text file, so you have to remember what they were. Also, the number of commands available in Scripsit is greater than the 40 or so available to the Electric Pencil user, making it much more difficult to understand and use.

Three of the most useful features of Electric Pencil are not available to the Scripsit user. The suffix /PCL is automatically added to all files created by Pencil, making them easy to spot on your disk directory. The command DI gives a directory of all files with the suffix on the drive specified; in Scripsit you have to exit the system to use the DIR command in DOS. You also must return to DOS in order to kill files.

Scripsit is cheaper and more flexible, but the Pencil is simpler; I tend to create files in Pencil and print them out with Scripsit. For real fancy printing, Scripsit is a must, but for ease of operation, Pencil can't be beat. Maybe somebody will come up with a Pencil compatible with the Radio Shack modification, or, perhaps, produce software that is as nice to use as Pen-

cil—but as advanced as Scripsit. (*Electric Pencil 2.0 is now sold by IJG Inc., 1260 Westhill Blvd., Upland, CA 91786; \$79.95 tape, \$89.95 disk. Scripsit is sold by Tandy/Radio Shack, Fort Worth, TX 76102, Model III cassette \$39.95, disk \$99.95, Model III SuperScripsit \$199, Model II Scripsit \$399, Color Scripsit cassette \$39.95, disk \$59.95.*)

Jim Perry  
May 1980

★ ★ ★ 1/2  
**Flextext  
Apparat Inc.**

★ ★ ★  
**Subscript 5.0  
ProSoft  
Model I and III**

These two word-processing enhancements let TRS-80 users take full advantage of the Centronics 737 capabilities.

Flextext modifies Scripsit on disk so that printer function commands embedded in text use almost every ability of the 737—proportional font right justification (mixed with elongated and compressed characters), underlining, superscripts, and subscripts—all anywhere within a document.

A special file called modify/SCP is entered in DOS to incorporate Flextext comments in Scripsit. The modification is stored as a separate file named SCP/LC or SCP/UC without any alteration of the original Scripsit file. SCP/xx has all of Scripsit's features.

The embedded-in-text printer commands always begin with the escape character (the less-than symbol), followed by a



primary action code (S or E for start or end), and a secondary action code to identify the 737 printer function to be activated.

One excellent feature actually overcomes a 737 shortcoming—stopping elongated printing at the end of a print line. Flextext continues double-width printing line after line if desired. Special print commands activate Flextext when it is time to print a document.

With Apparat's Flextext, Scripsit combined with the Centronics 737 becomes an outstanding word processor.

Subscript 5.0 is a Newsript-compatible word-processing enhancement that combines programs named Edit, Scrip, Prop, and Mininit into a Basic/machine-code package that is line oriented rather than character oriented like Scripsit.

You compose using the Edit mode, inserting print format commands as you go. This is done in Basic, so you must hit enter after each 255 characters, or at the end of each display line.

When your document is ready for printing, Script reads the printer function commands, and combined with machine-code Prop, formats and prints the documents using all of the 737's special features.

Mininit is an upper/lowercase driver that supports Radio Shack's modification. It also detects Basic's string compression mode and has a screen-print routine.

Subscript has some features not found in Scripsit or Electric Pencil. One is a darkness control code, which overstrikes a character, and a table of contents code, which builds a separate file of strings to be

printed as a table of contents with automated page and paragraph numbering.

Another useful feature, the alter code, prints special 737 characters that are not on the TRS-80 keyboard.

However, Flextext wins over Subscripts prop in right-justification ability. Prop can right justify single- and double-width proportional styles mixed on a line, but not a mixture of proportional and compressed formats. Flextext handles all combinations (excluding 10 characters per inch).

*(Flextext, Apparat Inc., 4401 S. Tamarac Parkway, Denver, CO 80237; \$29.95. Subscript 5.0, ProSoft, Dept. C, Box 560, N. Hollywood, CA 91603, has been replaced with Newsript, which sells for \$124.95.)*

John F. Rogers  
September 1981

### Hexspell

(See Page 239)

### ★ ★ ★ 1/2

### Lablmakr ETS Center Model I or III

Lablmakr is designed to create custom labels and should not be confused with mail-list-type label programs that are used for mass mailings. It is most useful for generating return-address labels or disk labels, or for generating a mailing list if your list numbers just a few hundred. The package can be used with an Epson printer with or without Graftrax.

The program has 16 label formats that run the gamut from the compressed to the jumbo

mode. The documentation is easy to use. The instructions take you through a step-by-step procedure for setting up an auto-booting disk for the various DOSes the program works with (TRSDOS 2.3, NEWDOS 2.1, and NEWDOS80 1.0 and 2.0).

Lablmakr is menu-driven with four user options. The first option lets you sample the files that come with the program of the files that you have created with option number four. The second option lets you print, correct, or add a new label. One handy function I liked was the program's ability to correct any label without having to retype the information from scratch.

Option three enables you to print the sample labels included on the program or your own personal creations. The fourth option is for adding new labels. You can create or correct a label, save it to disk, print it, or abort operation.

A minor problem with the package appears when you first initialize a data disk. You must break the program and type GOTO 1300; this accesses a routine that lets you name the disk for future reference. It's a good feature, but it should be menu-driven. Also, the program doesn't have an insert, delete, and reorder function. This makes it impossible to keep an up-to-date mailing list.

The program is self-prompting, easy to use, adequately error-trapped, and very versatile. Lablmakr is a good value for the money.  
(ETS Center, Box 651, 35026-A Turtle Trail, Willoughby, OH 44094; \$19.50 disk.)

Mark Sprague  
October 1982

**Microproof**  
(See Page 239)

**Miz'spell**  
(See Page 239)

★ ★ ★ ★  
**Model II Scripsit**  
**Tandy/Radio Shack**  
**Model II**

Scripsit for the Model II is Tandy's high-level entry into the word-processing arena for its top-of-the-line microcomputer system. Scripsit's features compare favorably with other state-of-the-art, sophisticated, screen-oriented machine-language wordmashers like Electric Pencil, WordStar, and Magic Wand.

I need flexible formatting features, but my primary emphasis is text editing. The commands should be simple and require a minimum of keystrokes. An integrated system from one manufacturer mates the software specifically with the hardware. It is thus with Scripsit and the Model II.

The arrow keys scroll and position the cursor. Some other programs require the control key and four different character keys. You can move the cursor one character at a time in any of the four directions indicated by the arrows.

Two special function keys, F1 and F2, serve as your single-stroke edit keys. F1 opens the text at any place you position the cursor to allow insertions. F2 deletes text.

Scripsit uses Model II's reverse-video feature to highlight selected text portions for bulk deleting, moving, and

duplicating. You do not have to use special text boundary markers.

The Define command, used to highlight text for manipulation, has several variations. Press W to define a word, S to define a sentence, P to define a paragraph, A to define all text above the cursor, and B to define all text below the cursor.

A special format line showing the format settings appears as a line of dashes along the bottom of the video display. Set the left and right margins on this line with (and). Typing O sets the standard outline tab to indent the first line of each paragraph.

Scriptit provides almost unlimited formatting variations within a document. You can reset the format line with new margins and change line spacing at any time for the complete document or any portion of it.

You can print text at any time while you are working on a document. You can print a document from beginning to end continuously, or pause after each page to allow insertion of another sheet of paper. Scriptit commands use all the Daisy Wheel Printer II's features. Some options include bold printing, underlining, overstrike (to create accented or other special characters), and superscripts and subscripts.

Scriptit uses only a portion of available RAM (about 8K) for text storage. You cannot fill computer memory. Each time you begin on a new page, Scriptit saves the former page to disk. There is a trade-off with a page-at-a-time system, however. Documents longer than one page produce the delays inherent in disk access. Scriptit is fairly fast

in its disk procedures, but transferring a page between disk and RAM requires a few seconds.

Scriptit is a completely integrated program; the editor and formatter are in the same program. You can print a portion in a specified format while you are editing.

Once you become familiar with Scriptit, it is easy to use. When you first sit down with it, you could be overwhelmed, especially if this is your first exposure to word processing. Because it is a rather complex package, Tandy includes an eight-lesson audio-cassette training course. Additionally, there's an on-screen Help menu to call up when you get bogged down.

Scriptit's page-at-a-time orientation can be frustrating. As such, Scriptit works within the relatively small RAM space of 8K. It does this quite well for same-disk operations.

If you want to make a back-up on another disk, the Copy utility copies the entire document one page at a time. Page one is transferred from disk to RAM, displayed on the screen, and copied; page two is transferred from disk to RAM, displayed, copied, and so on. In a two-drive system, this is done automatically. All you do is wait.

If you have a one-drive system, however, be prepared for some hair-pulling. You have to manually swap disks after each page is copied. A 20-page document requires 20 exchanges of disks! Using the back-up utility takes about 20 minutes and copies user and program material, overkill when you need a copy of one document. This, too, requires manual swapping

of disks several times on a one-drive system.

Tandy programmers should make a special provision to use all available RAM so as many pages as possible are loaded into computer memory, copied, then RAM completely filled again and copied. Copying only one page at a time is needlessly inconvenient and wasteful of a 64K system's capabilities.

Scriptit would be more efficient if its developers had taken a tip from Scripsit for the Model I and the other micro-based programs and provided a RAM buffer large enough to hold and manipulate 15 to 20 pages of text. The only advantage of the program in its present form is that, in the event of a power failure, you lose a maximum of one page.

Scripsit is a satisfying and sophisticated word-processing program with advanced, easy-to-use editing features and wide flexibility in formatting text. Scripsit will no doubt be the choice of more Model II owners than any other word-processing program. It is less expensive than some others, and it supports Radio Shack's Daisy Wheel Printer II.

*(Tandy/Radio Shack, Fort Worth, TX 76102; \$399.)*

Richard Harkness  
February 1982

★ ★ ★ 1/2  
**Newsript 6.0**  
**ProSoft**  
**Model I and III**

Newsript 6.0 is a word-processing program that offers a justified proportional font along

with its other features.

There are several programs; most are written in Basic for easy modification. The two chief programs are Edit and Script. Edit allows full editing of a document, including insert and delete, block copy, move or delete, and global search and change. Autosave is a convenient feature, allowing you to automatically save the document as often as you desire. It also has the ultimate command, Whoops. This restores the screen to its original content by wiping out all changes.

When you create a document, signal completion by typing End on the command line. This saves the file in process, and gives you the opportunity to stop, edit another file, or go into Script (default). The file must be saved. A Script is another program 13 grams long.

Newsript differs from earlier ProSoft products with some important changes. Mininit, an extra option on the earlier programs, is now included and supports Radio Shack's lowercase modification. This also speeds up the keyboard, providing repeat and debounce. You no longer need to hit enter with every line, and forced spacing is now possible.

The program is not perfect; it doesn't provide any means of hyphenating, and some of the commands aren't convenient. But, for those of you with a dot-matrix printer and a limited budget, this is definitely one program to consider.

*(ProSoft, Dept. C, Box 560, N. Hollywood, CA 91603. The current version of Newsript [7.0] sells for \$124.95 or \$139.95 with the Mailing Labels com-*

*panion program.)*

**Jon Boczkiewicz**  
March 1982

★★★★  
**Newsript 7.0**  
**ProSoft**  
**Model I or III**

Newsript is the best word-processing package I have seen. The well-written manual references chapters divided into Edit commands and Script (formatting) commands. The manual includes chapters on installing Newsript for your computer and printer and how to create indices, titles, tables, and points (bullets).

Because of Newsript's power and flexibility, it will take a little while to get the hang of it, but once you do you will find it to be a powerful word processor.

Newsript is written in Basic and machine language. The editor is fast. The upper limit is 750 characters per second. Basic is used primarily for I/O and to install the front ends of Newsript.

Newsript is broken up into several programs. Control is automatically passed among them as needed, freeing the user from the hassles of remembering which programs to run.

Newsript is, for the most part, menu-driven and optionally integrates with Microproof's spelling checker and the G.E.A.P. graphics editor. Newsript supports many printers as well as double- and single-width, italics, boldface, super- and subscripts, underlining, and backspacing, and it is unsurpassed in printer control.

Newsript's editor is fast, easy, and powerful, but where Newsript shines is in how it formats the words on the printed page. Newsript easily creates tables, indices, tables of contents, form letters (with either keyboard entry allowed at format time, or selectively read data from disk files according to key words), and multiple top and bottom titles.

It also lets you change fonts, center text, turn formatting on and off, and append and imbed files together for long documents. No other TRS-80 word processor can match Newsript's ability to format text.

Newsript comes on a single disk for Model III users, and both sides of a floppy disk for Model I users. It is supplied on a TDOS system disk (an abbreviated version of DOSPLUS), which is also quite good. Newsript can then use TBasic (a small Basic), leaving maximum space for your documents.

One of the most interesting and powerful attributes of Newsript is its ability to create form letters. You can create a data file and in it place a line of key symbols, a block of lines, followed by a number of variables. Another thing Newsript does well is link different files together to create finished documents.

Newsript is for professional writing. It would be overkill for lesser needs.

Newsript 7.0 is an excellent product. The system is large, but it needs to be to allow for the power packed by the Newsript package. Unless you are familiar with WordStar or with main-frame word processors, it will take you a few days before you

## SOFTWARE

---

can put out any sizable documents, but the result will be well worth it.

*(ProSoft, Dept. C, Box 560, N. Hollywood, CA 91603; \$124.95.)*

Bruce Powel Douglass  
October 1982

★ ★ ★ 1/2

### **Palantir Word Processing Designer Software**

★ ★ ★ ★

### **Scriptsit 2.0 Tandy/Radio Shack**

★ ★ ★ 1/2

### **Select Word Processing Select Information Systems Model II**

I found all three of these Model II word processors to be high-quality, professionally-made products. All three offer the usual cursor controls, scrolling, word wraparound, typeover, and insert and delete options. They also include text-formatting features such as left and right justification, block movement, lock, deletion, tabs, headers, footers, full format control, file merging, special printer controls, global search and replace, and so on. Two of the programs, Select and Palantir, require CP/M. Scriptsit 2.0 is supplied with TRSDOS. All three deserve high praise for ease of installation.

Palantir must be configured for a specific computer and terminal by the dealer before it can be used—a slight inconvenience for the mail-order customer, but also an assurance that everything is working correctly,

Palantir thoughtfully includes a keyboard map for 30 different computers, including the Model II. The instructions refer to specific keys by their function. You simply look up the function you want on the chart for your machine, and it tells you which key to punch. Palantir supplies drivers for a variety of printers, and these may be interchanged at any time.

Select encourages purchasers to have their dealers configure the program, but they also make it quick and easy for the end user to install. Unlike Palantir, Select requires that the printer be specified during installation, so you'll need two or more copies of the program if you have more than one printer.

Scriptsit, furnished with TRSDOS, is specifically designed for the Model II. Installation is a matter of putting the disk into the drive and flipping the reset switch. If you have a Line Printer IV or Daisy Wheel II, everything is hunky-dory, but owners of other printers may or may not be able to use all of their printer's special functions. If you are on good terms with Assembly-language programming you can write a driver to meet your printer's requirements.

All three programs do the job of educating the user reasonably well, although they use radically different means to achieve this goal.

The Select user is asked to put the manual down and call the Teach option, a 90-minute interactive tutorial.

The Palantir program also includes a step-by-step tutorial, although it is the manual that does the teaching. Scriptsit 2.0

uses tutorial cassettes and a professionally written, designed, and produced 80-page reference manual. The manual is perhaps the best I've seen included with any software. While it doesn't include more than the others, it makes the information easier to get to. Scripsit is also supplied with several well-designed sample files.

For the beginner, Scripsit and Palantir are generally the easiest to master. Select is more complex and takes a little longer to grasp; it tends to have greater appeal to the technically astute user.

Scripsit is the most elegant of the three. Whether or not it is efficient or easy to use seems to depend completely on individual experience. I find that most people learn the basics of Scripsit in a short period of time, without excessive reference to the manual.

This can be a drawback. For example, if you don't get far into the manual, you might never discover that you can cut the number of keystrokes needed by entering a direct command, or bypassing the display of one or more menus. In fact, you can chain several commands together through a single key, customizing your system for any particular need. Users who never continue their education beyond the fundamentals usually describe Scripsit as a slow, unwieldy system. But once a user has lived with it for several weeks, opinions tend to be very favorable.

One feature unique to Scripsit is the method of saving documents. Scripsit is page oriented, which means you display a single page at a time,

formatted just as it will appear when printed. Text is automatically saved for you but before leaving the program, you must remember to return to the directory. Failure to do so can result in loss of data.

Palantir superficially resembles Model I/III Scripsit in its basic functions, so veteran TRS-80 users tend to grasp it quickly. There are four simple commands used for disk I/O—Edit, Read, Save, and Backup. Edit creates a new file, Read loads a previous file from disk, and Save stores the current file on disk. Backup saves the file to disk, then reads it into memory again. It never erases the old file until the new one is successfully stored. I use this command every 15 minutes or so to reduce data loss in case of power failure.

While Select handles blocks in the traditional way—you place a marker at the beginning and end of the block, and then do whatever you want with it—Scripsit and Palantir allow you to place the cursor at the beginning of a block, and specify how much text you want in it using simple commands. Text in the block is highlighted by displaying it on the screen in reverse video.

Another feature of both Palantir and Scripsit is the user-definable keys. Each key holds up to 255 characters in Palantir, and 250 keystrokes in Scripsit. Scripsit offers 20 user-definable keys, and Palantir 36. Select can perform the same function in a limited fashion, but is not as convenient to use as the other two. Palantir does not recommend including control codes in user-defined key text, but Scrip-

sit has no such limitation.

Select's most important feature is the inclusion of a proofreading dictionary and sub-program. The Select spell program works well, and allows you to add your own vocabulary to the dictionary. Scripsit offers a proofreading program that must be purchased separately. Both Scripsit and Select allow merging of and access to data from other sources. The Scripsit merge option is the easiest to understand and use, but Select's is far more flexible.

To conclude, all three programs are top notch. Select and Scripsit are the most versatile of the three: Palantir is the easiest for beginners to learn and use, with Scripsit a close second. For creative writers who are primarily interested in the editing aspect of word processing, Palantir and Scripsit allow the greatest freedom from technical details during the writing process. From an economics point of view, Scripsit is the least expensive of the three, particularly if you don't yet own CP/M.

Whichever you decide on, I offer this advice to help you get the most from your system—read the manual!

*(The Palantir Word Processor sells for \$450, and is available from Designer Software, 3400 Montrose Blvd., Suite 718, Houston, TX 77006. The Select Word Processor is available from Select Information Systems, 680 Beach St., Suite 396, San Francisco, CA 94109 for \$595. Scripsit 2.0 is marketed by Tandy/Radio Shack, Fort Worth, TX 76102; \$399.)*

Paul Grupp  
September 1982

★ ★

### **Pensawrite Word Processor Pensadyne Computer Services Models I and III**

Pensawrite has five modules designed to work in a 16K single-disk system. Two are printout formatters, one for letters and one for reports. One receives formats and creates upper- and lowercase text. Another is used for editing. All four are invisible to the user and are called and controlled by the master menu and module.

Compared to most documentation I've received, the manual is wonderful. Commands and processes are described in detail, without being wordy. The summary page ignores two important text-building commands, but the program routines are documented within remark statements.

Had the programmers been as careful with their programming as they are with this manual, I would recommend Pensawrite. However, Pensawrite fails to live up to its potential. The 64-character input and use of printouts is a natural limit for efficient use of memory and random-access disk files. Memory is saved by keeping the formatting of routines at a minimum. The user types most special formatting, like special indentations, though it would be nice to signal some single-space sections within double-spaced text. If you have special needs, Pensawrite won't accommodate them.

The editor function is impossible because of its failure to use random-access files. Every correction, no matter how insignificant, rewrites the entire text file.



It is so poorly designed that I don't recommend Pensawrite. It can be used for short personal letters that don't require careful editing; it does create a nicely formatted title page. But if it is to be taken seriously as an inexpensive personal word processor, Pensadyne should take the TRSDOS manual and rewrite Pensawrite with random-access files.

*(Pensawrite is sold by Pensadyne Computer Services, 4441 W. First St., Vancouver, BC, Canada V6R 4H9; \$19.95.)*

Louis Zeppa  
January 1981

★ ★ ★ 1/2

**Poor Man's Text Editor**  
**Don Coon**  
**Model I**

If you need to do word processing, but don't want to pay a lot of money, try the Poor Man's Text Editor. It costs less than \$10, is surprisingly versatile, and might satisfy all your needs.

The text-editor program features upper- and lowercase, a nondestructible, easily-maneuvered blinking cursor, and editing and graphics options, and it interfaces with either tape or printer. You can add, delete, or repeat any regular or graphic character. Furthermore, you can hold up to nine pages of text in memory, and when printing you can select the line spacing.

The program requires a 16K memory to use all the options. To print or display lowercase, your TRS-80 must be modified using one of several conversion

kits available for \$20 or less.

With only 13 commands, using the Poor Man's Text Editor is simple, and the instructions supplied with the cassette are excellent. Once you've entered text, editing is easy. Move the cursor under the character you want to change and type the change.

You can't conveniently move blocks of text, or paragraphs, from one position to another; everything is accomplished with individual characters. Right-justification of the text is not automatic, but you can do it by inserting blanks on each line as needed. It's a good product for the price.

*(Don Coon, 1228 Alpine, DeWitt, MI 48820; \$9.95 tape, \$19.95 disk.)*

Fred Blechman  
February 1980

**Proofreader**

(See Page 239)

★ ★ ★ ★

**Qwerty**  
**Med Systems Software**  
**Model I and III**

Qwerty is a patch that expands Scripsit's word-processing capabilities. It is the best word-processing package I have used for typing technical documents and general-purpose manuscripts.

Qwerty is a machine-language program and comes in several versions for various printers. The supplier claims the program works with either TRSDOS or NEWDOS and possibly other DOSes. I reviewed Qwerty for the Line Printer IV (Centronics 737). Med Systems reports that ver-

sions are now, or soon will be, available for the Daisy Wheel II, the Line Printer VIII, and the new NEC printer.

Without a copy of *Scriptit/LC*, *Qwerty* is usciss. *Qwerty* uses most *Scriptit* commands plus four additional ones. *Qwerty* lets you compose pages with a two- or three-column format, using either a proportional or 16.7-cpi font. This lets you get a maximum of readable material on a single page. Another handy format feature of *Qwerty* is the ability to check the format of the text without printing anything.

The program is supplied on disk with versions for machines with 32K and 48K of RAM. It is shipped on a data disk without a DOS. *Qwerty* works fine with a single-drive system when copied onto a DOS disk, but a single-drive user will have to find a way to make the copy.

*Qwerty* works equally well on both the Model I and Model III with a Line Printer IV. The manual is well-written and complete, but assumes the reader is familiar with *Scriptit*.

The keyboard can generate about 76 special characters, including the ASCII characters not on the keyboard, part of the Greek alphabet, and a variety of mathematic and scientific symbols. *Qwerty* also provides superscripts and subscripts. If you need footnotes, *Qwerty* lets you place them anywhere in the text; the computer takes care of putting them where they belong.

One problem with *Qwerty* is that the horizontal centering function from *Scriptit* doesn't work well with proportional spacing unless the writer supplies extra spaces to adjust for the variations in letter width.

Switching to a monospaced type font sometimes confuses the computer as to character counts. The result is uncentered text.

Because *Qwerty* is a flexible and powerful word-processing program, it takes some practice to learn the commands. Potential users should feel comfortable with *Scriptit* before attempting to learn *Qwerty*. Don't be discouraged if the first few times you try typing some complex charts or formulas, the printouts look rather strange. Like any useful skill, *Qwerty* must be practiced to be appreciated. (*Med Systems Software, P.O. Box 3558, Chapel Hill, NC 27514; \$74.95.*)

D. Wilson Cooke  
October 1982

★ ★  
**Refware Thesaurus**  
**David C. Whitney**  
**Associates Inc.**  
**Models I or III**

Put a thesaurus on disk and save yourself some thumb wear. At first blush, it sounds like an unimpeachable idea. But unfortunately, it is an idea that lives and dies by its host—the microcomputer.

The *Refware Thesaurus* consists of three 5¼-inch disks, one each of nouns and adjectives and one builder allowing you to create a specialized thesaurus of up to 6,200 words. Each disk is sold separately; verbs and adverbs are in the prototype stage along with a Model II version.

All three programs are very easy to use, so much so that the three user's manuals seem

almost superfluous. The thesaurus displays or prints a list of 10 synonyms. However, unlike a conventional thesaurus, it also inserts the synonyms for the word you want replaced in your sentence. After you compile your first set of 10 synonymous sentences, the thesaurus asks if it should find a different group of alternate words for your noun or adjective. If you reply positively, the program determines if it has such a group and, if so, either lists or prints the set.

All this is done very quickly—faster than a search through a conventional thesaurus, if you find a suitable word in the first list the program gives you. Chances are, however, you won't. Here's why:

Under the word *agreement*, for example, a conventional thesaurus lists 13 synonym categories. Most times, you can immediately spot the category closest to the meaning of the word you are looking for. With Refware, the computer chooses the category. If it isn't the correct one, you must request another group of words. This adds to the time it takes to find a synonym. Also, you never know if there is another synonym set, so every time you call up one set, you must call up another to see if there is one and if it has a more suitable word.

While the microcomputer thesaurus is initially faster than finding a word manually, it often seems like it is taking longer because you spend time waiting, while time spent thumbing through a printed thesaurus is time spent doing.

The Refware Thesaurus, due to disk space considerations, excludes words with more than 12

letters. It contains 6,200 nouns and the same number of adjectives. This is a significant limitation when you consider the fourth edition of Roget's book, published by Harper & Row, contains 250,000 words.

Another limitation of the microcomputer thesaurus is that it cannot be accessed while you are in a word-processing program. If you are composing with a word processor and need a synonym, you have to break out of that program, load Disk Basic, load the thesaurus, find a synonym, break out of Disk Basic, reload the word processor, call up the file you were working on, and insert the word you were looking for. A cumbersome process, to say the least.

The eight utility programs in the thesaurus are sold separately as a builder enabling you to construct a reference work and possibly make some extra cash—if you compile a specialized thesaurus that may be useful to others in a given field or profession and Refware judges it to be commercially viable, they'll market it and pay you royalties. The builder lets you enter synonyms in the same groups-of-10 format, with "dummy" used to reserve spaces in blocks with fewer than 10 words; it also allows listing or printing the blocks or all the words in your thesaurus in alphabetical order.

According to Refware author Whitney, who has edited the *World Book Encyclopedia*, *Encyclopedia Americana*, and *Reader's Digest Almanac*, "With the ever-expanding memory size and storage capacities of microcomputers, it seems likely that massive

printed reference works like a (Roget's) thesaurus will soon go the way of the dinosaur." After viewing the Refware programs, Roget's heirs needn't sweat over loss of royalties. . . not for a while yet.

*(David C. Whitney Associates Inc., P.O. Box 451, Chappaqua, NY 10514. The Thesaurus 2.0 now sells for \$89.95 and consists of 24,800 words. The builder is sold separately for \$149.95, as are the noun, adjective, adverb, and verb packages for \$24.95. All are for the Model I and III with 48K, two disk drives.)*

John P. Mello Jr.  
April 1982

★ ★ ★  
**Scrip232 (Scripmod)**  
**Small Systems Software**  
**Model I and III**

Having trouble with your non-Radio Shack printer with Scripsit? Scrip232 (advertised as Scripmod) is a program that customizes Scripsit for use with your serial printer.

Installation is straightforward. Detailed instructions tell you how to use the tapedisk utility to create a file called Scrip232/CMD. On execution, the program asks eight questions which it uses to customize Scripsit to work with your particular printer.

You decide whether to modify the uppercase or lowercase version of Scripsit. The selected version of Scripsit is loaded into memory by Scrip232. Next, you can modify Scripsit for use with either the TRS232 or RS232-C interfaces.

Since some printers require a

line feed to advance paper, and since Scripsit sends a carriage return but no line feed at the end of each line, Scrip232 will modify Scripsit to add a line feed after each carriage return.

Additionally, Scrip232 adds handshake control, automatic line feeds, and characters to the serial driver contained in Scripsit.

In summary, I'm very pleased with Scrip232. I think it's a real bargain. Judging by reader letters in the computer magazines, many people are having trouble getting Scripsit to work with non-Radio Shack printers. If you have a serial printer and either a TRS232 or RS232-C interface, Scrip232 may solve the problem. *(Small Systems Software, P.O. Box 366, Newbury, CA 91320, could not be reached for update information. Scripmod originally sold for \$14.95.)*

John A. Records  
April 1981

**Scripsit 1.0**  
(See Page 243)

**Scripsit 2.0**  
(See Pages 246 and 250)

**Select Word Processing**  
(See Page 250)

★ ★ ★ 1/2  
**Spell 'N Fix**  
**Star-Kits**  
**Color Computer**

Spell 'N Fix is a program for finding and correcting spelling errors in Color Computer text files. It compares each word with entries in a dictionary file, identifies words that do not appear there, and displays or prints

them. You can mark suspect words for later correction in a new version of the text material, and you can customize the dictionary to reflect your vocabulary. Spell 'N Fix, an adaptation of Magic Spell, is unique in that it is also available in the Flex format.

Spell 'N Fix operates on ASCII files so it is compatible with most Color Computer text processors. I have used it with Color Scribes (using the print-to-tape option), CC Writer, Super Color Writer, and Telewriter. Telewriter doesn't produce ASCII material, but rather core-image (binary) files. Spell 'N Fix and the disk version of Telewriter include conversion programs to take care of the mismatch.

I reviewed the disk version of Spell 'N Fix. The cassette version is much less flexible, and I question its utility over the long haul; cassettes have too many restrictions for the kind of file shuffling that a program of this nature requires.

Spell 'N Fix is easy to run, although there are many options to declare. It has lots of screen prompts and an excellent manual to lead you along. The disk contains nine files, but only three are involved in an elementary spelling correction run: SPELLFIX/BAS, the loader and interface program; SPELLFIX/BIN, the machine-language spelling checker; and DICT/DAT, the dictionary file, which contains just under 20,000 words in a special compressed format.

Once the text file is located and opened, there is a prompt—you are asked to specify the kinds of words that Spell 'N Fix is to examine. The choices are

any group of characters (excluding certain punctuation marks) enclosed by spaces or carriage returns, or only what the program considers reasonable words.

Spell 'N Fix reads the text file, constructing in memory an alphabetized list of all distinct words to be compared with the dictionary. This is the key to processing large files. English text is very redundant; the number of different words in a large sample is much smaller than the total number of words. The manual claims that Spell 'N Fix can handle files of up to 400,000 characters in a 32K system. The program read my test file in about 1 minute, 50 seconds.

Next, Spell 'N Fix asks for the name of the dictionary file to be used; in the early going, this is the stock dictionary. You can add new words to the dictionary at this time. The program reads as much of the dictionary as fits into memory along with the list of distinct words, and when it has done so, it compares the two.

Words that do not match the dictionary are printed on the screen (and on the printer, if that option was chosen) in alphabetical order. The process is repeated in steps until all of the dictionary has been read and compared with the input text.

Several options are presented when the first misspelled word is encountered. You can ignore a given word, mark it for future correction, or mark all such findings in the remainder of the text. The latter option is as close as Spell 'N Fix comes to fully automatic operation; using it, my file was processed in just

under two minutes. The other options call for an operator decision every time a misspelling is detected. The decision to ignore such a finding is not always easy; the program recognizes specialized technical jargon, but proper names and control codes sometimes cause confusion.

Operating Spell 'N Fix is not as cumbersome as it sounds. Here is a summary of the entire process for a one-drive system, assuming that an ASCII text file is available at the outset:

Load Spell 'N Fix

Insert text file disk

Read in and process file.

Insert Spell 'N Fix disk. Read in dictionary and prepare a list of misspellings.

Insert text file disk. Reread original file and prepare a new one with errors changed or marked.

The dictionary furnished with Spell 'N Fix contains not quite 20,000 words. This limit was originally chosen so there would be room for the user to construct a second customized copy on a dedicated dictionary disk.

The Spell 'N Fix disk also contains a number of utility programs. In addition to Build and BINCON, there are SAMPLE/DAT, a short piece of text used for teaching the system; and LIST/BAS, which prints the contents of ASCII text files. RESET/BAS, which flushes everything and returns the computer to its normal wake-up state; EXPAND/BAS, which converts the Spell 'N Fix dictionary from its normal compressed form to an expanded form in which it can be edited. This is the only way to remove words from the dictionary, but the manual points out that Expand

may take several hours to perform the complete conversion.

The time Spell 'N Fix takes to process one file is not unreasonable, although its use on a single-drive system is unnerving. One solution to this problem was to copy my working dictionary onto each disk used for text files. This mini-mized the disk swapping, at the cost of 31 or 32 grants per disk.

Spell 'N Fix does its job and adds a little more professionalism to Color Computer text processing.  
(Star-Kits, P.O. Box 209, Mt. Kisco, NY 10549; \$69.29 cassette, \$89.29 disk.)

Scott Norman  
November 1982

### **Subscribe 5.0**

(See Page 244)

★ ★ ★ ★

### **Super Color Writer 1.0 Nelson Software Color Computer**

Veteran hackers, who remember the Cenozoic Era of personal computer word processing, must be scratching their heads over how fast software for the Color Computer has come of age. Nothing illustrates that coming of age better than this offering by Nelson Software.

Super Color Writer is a machine-language program allowing you to edit text anywhere on your screen. It contains most editing functions found in high-powered word processors. You can insert text by the line or character; delete by character, line, or block; locate, change, or delete any string of characters;

and copy, move, or delete text blocks.

In text mode, the screen is black with a green command line across the top. Hitting clear twice puts you in the command mode, where you can clear all text from the buffer, before or after the cursor. You also activate the locate function from this mode; a string may be located exactly as you type it in or you can "mask" the search so it will ignore uppercase/lowercase distinctions.

The only way to clear all the text in the buffer is through the Clear command or cutting the power to the computer. We reset our CC with text in the buffer and Super Color Writer and the text was still there after reboot.

In the text mode, in addition to the left-, right-, up-, and down-arrow key motions, there are nine additional cursor movements activated by pressing clear and a letter: bottom or top of CRT, start or end of text, start or end of line, back or ahead 15 lines, and pre-set tabbing at 5, 8, 16, 20, and 24 spaces.

Anyone using their computer with a black-and-white monitor will have difficulty using this program. It uses colored blocks to identify functions and modes—white when clear is hit, blue for character insert, orange for a block marker, and so forth. This makes it easy to find things when scrolling text and to keep track of what mode you're in. Most control functions take two keystrokes, which improves typing speed over word processors requiring two keys to be held down simultaneously.

Once we began typing, we found the major flaw in Super Color Writer: It has no lowercase

character set. Uppercase characters are displayed as light capitals on a black background and lowercase as dark capitals on a light background.

Super Color Writer's chief competitor, Telewriter (produced by Cognitec of Del Mar, CA), addressed that problem by incorporating a software-generated lowercase character set in its word processor. Dan Nelson said his firm rejected that option so there would be more room for text—8,446 characters available after loading in a 16K Color Computer, compared to 2,200 (less than two double-spaced typed pages) for Telewriter.

Super Color Writer's block commands are simpler and more powerful than its competitor's. Unlike Telewriter or Model I or III Scripsit, the program contains a true block move; when it moves a block, it erases the original block automatically.

The word processor automatically chains files in memory. If you have one file there, you can load as many more as you want until memory is full. This wrinkle proved very valuable in dealing with the vagaries of CLOADing.

Another feature allows you to write Basic programs, save them in ASCII format, and run them under Basic. This is handy because the full screen editing commands of a word processor are superior to the line-oriented ones of Basic.

Super Color Writer also gives you a programmable function, in which you can include up to 28 commands, keystrokes, or modes, and have them all executed up to 65,535 times automatically. For cassette file-linking, you can tell the com-

puter how many files you want loaded and printed and it does it without your touching the keyboard again.

There are functions for headers, footers, automatic centering, and including codes for your printer's special features. The printing section of the documentation is sketchy. The authors, Nelson explained, assume users



have intimate knowledge of their printer before using Super Color Writer.

The software has default values set for perfectly centering text on 8½-by-11-inch paper. However, you can set your values for margins, justification, page numbering, page length, and header and footer location. You can also pause printing by hitting the space bar during a print run.

Super Color Writer is designed to operate out of the serial port of the Color Computer. It will not work with Micro-Labs Inc.'s CPRINT parallel printer interface.

As state of the art as Super

Color Writer 1.0 is, Nelson Software has already released 2.0, which features a window mode allowing you to see text exactly as it will be printed, free-format cassette file-linking, disk file-linking for the disk version, headers and footers on odd or even pages, three programmable functions, a flush right margin, memory used and memory left display, and six programmable tabs. So Color Computer owners brace yourselves; a good word-processing program is going to get even better.

(Nelson Software, P.O. Box 19096, Minneapolis, MN 55419; \$99.95 disk, \$74.95 ROM pack, \$49.95 cassette.)

John P. Mello Jr.  
and Jake Commander  
September 1982

★ ★ ★ 1/2  
**Superscript**  
**Acorn Software Products Inc.**  
**Model I and III**

Scriptit, Radio Shack's TRS-80 word processor, has been on the market for over a year now, and many people have complained about their problems when interfacing Scriptit to a non-Tandy printer. Computerists said Radio Shack did not sell a printer that could be effectively used as a word processor: In fact, many were upset that Radio Shack did not assist them in writing driver routines for letter-quality non-Tandy printers.

Well, Acorn Software Products has released a short program that patches Scriptit so that it can be used with a variety of serial and parallel printers not



previously supported. And it fixes some rather troublesome bugs in it.

With Superscript you can get a disk directory that tells how many free grants are left on the disk without losing your text file; kill a file while in Scripsit; create files that can be read by TRSDOS or NEWDOS; insert text into unjustified lines while printing, a necessity for generating form letters; and protect drivers in high memory from being written over by Scripsit. Further, if your printer has the capability, you can superscript and subscript (using normal and reverse half-line feed), underline, print in boldface, slash zeros, and set the printer to 10- or 12-pitch (characters per inch) while printing.

In addition to these capabilities, Acorn has fixed Scripsit so that you need a file-specification typed with the L command, thus eliminating the problem of accidental text wipeout. Status messages printed at the bottom of the screen have been shortened, and the keyboard driver is faster without disrupting the video display. Custom drivers are supplied for the Diablo and NEC 5530 printers, both serial and parallel versions, and a custom driver lets you design your own, also serial or parallel.

Superscript's directory command works great, except that you cannot get a list of the invisible or system files or the file allocation specifications. Except for that, I have found little to complain about.

The drivers supplied on the disk are easy to use and well explained in the pamphlet. I use the recently introduced Radio Shack Daisy Wheel II printer,

and was upset to find that it has many features not supported by Scripsit: I was, however, very pleased to discover that it can use every Superscript option, though I did have to cheat a little on the boldface feature (using the command sequence 1B,01,08 and the custom parallel driver, since the Daisy Wheel II is too new to have a driver on the Acorn disk). I still cannot use most of the special characters on the printwheel, but at least I can use the additional brackets, braces, and caret symbols, thanks to Acorn. (Acorn Software Products Inc., 634 North Carolina Ave. S.E., Washington, DC 20003; 32K disk, \$50.)

Terry Kepner  
March 1981

★ ★ ★ 1/2  
**Telewriter**  
**Cognitec**  
**Color Computer**

With Telewriter, Color Computer word processing becomes a big-league affair. Telewriter offers full-screen editing with many of the cursor-control features found in professional text-editing packages. It has most of the print formatting and cassette file-handling features you might want, plus a few specialties of its own. Telewriter employs a software-defined character set to put 24 lines of 51 characters each on the video screen. That is a respectable fraction of a double-spaced text page and makes the full-screen editing meaningful.

Full-screen editing is one of Telewriter's outstanding

features. You can move the cursor, which marks your location for text entry or deletion, anywhere over the displayed text, at any rate of speed. You can also scroll the text in either direction. Finally, Telewriter doesn't distinguish between text entry, deletion, or editing modes. Type in your material and alter it as you go along; you can see a significant amount of it at any time.

This free-wheeling operating mode as well as the real lowercase character set, (which resembles 5-by-7 matrix printing), lets you direct most of your energy towards thinking and composing. There is relatively little need to be concerned with control keys and the computer until you format the material for printing.

Telewriter is a machine-language program requiring a minimum of 16K of RAM. Since Telewriter itself occupies over 6,400 bytes, 32K is better in terms of storage capability. A 16K machine can store just over 2,200 characters, a 32K enables you to store about 18,500 characters.

### Getting Started

The program is supplied on cassette, recorded once on each side. It also contains a text file for demonstration purposes. The documentation is quite complete and takes the form of two manuals: a 27-page tutorial and a 33-page reference manual. An additional sheet summarizing the commands is also furnished.

There are two menus: main and format. The main menu comes up when you load the program and give the EXEC command; it is used to read or

save a cassette file, create or edit text in RAM, and perform other chores.

The format menu, accessed by the F command, controls the printing format and instructs the computer regarding the number of files to be chain printed. You can also embed many of the format menu's commands in your text in order to change such parameters as line spacings, margins, and printing fonts on the fly. Menus are displayed in standard reverse video, and most commands require only a single-letter response. Where additional input such as a numerical value for setting a printed parameter is required, the familiar flashing Basic cursor appears at the appropriate spot on the format menu.

You enter text by using the main menu's C (Create) command. The display switches to a white screen with a flashing L for a cursor and a small black rectangle for an end-of-text marker. Anything you enter at this point will be in Telewriter's software-defined font. The usual shift/zero combination is retained for uppercase lock. The shift gives any arrow key a fast auto-repeat, while Clear is a special control key. It enables you to make the cursor skip to the top or the bottom of a text file using the up or down arrows, or to the beginning or end of a given screen line with the horizontal arrows.

Clear lets you define other keystrokes as commands, rather than text, in this mode.

If you place the cursor somewhere in the middle of a block of text and begin to type, the new material is added to the old; you don't type over

anything. In the same way, backspacing over text doesn't erase it. The Break key alone erases the character immediately to the right of the cursor. The Clear-Break combination erases the character to the cursor's left. There are also simple methods for erasing complete lines or large blocks of text.

Telewriter has many other commands for moving and changing text, and most of them use Clear to define a control function in a similar way.

### **The Cassette Handler**

The cassette-handler portion of the main menu provides the means for storing and retrieving text. Its commands allow you to read a cassette file into memory, append a text file from tape to the end of any file in RAM, write all or part of a file from RAM to tape, and verify that such writing was successful.

Cassette-handler commands consist of a single letter (or the percent sign, in the case of partial save). Each results in the appearance of the Basic's flashing block cursor next to the selected command as the program awaits a file name. You can skip the file name in all cases, but it is a poor idea to write an unnamed file to tape. The procedure for saving a portion of a file requires you to first go into the editor and mark the end of the desired portion by pressing Clear-E. You then move the cursor to the beginning of the segment as though you were deleting it, return to the main menu with Clear-M, and enter the /command.

The Verify command allows you to check a tape record to make certain that it can be read properly, without destroying the

text file in memory.

Another handy command, Clear-W, displays the number of words in the text buffer, along with a count of the number of lines into which the text is formatted at any particular time. It also tells you how many spaces (characters or carriage returns) are still available. The part of the menu that displays this information also gives the name of the last file read or appended from cassette. This helps you keep track of your file names when you are chaining many together for printing.

### **Formatting and Text Printing**

The main menu's F command brings up the format menu. This displays the defaults for the basic format parameters: line spacing and length; the left, top and bottom margins; the number of lines per page; and the baud rate (encoded according to a table in the documentation). You can reset any of them, and also specify several others: the starting page number of the file in memory; the number of files to be chain printed; and whether or not the printer is to pause at the bottom of each page. There is a special font command for MX-80 users as well.

The format parameters only need to be set once if you are chain printing several files. If you desire page numbers, make the appropriate non-zero entry for the starting page of the first file. Telewriter keeps track of the line count and the page number across file boundaries during printing.

Telewriter's embedded commands are defined by a special leading character—an upward

pointing caret—generated by pressing Clear and then the period. You can use everything mentioned so far, except for the MX-80 font and the number of files in the printing queue, in an embedded command.

A few print commands that don't control standard format parameters also use the embedded command syntax. You can center a title by prefixing it with a caret and an asterisk. You can print a header at the top of every page of a file but the first, using the II format code. One of Telewriter's very few bugs occurs when you use headers. If a header definition is followed by two carriage returns, it fails to print at the top of the second page; it doesn't begin until the third. This bug has now been eliminated, but Telewriter purchasers should check out their own copies of the program.

While Telewriter may lack some of the features of word processors designed to run on larger, business-oriented machines, it gives the Color Computer respectability as a document preparation center for the home or school. I feel that the combination of full-screen editing and the very respectable font give the program real advantages.

The Color Computer still lacks a word processor with a file-merging capability. It is not yet possible to write a letter with variable data in the address and salutation fields and merge it with a mailing list, for example, but Telewriter remains a fine program. At \$49.95 it is the most expensive of the Color Computer word processors, but it is well worth the price. (Cognitec, 704 Nob Ave., Del

Mar, CA 92014; \$49.95 cassette, \$59.95 disk.)

Scott L. Norman  
May 1982

★ ★ ★ ★

### **Word Micro Architect Model I, II, and III**

Micro Architect offers a low-cost group of programs for word processing under the group heading Word.

Word2 is a 16K cassette-based program designed for TRS-80 Level II. Word3 and Word4 are similar programs for a disk-based Model I, and Word-M2 is a Model II program. I ordered Word4.

The disk contains four files and enough TRSDOS programming to enable it to run on a one-disk system. Two of the files were Word3 and Word4; the other two were text files that either could be listed or formatted with Word, consisting of instructions on how to use Word. A 12-page manual that came with the disk is adequate only for someone fairly familiar with computers and basic word processing.

There are slight variations in different versions of Word, but all are based on the same philosophy. It is written in Basic so that the program is relatively easy to modify and use. It also runs more slowly than a program written in machine language.

Word takes a text file prepared independently of the processing program and formats it into a printed document. The text file

is prepared using Basic, with line numbers and all, using about 20 special format instructions appropriately interspersed within the text. The edit function allows text editing at any time.

Word4 is a slightly enhanced version of Word3. Both programs are for a standard disk-based machine using TRSDOS or NEWDOS. Word4 has the ability to generate lowercase characters sent to the printer even if they aren't displayed on the video monitor. Because of the software needed, Word4 runs more slowly than Word3, but where Word3 can accept only one text file and process it, Word4 can accept several text files and chain them together for printing longer documents. Word4 can also number lines in the formatted document so that they correspond with those in the text file.

The only serious problem I encountered was in using the format instructions incorrectly when I prepared the text file. It is possible to get Word hung in an infinite loop if the commands are not used properly, but a little experience will solve such problems. For the price, Word works well.

*(Micro Architect, 96 Dothan St., Arlington, MA 02174; Model II packages \$49-\$69, Model I and III packages \$20-\$69.)*

D. Wilson Cooke  
August 1981

**Word-IV**  
(See Page 43)

★ ★  
**Wordsmith**  
**ABS Suppliers**  
**Model I and III**

Wordsmith is an inexpensive word-processing program written in Basic in two versions: one for 16K machines without disk, the other for 32K machines with at least one disk drive. The 16K version lacks the disk version's search and kill functions, and requires the B-17 Tape Operating System marketed by ABS to run. The disk version comes on cassette and can be CLOADED under Disk Basic, then saved to disk.

Wordsmith might satisfy a hobbyist interested in playing around with word processing for casual letter writing or keeping a diary, but it will not suit those who have more demanding writing applications. Typists who work at moderate to rapid speeds will find that Wordsmith has difficulty keeping up with them, often losing characters and occasionally locking up completely until typing stops.

The instructions are clearly written and accurate. You should be able to use Wordsmith without difficulty the first time you load it.

Wordsmith allows a length of up to 60 characters and features word wraparound. Unfortunately, it does not allow simultaneous composition and editing, and its menu-oriented design is a definite disadvantage. Wordsmith's editing functions are cumbersome.

Once you have written your text, Wordsmith allows several means of formatting its appearance. Designed for use with the MX-80 printer, Wordsmith

provides several functions related to that printer's capabilities, including specified print density, emphasized strike, and double strike.

Wordsmith doesn't even have a way of exiting the program. In comparison with more expensive, more sophisticated word-processing programs, it has little to offer, but its low price is hard to argue with. If you want to make serious use of microcomputer word processing, though, look elsewhere.

*(ABS Suppliers, 3352 Chelsea Circle, Ann Arbor, MI 48104; \$14.95 cassette or disk Model I, disk only Model III.)*

Hugh M. Ruppensburg  
June/July 1982

## — MISCELLANEOUS —

★ ★ ★

### **Aircraft Instrument Approach Simulator J.C. Sprott Model I or III**

Do not look upon Aircraft Instrument Approach Simulator as a game. It is a simulation involving flight. An individual unfamiliar with instrument flying will not know how to make the simulator work and will probably crash often.

The program comes on cassette with a description of aviation terms as well as a diagram of a typical approach and an explanation of the screen display.

In spite of these materials, neophytes might find the screen

somewhat mystifying. The description neglects the artificial horizon, but after "flying" the thing for a while, that becomes obvious. Nonpilots might be confused by the localizer function, since the simple statements about flying to or away from the needle in front-course versus back-course situations will not make much sense.

However, this program is lots of fun, somewhat of a challenge, and a learning experience for a student or pilot who is familiar with instrument flying and wants some simulator practice.

The program has some drawbacks. You enter numbers from the keypad to direct the aircraft. Unfortunately, these are often combined, and it is sometimes unclear how to get out of a turn while climbing or descending and still maintain the climb or descent. The four arrows should have been used, and a small range of numbers should indicate power changes. Power can only be increased or decreased very slowly. This is artificial.

You specify the field elevation and the landing heading as well as wind direction, speed, and turbulence. I like programs that select all these things for me. It is more realistic, since even in a situation in which you know the wind characteristics on the ground and at altitude, you must always "feel out" wind conditions during an approach and experiment until the approach is set up. Knowing the exact wind throughout takes away some of the challenge.

Considering the price of the program and the amount of fun someone interested in flying is likely to have with the program,

I recommend it. It works and so will you!  
 (Addendum: Since this review was written, Mr. Sprott has incorporated some of Mr. Gorsky's suggestions.)  
*(J.C. Sprott, Madison, WI 53715; \$9.95.)*

Buzz Gorsky  
 February 1982

★  
**Astrology**  
**Tandy/Radio Shack**  
**Model I**

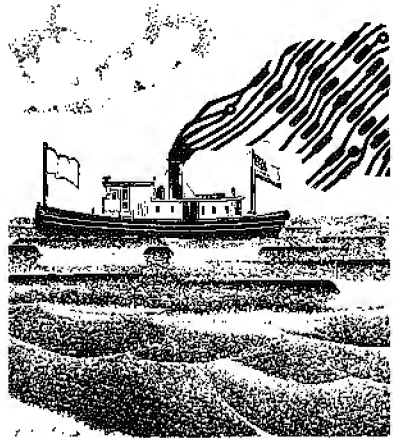
If your only previous experience with astrology is the column in the newspaper or the scales at the dime store, you probably won't be able to use most of the information produced by this software. However, if you are a professional charter, you're going to wonder what happened to some very important information.

The nonprofessional user, who is less likely to have a printer, will find it hard to assimilate the long lists of data. That's why charts were developed. Yet the documentation to this particular program doesn't tell you how to make one.

The professional, who is more likely to own a printer, knows how to make his own charts. So why write a program that only generates a chart on the printer?

If the program was written with the professional in mind, then why doesn't it give heliocentric (sun-centered) aspects along with the geocentric (earth-centered) ones? Where is the meridian sign? The sidereal (star-relative) time? Why are the calculations only accurate for the 20th century?

The program shortchanges



both the professional and the beginner. It appears that Radio Shack software developers only got halfway on this one.  
*(Tandy/Radio Shack, Fort Worth, TX 76102; \$19.95 cassette.)*

Dennis Thurlow  
 May 1981

★★  
**Autobasic**  
**Schneider Enterprises**  
**Model I**

Autobasic is a Basic program that writes Basic programs. It comes on disk with three programs, ABAS, ABRUN, and SUBR, and a minimal DOSPLUS operating system. This is nice for TRSDOS users, since TBASIC allows almost 40K of usable RAM for Basic.

After loading Tiny Basic and running the main program, ABAS, you are presented with a menu of several options, including entering equations from keyboard, loading program elements from or saving to disk, correcting or reordering elements, reordering the equations

for a subroutine, and entering a recursive conditional structure.

The program accepts Basic equations as input and writes the program using these. Autobasic distinguishes between simple and array equations; the former contain all simple variables and the latter contain one or more arrays. ABAS writes the file to disk and ABRUN compiles it into Basic form, routing all output to the printer, screen, or to a disk file. Once this file is read in (from PFILE), it is operated on and then written to disk (in WFILE). Then, the created program is loaded and executed.

Autobasic puts in a Let statement for assignment statements, making it difficult to use the finished program. For example, if you are trying to insert a DEFFN statement, you'll get a syntax error on "LET DEF FN F(X) = . . ."

There is no simple way to enter interactive For . . . Next schemes for non-array equations within Autobasic or for grouping array equations within a single For . . . Next loop. The conditional format only allows comparison of a variable to a constant. The created programs are not written in a time-efficient or memory-efficient manner for iterative programs, and you cannot use the program flexibly to create such a structure.

The program allows you to reorder equations but not to reorder and place in a subroutine. If you do, it forgets its new order. If you enter the subroutine first, when you reorder your program it forgets that you have a subroutine.

Autobasic has some good features that, if properly im-

plemented, are a great addition for a programmer to write error-free code. An example is SUBR which lets you enter subroutines. Since you will have trouble matching the variables, this routine allows a global editor to change all occurrences of that variable to some other variable, and will not make these changes inside Basic reserved words. You can, therefore, change T to G without changing GOTO to GOGO.

As implemented, you can use a full-screen editor better for Basic than Autobasic (and much cheaper, too), but if Autobasic is rewritten so that it is a usable product, you will appreciate this feature.

On a scale from 1 to 10, the manual rates a solid 1. It was totally unreferenced, with no table of contents, no index, and no page numbers. It is also poorly written and difficult to read.

More limitations: Only single-dimensioned array variables are allowed. Autobasic does not support DEF FN, and I/O is not supported to disk or cassette. The program is also not error-trapped; entering a routine out of sequence (the manual doesn't mention the correct sequence) often leads to redimensioned array errors causing the program to exit to command mode, and invalid data is not checked. The reordering doesn't work under some circumstances, and the program incorrectly parses some statements with parentheses, such as  $df = (d1 - d2)/h$ , and thinks that this is an array equation.

The program has great potential if these major problems are removed. Schneider Enterprises' approach is fine and potentially



useful, but the limitations are so debilitating that I cannot recommend its purchase. I hope these suggestions can be incorporated into an Autobasic revision. With some work, this could be an excellent product.

(*Schneider Enterprises, 1252 N. Brownslake Road, Burlington, WI 53105; \$95.*)

Bruce Powel Douglass  
September 1982

★ ★ ★

**Computer Downs**  
**Ernest H. Fellows**  
**Model I**

There are many lessons to be learned at the race track. Ernest Fellows and John Cater put together a TRS-80 16K, Level II program that runs on cassette to take a lot of the guessing out of picking out a winner.

The documentation is well organized. It asks for quite a bit of information about each horse. This is to your advantage, since the accuracy of the program's predictions depends upon the number of relevant factors considered.

The program is easy to use. If you make a mistake, the left arrow erases it. The more critical answers use error-trapping. The screen formatting is well organized. Above the input prompts, a table continually displays the previously input information and a calculated speed rating for each horse.

Although the program is designed well, runs smoothly, and seems to be free of bugs, it lacks a few features that would enhance its usefulness. A print-out option would be useful so

you could easily check your input data for errors.

Then, if the program had an editing option, you could return to the input mode and change the incorrect entry and recompute the results. An editing function would also let you give different weight to different factors by changing the input data.

Many factors should be considered in choosing horses, but Computer Downs uses most of the obvious ones. Several factors it doesn't consider are track condition, track surface, track surfaces of a horse's previous races, and whether the jockey is compatible to the jockey who rode the horse in the best of his last three races.

When balancing the strong points with these negatives, Computer Downs is still an excellent program. It has continually proven its accuracy at the track. But it is only an aid, and it does not profess to remove all risk from your decision. However, every bit of relevant information helps to give you a better idea of which horse is most likely to win.

(*Ernest H. Fellows, 9230 Bingham, San Antonio, TX 78230; \$30.*)

Joel Benjamin  
February 1982

★ ★

**Life List**  
**Manhattan Software**  
**Model I and III**

My Life List program tells me I saw a Double-Breasted Tuna Grinder in Antrim, NH, last weekend. The weekend before, I ran across a Black-Billed Boot Stomper behind Dr. Fred's Auto Clinic. According to my notes, I blew the dirty little thing away

with my grenade launcher.

How I managed before Manhattan Software came up with this bird-list program I will never know. My closets were full of shoeboxes stuffed with matchbook covers, napkins, bank deposit slips, grocery store receipts—anything to scrawl a note on when I unexpectedly spied a Double-Shagged Throw Rug or Red-Eyed Fly Hog.

Naturally, my notes got confused. Was it a Clip-Winged Tarsucker that I flattened with my moped? Did I really see the rare Tie-Dyed Burger Bun while floating in my sensory deprivation tank? And where is that recipe for White-Crowned Sparrow fritters?

Life List takes care of all my problems. It lets me record the names of some 450 birds along with the date and the place sighted and any extra notes. I can sort the list alphabetically and get a complete printout with the touch of a button. I can perform a special search for any bird, using only part of its name.

Of course, the people who might possibly be interested in this program would probably fit into a phone booth, with room left over for an hors d'oeuvres table. But Charles Leedham of Manhattan Software says this is the sort of specialized offbeat program that microcomputers were made for.

After transferring my notes to disk, I was faced with another dilemma—what to do with all those shoeboxes. It finally dawned on me. They would be perfect for storing field specimens. At last my roommate has stopped complaining about those stuffed birds under the kitchen sink.

*(Manhattan Software, P.O. Box 1063, Woodland Hills, CA 91365; \$24.95.)*

Eric Maloney  
May 1982

★ ★ ★

### **Morse Code Transmit and Receive Program Richcraft Engineering Model I**

This program will send and receive Morse code at up to 25 words per minute, using no peripheral devices whatsoever.

The cassette motor control relay, K1, is used for the keying relay. The cassette EAR plug line is used for receiving Morse code audio derived from the station receiver's speaker terminals. Since K1 will only handle very low power levels (about 6 volts at 400–500 mils) it is strongly recommended that a 7406/7507 TTL buffer chip or Radio Shack #275-004 relay be used as a buffer between the TRS-80 and the station transmitter.

The initialization segment allows you to choose either alphanumeric or Morse code readouts on the video and to choose the Morse transmit speed (receive speed is automatic). An error-trapping function can be used to obtain immediate return to the transmit mode.

In that mode all generally accepted Morse characters are provided plus EOM (end of message) and EOW (end of work) by using the # and & symbols respectively. Should a keyboard character which has no Morse equivalent be entered, an error-trapping subroutine skips it. Twenty prepared Q signal and message formats are given. There is no limit,

except available memory, to the number of additional messages you can add.

There is also a speed subcommand that can change the transmit code speed without reinitializing the program and losing the data stored in the automatic logbook. Solo operators can practice on alphabet alone, alphanumerics, or alphanumerics and punctuation.

*(Richcraft Engineering, Drawer 1065, Chautauqua, NY 14722.)*

Robert M. Richardson  
July 1980

### ★ ★ 1/2

#### **Moving Signboard Circle Enterprises Model I**

Moving Signboard is a machine-language program that stores a message and then displays it in a horizontal line that moves from right to left. A message of up to 1,024 letters is first typed on the screen and then stored in memory.

Then, another message is typed on the screen and then stored in memory. Then, another message is typed on the screen, leaving at least one horizontal line vacant. When Signboard is activated, and the second message is constantly displayed on the screen while the first crawls across any chosen line.

This program is mostly a novelty; its greatest potential is as a store window display. *(Circle Enterprises, P.O. Box 546, Groton, CT 06340, could not be reached for an update.)*

Rod Hallen  
February 1980

### ★ ★ ★ ★

#### **TRS-232 Formatter Small Systems Software Model I**

The Formatter, for use with the TRS-232 printer interface, comes complete with 17 pages of documentation.

When you type RUN, the program POKEs into high memory and cycles through the various options. Once it modifies the machine-language program in memory to reflect your choices, you can delete Formatter and load or run any of your own programs.

The Formatter accepts 10 baud rates and can be modified to support nonstandard baud rates. It generates form feeds, a line feed after a carriage return if needed, and it lets you set the maximum line length.

Another great feature of the program is the Line Length for Early Termination feature. You can also set the number of lines per page and the number of spaces between pages. You can view the information as it is printing on the video screen and disable interrupts from the disk controller.

The Formatter program even inserts an automatic keyboard debounce routine. To top it off, it contains four special features that make the program worth its price for their inclusion alone. The features allow you to stop printing at any time by simply pressing the space bar, stop listing and return to Basic by pressing the break key, use the clear key to reset the lines-per-page counter (if using the keyboard debounce routine). If you didn't request this routine, you can still reset the line

## SOFTWARE

---

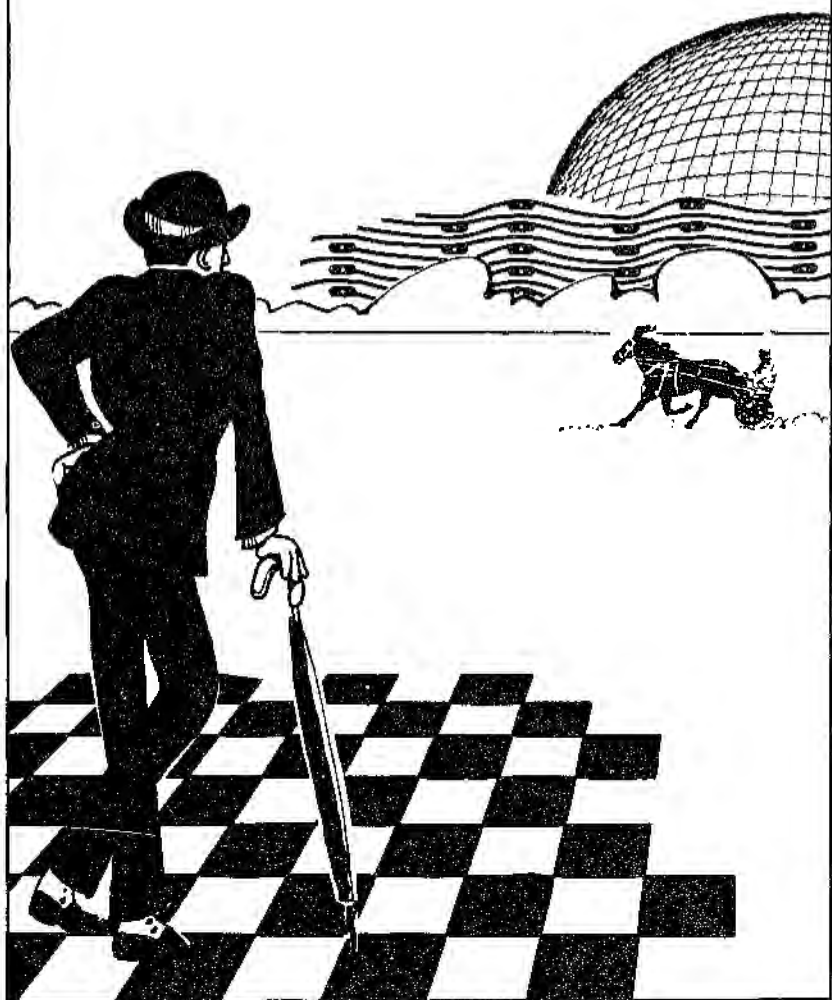
counter by typing LPRINT CHR\$(3). The Formatter also sends to the printer any information on the screen (barring graphics) making intelligent use of the TRON/TROFF functions of Level II Basic possible.

My only complaint about Formatter is that you cannot set

left-hand margins. Considering the complexity of the package, I wonder why they forgot it. (*The TRS-232 Formatter is sold by Small Systems Software, P.O. Box 366, Newbury Park, CA 91320.*)

Hugo T. Jackson  
October 1980

# HARDWARE





## DISK DRIVES

★★★★

**Color Computer Disk System  
Tandy/Radio Shack  
Color Computer**

The TRS-80 Color Computer Disk System is for everyone, or at least for anyone that can afford disks. The system is easy to operate and requires a 16K Color Computer with Extended Basic. The 92-page manual is written in a style similar to Radio Shack's Color Basic manual, including sections for the programmer and newcomer to programming. Its simple and direct language takes you slowly, and clearly, through the disk operating system (DOS), and features examples of the commands and Disk Basic.

The system consists of the disk interface in a plug-in program pack cartridge, a cable, and a 5¼-inch disk drive. The cable connects to the disk interface and the disk drive with one free connector allowing you to connect a second drive, if desired. The interface controls up to four drives, but an additional cable is required when

adding two more drives. The 35-track double-density 18-sectors-per-track disks hold about 161K bytes of data.

The DOS uses only 2K of your system's RAM leaving you 6.4K; when you PCLEAR1, you have a little over 11K of RAM left. (Tandy/Radio Shack, Fort Worth, TX 76102; \$599.)

Howard Berenbon  
June/July 1982

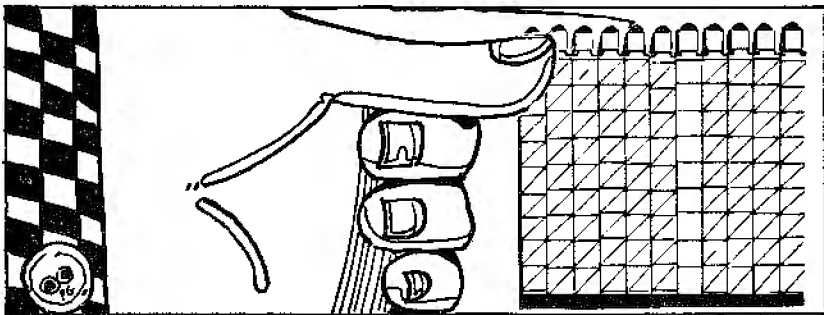
★★★½

**Color Computer Disk System  
Tandy/Radio Shack  
Color Computer**

Color Computer owners that have been struggling with tape systems need struggle no longer. I've been trying out the Color Computer disk system for over two months and consider it to be a reliable piece of equipment.

The main benefit to any disk user is the ability to perform loads and saves reliably at high speed. Although the tape system used in the Color Computer is fairly reliable (and faster than the Model I), it suffers from the inherent limitations faced when recording any digital information on an audio machine.

Also important is the ability to



perform random access I/O on a disk file (more correctly referred to as direct access in the manual).

The Color Computer Disk System contains a disk drive, a connector cable, a blank floppy disk, an owner's manual, and a cartridge that fits in the slot at the right side of the keyboard. You won't be able to use any peripherals that use the cartridge slot as it is commandeered by the disk I/O cartridge; I'm puzzled as to why Radio Shack didn't continue the bus on from the back of the cartridge to allow further expansion to take place.

Hooking up the drive is simple. Turn the computer off, insert the cartridge, connect the cable between cartridge and drive, and plug in the drive and switch it on. All that remains is to switch the computer on. You will be greeted by a new message telling you that you're now running under Disk Extended Color Basic.

For users of other microcomputers, the first surprise on powering up is that the familiar sound of the disk loading a bootstrap loader is missing. The Color Computer's disk operating system is already in the ROM contained in the cartridge. This has a couple of advantages such as not wasting disk space with the operating system code and keeping the DOS inviolate against any reckless POKES or errant machine-code programs. The disadvantages are that the DOS is unchangeable (being in ROM) and the memory used by it is unavailable for any other use.

The drive unit itself is manufactured for Radio Shack in Japan and no labels on the

outside give away the producer. However, the drive has a good solid feel about it and I've had no problems in the time I've used it. I also have a second Shugart drive hooked up to the computer and this has also performed without error.

It's possible to have up to four double-density, 35-track drives connected, although the cable supplied is only configured for two. A second drive is more of a luxury on the Color Computer than it is on the Models I and III because the same amount of room is available on the first drive as on the others: 35 tracks of 18 sectors at 256 bytes per sector—that's a total of 156K (excluding the directory track). Of course, if you have large files or want the convenience of backing up without having to swap disks, you may want more than one drive.

The disk system manual is extremely well written—though it's in standard 8.5 by 11-inch format while the other Color Computer manuals measure 11 by 8.5. It's definitely aimed at the novice and patiently goes through the A to Z of disks in a most understandable manner. Plenty of cartoons break up the monotony and also make the manual a little difficult to wade through as a quick reference, though there are eight appendices and an index.

The Color Computer Disk System is reliable, easy to use, a wee bit expensive, but the only way to go if you're serious about getting the most out of your Color Computer.

*(Tandy/Radio Shack, Fort Worth, TX 76102; \$599.)*

Jake Commander  
September 1982



★ ★ ★ 1/2

**Percom Disk Drives**  
**Percom Data Co.**  
**Model I or III**

When we began looking for a disk system, we wanted a system that had shown hardware and software success with other computer systems as well as a firm able to back up its product. We chose Percom Data Company's TFD-100 drive.

It offers 40-track operation and 20-millisecond track-to-track stepping time. It also records on both sides of a disk without needing additional holes.

Best of all, Percom offered its own disk software, MicroDOS, which could be loaded into memory leaving the entire disk free for data and program storage. We purchased one TFD-100 drive, a two-drive cable, and MicroDOS.

Essentially, MicroDOS stores data and programs. It doesn't have disk directories, passwords, or clock functions as TRSDOS has, but it does the job. Furthermore, it is completely resident in less than 7K of memory.

The manual seemed clear, but it wasn't complete. We contacted Percom, and the author of MicroDOS, James Stutsman, answered our questions.

Since setting up the system, we've had no hardware problems. But we've had several software questions, all answered by Percom and Mr. Stutsman. The only lingering problem is the incomplete manual, but Percom has promised a new manual—the perfect addition to a fine disk system.

*(Percom Data Co., 11220 Pagemill Road, Dallas, TX 75243; \$399 Model I, \$599*

*Model III. MicroDOS is no longer available.)*

David Buffington and  
 Cindy Wagner  
 February 1980

**PMC SFD-51-A**  
 (See Page 283)

★ ★ ★ ★

**QuCeS Hard Disk Subsystem**  
**Quality Computer Services**

Quality Computer Services manufactures several hard-disk units. I reviewed the DSK52, a 5-inch MiniScribe hard disk, which is available for the Models I and III. The system consists of a cabinet containing all the hardware (drive mechanism, platters, motors, power supply, and cooling fan), a cable, and one of two operating systems on a floppy disk.

The standard unit uses floppy disks for back-up, but QCS will have an optional 10-megabyte tape drive back-up in the same cabinet available by the end of 1982.

The available operating systems are Micro-System Software's DOSPLUS 4.0/g, which comes with a full operating system plus utilities and manual, or Logical Systems Inc.'s LDOS. The free LDOS system comes with hard-disk drivers only—you must purchase LDOS separately—and DOSPLUS costs an extra \$225. LDOS is easier to install, but DOSPLUS runs slightly faster.

One noticeable difference between a hard-disk and a floppy-disk system is sound—a minor, but psychologically significant, difference. Other than the steady

white noise made by the cooling fan, hardly a sound comes from the hard disk as it reads and writes information.

The most startling discovery is the speed of the system. You will be astounded at how fast the system performs a series of read and write operations. The QCS hard disk can move data at the rate of 5 megabits per second.

And all that room! Imagine reading a directory and finding 5,000K free on each of two hard-disk platters. The QCS 10-megabyte drive holds the information on 45 double-density or 125 single-density disks. Another advantage of hard disks is that they extend the life of your floppy-disk drives.

Most of your software will work with the hard disk. Exceptions include Adventure International's Maxi-Manager and Radio Shack's Profile. This software has routines that call specific disk drives and will not run on the hard disk.

The QCS hard disk performed flawlessly in the two months I tested it. The manual contains step-by-step instructions to install the hardware; I had the system up and running in less than a half hour.

One serious omission in the manual is the fact that a hard-disk unit takes about a minute to warm up, obtain the necessary rpm, and stabilize. Attempting to load an operating system loader before the drive is ready causes an error.

Both operating systems require you to engage a specially configured floppy disk when you turn on the system. The hard disk then becomes the master drive, and all operations originate with hard-disk pseudo-

drive number 4. (LDOS makes the hard-disk platters function as drives 0 and 1, while DOSPLUS calls them 4 and 5.)

The hard disk lets you run your programs one after the other without having to remember on which disk you stored them. After all, on the hard disk you can store hundreds. The fast access time is also convenient. However, when working with data files of less than 32K, word-processing files, or even Basic programs, the increase in speed is merely a minute or two per work session.

But, if you've got mountains of data, or mountains of money and a yen for state-of-the-art hardware, here's a new machine to solve your problem.

*(Quality Computer Services,  
178 Main Street, Metuchen, NJ  
08840; \$3,195.)*

G. Michael Vose  
October 1982

### **SAS00 Eight-Inch Disk Drives** (See Page 293)

### **TFD 100-1 Percom Disk Drives** (See Page 283)

★ ★ ★ ★

### **TFD-100 Percom Disk Drives** **Percom Data Co.** **Model I**

As near as I can tell, Percom has the perfect disk-drive system. It includes the drive(s), power supplies, enclosure, shipping and, until recently, the connecting cables. Even delivery is prompt.

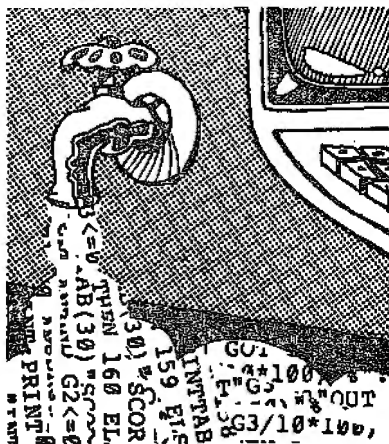
I ordered the TFD-100 system consisting of two WANGCO

Model 82 drives, although you can receive Pertec FD-200 drives. Both support 40-track operation and have dual sense lamps so you can record on both sides of your disks without punching extra holes in the jackets. Some assembly is required, but the instructions make it a snap.

With each system, Percom also sends their PATCHPAK disk which tells DOS how to handle 40-track drives. This PATCHPAK requires two drives to run, but once updated DOS handles 40-track drives just as well as it did 35-track drives.

*(The TFD-100s have been replaced with the TFD-42 for \$499. Percom Data Inc., 11220 Pagemill Road, Dallas, TX 75243.)*

Walter C. June  
May 1980



number of ways, two of which are: develop a dramatic innovation on a useful product or simply build a better mousetrap.

Percom's new double-density system will revolutionize the world of disk operating.

The Percom system has two parts. The hardware is a printed circuit board that plugs into the disk-controller chip socket in the expansion interface, a simple (five minute) installation. The hardware is totally invisible to your single-density system except that it contains Percom's data separator, which provides error-free reading of your disks. The magic begins when you boot up the double-density operating system.

The Doubler is supplied with Percom's DBLDOS. TRSDOS users will have no problems; all TRSDOS commands are available plus the following enhancement: the copy command will work with one drive and copies single to double, double to single, as well as like to like. It will not format or back up anything but double density. This is a source of aggravation. Should

## DISK DRIVE CONTROLLERS

**Disk-80**  
(See Page 288)

★★★★★  
**Doubler**  
**Percom Data Co.**  
**Model I**

★★★★★  
**DOSPLUS**  
**Micro-System Software Inc.**  
**Model I and III**

One may grow rich in a

you want to supply a single-density disk to a friend, you must format the disk with some other DOS. Further, you must copy each file separately with the DBLDOS copy command.

Nevertheless, the hardware is first class. Percorn could safely offer a money-back guarantee for life, because once you use double density, the old way seems like the stone age.

The DBLDOS software is okay. It worked just as the manual indicated. Only seven pages are devoted to the operating system, which is sufficient because there is little to add to the well-written TRSDOS manual. One of my friends even figured out how to install the hardware without taking the bottom off the interface (not recommended).

DBLDOS, like TRSDOS, is satisfactory until one finds something that is far superior: Enter the better mousetrap.

Micro-Systems Software's DOSPLUS double-density system comes with a clear, concise, and mostly complete 28-page manual. There are several functions which are not explained in enough detail. I would grade the manual a B-, and give the operating system mostly straight A's. Still, is DOSPLUS worth \$99, considering you get DBLDOS free with the hardware? Here are some of the features. You can decide if they are worth the money.

DOSPLUS double density is fully compatible with TRSDOS, and allows single-drive backup, copy, and so on. If you have lowercase hardware installed, upon booting up, your keyboard recognizes it and works like always. When you shift a lowercase letter appears. If you want

typewriter format, shift-O does the job. The keyboard repeats without keybounce, nice when one is editing.

The transfer mode allows one to copy all the visible files from one disk to another, which can save a tremendous amount of time. The system automatically recognizes disk density and supports over 120 files; with an 80-track double-density disk, one will probably run out of file space before disk space.

A unique HBasic in the DOSPLUS system lets you go from Basic to a command operation and, when the operation is completed, return to exactly where you were in the Basic program. This feature, plus the Build, Auto, and Do utilities, allows you to create some remarkable routines.

The new printer driver allows forms and paging perimeters to be set and changed at will. Restore brings back files that have been accidentally killed (as long as you haven't subsequently written over those sectors); Purge allows you to delete selected files one at a time. Diskdump is a display sector/modify program that works with filespecs. Diskzap is an easy-to-use single/double disk editor. Crunch deletes all spaces not in quotes and REM statements, helpful when a Basic program won't quite fit in available memory.

That may not happen often. A formatted 40-track will give over 170K usable space and a formatted 80-track over 350K. A single 80-track drive system will now do the work of four 40-track single-density drives (with some limitation). Four 80-track drives provide over 1.4 megabytes

on line!

The advantages of the Doubler are immediate and obvious. It's love at first sight. To fully appreciate the value of the DOSPLUS system, one must sit at the keyboard and use it. (Percom Data Co., 11220 Pagemill Road, Dallas, TX 75243. The original DOUBLER has been replaced by the DOUBLER II which provides, in addition to the original features, a Model I with the hardware to run Model III disks; \$169.95. An updated DOSPLUS from Micro Systems Software, 4301-18 Oak Circle, Boca Raton, FL 33431, Model I and III version 3.5, is now being marketed at \$149.95. DOSPLUS II is now available for the Model II and DOSPLUS 4.0 is available for the hard-disk system.)

Stewart E. Fason  
March 1981

★ ★ ★ 1/2  
**Drive Control Unit**  
**Optronics Technology**  
**Model II**

It has long bothered me that my eight-inch disks continue to spin in the drive after I have accessed the files I need. The wear on the disk drive, and shortening the life of the floppy disk, seems unnecessary.

Now, eight-inch drive systems can be adapted to turn the drives off when they are not needed. Optronics Technology's Drive Control Unit (DCU) is available assembled and tested, or in kit form. The DCU can be installed in existing eight-inch drive enclosures with a minimum of effort.

The kit comes with a six-page manual that includes two pages of drawings and diagrams and a photograph showing how the unit mounts in the rear of an eight-inch Shugart drive. The printed circuit board is about 2<sup>3</sup>/<sub>8</sub> by 3<sup>1</sup>/<sub>2</sub> inches in size and comes predrilled. There are about 20 parts to mount on the PC board. No one who has had any experience at the Heathkit level should have any problems assembling the board.

Upon installation in my system (with a CCS disk controller and Shugart 801 drive), the Drive Control Unit worked fine the first time without fine tuning or troubleshooting. The lack of drive noise and continual disk spinning is a delight!

One problem that was probably not anticipated by the designers is in copying from one disk to another using a single-disk, file-transfer program. If the nine seconds expire before you change the disks, you'll be left staring at the error message "Drive A Not Ready." Nothing too serious, though; you just have to reboot and try again.

This product serves a needed function. The price is reasonable and the device works with no debugging. The disk-drive motor now shuts down nine seconds after being disengaged. Disk and drive wear has been greatly reduced. The unit mounts easily in the rear of a Shugart eight-inch drive without modification to the drive. You must solder three wires to the Shugart PC board to complete the installation. The instruction sheets are detailed enough to guide even a novice.

(Optronics Technology, 2990 Atlantic Ave., Penfield, NY

14526: \$39.95 assembled,  
\$29.95 kit.)

David DuPuy  
December 1982

★ ★ ★  
**Floppy Disk Controller Kit**  
**A.M. Electronics**  
**Model III**

The popularity of the Model III has been accompanied by a veritable host of companies supplying after-market equipment, including disk drives and controllers. These typically offer several advantages over the Radio Shack-installed drives, including lower price and greater flexibility. Most of the available controller kits support double-sided drives.

The A.M. Electronics Floppy Disk Controller Board kit includes the board and all mounting hardware, cables, and power supply, but not the disk drives. I chose two Tandon double-sided, 40-track drives, without case or power supply. I also purchased DOSPLUS 3.3 from Micro-Systems Software. The total cost for my disk system was about \$1,170.

The installation instructions were thorough and easy to follow. A few nuts and bolts for mounting the controller board were missing and the description of the power supply cable did not match the cable I found. I determined the proper orientation by referring to the pin numbers. There were no traces to cut and no soldering was necessary; I was impressed with the way the kit fit perfectly into the Radio Shack design. The most difficult operation was breaking out the plastic face plates that cover the

disk drive openings in the computer housing.

The disk drives must be configured to the system. I called A.M. Electronics and was told to remove the terminating resistor packs from both drives and break certain DIP jumper connections on the drives; which connections to break will differ with different drives.

Upon attempting a backup I ran into trouble—DOSPLUS declared about half the tracks on the disk flawed and rejected the backup. At this point I learned one of the disadvantages of assembling your own system: if something goes wrong, it may be difficult to determine which part is faulty. I ruled out disk failure because it rejected the same tracks every time, even with high-quality disks; I ruled out DOSPLUS failure as TRSDOS did the same thing. I called A.M. again and was told it was probably a faulty controller board.

With the new board installed the system worked perfectly. It was particularly gratifying to be able to ask for DIR:0B (back-side of drive zero). I now have the equivalent of four disk drives with a capacity of roughly 720K.

The A.M. controller kit is compatible with all existing Model III DOSes, although TRSDOS will not support the double-sided capacity. I highly recommend the kit, along with double-sided drives and DOSPLUS as a reliable, easy-to-assemble, large-capacity system. There is a substantial trade-off—Radio Shack will not service, much less honor the warranty of, a system with after-market equipment installed; it is easier to find a non-Radio Shack repair facility

than to disassemble the unit and remove the disk system before going to RS service—but this is an acceptable price to pay for the increased flexibility and lower system cost.

*(A.M. Electronics, 3366 Washtenaw Ave., Ann Arbor, MI 48104; \$390 for the kit; \$205 - \$405 additional with drives.)*

John Ratzlaff  
April 1982

★ ★ ★

**LN Doubler 5/8**  
**LNW Research Corp.**  
**Model I**

★ ★ ★

**Percom Doubler II**  
**Percom Data Co.**  
**Model I**

★ ★ ★ ★

**MULTIDOS**  
**Cosmopolitan Electronics Corp.**  
**Model I and III**

★ ★ ★ 1/2

**NEWDOSSO 2.0**  
**Apparat Inc.**  
**Model I and III**

★ ★ ★ ★

**DOSPLUS 3.4**  
**Micro-System Software**  
**Model I and III**

★ ★ ★ ★ 1/2

**LDOS 5.1**  
**Logical Systems Inc.**  
**Model I and III**

★ ★ ★

**TFD 100-1 Percom**  
**Disk Drives**  
**Percom Data Co.**  
**Model I**

★ ★ ★

**PMC SFD-51-A**  
**Personal Micro Computers Inc.**  
**Model I**

Double-density disk I/O is actually more reliable than single-density. But, how do you get double-density and which disk controller and DOS should you choose? There are a wide variety. I contacted many manufacturers of double-density products, and those that responded are represented here.

Two double-density disk controllers, four DOSes, and four disk drives are reviewed. Although most of this information is applicable to Model I users only, the DOSes have Model III versions.

### The Disk Controller

The Western Digital disk controller that accompanies the Radio Shack Expansion Interface is a single-density disk controller, but it can write several kinds of data address marks. To allow compatibility, both the Percom and LNW disk controllers allow your computer to use the original disk controller as well as the double-density disk controller.

I reviewed the Percom Doubler II and the LN Doubler 5/8. The Percom Doubler comes with easy-to-follow instructions and Percom's own DBLDOS (essentially a TRSDOS patched to run double-density with a few corrected flaws).

The LN Doubler 5/8 comes with better instructions and an even easier installation manual. The manual includes the Western Digital technical manual for their disk controllers.

## HARDWARE

---

It also supports 8-inch floppy drives and comes with DOSPLUS.

I was unable to demonstrate a significant difference in any disk-controller functions between the two and reliability testing showed no significant differences in granules locked out or in read/write errors.

### The DOSes

The aspects of a DOS that I consider important are: ease of use; speed of operation; accompanying utilities, such as editor/assemblers, file utilities, debuggers, monitors, and languages; readability and completeness of documentation; compatibility with other disk formats; and cost.

### MULTIDOS

This is the most inexpensive of these DOSes and has many features its more expensive competitors don't have. It is the only totally compatible DOS around. It will read any of the other disk formats, including DBLDOS. It is also a "no-hang" DOS.

The only major problem with this system is that it only checks four times on a read/write. My old Pertecs showed many more "Data Record Not Found" errors with MULTIDOS than the other DOSes, which check 10 times before returning an error.

The manual is a terse 66 pages. It lacks an index, but it does cover the features of MULTIDOS adequately, provided that you already have the TRSDOS manual.

MULTIDOS has many useful library commands besides those provided in TRSDOS, and has a number of system utilities that include the following: a disk-based version of Radio Shack's

Editor/Assembler, a RAM Scanner to locate a 1- or 2-byte word in memory, a graphics utility to allow direct keyboard entry of graphics characters, a printer spooler, and best of all, MULTIDOS's Versatile File Utility (VFU).

VFU is a utility for frequently needed disk operations, including purging files, printing a disk directory, multiplying file copies, and executing menu-based programs. The Purge option gives a screen of the available files, as do the Copy and Execute commands.

MULTIDOS has the best Basic around. SuperBasic. It also has a programming and debugging Basic called BBasic with Boss Basic utility program, built into it.

### NEWDOS80 2.0

NEWDOS80 2.0 is a double-density DOS that writes and reads a specific disk format differently from the other DOSes. The familiar gran for granule has been replaced by lumps. What's a lump? NEWDOS80's documentation doesn't go out of its way to tell you. It is definable with the PDRIVE GPL parameter, so there are between two and eight granules per lump. You can define the granules-per-lump (GPL) parameter as eight. Once you do so, the DOS's features are about the same as the others.

This system will read other formats, but they must be configured with the PDRIVE each time a different disk is used, particularly if the disks are in different densities.

It will take you longer to get started with NEWDOS80 than with the other operating systems



presented here. This is partially due to the manual's technical approach, and partially to the approach taken by the DOS in its performance of the various functions.

NEWDOS80 has a large number of library commands, that include the following: Break; Chain and Do; JKL, which sends the current screen contents to the printer; HIMEM, which sets the top of memory from DOS; LC and LCDR; Pause; PDRIVE, which assigns the default attributes to a physical disk drive; Print; R; and Route.

The operating system also sports a miniDOS not found in the other DOSes. The miniDOS allows the execution of DOS library commands except Append, Chain, Copy, or Format. Since it is interrupt-driven, it can be used from almost any program that keeps interrupts active and uses the NEWDOS80 keyboard driver.

The utilities provided by NEWDOS80 include the cver-popular Superzap. It also supplies a patched version of Radio Shack's Editor/Assembler and a disassembler. In addition to this, NEWDOS80 handles the standard sequential and random, as well as marked and fixed items.

NEWDOS80 is a good DOS. The Basic is not as good as MULTIDOS's, but it operates well. It has a disassembler, Superzap, and other powerful utilities. Apparat supplies zaps free to registered owners and has provided good customer support in the past. Unfortunately, it does not have automatic density recognition.

### DOSPLUS 3.4

DOSPLUS comes with the LN

Doubler 5/8. It has automatic density recognition, although it cannot read DBLDOS format. It has an automatic repeating, lowercase keyboard driver and DO file capability. The quick format command verifies the tracks backwards. Model I and III double-density versions are directly compatible, meaning that a Model I DOSPLUS disk can be read by a Model III DOSPLUS. DOSPLUS does come with a Convert command to transfer files from a Model III disk to a Model I DOSPLUS disk, somewhat like NEWDOS80.

The program's library functions perform like the other DOSes described. The DOSPLUS utilities include a printer spooler, single-drive copy, Crunch (a program that compresses Basic programs saved on disk), Diskdump and Diskzap (floppy-disk editors similar to Superzap), Map (shows file locations on a disk), Restore (recovers killed files), Sysgen (creates nonstandard system disks), Tape (similar to NEWDOS80's LMOffset), and Transfer (for multiple file copies from one disk to another).

DOSPLUS enters Basic differently than MULTIDOS and NEWDOS80. The Basic is enhanced as well and has a large number of abbreviations.

DOSPLUS comes with a tiny Basic called TBasic. TBasic occupies slightly more space than MULTIDOS's SuperBasic, but has no file space allocated. TBasic lacks many of the enhancements found in regular DOSPLUS Basic, but is useful because it will still run the full set of Basic commands. TBasic displays abbreviated error messages in a further attempt to free up memory.

### LDOS 5.1

LDOS recently reduced its price from \$169 to \$129, due to the large volume of sales. Registered LDOS users receive the quarterly magazine, *The LDOS Quarterly*, support hard to match. The operating system comes with a rather imposing 300-plus-page, well-written manual. The LDOS manual is the best of the DOSes reviewed here.

The library commands in LDOS are similar to those available in the other DOSes. Trace supports and displays the PC register in the upper right hand corner of the display, and Filter is similar to the Route command on the other DOSes but better. It establishes a program to filter the I/O path of some device. Several filter programs are supplied with the LDOS disk.

One of the most powerful features of the LDOS operating system is its job control language (JCL). The JCL is actually a compiled language. It can handle conditional statements and support variables. LDOS allows concatenation of variables and logic expressions using NOT, AND, and OR. In addition to this, the JCL allows macros in its programs. This means that you can specify a macro name and some complex function will be performed.

Other utilities include CMDFILE (a more powerful version of NEWDOS80's LMOffset that allows appending patches to machine code, appending machine-code programs together, saving them to disk or tape, moving them around in memory, and so on) and LCOMM

(a communications program that allows your TRS-80 to talk to other TRS-80 computers or to bulletin board systems, such as Forum 80).

LDOS is a good DOS, but there are a few minor inconveniences. The Basic could be more powerful, and it would be helpful if the top of memory were free. The manual is quite good, and the Filter and JCL features make this the most powerful microcomputer DOS around. Finally, the LDOS support is unprecedented in quality and ease for the registered user.

### In Summary

As you can see, all these DOSes have their good points. MULTIDOS has its low price tag, versatile file utility, no-hang ability, virtually total compatibility, and the excellent SuperBasic. The manual is a bit terse and MULTIDOS only checks a read/write four times.

NEWDOS80 has good utility programs like LMOffset, a disassembler, and Superzap. The manual, however, lacks a tutorial and NEWDOS80 needs automatic density recognition. DOSPLUS has Diskdump, Diskzap and automatic density recognition. LDOS has its excellent manual, great customer support, very flexible filtering of I/O, and a powerful JCL. Which you choose depends very much on your applications.

### The Drives

The Percom is the only one of these drives that I reviewed that is flippy, meaning that you can remove the disk from the drive and write to the other side of the disk. This effectively doubles the amount of storage you have in

your disk library.

The Percom drive also has the drive-cable card extending out from the rear of the machine, so it is easily accessible without removing the case, a nice feature. It has a stepping-motor speed that allows you to seek tracks at 12 milliseconds. This is faster than the Teac (at 30 ms) and my Pertecs (at 20 ms), but slower than the PMC and the Trak (both at 5 ms).

The PMC drive also has the drive-cable card extending out the rear of the drive. It has a fast-stepping motor with 5-ms track seeking. It allows writing only to one side of the disk.

The Teac drive was the slowest of the bunch, requiring a minimum of 30 ms to seek a track. In order to change the drive cable, you must remove the cover. The card for the drive is inconveniently spaced, meaning the drives must be slightly closer than recommended by Radio Shack when using their four-drive cable.

The Trak drive was another fast drive, requiring only 5 ms to seek a track. The card for the drive cable was even more inaccessible than the Teac drive, and to use it with the four-drive cable, you must run the drive with the cover removed, use the last connector on the cable, or use a disk-drive extender cable, which costs about \$10. It comes with a single-drive cable already connected to it, so single-drive users won't need to get a cable when purchasing this drive.

A nice feature of this drive is that the disks are spring-loaded. The disk springs out half an inch or so for easy removal when you open the drive door. This is also the only drive with a power-on

light as well as a drive-busy light.

All the drives worked well and deserve a recommendation. (Percom Data Co., 11220 Pagemill Road, Dallas, TX 75243; Doubler II, \$169.95. The Percom TFD-100-1 floppy drive has been replaced with the TFD-42, which sells for \$499. LN Doubler 5/8, LNW Research Corp., 2620 Walnut, Tustin, CA 92680; \$219.95 with DOS, \$199.95 without DOS. MULTIDOS, Cosmopolitan Electronics Corp., Box 234, Plymouth, MI 48170; \$79.95. NEWDOS80 2.0, Apparat Inc., 4401 S. Tamarac Way, Denver, CO 80237; \$149. DOSPLUS 3.5, Micro-System Software, 4301-18 Oak Circle, Boca Raton, FL 33431; \$149.95. LDOS 5.1, Logical Systems Inc., 11520 N. Port Washington Road, Mequon, WI 53092; \$129. PMC-SFD-51A, Personal Micro Computers Inc., 475 Ellis St., Mt. View, CA 94043; \$355.)

Bruce Powel Douglass  
December 1982

★ ★ ★  
**Micro Mainframe Disk  
Controller  
Micro Mainframe  
Model III**

At \$279.95, the Micro Mainframe Model III floppy-disk controller is about \$115 less than many other controllers listed in *80 Micro*. Micro Mainframe is not a well-known company, but \$115 is a substantial price difference, so I decided to order their unit.

The first difficulty was finding a dealer that stocked the kits. Micro Mainframe in California

put me in touch with three dealers. BT Enterprises of New York had them in stock at the same price. The unit was delivered one week after I placed the order.

My initial uncertainty about the MM controller kit proved to be unfounded. The circuit board is well made, and is obviously well designed. You get the controller board, a switching power supply, almost all mounting hardware, all cables needed for installing two drives, and instructions. Only the four 32 by 1/4 mounting screws for the drives are not supplied. I bought a Tandon drive from Software Sector Marketing; The Model III Service Manual contains service instructions for it.

The installation takes about an hour and a half. It's a very simple job. The instructions, unfortunately, are supplied on faded photocopies, difficult to read. There are at least five serious errors, and the diagrams frequently don't match the text. In spite of these errors, anyone with common sense can complete the installation. The drives fit well into the cutouts of the Model III, and nothing is force-fitted into place.

There is one ambiguity in the instructions. This is with configuring the disk drive itself. The diagram in the instructions implies that the socket on the Tandon drive into which the shunt is placed has seven positions. Actually, there are spaces for eight.

The system worked immediately. Data transfer to and from the drives seems to work well; I haven't encountered any disk errors. I've found one commercial program that refuses to

load, though it loads on a Tandy controller. The program uses tricky encoding techniques, however, and may not be a fair test. All the Radio Shack software I have loads perfectly.

There is one serious drawback. No service manual is available for the unit, and no schematic diagram. Service is available for a \$45 repair fee, but if you like to fix things yourself, you'll have to trace the circuit diagram on your own. Hopefully the company will change this troublesome policy as they gain experience.

*(Micro Mainframe, 11325 Sunrise, Gold Circle, Building E, Rancho Cordova, CA 95670; currently \$179.95.)*

Jerry O'Dell  
November 1982

### **Percom Doubler II** (See Page 283)

---

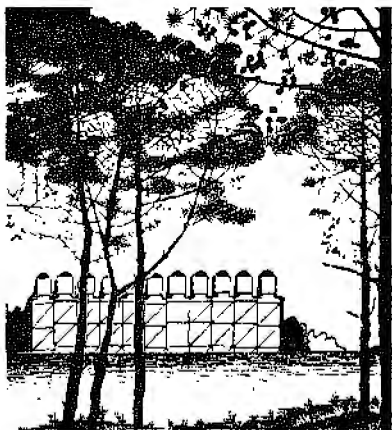
## EXPANSION MODIFICATIONS

★★★★

### **Disk-80** **Micromint Inc.** **Model I**

Disk-80, designed by Steve Garcia, is a disk-controller interface and memory expansion for the Model I. It includes the usual complement of expansion features: floppy-disk controller, 32K memory expansion, Centronics-type parallel printer port, and an interrupt-driven real-time clock. It is compatible with all TRS-80 software.

Disk-80 is not unique; in fact,



it is mundane. But uniqueness is rarely the criterion for Model I system expansion. Rather, transparent operation and high reliability are essential, and these qualities are admirably present in Disk-80.

The disk control system is a pleasure to use. Disk-80 incorporates a hardware data separator of standard design as standard equipment, and it works with the precision of the Percom add-on for the Radio Shack interface. Disk read errors were common to TRS-80s until designers went back to the drawing board and read Western Digital's recommendations for their own disk controller chip—use an external data separator. Disk-80 does this and again uses Schmitt triggers for signal reshaping on information incoming from disk drives. You can add the usual four drives with Disk-80.

The real-time clock also includes design upgrades, including CMOS integrated circuits to lower power consumption and noise and increase precision. The parallel printer port (incorporated in the power-supply board) is standard. The

power supply itself is adequate and concisely designed. The entire expansion unit is packaged in a small plastic case less than half the size of the Radio Shack expansion system.

Excellent documentation and full schematics are provided with Disk-80, and the construction article (*Byte*, March 1981) describes details of the circuit operation and unveils many of the mysteries of expansion-interface operation.

Reviewing Disk-80 is almost incongruous, because any comments can be summarized with the sentence, "It works."  
(*Micromint Inc.*, 917 Midway,  
Woodmere, NY 11598; \$275.)

Dennis Bathory Kitz  
March 1982

★ ★ ★ ★

### **The Internal Memory Holmes Engineering Model I**

Occasionally a product comes along that you could consider the greatest since the microcomputer. This time it is the Internal Memory (or the IM) from Holmes Engineering, an attachment for your TRS-80 Model I.

The IM is a small circuit board that contains 16 RAM sockets (plus the necessary decoder circuits), and is plugged into the eight RAM sockets in your keyboard. The original eight RAMs (they must be 16K) are then plugged into the IM. This leaves you eight more sockets for either 4K or 16K RAMs.

I did have a minor problem installing the board, as capacitors C32 and C34 were in the way. I merely moved them to the other side of the CPU board and, voila,

## HARDWARE

---

32K of user RAM (actually 31956 bytes). Larry Holmes has indicated that he is aware of the problem with the capacitors, and is already planning a slight redesign so they will not present further installation problems.

The IM also requires four connections to the CPU board, of which only three are required if you are adding 16K. These are logic connections, as the IM acquires its power from the RAM sockets. Holmes Engineering has come up with some miniature clips that are small enough to attach to the pins of the other ICs on the CPU board. Thus, there is no need to solder or cut traces, and the IM is easily removed when service is required.

The IM is strictly a hardware modification and does not require software work. Just install it and you are ready to go. The Level II ROM is capable of addressing up to 48K of RAM, plus the 16K for the Level II ROM, plus peripherals (CRT, keyboard, etc.). This is a total of 64K for a Z80 system. Holmes Engineering provides total support for the IM and a one-year guarantee. Those of you who have limited space and funding; the Internal Memory is for you.

*(The revised IM II is sold by Holmes Engineering, 3555 South, 3200 West, Salt Lake City, UT 84119; \$119.50 for 16K, \$139.50 for 32K.)*

Eric Keener  
December 1981

★ ★ ★ 1/2

**LNW Expansion Board**  
**LNW Research Corp.**  
**LNW80**

The LNW Research Expansion

Board has provisions for up to 32K extra memory, floppy-disk controller, parallel printer, dual-cassette port, real-time clock, RS-232 port, on-board power supply, and a bus for expansion. All features are optional. The manual includes itemized parts and lists for each option, enabling you to build only the sections you want.

The manual is well written, and it gives details for construction, configuring the system, and the theory of operation. Data sheets for the UART and the disk controller (1771) are included with software for using the 80 as a terminal. Software is also provided to drive a serial printer through the RS-232 port or through the parallel port, with certain board modifications.

The first job is to look for the necessary parts. Although the manual does mention parts substitutions, it doesn't cover the UART, a Western Digital TR1602B. I have used both an AY-5-1013A and an AY-3-1014A with fine results.

The RS-232 port is well provided with handshaking, and requests to send (RTS), clear to send (CTS), data terminal ready (DTR), data set ready (DSR), carrier detect (CD), and ring indicator (RI) are all implemented, as well as a provision for generating a true break signal. In addition, the serial data appears at 1 bit (IN OE8H, bit 1), and the port can be wired for 20mA operation if desired.

Although the TRS-80 can be set up as either a data terminal (DTE) or data communication device (DCE), the handshaking lines are fully implemented only for DTE. There is no provision for the UART to generate inter-

rups, nor is the interrupt line easily available to those who have time-critical applications.

Interrupts are definitely usable on the TRS-80, since the vectors are stored in RAM. Many of the ROM routines depend on the interrupt vectors, and you may have to write your own keyboard and screen drivers depending on what interrupt mode you use.

A few words of warning are in order. The bus expansion provided should be used with due caution. It is a simple, unbuffered extension of the TRS bus, and any expansion requires more buffering with high impedance input buffers (not 7400 ICs).

Using disks, you can patch machine-language programs to correct errors, and reassemble with ease. Data files are child's play, and long programs load in seconds.

The LNW Research board is well laid out and inexpensive. The documentation is satisfactory. With disk drives, 48K of memory, and Electric Pencil with lowercase modification, it makes a good machine. (LNW Research, 2620 Walnut, Tustin, CA 92680; \$69.95, hard-to-find parts kit \$27.50.)

Ian Hodgson  
May 1981

★ ★ ★<sup>1/2</sup>

**LNW System Expansion Board**  
**LNW Research Corp.**  
**Model I**

LNW's System Expansion Board claims to have all the features of the Radio Shack expansion chassis and then some.

Also mentioned is software compatibility and "quiet bus" design that eliminates some of the memory problems. I spent some time comparing LNW's circuitry with that of Radio Shack. Anyone building the LNW board should purchase the expansion interface hardware from Radio Shack.

The manual has a good parts list. Actually, the parts are listed in several ways. The system allows partial construction for those who don't need all the features. Parts are sorted alphanumerically to make it easy to order and inventory.

The manual suggests buying another Radio Shack power module similar to the one that powers the keyboard. Check at your local Radio Shack to see if they have a bad one laying around. Pry it apart and replace the fuse. A schematic is included in the construction manual. I bought separate transformers and mounted them internally with a power-line filter.

First I mounted the board, the power transformers, and cut the slots for the ribbon connectors. Next, I assembled the board per the guidelines of the manual. I used sockets for all chips. The pins on the sockets were tapered so that they barely extended through the board. This made inspection of soldering hard, but I managed by going slower than I normally would. There was nothing else tricky about assembling the board.

While the manual directs you to plug in the chips prior to preliminary power checks, certain jumpers are not installed. With these removed, no power is applied to the chips. Follow the directions exactly. This section

performs voltage checks to find errors or bad parts that could cause damage. I installed the jumpers, turned on the chassis, and rechecked the power-supply voltages. I then removed power and connected to my keyboard.

There are two 40-pin edge connectors on the LNW board. Either one can be connected to the TRS-80. The other is available for devices that work directly off the expansion port, such as a page printer. The bus termination resistors are installed near one of these connectors; I installed them and have experienced no problems.

The real-time clock is only of limited use without a disk operating system. It has proven useful in disk operation for timing program events in Basic programs.

Cassette operation is slow, but dual cassettes open up a new world for data handling if you are interested in editing data files. The LNW board has the decoder and relay driver. All that is required in addition is an external relay.

The serial port uses a common chip called a UART (universal asynchronous receiver/transmitter). This chip is almost a magic device. You inject parallel data, and out comes serial data, or input serial data and out comes parallel data. The chip can do both jobs simultaneously even with unrelated data of different baud rates. I like this serial port so well that I wish the LNW board had two of them. It would be nice since it is possible that one might want one port for a line printer and one for a modem.

The line printer port, like every other LNW feature, works

like Radio Shack's and is designed to work with a parallel printer, using the Centronics interface. For those using a serial printer, the serial interface can drive it. With no modifications to the printer or the LNW board, you can set up the serial interface for the right baud rate and punch in a software routine that will link the LLISTs and LPRINTs to the serial interface. The driver software program is included in the manual.

A better way is to follow the instructions of Chapter 6, which route signals that would normally go out the normal parallel printer port through the serial port. A still better approach would be to build a separate parallel to serial converter, which would plug into the printer port and convert to the type of serial level required by the printer.

The TRS-80 generates a lot of energy in the radio-frequency spectrum. It seems to be the worst in the 40-meter region. It seems that every accessory added causes the noise level to come up. I have done nothing to correct the radiation caused by the TRS-80, but I do not observe any increase in noise when the expansion interface is connected or disconnected.

The LNW board performs. It is totally hardware and software compatible with Radio Shack products designed for the TRS-80. While the manual is good, the project is such that I recommend it only for the advanced kit builder.

*(System Expansion is sold by LNW Research, 2620 Walnut, Tustin, CA 92680; \$69.95.)*

Bob Couger  
February 1981



★ ★ ★ ★

### **LX-80 Expansion Interface SA800 Eight-Inch Disk Drive Lobo Drives International Model I**

Though Lobo Drives' LDOS may already be a familiar name to disk operators, the equipment for which it was designed—the LX-80 expansion interface and the SA800 dual eight-inch disk drive—remain unexplored.

Why bother with a more expensive alternative to the Radio Shack expansion interface and drives? First, an unmodified Radio Shack interface won't accommodate eight-inch drives, whereas the Lobo will. Also, since production of the Model I has ended, it's anyone's guess as to how long the attendant interface equipment will remain available. Despite this threat the Model I is thriving; manufacturers of ancillary gear are not merely continuing to support the hardware, but in some cases are offering superior alternatives.

I've been operating the Lobo equipment on my system for over two months without a single glitch. This reliability, reflected in the stiffer price, has also been built into the equipment; instead of a plastic case, both interface and drive units are enclosed in one-eighth-inch thick steel. Not that many users are likely to try, but you could quite literally drive a car over the interface without damaging it.

The unit measures just under three inches high (lower than the RS expansion interface) by 19 inches wide by 12 inches deep. The LX-80 comes with a user manual that not only describes set-up procedures but offers sim-

ple step-by-step diagrams. An owner who dives in without reading can plug in either the five- or eight-inch drive connector cables the wrong way (neither the multi-pin plug nor the socket have a keying notch).

The manual also tells you how to change the LX-80's parameters to fit your needs, from using either 117V ac power at 60 Hz or 235V ac at 50 Hz to opening the box and installing up to 32K of your own RAM. The only criticism I have of the manual is that there is no circuit schematic. I don't care how secret Lobo's design is; if someone wants this sort of quality and is willing to pay this sort of price, he should have the option to maintain his own equipment. This is obviously impossible without the circuit diagram.

If you want to use four five-inch or four eight-inch drives, double or single density, single- or double-sided, or any of the above in any combination, the LX-80 will support it. Not only that, but it's possible to configure the interface to boot up from either five- or eight-inch drives, or from Lobo's 1850T dual fixed/floppy hard disk. The point is merely to tell the interface which is drive zero, and this is done by setting small DIP switches at the rear of the unit.

I need to add one small caveat to this glorious mixing of drives. The LX-80 doesn't map its disk input/output in the same way as the Radio Shack expansion interface, addressing the drives via I/O ports instead of memory-mapping them. The result is that the few pieces of TRS-80 software that don't perform disk I/O through the disk operating system, such as Super Utility

## HARDWARE

---

and certain adventure games, will not work.

This is the reason Lobo needed to have a DOS developed especially for their hardware. All Basic programs and machine-code programs will work fine, as they access disk files via the operating system, in this case, LDOS.

An interesting feature of the Lobo setup is the ability to override the keyboard ROM. A switch at the rear of the unit replaces the Level II Basic ROM with an alternate set that can be plugged into three sockets inside. Numerous sorts of ROM can be added by reconfiguring a set of jumper wires near the alternate sockets. This should allow the whole TRS-80 to operate with any dedicated application in mind. Possibilities include a Pascal or Pilot that can be available on power-up, or any number of industrial or mechanical applications.

The LX-80's real-time clock and expansion port work in the same manner as the Radio Shack interface parts and a Centronics-type parallel printer port is memory-mapped to the same address, so printing is unaffected. Two serial output ports will drive a serial printer; they can be configured by the user for custom I/O. Baud rates can be set from software, and range from 12.5 baud to 316.8 kilobaud. That should cover just about every possible serial device imaginable.

Access couldn't be simpler to the 10.5- by 14-inch printed circuit board for adding RAM or changing jumpers. The board is rigid thick glass fiber held down by five screws which should eliminate any flexing problems.

Though the LX-80 steals the show, the dual SA800 eight-inch single-sided disk drive offers over one megabyte of storage in double density. The units are tried and trusted Shugart SA800 soft sector drives which boast a head life of 15,000 hours and a disk life of a phenomenal 3.5 million passes per track. The 17.5- by 22 by 4.5-inch cabinet uses the same hefty construction techniques as the interface; Lobo installed a cooling fan at the back, since eight-inch drive motors are constantly running, whether or not the drive is selected. Unfortunately, for my taste, the fan was noisy to the point of distraction.

The drives themselves function so perfectly it's almost boring. I haven't noticed a soft error in two months.

Though the Lobo expansion interface offers much greater versatility than others currently available, their drives just outperform, or are equal to, others now on the market. However, both units are of the highest quality and seem to be aimed at the professional microcomputer user.

Okay, Lobo International, when are you going to come out with a TRS-80 compatible keyboard unit?

*(Lobo Drives International, 354 South Fairview, Goleta, CA 93117; LX-80 interface \$460, complete with smallDOS \$510. LDOS is an additional \$129. The RS-232 option is an additional \$100. The SA-800 drives are now available for the Model I and III and cost \$1225. The SA-850 drives cost \$1725.)*

Jake Commander  
July 1981

★ ★ ★ 1/2

**MDX-2 PC Board & User  
Manual  
Micro-Design  
The MDX-2 Complete Kit  
of Parts  
Computex  
Model I**

If a direct-connect mode, 4K EPROM, serial port, plus all the features of the Radio Shack Expansion Interface sound interesting, consider the Micro-Design alternative.

The interface is available either as a kit or assembled. The kit is not intended for novice assemblers; knowledge about electrical components and their markings is needed as well as experience in soldering techniques. You can purchase only the PC board and manual from Micro-Design and individually buy the parts, or buy the parts in module form from Computex and build only particular features.

The interface is completely software compatible with the Radio Shack interface. The added features are not available on any other expansion board currently on the market, though the advertising is a little misleading. The RS-232 port must be used for modem operation and is unavailable for other use. The dual cassette port is nothing more than an output telling when the cassette drive has been accessed. Additional external circuitry is required for proper operation. Finally, the direct-connect modem needs an off-board coupling transformer and switch to function properly.

The kit goes together fairly well. There are a few minor errors in the instructions and

schematic, nothing that a little common sense wouldn't fix. There does not seem to be any rhyme or reason to the component labels silk-screened on the board; R1 might be found on one side of the board next to R53. With a little searching all components can be found.

The edge connectors, like those on the Radio Shack interface, are not gold-plated. Only time will tell if this deficiency will cause a reliability problem.

Two resistor packs are required in the board layout, neither provided in the Computex kit. Instead, instructions for building your own are given; no headers are supplied to facilitate the task. Another inconvenience is the lack of Radio Shack power packs; a transformer assembly of the power cord, external diodes, and wiring has to be constructed.

The printed circuit board layout is well designed, incorporating several features to make the kit nicer to build and use. The use of DIP switches for programming baud rates is a plus, as is the inclusion of the originate/answer switch on board, along with a DB25 connector.

To fit the modem on board, components have to be stood on end, making assembly awkward. There is not enough room for the coupling transformer, either. It would be nice if the on-board power regulators could provide enough power for at least one disk drive. Perhaps the next generation board will include a double-density disk controller and 1,200-baud modem capacity with auto answer and dial. Even without these features, the MDX-2 is impressive.

The manual is severely deficient if the board does not work the first time. A 24-hour hotline takes messages for technical questions; technical assistance was available after 5 p.m. Texas time. They were helpful when I called.

The line printer, modem, and memory all function flawlessly. Your printer must have an auto line feed on receipt of a carriage return to work properly; this has nothing to do with the interface, but is a result of the Radio Shack software. A better data separator circuit should have been included; the disk controller has a tendency for read and parity errors on the inner tracks.

The Micro-Design expansion interface alternative is well worth the money. But be prepared to spend some time building and troubleshooting the board.

*(Micro-Design, 6301 Manchaca Road, Manchaca, TX 78652.*

*The MDX-2 board sells for \$74.95, completely assembled for \$469.95. Computex, 17321 El Camino Real, Houston, TX 77058; complete kit \$269)*

Ronald Cangro  
April 1982

★ ★ ★ 1/2

### **MDX-3 Interface PC Board Micro-Design Model III**

Many companies offer whole or partial kits to upgrade your Model III for floppy disks. These kits usually include one disk drive, an additional 32K of RAM, a floppy-disk controller interface, and a power supply for the floppy disk. Micro-Design's MDX-3

interface board not only has these standard items incorporated, but an RS-232 and a modem as well.

The MDX-3 is a bare printed circuit board that you populate with the required components. When fully constructed, the board's options include the RS-232, floppy-disk controller, and a 300-baud answer or originate modem.

The MDX-3 fits inside the Model III's case in the same place that is normally occupied by the RS-232 and Radio Shack's floppy-disk controller card. All mounting hardware for the installation is included with the board. The modem requires the RS-232 for operation and cannot be constructed as a stand-alone section. However, the floppy-disk controller or RS-232 can.

If you can solder neatly and follow instructions carefully, you'll have little trouble constructing and installing the MDX-3 interface board. Should you encounter problems, Micro-Design seems more than willing to help. They have technicians available for this purpose after business hours and also have service centers should the problem be unresolved.

The cost of constructing the interface and buying the disk drives yourself is very low. Combine this with the fact that the interface also contains an RS-232 and a modem and you have an extremely low-cost system. It is this, plus the



support of Micro-Design, that makes the MDX-3 a worthwhile investment.

*(Micro-Design, 6301 Manchaca Road, Austin, TX 78652; \$74.95.)*

Mel Patrick  
October 1982

★ ★ ★

### **The Memory Box Displayed Video Model I**

The Memory Box consists of a small plastic box with a TRS-80 standard 40-pin edge connector emerging from a slot in one side, and a plug-in 5V power supply permanently attached by a five-foot cord. The power supply has a switch that permits operation on either 110V or 220V circuits.

Displayed Video provides a short cable that has a 40-pin edge plug attached to each end. In normal use one end of the plug is connected to the edge connector on the Memory Box, while the other is connected to the expansion edge connector on the rear of the TRS-80 keyboard unit.

The box provides 1K of additional RAM in memory locations 3000H-33FFH (12288 decimal-13311 decimal). Because the TRS-80 (and most programs) do not expect any memory to be present at these locations, anything stored in this area is not likely to be overwritten or changed. The Memory Box retains the contents of its memory even if the power to the TRS-80 keyboard unit is switched off.

The Memory Box can also be used to store variables for use by more than one Basic program, as

in a case where one program chains to another. Basic PEEK, POKE, or machine-language USR functions can be used to move the values of certain variables to or from the Memory Box.

I do not use the cable supplied with the Memory Box. I purchased a 40-pin plug from Radio Shack (catalog number 276-1558), and using a bench vise, I pressed it onto the TC-8 cable about three inches from one end. The result was two plugs near one end of the cable, and one plug at the other end. Now I can plug the end with one plug into the back of the TRS-80 keyboard unit, and the plugs on the other end connect to the TC-8 and to the Memory Box.

Once connected, my Memory Box has never failed to work reliably, even with a high-speed modification (200 percent of normal speed) on my TRS-80.

### **Improvements**

Occasionally a machine-language program goes bonkers and writes random bytes into various places in memory. For this reason, the memory Box should have a write-protect switch—flip it on and the memory inside the Memory Box would suddenly look like ROM to the CPU.

Any writes to addresses contained within the Memory Box would be ignored, so that a loss of power would be about the only thing that could destroy the memory contents. With the switch in the opposite position, normal operation (read or write allowed) is resumed.

Displayed Video warns against operating the CPU with the power disconnected from the

Memory Box, as this causes abnormal bus loading and reduces the reliability of the TRS-80 system. Therefore, disconnect the Memory Box from the computer if you have to pull its plug.

The Memory Box uses the same area of memory as the PROM operating system of the Exatron Stringy-Floppy. Stringy-Floppy users would not be able to use the Memory Box while the Stringy-Floppy is connected. I do not know of any other major add-on devices for the TRS-80 that use this area of memory, so unless you own a Stringy-Floppy, you should find the Memory Box a useful addition to your Model I TRS-80.

*(Displayed Video, III Marshall St., Litchfield, MI 49752. The Memory Box is no longer available.)*

Jack Decker  
May 1982

★ ★ ★

### **MM+ (Memory Expansion) Exatron Model I**

If you do a lot of programming on a 16K Model I you rapidly learn what an OM error means; programs always seem to grow and fill or exceed available memory. Most of you turn from a cassette-based system to disks or a Stringy Floppy. In either case, more memory and/or a floppy controller is needed. Exatron's MM+ (Memory+Interface) is a quality alternative to the Radio Shack expansion box and could cost you less depending on your situation.

The unit is designed to fit under the TRS-80 monitor and comes fully assembled. Standard

features are: 32K of memory, built-in power supply, RS-232C serial printer port, real-time clock, light pen port, parallel printer port (Radio Shack/Cenronics compatible), and a general parallel port (IBM Model 50 compatible). A floppy interface was not included since the unit was designed for TRS-80 owners who need more memory but do not own a floppy disk; a floppy controller and an additional 32K will be the first options available on a second circuit board which will fit in the present chassis.

The light pen port is a unique feature of the MM+. Most light pens can be used with a cassette recorder serving as an amplifier, but the MM+ port is more convenient and leaves the recorder free. The port was designed for MicroMatrix's Photopoint, but should work with any light pen that is designed to connect to a recorder.

Exatron offers a speed-up kit for the TRS-80 that lets you run the computer at a 50 percent or 100 percent increase in speed. The MM+ is guaranteed to handle the 100-percent increase if your CPU board and memory will run at the 3.55 MHz frequency.

If you have no immediate need for a floppy interface and want more memory, this unit deserves your consideration. I have used mine for several months with no problem, even after overnight memory tests. It works well with disk alternatives such as the Beta-80 and the Stringy Floppy. It should satisfy your memory requirements for the time and give you many additional useful features.

*(Exatron, 181 Commercial St.,*

*Sunnyvale, CA 94086. MM+ is no longer available.)*

Harley Dyk  
July 1981

★ ★ ★ ★

**MT-32 Printer/Interface  
Module  
Microtek Inc.  
Model I**

Until Microtek introduced their MT-32 Printer/Memory Module, there was no way of adding just additional memory and a parallel printer interface to your system short of building one yourself. The MT-32 provides both of these features for less than \$125 in its most basic form. Like Radio Shack's interface, it sits under the video monitor so it takes up no additional desk space.

The documentation and instructions are clear. Memory installation couldn't have been made any easier. All that is required is to remove two screws on the back of the module, slide the cover off, insert the chips, slide the cover back on, and replace the screws.

The only disadvantage that I could find was that the MT-32 has no extension of the CPU's bus for connection to other peripherals. Fortunately, this problem is easily solved by the installation of any of the commercially available "2 for 1" bus splitting cables between the MT-32 and the keyboard/CPU unit.

One of the major advantages of this interface is that since it has no disk controller, pressing the reset button to stop the cassette recorder or the printer will not result in a hangup or

return to the memory-size prompt, but will return to the ready prompt as it would without the interface. Another great advantage is that the interface draws its power from the keyboard's power supply, so that another cord needn't be added to the already impossible tangle of cords behind the computer.

The MT-32 is a great, low-cost alternative to the \$300 plus price of the Radio Shack Expansion Interface. I have experienced no problems with it in the three months that I have been using it. Even if you don't want a printer but need the extra memory, the MT-32 is the lowest-cost product for memory expansion.

*(MT-32 Printer/Interface Module, sold by Microtek Inc., 9514 Chesapeake, San Diego, CA 92123 is no longer being produced.)*

Fritz Milhaupt  
June 1981

★ ★ ★ 1/2

**RS-232C Board  
Tandy/Radio Shack  
Model I**

The RS-232C board from Radio Shack gives the user a standard way of connecting a variety of peripheral devices, including a serial printer. The board installs under the large access cover in your existing expansion interface and comes with a ribbon cable, instruction manual, and machine-language program that allows your TRS-80 to emulate a standard asynchronous terminal (provided a modem, acoustical coupler, or direct connect is available). The board plugs into a 42-pin

connector that is part of the expansion interface.

The early version of the instruction manual has a number of errors (although the quality of the rest of the feature is excellent). For example, the DIP switches are labeled on the board S1-S8), but the user isn't told in which direction to push a DIP switch to open or close the circuit.

To open, push each switch in the direction of the UART (large IC chip). Most of the remaining errors are minor and all remaining implementation instructions are clear.

*(Tandy/Radio Shack, Fort Worth, TX 76102; \$99.)*

Roger L. Hicks  
March 1980

★ ★ ★ 1/2  
**TRS-80 Serial I/O Board**  
**Electronic Systems**  
**Model I**

As an alternative to an expansion interface and an RS-232 board, I purchased the TRS-80 Serial I/O board without parts. You can buy a complete kit (without power supply or case) or the assembled unit (no power supply or case). Cables and connectors to the TRS-80 expansion bus and the modem of your choice are extra. Even with the extras the cost is only about \$150, and the darn thing works!

There are some sacrifices you will be making for the sake of saving some money. The TRS-80 Serial I/O will not work with any Radio Shack terminal software. The TRS-80 I/O board uses port 37H for input and address 37F8H for output.

Although the assembly instructions are brief, anyone with some knowledge of soldering can assemble it in an evening. Once I set my unit up, I never had to go back and readjust anything.

The TRS-80 I/O board comes with a machine-language line-printer routine, terminal program and a routine to send Basic programs to another computer. The latter program didn't work, but I figure two out of three isn't bad. I assembled the terminal program using Radio Shack's Editor/Assembler and made my first call to another computer within minutes without a hitch!

The Electronic Systems TRS-80 Serial I/O board is great for those who have time to tinker and are also short on cash. I have not yet touched on the product's usefulness as a serial printer interfacing device, which may be your reason for purchasing it. But I hope you find it as useful and educational as I have.

*(The TRS-80 Serial I/O Board is sold by Electronic Systems, San Jose, CA. They could not be reached for an update.)*

Jim Cambron  
February 1981

★ ★ ★  
**Wolfbug 64K Color Computer Upgrade**  
**Atomic City Electronics**  
**Color Computer**

Upgrading your 4K, 16K, or 32K Color Computer to 64K is relatively easy when using Atomic City Electronics' 64K upgrade. The upgrade consists



of two major components: a monitor called Wolfbug, which has 12 single-key commands, and a 64K RAM adapter card that reconfigures the old RAM socket to accept the new 4164 RAM. The uniqueness of this upgrade is its simplicity. Traces don't have to be cut; just solder in a couple of wires for power, address lines, and some jumpers, and you have 64K of usable RAM.

The instructions, which are very thorough, consists of two parts. One is for the EPROM adapter card, which holds Wolfbug in a 2716 EPROM, and the other is for the 64K RAM adapter card.

Wolfbug is a machine-language monitor that resides at SF800 to SFFEA. It features 12 single-key commands that include the following: an ASCII dump of any memory location, a go-to-a-machine-language program command, a floppy disk boot, a command to route the output to the display, a hex dump of memory, a command to transfer Basic to RAM in the 64K mode and run it, permits typing text to the screen, and a command that switches you to MAP-1 mode.

Wolfbug is an excellent upgrade from its installation to fine documentation. You should have no problems with it at all.

*(Atomic City Electronics, 3195 Arizona Ave., Los Alamos, NM 87544; Wolfbug with adapter card \$55, 64K RAM card without memory \$44.95, 64K RAM card with memory \$144.95.)*

David L. Wasler  
December 1982

---

## MODEMS

---

★★★★

**Hayes Stack Smartmodem  
Hayes Microcomputer  
Products Inc.**

★★★★

**Automated Communications  
Exchange  
Ace Computer Products  
Model I or III**

What makes the Smartmodem think it's so smart? It's the special Z80 processor, programmable in any language by ASCII character strings. The command buffer holds 40 characters that specify the modem's program.

Eight DIP switches define the default parameters for the modem upon turning it on, but can be overridden by the modem's programming. These capabilities open up some convenient options, of which Ace Computer Products of Florida took full advantage in their Automated Communication Exchange (ACE) software package.

The Smartmodem has a two-year limited warranty. It is a direct-connect modem, which means that it is "cleaner" than other acoustical modems that require transmission through a microphone, and it is fully approved by the FCC. It requires an RS-232C cable and interfacing software and has an adjustable rate of 0-300 baud. The Ace software requires 48K and one disk drive.

The Smartmodem documentation is well-written and includes helpful diagrams. Setting up the modem should be a breeze for

even the greenest electronic communicator. Simply plug the modular phone cord into the back of the modem, connect the RS-232C cable to the modem, and you're in business.

Your telephone or other Hayes Stack hardware fits comfortably on top of the Smartmodem, merging compactness with attractiveness. The front of the Smartmodem sports seven LED indicators to show the state of the modem at any time. These include modem ready, send data, carrier detected, off hook, and others.

The feature that adds the most flexibility to the system is its programmability. With the right software you can program the modem to dial or answer and send or receive ASCII files at some future time, even when you won't be there! One sin of omission is the inability to connect your phone to the modem in series.

The Smartmodem is a great device, and we have had no

problems with it. It is simple to program (much easier even than Basic), easy to use, and comes with good documentation. We highly recommend it for reliable and simple, yet sophisticated data communication.

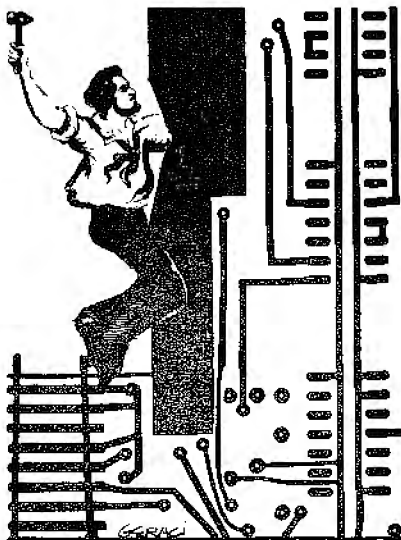
The Automated Communications Exchange (ACE) was designed exclusively for the Hayes Stack Smartmodem. ACE is a collection of interconnected programs designed to give you communication ease and power when used with the Smartmodem.

ACE allows some unprecedented clout for communication across the phone lines. It can look up a phone number, dial it, log on to the system, upload ASCII files, download ASCII files, log off, wait until another day and time and call another system, download a few files, and so on—all from a completely unattended TRS-80!

The author, Mike Moore, thoughtfully included JCL (also known as DO) files, for those DOSes that support it, to let you easily boot up ACE.

The documentation for ACE contains all the information needed and is reasonably well-written. Once you have played with the program and understand its operation, ACE is a joy to work with. It is simple to operate and control flows smoothly among programs when necessary with a minimum of user interference.

The ACE software is excellent overall and takes full advantage of the Hayes Stack Smartmodem's unique characteristics. The Smartmodem and Automated Communications Exchange software together make an ex-



cellent communications system. (*Hayes Stack Smartmodem is available from Hayes Microcomputer Products Inc., 5835 Peachtree Corners East, Norcross, GA 30092; \$279. Automated Communications Exchange is from Ace Computer Products, 1640 N.W. 3rd St., Deerfield Beach, FL 33441; \$79 alone, \$39 with purchase of Smartmodem.*)

Bruce Powel Douglass and  
Doris Christine Minnerath  
October 1982

★ ★ ★ 1/2

**The MicroConnection  
The MicroPeripheral Corp.  
Model I and III**

Dial-up systems for the computer hobbyist are relatively new. Such a system used to require an expansion interface, the optional RS-232 board, and then an expensive acoustically-coupled modem. This adds up to at least \$650 just to get on line.

The MicroConnection is an FCC-accepted modem that connects directly to the phone line and does not require an expansion interface or any sort of RS-232 interface. For that matter, it has its own RS-232 port for driving a serial printer or operating as a stand-alone modem between a serial terminal and the phone lines.

The modem operates at 300 baud, but can be converted to run at 110 baud through a simple hardware modification described in the owner's manual. The word protocol is under software control; thus, you set the MicroConnection to operate with even, odd, or no

parity, a 5-, 6-, 7-, or 8-bit word, and 1-, 1.5-, or 2-stop bits. (The total of the parity bits plus the word bits can only equal eight, as I found when I tried to transmit an 8-bit word with even parity.) The unit also provides an input and output to be used with your amateur radio equipment for ASCII Bell 103 standard teletype.

A dumb terminal program, supplied on cassette, provides redefined keys to transmit escape, left and right brace and bracket, back slash and apostrophe, wave, and vertical broken symbols; the up-arrow works as a control key so you can transmit any other control code. The manual provides addresses if you wish to redefine the special characters.

The program also provides for printing the screen on a parallel printer. The command mode is accessed using a shift up-arrow (P turns on the printer, S turns it off, E returns you to Memory Size?, and I returns you to the initialization routine for setting half or full duplex).

The MicroConnection provides an easy and efficient method of checking into the various bulletin boards (Forum-80, CBBS, and so on) as well as CompuServe and The Source. The MicroPeripheral Corporation has a whole line of smart terminal programs as well as MicroConnections for other models. It is well worth the investment to get into this exciting and interesting facet of computing.

*(The MicroPeripheral Corp., 2643 151 Place N.E., Redmond, WA 98052; The MicroConnection is available in two models. The Model II sells for \$209 and*

*the Model TI-A sells for \$259.)*

Eric Keener  
July 1981

★ ★ ★ ★

**Modem I**  
**Tandy/Radio Shack**  
**Model I and III**

The Modem I, Radio Shack's first direct-connect modem, has some great advantages over its competition. Most important to small computer owners, the Modem I can work from the cassette port without an RS-232C. You don't need an expansion interface and, since the computer acts as a dumb terminal, there are no minimum memory requirements. If you have the E.I. and RS-232C you can go that route too.

The modem uses a modular phone jack to connect to the phone lines and features both answer and originate modes. My modem has worked 100 percent of the time. If you have a party line or multi-line phones, you may, however, experience some interference in the form of misprinted screen text.

A warning seems appropriate here. Be sure that the Modem I is compatible with the system you wish to hook up to; RS-232C compatibility does not guarantee modem compatibility.

If you have an RS-232C then you have all the cables and software you need. If you have a cassette system, you must buy a cable to connect your keyboard to your modem.

I have one minor complaint. The original advertisements stated that the Modem I would handle a baud rate of 600. Actually, the Modem I has a baud capability of 0-300.

*(The Direct-Connect Modem I is available from Tandy/Radio Shack, Fort Worth, TX 76102, for \$149.)*

Richard C. McGarvey  
January 1982

★ ★ ★ ★

**Modem 80**  
**ICROM Enterprises Ltd.**  
**Model I and III**

The increasing number of computer communication networks such as CompuServe, The Source, MicroNet and the standard CBBS (Computer Bulletin Board Service) have made modems and terminal programs popular with computerists. After looking at the software support that several setups had to offer, I chose the Modem 80.

The Modem 80, direct-connect modem, is a complete unit and includes all cables, instruction manuals, and, most important, a tape containing six programs for software support.

The modem attaches to either the expansion interface or to the back of the Model I keyboard's extension bus. The unit is also compatible with the Model III (Model I uses a 40-pin bus, Model III a 50-pin bus).

The unit measures 8 by 3½ by 1½ inches. On one end are two modular phone jacks for connecting to the phone line and the telephone, although the latter is not required for operation. The opposite end houses a ribbon cable and an edge-card connector. A special Y power cable extends from this end as well. One end of this cable connects to the power socket on the computer, the other connects to the jack on the other end of the power pack. Two LEDs on the top of the unit

indicate on-line and carrier-detect conditions.

Two pages of instructions are included for initial setup and a self test to ensure proper connection. Two manuals also accompany the Modem 80. One describes operation from Basic, and the other describes all the command functions in the terminal mode.

### The Software

The Modem 80's strongest point is its software. Four of the programs supplied on tape are for Basic operation between two Modem 80-equipped systems. Two of these programs are in machine language for the actual modem input and output. The other two programs are host programs which work in conjunction with the machine-language programs to support either tape or disk.

The last two programs on the tape are machine-language terminal programs. These smart terminal programs allow users to access computer information networks.

There are 32 command functions in the terminal mode. Capabilities include: eight programmable function keys; an uppercase/lowercase switch; baud rate of 25 to 300; no limit on the number of digits you can dial; redial-phone-number feature; set parity at even, odd or no parity; full or half duplex; write/load buffer command; transmit buffer command; save/load function keys; scroll buffer command; line-printer buffer command; and place-markers-in-the-buffer command.

### Hardware

The Modem 80 is designed

around the Motorola modem IC (MC14412). While this chip is capable of either answer or originate modes, the Modem 80 can only send in the originate mode, but can receive in either mode. Any mode switching is done by setting a bit in a control word and then sending it to the Modem 80's port (port 120). The remaining ICs on the board are for decoding input/output port control words and the active filters to respond to only the other modem's signal transmitted over the phone line. A filtered power supply is also on board for power requirements.

Modem 80 has proved to me that reliable communication between systems is possible for a low initial cost.

*(ICROM Enterprises Ltd., 1240 Bay St., Suite 205, Toronto, Ontario, Canada M5R 2A7; \$289.)*

McI Patrick  
June/July 1982

## —MUSIC AND VOICE— SYNTHESIZERS

### ★ ★ ★ Cognivox Voicetek Model I

I recently decided to try a voice I/O peripheral called Cognivox. The advertisement, as well as the instruction manual, said the Cognivox was fully assembled and contained all necessary equipment to "plug in and use." This is not totally true—lacking in the package (but available at extra cost) are

## HARDWARE

---

the ribbon cable and 40-pin edge connectors needed to hook up the unit. Other than that one point, Cognivox is ready to use on arrival.

The Cognivox comes with a power supply, one microphone, and a cassette containing the software to use Cognivox. The software is designed for a Model I, Level II computer with at least 16K. The operating software is very small but the memory requirements for the digitized voice data are so great (approx. 1.5K per second of speech) that a 16K machine has less than 4K left for a Basic program. A 16K machine will hold approximately 11 words or phrases unless they are very short: word or phrase length is limited to a maximum of three seconds duration. Fortunately, three seconds is a long time in speech, so 32K can store a usable vocabulary and 16K can be functional though moderate.

The software package contains a driver called VOX2, and two demos called PROG2 and Dialog. Also included are some games, a music demo and a program called VDUMP that will give a verbal output of a memory dump in hexadecimal.

It is obvious that for \$149 you are not going to get high-fidelity voice output from Cognivox. By comparison, the voice output is a little less accurate than a well programmed synthesizer, such as the Radio Shack model. However, Cognivox does not require phoneme programming and is therefore easier to use. Also it responds with and to your accent, not that of a machine-sounding synthesizer.

Training Cognivox is a three-pass operation that is simple to

program. The manual fully describes all of the variables and addresses needed to institute each training pass as well as the single-response pass. Most synthesizers require that each word to be spoken is first broken into phonemes. Cognivox, on the other hand, is programmed by speaking the vocabulary into the microphone and thereby training the Cognivox to repeat or recognize the vocabulary. Once spoken into the microphone three times, the vocabulary is memorized and stored for use.

The manual is very well done and supplies all the information needed for integrating Cognivox into your programs. It does assume that you can program in Basic. The manual also shows the simple steps for creating single-voice music and sound effects.

A program that relocates rather than just moves code will considerably reduce work. Your first step should be to make a backup of VOX2 and work from that. If you do not understand Assembly programming or if you don't know what I mean about disassemblers and relocators, then send for VOX 2.1 and use your cassette. Voicetek will come out with a disk-oriented system in the near future.

There are very few limitations on vocabulary. You can use a maximum of three seconds as long as you do not pause for more than 150 ms. The phrase "How are you" easily fits into a three-second input. In fact, phrases are reproduced more accurately than single words.

Choosing a vocabulary can be difficult. Words (especially short words) are easily confused by

Cognivox. Voicetek claims up to 98 percent accuracy in recognition, but I have found that with a well chosen vocabulary and a little practice I can get 100 percent accuracy in most of my applications.

Cognivox is also the easiest voice I/O unit to program and to reprogram. Some voice quality is sacrificed, but with practice and a well chosen vocabulary, clarity and recognition improve. However, everyone does not have a Cognivox, so your programs will be used only on your machine. That makes Cognivox an expensive toy but it is well made and well worth the money if you have a need for it.

*(Voicetek, P.O. Box 388, Goleta, CA 93116, could not be reached for an update.)*

Richard C. McGarvey  
December 1981

★ ★ ★ ★

**Isolated Word-Speaker  
Trained System  
(VOXBOX)  
Tandy/Radio Shack  
Model I**

Radio Shack now produces a speech-recognition unit that it calls an Isolated Word-Speaker Trained system. To use it, you read into the microphone, which is provided with the VOXBOX, the words you want to install in memory.

The box has a limited vocabulary of 32 words at one time. The computer returns an index number for each word in its memory whenever it hears it and returns an error code when it hears a word it doesn't understand.

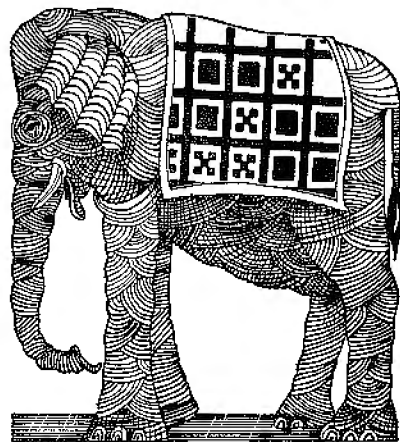
From the return code index

you can program the TRS-80 to do remarkable things. These numbers can control software just as if you entered data via keyboard. And almost any program that asks for keyboard input can be changed to accept a vocal input through the microphone.

Included with the unit are three demonstration programs. One is called VOXPLOT, and herein lies the key to the speech-recognition unit. Running VOXPLOT yields a graph of the four parameters making up the identification pattern of each vocabulary word or phrase.

When you enter a user program or one of the demo programs, you first train the computer by storing in the driver program the vocabulary that must accompany the user program. The documentation isn't as clear as it should be at this point. These steps must be done each time you turn the computer on.

Besides the obvious demonstration programs (they work well and are guaranteed to boggle minds), it isn't difficult to



write a simple calculator program. Speak in a number, say the word "plus" and then another number, and then say "equals." The answer printed on the screen will cause nonbelievers to start searching the floor for hidden buttons. (*Tandy/Radio Shack, Fort Worth, TX 76102. The VOXBOX is no longer available.*)

Art Welcher  
May 1981

★ ★ ★

### **Micromouth Micromint Inc. Model I and III**

Micromouth is certainly a treat to hear. One of the most convincing speech attachments for the home computer, its original speech was done by a human, then digitized. The voice was placed into ROM and made part of the Digitalker product set manufactured by National Semiconductor. The first complete TRS-80 (or Apple) attachment using the National electronics is Micromouth.

A plastic box encases Micromouth and is complete with cable and power supply that attaches easily to the TRS-80: an amplifier is built in, but a small speaker is needed to plug into it. Then it's ready to go. Typing OUT 127.0 calls the sign-on phrase (the only one in a female voice). "This is Digitalker."

A vocabulary of 143 elements is provided with the unit (31 numbers and number parts; 26 letters, 79 words, five silences, and two tones); other vocabularies are being developed by National Semiconductor. Additional words can easily be made from

the basic set ("endangers" as  $N + \text{dangers} + \text{ss}$ , for example), allowing over 200 words plus all possible numbers from 0 to 999,999,999. Most words needed for Basic are there (an unfortunate exception is Print), as are many simple words used in bookkeeping or clerical work.

Micromouth has a self-contained, crystal-timed clock and all latching circuitry necessary for operation. A quick command produces the entire preprogrammed phrase; a running program can continue as Micromouth speaks. There are no software drivers or other programming baggage beyond commands for the words the user needs.

The device is provided with good documentation and full schematics. (Detailed information was published in *Byte*, since Micromouth was designed by columnist Steve Ciarcia.) The vocabulary ROMs (two 64-kilobit chips) are socketed, so new vocabularies may be inserted at any time.

There is one amusing feature of Micromouth. Because of the electronic arrangement, any data greater than 143 sent to port 127 produces garbage. Normally, garbage isn't very useful, but spoken garbage is hysterical gibberish—or a subtle way to tell folks they just entered some pretty dumb input. Also, Micromouth can be made to stumble and stutter by interrupting the word being spoken at random intervals with the same word or another. It starts over until allowed to finish.

Without the immediate availability of exchangeable vocabularies, Micromouth is limited to simple prompting tasks ("2 high,



please try again") or use in games or as an all-purpose, amaze-your-friends toy. Nevertheless, it is an excellent example of the progress of speech synthesis, attractive for specialized applications and experimenting, and very easy to use. (*Micromint Inc., 917 Midway, Woodmere, NY 11598. Micro-mouth has been discontinued.*)

Dennis Bathory Kitzsz  
April 1982

★ ★ 1/2

**The Music Box  
Newtech Computer Systems  
Inc.  
Model I and III**

The Music Box claims to be "a sophisticated facility for the composition and performance of music operating on the TRS-80 Microcomputer." Unfortunately, while it does fulfill this claim, it falls a bit short of actually being a truly versatile tool for the serious music student.

The software manual is confusing and incomplete. By contrast, the hardware manual, which gives instructions on how to hook The Music Box up to your computer, is impeccably prepared.

The Edit program is used to enter music into the memory of the computer and to change music already composed. The editor is very straightforward, supplying cursor, page, measure-step, and home functions. Unfortunately, this straightforward approach is a liability, being a little too clumsy for any real professional or even a beginning student of music.

The Editor imposes one major restriction: If more than one

voice is selected, all the voices called to play at the same time (i.e., all voices on the same line number) must play for the same duration. If voice number one has an eighth-note duration, voices two, three, and four are played for an eighth-note duration, even if the music calls for something else.

The Compiler converts the music from the notation used by the editor to the numerical form required by the Play program. It is a slow and tedious process to wait through.

The Wave program creates and stores new sound waveforms (also called colors or instruments). These are used by the Play program as the sound that generates notes. Unfortunately, unless you have had courses in phase angles and relative amplitudes, the Wave program will be nothing more than a series of puzzling numbers.

The Play program brings everything together and gives you three modes of operation. The first is simply play the music, the second is rehearsal, and the third is interactive. Rehearsal lets you use the computer as a musical practice partner. Interactive mode lets you assign refrains of music to certain keys on the keyboard. This is very, very good for composing avant-garde pieces.

The main advantage of the program lies in its technical versatility. Microtones are useful in achieving Eastern musical effects, and along with the wide variety of possible note lengths, can be extremely valuable in creating original, unconventional compositions. In addition, the rehearsal and interactive modes

of the Play program greatly enhance its usefulness in the creation of music.

The main disadvantage is the poorly written manual. I must warn the noncomputerist that this package is not easy to learn.

Yet, if you are a computer music enthusiast and you want to experiment with unusual sounds and special effects, you might want to try this package. (*The Music Box is no longer available. Newtech Computer Systems Inc., Brooklyn, NY 11201.*)

Terry Kepner and Erich Whitey  
May 1981

### **Orchestra-80** (See Page 194)

### ★★★★ **Orchestra-85**

### ★★★★ **Orchestra-90** **Software Affair** **Model I and III**

The Orchestra-85 (Model I) and 90 (Model III) systems consist of a machine-language program and a small circuit board. The new system maintains all the features of Orchestra-80 and adds many more, including stereo sound, percussion, optional fifth voice, new editing features, and improved sound quality. The system is downward compatible with Orchestra-80, but the Orchestra-85 files that use stereo and voice-altering features will not work with Orchestra-80.

The hardware consists of a 2¼-by-3-inch circuit board con-

taining nine ICs and two RCA-type phono jacks. The board is connected to the rear of the TRS-80 keyboard or the expansion interface bus extension and any external stereo amplifier. No additional power source is needed.

The software consists of a machine-language program that requires a minimum of 16K. About 8K of memory remains for music entry in the 16K Level II tape version. Both tape and disk versions are supplied on a high-quality cassette. The program consists of five major parts: a digital synthesizer, a music language compiler, a text editor, a file manager, and an initialization routine.

The synthesizer features a six-octave range and either three, four, or five simultaneous voices. The three-voice synthesizer gives the best sound quality. The five-voice synthesizer should only be used with TRS-80s with a high-speed CPU clock modification.

The synthesizer voices are pre-programmed to simulate, with marginal accuracy, an oboe, a trumpet, a clarinet, and an organ. You can also define a voice as a percussive. Any voice or voices may be altered at any time within the music file.

Orchestra-85 and 90 use a symbolic language to enter musical pieces. The language works well and has the capability of producing a wide range of musical effects.

The compiler accepts music written in any key, any time signature and any note value from whole to sixty-fourth notes. Notes may be single, double, or triple dotted, or played as triplets. Accidentals, staccato, and

pizzicato note forms are also available. Two forms for articulation are provided, as are the capabilities for repeats, second endings (with or without retard), and modulation.

The program's text editor is very useful as is the file management system. Also useful is the initialization routine, which allows you to custom configure the program to your system.

Software Affair has kept up their tradition of fine documentation. The 43-page nicely typeset and printed manual provides full instructions on set-up and use of the system, including an expanded section for non-musicians.

Registered owners of Ochestra-80 may upgrade by sending their system to Software Affair with \$69.95 and \$2 for shipping and handling.

There is one area of the system that needs improving. It would be much easier to enter music if the screen was turned into a musical staff—each note could then be entered just as it appears on sheet music. While the graphics capability of the TRS-80 is somewhat limited, some creative programming could undoubtedly solve the problem. And Jon Bokelman has proved himself a creative programmer.

These products are, however, a step forward in TRS-80 music synthesizers, and remain the best buys in computer music systems.

*(Orchestra-85 and 90 are marketed by Software Affair, 858 Rubis Drive, Sunnyvale, CA 94087; Orchestra-85, \$129.95, Orchestra-90, \$149.95.)*

Jim Heid  
January 1982

★ ★ ★

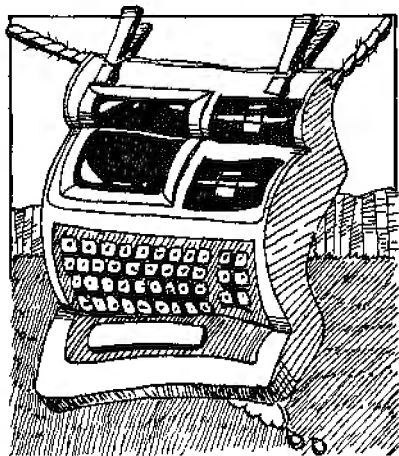
### **Speak-2-Me-2 Percom Data Co. Inc. Model I**

Percom's advertisements say that the Speak-2-Me-2 interface device, when used in conjunction with a modified Texas Instruments Speak & Spell, can be the voice of a computer through the use of a few Basic program lines. An external power source is needed, as is either an expansion interface or printer cable adapter.

My Speak-2-Me-2 package was well presented and included a small printed circuit board, a 22-page manual, and a ribbon cable with connectors at both ends. Unfortunately, however, I could not tell from the advertisement, the documentation, or any marking on the hardware which cable assembly I had. In any case, it eventually worked with the expansion interface.

Initially I was disappointed because there seemed to be a very small amount of hardware for \$70. I eventually realized that more than half of what I paid for was the knowledge and facility to get to the heart of the Speak & Spell and make it work with my system.

The manual is quite thorough. The mechanical and electrical detail was precise and understandable, and there are some Basic programs and a machine-language program to run on the new system. The programs were easy to type in and use, as soon as I realized I was under TRSDOS Basic and had to use DEFUSR instead of POKEing the starting address of the machine-language portion into the locations.



After opening up the Speak & Spell box and getting access to the circuit board on both sides, the trick is to locate a pattern on the board identical with that in the diagram. I made a false start with one diagram before finding one (in an addendum) that exactly matched my unit; if your Speak & Spell is in any way different from that depicted in the manual, it and the whole package can be returned to Percom with \$25 for the required modification.

I also tried a home-brew 10-volt supply to run the unit after finding that a standard 9-volt battery did not have sufficient current capacity. The manual suggested using the Radio Shack PN 274-251 power adapter. I purchased one and found it to work very well.

After these initial problems, I had added a useful new peripheral to my system. Percom offers a separate software package to give an expanded vocabulary. For myself, I much prefer the challenge of experimenting on my own. (Percom Data Co. Inc., 11220

Pagemill Road, Dallas, TX 75243; Speak-2-Me-2 is no longer available. It originally sold for \$69.95.)

Edward Louis  
July 1981

★★★★  
**TRS-80 Voice Synthesizer**  
**Tandy/Radio Shack**  
**Model I**

Radio Shack's Voice Synthesizer is a completely self-contained unit. Like the TRS-80, it uses a separate in-line, step-down transformer. The synthesizer can be used on any Level I or II TRS-80 and attaches by a ribbon cable and 40-pin connector expansion port.

Radio Shack includes a demonstration cassette tape with the synthesizer. The manual provides adequate information for anyone who can program in Basic. All the programming is done in Basic, which makes using the synthesizer very simple.

Based on the program description, it appears that the voice synthesizer acts as a 32-byte block of memory paralleling the video monitor locations 16352D to 16383D. This block is opened and closed by the control character, ?. The correct programming method for the synthesizer is to type the ASCII character that represents the phonetic sound desired. The manual includes a table that cross-references the phoneme symbol and its ASCII character.

The TRS-80 synthesizer is a marvelous invention, but several areas can be improved. The first is the monotone computer sound. Second, some control is

needed over the speed. Some of the sounds are not perfect, but the synthesized voice is, for the most part, very intelligible. I'd recommend the synthesizer for most any application where an audible prompt is desired.

*(The TRS-80 Voice Synthesizer, Tandy/Radio Shack, Fort Worth, TX 76102, is no longer available.)*

Jim Wright  
September 1980

## — PRINTERS/PLOTTERS —

★ ★ ★

### **Anadex DP-8000 Printer Anadex Inc.**

The Anadex DP-8000 printer attaches to the TRS-80 through either of two connectors. One provides reference RS-232C and 20/60-milliamp current loop interface, and the other provides a Centronics-compatible interface.

The printer prints 80 columns at 112 characters per second bidirectionally, giving you a throughput of 84 lines per minute. The 96 characters (upper/lowercase) are formed by a 9-by-7 dot matrix (no descenders).

Additional features include top-of-form control, skip-over-perforation control, eight programmable vertical tab positions, double-width printing, and a 1,024-character buffer that can be expanded to 2,048.

Two idiosyncrasies of the printer should be noted. Top-of-form control is implemented using DIP switches to set page

length. Normally, this works fine; however, while at top-of-form, all line-feed commands are ignored. I even set the form length to zero, hoping this would disable top-of-form, but all line feeds were still ignored.

The TRS-80 places characters in the 1,024-byte buffer of the Anadex until a carriage return is received. After receiving a carriage return, the line is printed while the next one is entering the buffer.

The TRS-80 is finished with your LPRINT and is continuing with your program while the printer is still running. There is a dead-man timer within the printer, so if nothing comes from the TRS-80 in 10 seconds, any residue left in the buffer is erased.

Despite these problems, the only honest complaints I have are with the Anadex ribbon and DIP switches for setting page length and skip-over-perforation. The ribbon is only available from Anadex suppliers and only comes in intense purple. The DIP switches are inside the case, so you must remove the paper and printer cover to make changes.

Overall, the 80-column DP-8000 is a reliable printer at a reasonable price.

*(The DP-8000 is now the DP-9500. Anadex Inc., Chatsworth, CA 91311; \$1,650.)*

Walter C. June  
February 1980

★ ★

### **Anderson-Jacobson Selectronic 841 Anderson-Jacobson Inc.**

The Anderson-Jacobson 841

## HARDWARE

---

with parallel interface is a reconditioned IBM Selectric. This printer is less expensive than a daisy-wheel; its stand-alone diagnostic isolates problems to it or the computer; it's compatible with the TRS-80, although Radio Shack does not support or recommend it, and it works well with the cables available from the A-J distributors and Radio Shack.

The inexpensive system has some deficiencies. There is little memory available for text after you enter Scripsit—4,100 characters in a 16K system, less than three double-spaced pages. You can overcome this shortcoming by frequently saving edited material on the cassette tape, or add more memory to the system. You can also add disk drives.

The lesser consequence to writers is the slow printer speed and the TRS-80/Selectric line-feed problem. The latter involves single-spacing the paper when the program requests a double space. Setting the line space lever on the printer to "double" solves this defect as manuscripts are double-spaced anyway.

In offline mode, the A-J printer functions as a stand-alone secretarial model with a variety of print fonts available by changing the type ball. Most word-processor printers cannot be separated from the system and have no separate keyboard.

*(Anderson-Jacobson Inc., 521 Charcot Ave., San Jose, CA 95131. The Selectronic 841, which originally sold for \$1,095, is no longer available.)*

Robert A. Batty  
April 1982

★ ★ ★  
**Base2**  
**Base2 Inc.**

★ ★ ★  
**Centronics 737**  
**Centronics Data Computer Corp.**

★ ★ ★  
**Epson MX-80**  
**Epson America Inc.**

★ ★ ★ 1/2  
**IDS 445**  
**Integral Data Systems**

★ ★ ★ 1/2  
**Microline 80**  
**Okidata Corp.**

I reviewed five printers, made by Base2, Centronics, Epson, Integral Data Systems, and Okidata. All are dot-matrix, have upper- and lowercase, accept 8.5-inch (or wider) paper, and are in the \$600–\$1,000 price range. These five printers are among the most sought-after and best values for the hobbyist and small businessman.

The 850 is made by Base2 Inc., a small California company; it replaces the 800, which had some problems with print-head reliability. The new unit has a continuous-duty printer head.

The 850 is unique in that it comes standard with a 2K FIFO buffer, bit graphics, RS-232, 20 mA current loop, IEEE-488, and parallel (Centronics-compatible) interfaces. This array of interfaces should match any change in hardware configurations. The buffer is necessary for graphics, since a full line of data must be sent to the printer before printing begins. It also allows the use of RS-232 at 600 baud without

handshaking, for most applications. Another unique feature is that paper can be fed into the printer from the bottom, front or rear. I found the 850 to be the most difficult for paper loading, probably due to the close tolerances used for the friction-feed feature.

The 850 prints an average-quality-dot-matrix character, and offers its best quality in the elongated 132-character-per-line mode, yielding 66 characters per line.

The 850 was the only printer capable of inverse printing. Its processing time was good, placing it third in this category. It was not the most quiet in the group, falling behind the Epson, Okidata, and Centronics printers. The 850 includes most features as standard and has only one accessory, a roll-paper holder (for friction feed) and wire tray combination for \$25.

The Centronics 737, also known as the Line Printer IV from Radio Shack, tied with the MX-80 as the lightest printer. It was of average noisiness and had the only gear-driven print head of the five. The print head is massive (comparable to the IDS 445's) and this, plus its not being bidirectional, accounted for the fact that the 737 was the slowest printer I tested. It compensates for the slower processing time, however, with its quality; in the proportional mode it has the most professional-looking print I have seen in a dot-matrix printer.

The proportional mode uses 1,185 dots for the 80-character line. It is ideally suited for right-justification, since spaces the width of dots can be inserted between words and letters. The

737 is the only printer of the five that can print subscripts and superscripts and do true underlining.

The 737 has friction feed and handles only 9.5-inch pinfeed paper. The paper carriage is at the rear of the unit, making the inclusion of tractors impractical. The 737 does not have a paper-out indicator and is missing a hardware top-of-form advance—features I would expect to find on a printer of this price. It is the only printer of the five that offers no form of graphics.

The Epson MX-80 is designed to be compatible with the TRS-80 and supports Model I and III block graphics. With its programmable line spacing, the MX-80 can print some graphics that can't be produced on the screen. It has a disposable print head that can be replaced for under \$30, well under other print-head prices.

The MX-80 tied for being the lightest in the group (12 lbs.), came in second for quietness and speed, has descenders, is bidirectional, is the least expensive, and offers the boldest print option. It also has the best documentation, if you get the manual by David Lien rather than the standard manual.

Up to 12 character variations can be produced under software control. The emphasized mode gives an excellent print, very close to typewriter quality. In addition to the variations on standard letters, the MX-80 offers some special Japanese, German, French, English, and American characters.

Line feed is fairly slow, and the lack of a friction-feed feature makes use of letterhead impossible. Another criticism is that I

have observed characters not being placed exactly vertical from each other on both the MX-80s I have tested; this happens in the bidirectional mode and could probably be adjusted. New options for the MX-80 include high-resolution bit graphics, serial interface, and an IEEE-488 interface.

The Integral Data Systems 445 Paper Tiger is a replacement for the 440. It receives the prize for the largest, heaviest, noisiest, and fastest printer of the group. It has an instantaneous print speed of 198 characters per second in compressed mode, so in spite of being unidirectional, it has a short processing time.

The 445 comes standard with a 256-byte buffer; a 2K buffer is included with the graphics option. DIP switches can be used to select a serial (up to 1,200 baud) or a parallel interface. Character densities are software- and hardware-selectable, and form-length control as well as one-inch perforation skip features are included.

The 445 prints a good, standard dot-matrix quality and gives the densest print in the 132-character-per-line double-width mode. It will handle up to four-part forms with no adjustment, and more by turning an adjustment knob. The 445 has a very fast line feed and, in operation, seems to be a very beefy commercial-quality printer. It has internal room for a 4.5-inch roll of paper, assuming one can find tractor-feed roll paper.

The smallest and quietest of the five printers, the Okidata Microline 80 prints TRS-80 block graphics and seems to do a slightly more uniform job than the Epson. It has a pinfeed roller

fixed at the 9.5-inch paper width. A tractor option can be snapped on or off very easily, so one can use friction-feed roll paper for keeping costs down, and switch to fanfold for the more important jobs. When in the 80-character-per-line mode, the Microline can be set to 64 characters (still at 10 cpi) to give a convenient left and right margin.

It prints 80 cps in the 10-cpi mode, as does the MX-80, but the Microline 80 is unidirectional. This means it will take about 34 percent longer to print a typical Basic program. (Base2 Inc., P.O. Box 3458, Fullerton, CA 92631; \$799. Centronics Data Computer Corp., 1 Wall St., Hudson, NH 03051; \$995. Epson America Inc., 2384 Hawthorne Blvd., Torrance, CA 90505; \$645. Integral Data Systems, Milford, NH 03055; \$795. Okidata Corp., 111 Gaither Drive, Mt. Laurel, NJ 08057; \$800.)

Harley Dyk  
April 1982

★ ★ ★ ★

**C. Itoh ProWriter 8510**  
**C. Itoh Electronics**

The ProWriter is a dot-matrix printer with high-quality print for the serious user, and a full repertoire of tricks for the hobbyist. It is made by TEC and marketed in this country by C. Itoh, a firm known for its daisy-wheel printers.

This printer is also being marketed as the ADS 8001 and, with a slightly different body style, as the NEC 8023A.

The 8510 accepts single sheets, roll paper, or fanfold, and



can produce up to three carbons. Printing speed is 100 characters per second (unidirectional or bidirectional with logic seeking). A quick-cancel feature moves the print head faster when it is repositioned for the next line. Paper feeding can be forward or reverse. Line-feed spacing is variable from 1/144 to 99/144 inch, making superscripting and subscripting easy.

It has a dot-addressable graphics mode for designing your own characters or making fancy drawings. The dots can be printed with a resolution of up to 160 dots per inch horizontally and 144 vertically.

Sixteen DIP switches let you set up a variety of default conditions. Other standard goodies include: a typewriter-like proportional character set; a self-test function; and a 1,500-character buffer that allows the ProWriter to accept data while printing.

Standard pushbuttons for line feed, form feed, and select/deselect are located in a recessed portion of the front panel, as are three LED lights to indicate power-on, select, and paper-empty conditions.

At nearly 19 pounds, the ProWriter is hefty compared to some other small printers. It is the steel in the machine that accounts for the weight. It also tends to give a feeling of confidence in the durability of the machine. Most of the plastic is on the outside. Its internal parts are held together by screws, not glue.

The print head is attached with two screws and can be easily replaced. An upcoming version of the 8510 will have a snap-in print head. The estimated life of the head is rated

at 100 million characters. A replacement costs about \$75.

The ProWriter uses a cartridge-type ribbon that simply snaps in place and doesn't require any threading. A new cartridge sells for about \$12 and should be good for 3 million characters.

The printer can be ordered with Centronics parallel, RS-232-C serial, or Apple computer interfacing. A TRS-80 connecting cable is not included, but the 8510 with parallel interface works well on my Model I using a standard Radio Shack printer cable, catalog number 26-1401.

The ProWriter's graphics characters are not the same as the block graphics generated on the TRS-80 screen. Those characters can, however, be built using the printer's dot-addressable graphics.

The manual contained a few erroneous or missing decimal values for some of the ASCII control codes. Other than this, and some phrases that must have been poorly translated from Japanese, the documentation is good.

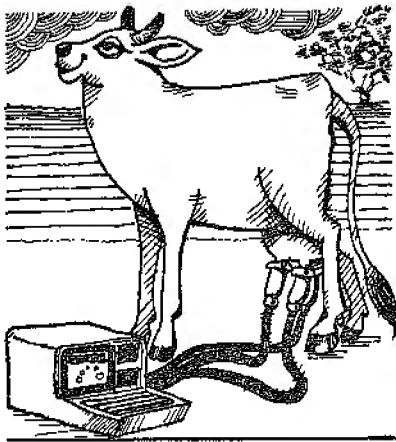
Most of the control codes are straightforward, although some require a numeric value be sent in its ASCII form.

Fanfold and roll paper are threaded into the rear of the printer and exit through the top. A separator is not necessary. Single sheets are fed through a separate opening in the top of the printer. (I would ask for an easier single-threading technique.)

One problem normally encountered in single-sheet printing is that the printer may indicate that it is out of paper when there are still a few un-

printed lines at the bottom of a page. The ProWriter lets you override this false paper-empty signal by repeatedly pressing the select switch and thus print to the bottom of the sheet.

Sophisticated forms control is possible on this printer. You can define page length, top and bottom margins, left margin, and up to 16 horizontal tabs. The



8510 can also be programmed to remember five separate page formats, each page having its own vertical tab settings.

The 8510 has standard and proportional print. Both styles contain the full ASCII set with the lowercase and descenders, as well as an assortment of graphics, Greek letters, and some useful symbols. A set of Japanese characters is also included.

Standard (uniformly spaced) letters can be printed in six sizes ranging from 40 to 136 characters per line, and proportional letters (actually a separate set of very nicely shaped characters) can be printed in two sizes. Both print styles can be enhanced

with bold printing and solid underlining. There are 32 different ways to represent a given character. You can change from one combination to another in the middle of a line. This printer doesn't arbitrarily cancel a selected mode at the end of a line.

No one printer can be everything to everybody. I find the VFU cumbersome to program and usually don't even bother with it, except for channel 1 now and then. If you run programs requiring fancy vertical forms control, you will find the VFU a blessing, but I prefer an easier method of setting vertical tabs.

I have both hot and cold feelings about the 1.5K buffer. Although it is sometimes great to have immediate access to my computer after executing a long LPRINT list, it can be a curse, since there is no way of purging the buffer if you change your mind after the printing is underway. I would like a button to clear the buffer without canceling other activated functions.

Objections notwithstanding, I would not trade my ProWriter for any of the other printers on the market within several hundred dollars of its price.

*(C. Itoh Electronics Inc., 5301 Beethoven St., Los Angeles, CA 90066; parallel \$795, serial \$845.)*

Mike Keller  
May 1982

★★★

**C. Itoh Daisy-Wheel Printer  
C. Itoh Electronics Inc.**

The C. Itoh company is the parent company marketing a

line of daisy-wheel printers under at least four brands: C. Itoh, Starwriter, Tokyo Electric Company (TEC), and Vista. Apparently, all four are identical. They are available in either 25- or 45-characters-per-second versions. Both speeds are available in either an RS-232 serial port or a Centronics-type parallel port.

Print quality is excellent. The C. Itoh printer uses Diablo-compatible 96-character print wheels that are readily available and inexpensive. Wheels are available in many fonts of both elite and pica sizes. Ribbons are in a cartridge, and they come in either regular fabric or carbon film.

There are two modes of operation: line and serial. The line mode types bidirectionally and prints only on receipt of a carriage return (ODH). No special features are available in this mode. The serial mode prints each character as it is received.

Special features, such as underlining, superscripts, and subscripts, are possible by sending an ESC code (1BH) followed by one or more characters. You can set letter spacing at 1-120 characters per inch. Vertical line spacing ranges between 1/48 inch to 1 inch. It can jump to tabs in either forward or reverse.

I did notice a few problems. The manual is not too bad, but is typical of Japanese translations. It is necessary to reread some sections several times to understand. However, the necessary information is there.

The platen is 15 inches long and will print a line until receiving a carriage return (ODH). There is no way to make a line wrap around. If you are using 8½-inch paper and LLIST and a Basic program, part of your data

will print on the rubber platen. Another DIP switch to set line length would be helpful.

My last complaint might seem trivial, but it is bothersome. The carriage return is slow. It takes a full second to return 15 inches, but this is only noticeable in the serial mode.

Overall, this is an excellent printer.

*(C. Itoh Electronic Inc., 5301 Beethoven St., Los Angeles, CA 90066; \$1,995.)*

Patrick Morgan  
February 1982

★ ★ ★

## **Centronics 730 Centronics Data Computer Corp.**

Interested in a printer that handles the full 96-character ASCII upper- and lowercase character set at 31 lines per minute takes pin-feed, single sheets, or cheap roll paper; comes in parallel or serial interface versions and weighs less than 10 pounds? This offer from Centronics may be just what you're looking for.

Since the printer is unidirectional—the carriage takes almost as long to return as it does to print a line—the advertised 50 cps (instantaneous) speed is closer to 41 cps. Characters are formed by a 7-by-7 dot-matrix impact print head and can also be printed in double width.

A print buffer allows a maximum line of 80 characters, printed 10 characters to the inch (six lines per vertical inch). An additional separate line feed buffer can store up to 255 line feeds, each of which moves the paper

1/6 of an inch. The TRS-80 parallel interface version (730-1) also has a built-in automatic carriage return and line feed, so no extra control characters need be sent to get proper listings.

While on the subject of line feeds, there is one quirk. You cannot skip an extra line on the printer by typing LPRINT alone as you do on the screen. Instead you must print a blank (LPRINT" ") or LPRINT CHR\$(138).

The ribbon is a nightmare on first sight—yards and yards of ribbon (20 to be exact) lying loose in a tray. No reels! There is a Mobius twist in the ribbon, so that both sides will be used. A small drive roller pulls the ribbon and can be twisted manually counterclockwise to remove slack. Ribbon replacements come in a zip-pack which is placed in the compartment and pulled off the ribbon in a sort of "yank the tablecloth out from under the plates" routine. I understand later ribbons will come in an easier-to-use cartridge.

The 18-page manual, definitely a rush job, is supposed to be replaced by a better one. The main text and drawings are legible, but the schematic is not. There is no exploded diagram with accompanying part numbers, so if something goes wrong, it will be difficult to repair.

If you've read this far, you've probably gotten the impression that I'm sold on my new printer, and you're right. I think it's a darn good buy for the money. (*Centronics Data Computer Corp., 1 Wall St., Hudson, NH 03051. The 730 is no longer available.*)

Louise H. Frankenburg  
July 1980

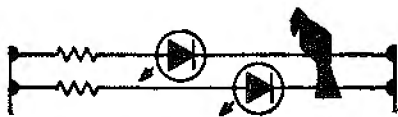
★ ★ ★

### Centronics 739 Centronics Data Computer Corp.

The Centronics 739 is a dot-matrix printer. At 14.5 by 11 inches, this compact printer fits neatly alongside a Model I, II, or III. The 739 accepts fanfold paper up to 9.5 inches wide, and roll or single-sheet paper up to 8.5 inches wide. It prints 40 characters per second (the manual claims 100 cps but my tests showed it to be slower than my NEC Spinwriter, rated at 55 cps) with vertical spacing of six lines per inch and horizontal spacing of either 10 characters per inch (cpi) or 16.7 cpi.

The 739 has a primary and a secondary character set. The primary character set is the standard 96 U.S. ASCII monospaced characters. The secondary character sets include switch-selectable sets for France, United Kingdom, Germany, Italy, and Sweden/Finland. The 739 is available for either parallel port or serial port configured applications. Both printers link up with the host device via 40-pin edge connector cables.

The 739 uses a unidirectional print head that slows it substantially. But it has some excellent features including support of underlining, proportional spacing for justification, elongated characters (in both the 10 cpi and the 16.7 cpi modes), backspacing and half line feed forward and reverse for subscripting and superscripting. It



also has a select graphics mode. This mode provides 75 horizontal and 72 vertical dots per inch for good quality resolution.

The 739's most obvious shortcoming is its lack of speed. All of its control knobs are located at the bottom along the front portion of the cabinet where they can be difficult to see and manipulate. The paper advance control allows single line advance or continuous advance; there is no form-feed control. The operator's manual commits a series of sins of omission.

The Centronics 739 will be a solid seller because Centronics will stand behind the machine with service and back-up support. If this is important to you, the 739 is a good buy. There are other printers on the market, however, packed with features and much less expensive. *(Centronics 739 is sold by Centronics Data Computer Corp., 1 Wall St., Hudson, NH 03051; \$995 parallel port, \$1,045 serial port.)*

G. Michael Vose  
December 1981

★★

## **Comprint 912 Computer Printers International**

The Comprint 912 is a matrix printer that can print a line of 80 characters in about 1/3 of a second. Whereas most other printers use a 5-by-7 or 7-by-9 matrix, the Comprint uses a 9-by-12 matrix. It also features lowercase letter descenders.

The printer comes with a 100-foot roll of thermal paper. A 300-foot replacement roll costs \$8. You can write on the aluminized paper with either

pencil or ball-point pen, the bonding is good, and no aluminum dust wears off onto your fingers.

To connect a Comprint 912 to your TRS-80, be certain that you specify model 912-GP. An older parallel version, 912-P, is complicated to connect. The model GP requires you to pull out one jumper plug and plug the printer cable into your expansion or printer cable interface. The only tool you need is a screwdriver. *(Computer Printers International, Mountain View, CA 94043 could not be reached for an update. The Comprint 912 originally sold for \$660.)*

Mike Aronson  
November 1980

★★★1/2

## **Coosol 101B-80E Printer Coosol Inc.**

The printer mechanism of the Coosol 101B-80E printer kit is a C. Itoh 830 with pin-feed platen accepting 1-9 1/2-inch paper. The controller board provided by Coosol uses a Cybernetic Microsystems CY-480 printer controller chip. This controller chip is a masked ROM microprocessor designed for a variety of dot-matrix printers. The basic features are a 5-by-7 dot-matrix character generator, full ASCII 96-character font, internal 48-character line buffer, graphics capability, 32 system level commands, selectable baud rate (110-9600 bps), built-in self test, and parallel port with handshaking.

Coosol uses two CY-480s in its controller board which gives a preset line buffer of 88 characters. The only problem is

## HARDWARE

---

that there is no overrun protection in the buffer. If you output more than 88 characters without a line terminator (CR or FF), you lose the additional characters up to the terminator.

The construction manual is very complete and easy to follow. It takes you step by step through the construction of the printed circuit board and the wiring of the power supply. The printer is pre-wired and requires only the drive motor and print-head cables to be plugged into the controller board.

The reference manual provides you with all the necessary functional and operational information pertaining to the printer. It also gives examples of the software driver routines to be used with six different microprocessors (including the Z80). One item not covered in the manual clearly is that the strobe pulse must be a minimum of 4 microseconds in length. The TRS-80 only outputs a 1.6 microsecond pulse, so additional circuitry must be added to provide a pulse stretcher.

I am most pleased with the addition of this printer to my system and would recommend it to anyone wanting reasonable print capability at minimum cost. (*The Coosol 101B-80E Printer Kit is sold by Coosol Inc., Anaheim, CA 92803; \$555.*)

Peter E. Noeth  
December 1981

★★★★ 1/2  
**Daisy Wheel II**  
**Tandy/Radio Shack**

One of the most promising products that Radio Shack announced in late August 1980

was the Daisy Wheel II printer. It represents a pricing breakthrough in word-processing accessories.

The Daisy Wheel II is well designed and carefully constructed of heavy-gauge cast aluminum. The sparse, but functional front panel displays a power light and two switches—on-line/off-line and pitch control. There are three possible pitch modes: 10 characters per inch, 12 characters per inch, and proportional spacing.

The wheel and print wheel were designed so that they are easy to remove and replace. The interior controls are equally simple; impression intensity of the print is controlled by a simple, three-position switch inside the printer.

At the rear of the printer are two switches—power and self-test. The self-test reveals characters that can't be accessed by either Electric Pencil or Scriptit, the two best-known premium TRS-80 word processors. But don't let that throw you; the Daisy Wheel II seems capable of printing both the French and German alphabets, if you have the software to generate them.

With a print speed of 43 characters per second, carriage return speed of 300ms/13.6, and line-feed speed of four inches per second, the Daisy Wheel II can compare with more expensive units.

For anyone who wants letter-quality word processing, the Daisy Wheel II can provide it at a fraction of the cost of other systems.

(*Tandy/Radio Shack, Fort Worth, TX 76102; \$1,995.*)

Bob Liddell  
January 1981

★ ★ ★ 1/2

**DIP-81**

**Data Impact Products Inc.**

The DIP-81 is an impact printer capable of printing 100 characters per second bidirectionally on an 80-character line. The normal print mode is a 7-by-7 dot matrix but can be expanded under software control to a 14-by-7 matrix.

The DIP-81 is not a marvel of advanced technology. It uses proven parts to accomplish a simple task—print. The print mechanism is manufactured by the Two Day Corporation and uses a single ac synchronous motor to drive the print head and paper feed mechanism. The print head itself is a seven-wire design that has a rated life of 100 million characters.

Connecting my DIP-81 with the parallel interface to my LNW Expansion Interface was no problem. The unit turned on and operated properly the first time I tried it. I am quite satisfied with my DIP-81. While I wouldn't recommend it for highest-quality business applications, it makes an excellent printer for the personal-computer owner. *(The DIP-81, once marketed by Data Impact Products Inc., 745 Atlantic Ave., Boston, MA 02110, has been replaced by the D-92 Dual Mode Printer, which sells for \$339.)*

David Tinis  
October 1981

★ ★ ★ ★

**DB-9500 Line Printer**  
**Anadex Inc.**

The DB-9500 is one of the better designs among dot matrix printers on the market. It in-

cludes built-in parallel, serial, and current loop interfaces; two type fonts available (9-by-9 and condensed 7-by-9) for 10, 12 and 13.3 characters per inch, and six or eight lines per inch; individual dot-addressable graphics; and 150-200 cps speed.

All parameters (except I/O format) are software- as well as switch-selectable. A ready line prevents data from going to the printer prematurely so that all characters are printed.

The print quality is crisp and clear in both fonts. A lever with detent stops controls the distance of the print head from the platen. Set the lever for maximum separation to load the paper more easily and then set it back for the correct impact pressure. This is a much better arrangement than the fixed-gap method used by most printers.

Its operation is almost flawless. Hooked up to our TRS-80 word processing system, the printer performed correctly on the first run. Running a mail list, the top two lines on each page did not have the correct separation with the perforation skip set at one inch. This might have been caused by paper drag as the paper came out of its box; when half an inch skip setting was used, there wasn't a problem with paging.

*(Anadex Inc., Chatsworth, CA 91311; \$1,650 Model I compatible.)*

Edward E. Umlor  
October 1980

★ ★ ★

**Dynatyper Keyboard**  
**Actuator**  
**Rochester Data Inc.**

A keyboard actuator is an ar-

ray of solenoids that fits over a typewriter keyboard. In response to LPRINT or LLIST, the actuator drives a plastic arm down onto the appropriate key and produces honest-to-goodness letter-quality output. If you already own an electric typewriter, then an actuator is an inexpensive alternative to the daisy-wheel printer.

The big disadvantage is speed. A Dynatyper averages about eight characters per second. You can boost this to 15 or even 20 characters per second, but you may lose some letters as the typewriter fails to keep pace with the mechanical typist. Solenoid heating can also be a problem at high speeds. Since repeatedly hitting the same key without an adequate cooling period can be damaging, typewriter graphics are limited.

Installation is simple. Two supports are mounted on the outside of your typewriter. You remove the cover of the Dynatyper and manipulate four nylon screws to adjust the height of the unit over the keys. When each plunger activates its key near the bottom of its stroke, you replace the cover and hook up the cables.

Each time you use the Dynatyper, you load and run a Basic program. The program takes over your top 306 bytes of RAM, POKEing a machine-language driver routine into that space. There is some inconvenience both in lost memory and in the time required to load the 4K Basic program. This last problem is mitigated if you have a stringy-floppy or disk. Loss of the strategic top bytes may interfere with your own machine-language programs.

The Dynatyper's poor documentation is its weakest feature. The would-be modifier of the driver program gets little support beyond a listing and a brief explanation of the operating philosophy. Worse, the Assembly listing of the machine-language program that resides in high memory does not use the standard Z80 mnemonics familiar to EDTASM fans. Fortunately, there are enough comments to permit comprehension.

The Dynatyper is not a "set it and forget it" device. Its height must be readjusted periodically. The solenoid plungers must be cleaned regularly, too. Fortunately, it is a simple job.

The low cost and flexibility of the Dynatyper make it an attractive product. The one-year warranty is a big plus. I am happy with mine after several months' service.

*(The actuator by Rochester Data Inc. of Rochester, NY 14601, originally sold for \$499. This company could not be reached for an update.)*

Paul Snow  
December 1981

### ★ ★ ★ **Escon Selectric Interface Escon Products Inc.**

The Escon Selectric Typewriter Interface Kit enables any Selectric typewriter to operate as a printer. The kit includes an electronic interface (blue box) and electro-mechanical components.

The kit works with any Selectric and operates at a maximum speed of 12.5 characters per second. The electronic interface accepts data in RS-232 or TTL



parallel form. You may use 110, 150, 300, 600, 1,200, 2,400, 4,800, or 9,600 baud rates.

The interface has a 96-character buffer and will accommodate various forms of handshaking. Instructions are included for interfacing to the Model I (with or without Expansion Interface) and the Model II and III.

The early versions of the Model I Selectric with serial numbers beginning with 4 have a small problem. A spring that debounces the space bar gets in the way of solenoid mounting. You can remove the spring and the unit will not suffer as a printer, though it will be very poor as a typewriter. As an alternative, IBM will modify the typewriter to operate like the later models and it will then perform well as a printer or typewriter.

Operating this unit from a TRS-80 Model I without an Expansion Interface requires a printer interface cable (Radio Shack number 26-1411), which you must then wire to the blue box of the interface unit.

The instructions are quite clear on how to adjust the linkages; they caution to expect imperfect copy until you have made adjustments. Documentation on the system is good and would allow a user to repair

or modify the unit with relative ease.

Solenoid installation requires work and patience and the nerve to pull guts from that costly Selectric. Escon will install the solenoids (for a fee) if the job seems to be formidable.

For letter-quality printing on a reliable, sturdy (although slow) machine, the Escon interface and a Selectric typewriter combine to form an attractive printer alternative.

*(Escon Products Inc., 12919 Alcosta Blvd., San Ramon, CA 94583; \$599.)*

Mike Rigsby  
June/July 1982

★ ★ ★

**IDS 440**

**Paper Tiger Printer**

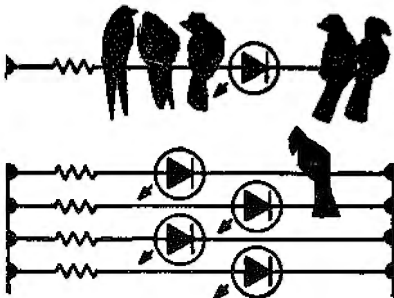
**Integral Data Systems Inc.**

The IDS Model 440 Paper Tiger is a 7-by-7 dot-matrix, upper- and lowercase printer in the 50-100 characters per second range.

The printer comes in an attractive, removable plastic case. All the controls are reached through cuts in the case or mounted on the metal back plate. There are no unguarded openings, so even a clumsy fellow like me can reach behind the printer for the fuse socket or main power switch with relative safety.

Both paper tractors are continuously adjustable for paper widths up to 9.5 inches. A paper roll holder is available as an option.

Seven DIP switches control the other adjustable features. Though plainly marked, they are small and close together, and



you might easily move more than one switch at a time. It is a good idea to turn off the main power switch when reconfiguring any of the DIP switches.

Switches 1 and 2 on the left bank control the print sizes—8.3, 10, 12, and 16.5 characters per inch, giving an 8-inch line length of 66, 80, 96, or 132 characters, respectively. Switch 3 selects six- or eight-line vertical spacing, and switch 4 enables or disables a one-inch skip at form boundaries.

Switch 5 enables or disables an automatic line feed with carriage returns. If a program was originally written for a printer without an automatic line feed, the necessary feeds were probably incorporated in the program, and this switch must be disabled. It should definitely be off for Electric Pencil.

Overall mechanical performance of my Paper Tiger is excellent. I have had no mechanical breakdowns that required service.

I have had certain software problems operating the Tiger in conjunction with my TRS-80, however. I found that TAB characters do not work past the 63rd character in a printer line\*, and once my Tiger hung up on a long, multi-statement line in a program listing. I simply broke the line into several lines of Basic and had no more trouble of this nature.

*(Integral Data Systems Inc., Milford, NH 03055. The IDS 440 Paper Tiger is no longer available.)*

*\*If he's using a Model I, Level II or TRSDOS, then the fault in TAB is the computer's!*

James H. Sheats  
October 1980

★ ★ ★

### **IDS 460 Printer Integral Data Systems Inc.**

The 460 is a microprocessor-controlled, bidirectional, dot-matrix line printer, available with a 2K character buffer and raster scan graphics option. It has a nine-wire ballistic print-head with true descenders. It handles paper widths from 1.5 to 9.5 inches (tractor feed only; an optional single-sheet feeder is available for 8.5-by-11-inch paper).

The printer can underline text with a solid line. Paper can be moved under software control, allowing subscripts, superscripts, or equations. Up to eight vertical tabs can be programmed. The user controls margins, tabs, and inter-character spacing. Character widths are available in 4.8, 5.8, 8, 10, 12, and 16.5 characters per inch.

The 460 has constant pitch and proportional spacing modes for all character widths. The proportional mode prints letters with a constant space between the end of one letter and the beginning of the next, which looks more pleasing than fixed spacing from center to center.

Page length can be set at power-up to one of eight lengths. Under software control, the top, bottom, left, and right margins can be set in increments of 1/48 inch vertically and 1/20 inch horizontally. Line spacing can be set to six or eight lines per inch at power-up, but set to any spacing in 1/48-inch increments via software. Right margin justification is built in; the right margin can be made straight under printer control, even in the pro-

portional spacing mode.

The 460 has both serial (RS-232) and parallel (Centronics-style) interfaces built in. The interface is selected by moving jumpers in a socket on the main circuit board. All other logic default controls, including on/off of auto line feed with carriage return, fixed or proportional spacing, parity select and baud rate (for serial interface), form size, line spacing, and automatic paging at form boundaries, are selected by switches at the top of the printer.

The unit's footprint is 15.75 inches wide by 12.5 inches deep. It is over 12 inches high, allowing room for over 500 sheets of paper under the printer and for straightforward placement of the main circuit board behind the paper path.

I have printed over 4 million characters with my 460 so far. I am on the second ribbon (even though the first one wasn't too light when I replaced it—IDS suggests about 5 million characters per ribbon). The ribbon costs about \$13, is not too messy to replace, and produces even inking across the page.

The 460 has performed faultlessly, save one problem. Mine tried to self-destruct when I sent it one code sequence, blowing a fuse in the motor power supply. I called IDS and learned the problem occurs when an escape code (decimal 27) is sent, putting the printer in the programming mode, and then a decimal 3 is sent. This is not a legal sequence and should not occur in practical use.

I have had no other problems with my printer. Its capabilities optimize the operation of several word processors and operating

systems. Its form feed, line feed, and forward/reverse feed are conveniently located and logical in operation. (The print head-to-platen spacing adjustment is less convenient, but needed only when a considerable change in form thickness is made.)

The printing rate varies depending on character size. At 10 cpi, proportionally spaced, the 460 outputs about 150 characters per second. Since it prints in both directions, the throughput is very nearly this great. In enhanced modes it outputs about 80 cps.

The graphics mode provides complete individual dot control. In this mode it prints unidirectionally to assure better line-to-line synchronization. At least one company has a graphics driver available for high-resolution (84-by-84 dots per inch) plots.

The IDS 460 is not the least expensive matrix printer on the market, but it has about every feature anyone would want and prints the finest-looking font of any printer I have seen for under \$1,500. A wide-platen version (IDS 560) is available for 15-inch paper capability.

*(Integral Data Systems Inc., Milford, NH 03055. The IDS 460 is no longer available.)*

Dennis J. Wilkins  
April 1982

★ ★ ★

**KGS-80 Keyboard Actuator  
Kogyosha Company  
NIK International  
Trading Inc.**

The KGS-80 Keyboard Actuator turns an IBM Selectric or

similar typewriter into a high-quality printer without modifying the typewriter.

It takes only minutes to position the unit on the keyboard, plug it into the expansion interface, or directly into the CPU using the Radio Shack printer interface cable, and make a few minor adjustments. Because the unit does not require software to operate, it is fully compatible with the Electric Pencil, and should work equally well with Scripsit or other text editing programs.

If you don't have a Selectric, the KGS-80 will need to be realigned to the new keyboard by removing the top cover and adjusting four set screws. The set screws position the tips of the solenoids to the proper height above the typewriter keys. A gap of about 0.02–0.08 inches seems to work best. After making adjustments, you can lift the KGS-80 from the keyboard and replace it in a matter of seconds.

The original factory speed setting can be changed from 10 characters per second to 20 characters per second, excluding the shift lock/release and carriage return, by rotating controls VR2 and VR3 on the circuit board.



Control VR4 adjusts the delay time between the carriage return and the first letter of the next line, and is factory set for a standard 12-inch carriage. The delay time can be increased to accommodate the return of extra long carriages, however. All the above mentioned adjustments can be made manually without the need for special software.

All characters are printed in upper- and lowercase. With a special IBM typing element the ASCII signs can also be printed. The operative ASCII input code is 13, 32 through 92, 95, and 97 through 122. A line feed is automatically inserted with each carriage return.

The KGS-80 will not take the place of a heavy-duty line printer if continuous use is required. The solenoids are not designed to exceed 400 successive repetitions without other letters and symbols being activated. This could be a problem if you wish to use it as a screen printer with very little on the video monitor. The successive activation of the space bar solenoid might cause it to overheat and lock-up.

The lock-up characteristic is a safety feature of the Kogyosha solenoids to prevent burnout. Models that became available after September 1980 have two solenoids, instead of one, which allow more continuous use of the space bar.

Another modification available is a buffer circuit that will allow the KGS-80 to print stored text while the computer is free to continue processing programs.

If you own a good electric typewriter, the KGS-80 is an excellent way to add a high-quality printer to your system at a very affordable price.

*(The KGS-80 Keyboard Actuator is now being sold by Personal Micro Computers, 475 Ellis St., Mountain View, CA 94043, by the name of Electronic Typing Fingers; \$595 with power supply, mounting hardware, and manual.)*

Ted I. Blumstein  
September 1981

★ ★ ★ 1/2  
**Line Printer III**  
**Tandy/Radio Shack**

I wanted a printer that could handle numerous forms without locking me into any one type of paper such as thermal or roll stock. While I have nothing against these printers, I wanted to be able to print labels, envelopes, and different kinds of multi-part forms. All the requirements I desired were found in Radio Shack's Line Printer III.

The printer has numerous features, including on-line/off-line modes; upper/lowercase; bidirectional carriage; expanded character capability, usable with or without expansion interface; the ribbon in cassette form to allow easy replacement; and a 9-by-7 dot-matrix print head. The one feature lacking is graphics.

Several negative characteristics came to light during my first week's ownership. The printer automatically prints 132 characters per line unless a carriage return is received; long lines require a 15-inch form. You cannot backspace the print head or overstrike characters, such as when you wish to underline. Also, the first form fed in is wast-

ed because of the print area's design.

Setup can be accomplished in 15 to 30 minutes; the instructions are straightforward. Diagrams of the controller logic, power-supply logic, driver logic, and motherboard logic circuits have been included.

Radio Shack has a great printer here; positives far outweigh any negatives. I would not hesitate to recommend the purchase of a Line Printer III to anyone. With proper care and maintenance, you will experience trouble-free operation. Happy printing!

*(Line Printer III, Sold by Tandy/Radio Shack, Fort Worth, TX 76102, has been replaced by the DMP-100 \$399, DMP-200 \$799, the DMP-400 \$1195, and the DMP-500 \$1795.)*

Robert James Lloyd  
February 1981

★ ★ ★  
**Line Printer IV**  
**Tandy/Radio Shack**

Radio Shack's Line Printer IV is basically a Centronics 737 repackaged in the familiar Radio Shack black and silver color scheme. It is capable of printing on either form-feed, roll, or single-sheet paper. It is a breakthrough in the low-cost quality printer market.

The primary character set is 10 characters per inch, monospaced (in this case, all alphanumerics have the same width).

The secondary character set calls for proportionally spaced characters. It takes advantage of the fact that different characters often have different widths. In

this mode, the output is close to letter quality.

There is also a 132-character-per-line typeface. It has the same spacing characteristics as the primary character set, and is also suitable for letter-quality footnotes.

All typefaces have upper- and lowercase, with descenders, and can be printed in elongated characters with underlines by printing other control codes. Line feeds may be either half or full, forward or reverse (this last feature lets you use sub- and superscripts).

There are no sense switches for out-of-paper or cover-removed conditions. There is also a tendency for the first sheet to wrap around and get dragged back into the feeder mechanism.

Contrary to the first ads I saw for the printer, this is not well suited for use with Scripsit. From Scripsit you can't activate the underline facility, the superscripts, or the subscripts. When using proportional print, line length assignments become almost meaningless. The proportion of a letter is totally ignored by Scripsit, and it is that typeface that produces near letter-quality print.

*(The Line Printer IV has been replaced by the DMP-100 \$399, DMP-200 \$799, the DMP-400 \$1195, and the DMP-500 \$1795; Tandy/Radio Shack, Fort Worth, TX 76102.)*

William O'Brien  
January 1981

★ ★ ★ ★

### **Line Printer VI Tandy/Radio Shack**

If you've been looking for a

line printer, you know what a jungle the business can be. Prices range from a couple hundred to a few thousand dollars, and features and functions of the printers are just as varied.

What are the alternatives? You can do without, or you can risk buying a cheap printer and pray it works without vaporizing your computer. You can go all out and buy a \$3,000 word-processing printer. Finally, you could get Radio Shack's new Line Printer VI. For the money, I believe it is the best choice.

I wanted a lot of features: tractor and friction feed, graphics, a print font that would be acceptable for word processing, a paper-out warning, bidirectionality, adjustable width that would handle everything from labels to 15-inch computer paper, and a good print rate. I didn't want to pay more than a thousand dollars.

The LP VI turned out to be much more than I expected. Well-designed and fast, it had all the features I wanted and more.

I wanted a tractor feed that was adjustable from 2½ to 15 inches. I also wanted single-sheet friction feed. The Line Printer VI has that and the added feature of a removable tractor. The tractor, which is almost flush with the top of the printer, snaps in and out.

The Line Printer VI has a paper-out alert that stops the printer without losing data when the paper runs out. This means I can leave the printer unattended while it prints long data runs without worrying about returning to find my carriage receiving a coat of ink. The paper-out warning works whether the paper is fed from the bottom or

from the rear of the printer. After refilling the paper and resetting the printer, it takes up where it left off with no data lost.

The Line Printer VI is not small; at 24.2 inches wide by 6.3 inches high by 13.3 inches deep, it does require some room. A separate sturdy desk is recommended. Since the tractor is not a large superstructure, the weight and height are not prohibitive. Remember that a 15-inch-wide carriage requires a fairly large printer.

There are four print fonts available. Power-up mode is 132 characters per line (at 15-inch paper size). This will print at 100 characters per second and 33 lines per minute. The normal characters can be elongated to double width, or compressed to 120 characters per second, 37 lines per minute. The compressed mode can also be elongated into the compressed-elongated mode. In either elongated font, the bidirectionality of the printer doesn't work. It does function fully in either compressed or normal modes.

I was especially impressed that once a particular font is selected it will remain active, unlike many printers on which you must call special fonts after each carriage return. Also, when a print font is cancelled, the printer returns to the font that was active previously, not necessarily the power-up mode.

Line spacing, called pitch, is another important consideration. The Line Printer VI powers up at six lines per inch. There is also an eight-lines-per-inch mode that is software selectable. Finally, the 12-lines-per-inch pitch is available for graphics. The pitches remain active until

cancelled.

In the elongated modes some of the special symbols are not clear. The printer has full upper/lowercase. Like most printers, it prints a bracket for an up arrow. The print is satisfactory for word-processing use; it is not as fancy as an impact printer, but the letters are neat and clean.

Most printers available to hobbyists are slow. Forty characters per second is not uncommon. In the normal mode, the Line Printer VI will print 100 characters per second. Since it is bidirectional, it prints on the carriage return as well as left to right travel. Also, when printing tabbed data, the print head moves to the tabbed position the first time and then returns to the tabbed start position only as long as more tabbed data is available. This means that no time is wasted returning full left and then to the tabbed position with each line. The same is true at the other end of travel; the printer will print only until the line end and then returns.

Everything considered, the Line Printer VI is faster than other printers that claim the same or higher characters-per-second rates. Speed may not be a requirement for you, but you have to admit that it would be a welcome extra. For speed and economy, the Line Printer VI can't be beat.

*(The Line Printer VI, sold by Tandy/Radio Shack, Fort Worth, TX 76102, has been replaced by the DMP-100-500 series of Radio Shack printers; \$399-\$1,795.)*

Richard C. McGarvey  
June 1981

★★★½

### Line Printer VII Tandy/Radio Shack

The two factors that most influenced my choice of the Line Printer VII were size and price. The VII requires a level surface measuring at least 16 by 8.25 inches to rest upon; it intrudes no more on my living room decor than the rest of my system.

Setting up the LP VII was no problem at all, even for one as mechanically disinclined as myself. The instruction book is short (18 pages) and has numerous misspellings and grammatical errors, but does give clear information on how to get your printer working. The diagrams are most helpful.

The printer can be used with the parallel interface or with 7- or 8-bit serial interfaces. It uses any tractor-fed paper 4.5 to 9.5 inches wide. The unit prints the entire 96-character ASCII set; I can even print lowercase from my uppercase-only keyboard. If US is an uppercase letter, `LPRINT CHR$(ASC(US) + 32)` sends the corresponding lowercase letter to the printer.

Though I was impressed by the good print quality the 5-by-7 dot matrix produced, I was disappointed that lowercase letters did not have true descenders. The lines are printed at 80 characters per line, 30 characters per second. Though slow, this is fine for anyone who only needs a printout occasionally. The print fades toward the end of long listings, though not significantly.

The LP VII can also print double-width characters at 40 characters per line. You can

select this feature dynamically via the `LPRINT CHR$(31)` command; `LPRINT CHR$(30)` returns you to the 80-character mode. `LPRINT CHR$(18)` selects the printer's graphics mode.

Graphics characters are printed by a 6 by 7 matrix, the sixth column forming the space between characters. In the graphics mode, the LP VII can darken any seven dots in any of the 480 columns on an 80-character line—with 66 lines per sheet of paper, a 480 by 482 grid compared to 128 by 48 for the video monitor. Only graphics characters can be printed; all ASCII characters except control codes will be ignored.

All things considered, I have been very pleased with my LP VII. It has not needed service or a new ribbon. Two problems have occurred on occasion. When it first starts printing, sometimes the ribbon jams the print head and prevents a return to the left margin. Also, the cable connection at the keyboard sometimes comes loose. Wiggling the ribbon or connection a bit cures the problem.

The LP VII makes no idling noises at all, prints a readable clean copy, has outstanding graphics capabilities, and comes at a price low enough for the hobbyist to consider seriously. (*The Line Printer VII has been replaced by the DMP-100-500 series of printers: \$399-81,795. Tandy/Radio Shack, Fort Worth, TX 76102.*)

James E. McKenna  
April 1982





★★★★

**Line Printer VIII  
Tandy/Radio Shack**

Radio Shack's newest printer is certain to become one of the standard accessories for TRS-80 computers. It not only has all the features of the discontinued Line Printer IV (including condensed, elongated, and proportional characters, subscripts, superscripts, and underlining), but also includes bit graphics and sells for \$100 less than the old machine.

The VIII operates in three printing modes: data processing, word processing, and bit graphics. The only real difference between the first two modes is how the printer responds to a line-feed command. In data-processing mode, any command affecting line feeds, such as reverse line feeds and setting different line spacings, is stored in memory until the printer receives its next line-feed command. In word-processing mode, line-feed information is acted on immediately, allowing superscripts, subscripts, and other items requiring half line feeds. These additional capabilities slow printing speed slightly.

In the bit-graphics mode, many control codes available in the other modes are ignored. Alphanumeric character sets are not accessible from the bit-graphics mode, but the printer can easily be switched in and out of this mode, so graphics and text can be mixed within the same line.

The LP VIII prints all 96 ASCII characters, 30 block graphics characters, and 32 European and special characters. A 9-by-8

dot matrix is used to print the 10-character-per-inch ordinary set at 80 characters per line, the 5-cpi double-width elongated ordinary set at 40 cpl, the 16.7-cpi condensed set at 132 cpl, and the 8.3-cpi elongated condensed set at 66 cpi.

The proportionally spaced character set prints with a 9-by-n dot matrix, where the character width (n) varies from 9- to 23-dot columns. Dot density is the same as in the 16.7-cpi condensed set, 1,600 dots in an eight-inch line; the length of a proportionally spaced line varies from 69 to 177 cpl. The proportional set is not as good as an IBM Selectric typewriter, but is certainly adequate for most reports and correspondence. Block graphic characters are printed in a 6-by-12-dot format.

The printer has a parallel interface that connects to the printer port on the Models I (directly to the CPU or through the expansion interface), II, and III. Easily operated switches select serial (7- or 8-bit) or parallel interface. It can be used with any computer with an RS-232 port. The serial transmission rate can be set to either 600 or 1,200 baud. (The Color Computer's 600-baud rate can be increased to 1,200 by executing a POKE 150,41.)

The printer is 15.4 inches wide, 11 inches deep, and 4.7 inches high; it weighs only 16.5 pounds. It accepts standard 9.5-by-11-inch fan-fold tractor-feed paper, and has a friction-feed platen for using single sheets with up to two carbons. It is equipped for using roll paper as well.

Normal printing speed at 10 cpi is 80 characters per second

and 23 lines per minute with an 80-character line. Condensed characters print at 100 cps and 18 lpm with 132-character lines. Elongated ordinary characters print at 40 cps, and bit graphics at 480 dots per second. The printer senses when a line contains fewer than 80 characters, stops the print head at the last character, and line feeds to the next line. It will only move toward the left margin as far as necessary to print the first nonblank character in a line. These features permit fast short-line printing.

Decimal ASCII codes 225–254 generate 30 block graphics designs; for more detailed graphics work, the bit-graphics mode lets you print any combination of seven dots in a dot column. A line feed moves the paper up approximately 1/10 inch, allowing for continuous printing without spaces between lines. Density is 960 dots per line with 480 addressable dot-column positions.

Any character or bit-graphics image can be printed up to 256 times, which is particularly useful for plotting points on a line. The bit-graphics commands are completely compatible with the LP VII's, except for the VIII's lack of a character column addressing command. Since that function can be performed by dot-column addressing in both printers, any program for the LP VII can be written to work without modification on the VIII.

Both block graphics and word-processing applications benefit from the variable line spacing allowed—normal line spacing is six lines per inch, but both eight and 12 lines per inch are available. Forward and reverse

line feeds are program-selectable when six or 12 lines per inch is set.

The 49-page instruction manual is comprehensive and detailed—unfortunately, so much so that it reads more like a technical manual than a “how to” guide. The sections on proportional spacing, repeated printing, absolute positioning, and bit graphics are difficult to understand without some background in bit/byte addressing. Another irritating aspect of the manual is the preponderance of typographical errors, some rather serious. There are some helpful examples, but the booklet is not as useful as it could be.

One of the most positive features of the LP VIII is the overwhelming similarity between its control codes and those of the LP IV (or its Centronics equivalent). Most programs written for the LP IV, such as patches to Scripsit, should work without modification. Some will be easier to implement on the VIII because of its graphics capabilities (adding Greek letters, math symbols, and equation capabilities, for example).

The Epson MX-80 has all of the character spacings available in the LP VIII except for proportional characters. The MX-80 achieves letter quality by either double-striking characters or striking them with more force; both of these modes yield very good results and have a significant advantage over proportional spacing. The MX-80 does not, however, have such an extensive block graphics set, the European characters, or the bit-graphics mode.

With the exception of some

difficulties dealing with existing word-processing software—none of the fancy word-processing features of the LP VIII such as super- or subscripts, underlining, block graphics, and much of the ASCII set are accessible from Scripsit—there is no question that the LP VIII is an exceptionally useful printer. It should have no difficulty capturing the large market once filled by the Line Printer IV.

*(The Line Printer VIII is now the DMP 400. Tandy/Radio Shack, Fort Worth, TX 76102; \$1,195.)*

Richard K. Wallace  
April 1982

★ ★ ★ 1/2  
**Microline 80**  
**Okidata Corp.**

The Microline 80 printer is impressively small, measuring 13.4 by 9.4 by 4 inches. It produces 162 lines per minute (80-column format) and can print either six or eight lines to the inch.

Extremely clear characters are printed in a 9-by-7 dot matrix. The character set is the ASCII standard 96 characters (upper- and lowercase) and features block graphics, printed at 40, 80, or 132 columns.

The printer comes standard with friction and pin feed (10-inch pins). A \$140 tractor-drive mechanism simply snaps into place when needed, letting you use inexpensive roll paper for your other work. Speaking of supplies, the Microline 80 uses inexpensive Okidata ribbons that have two special coatings to prolong printhead life.

Using the Microline 80 is a real dream. There were no installation problems whatsoever, the ribbon connector sent with the

unit fit perfectly, and the connection pins required no rewiring. The quality of the print really helps when looking for that hard-to-find bug—though the lowercase letters do not have below-the-line descenders, the output is definitely of letter quality. *(Okidata Corporation, 111 Gather Drive, Mt. Laurel, NJ 08054; \$449.)*

Gary L. Osburn  
October 1980

**Microline 80**  
(See Page 314)

★ ★ ★ ★  
**Microline 82A**  
**Okidata Corp.**

The Okidata Microline 82A is a fantastic little printer that I believe is superior to anything in and around its price range.

I bought mine for \$499 plus postage. I have seen the list price as high as \$879, so it pays to shop around. Interfacing the printer can be done in either serial or parallel. The place where I bought the printer extracted several pins to make it compatible with the TRS-80. You can do this yourself since the manual informs you how the pins are arranged for both serial and parallel interfaces.

The 82A's print head is rated at 200 million characters and costs about \$143. This compares well with Epson's disposable print heads costing \$30-\$40 that are rated at only 50 million.

The printer uses a \$38.40-per-dozen ribbon on a spool rather than the often messy \$14.95 mobius ribbon. Although the mobius ribbon is rated at 3 million characters and the

Okidata ribbon is only rated at 1.5–2 million characters, there is still an impressive savings with the Microline 82A. Having experienced no difficulty with the printer, I do not have any information regarding various service charges.

The printer measures 14.2 inches wide by 12.9 inches deep by 5.2 inches high. It weighs 19.9 pounds and the case is made of aluminum. Its nine-pin print head prints at 120 characters per second. The printer is both bidirectional and short-line seeking.

Pin feed and friction feed are standard but a tractor feed option is available for \$50. The paper (8.5 inches wide, single sheet or roll, or 9.5-inch wide sprocket) is fed from the rear. If you use sprocket paper you can feed the paper in from the bottom.

Controls on the front panel allow you to select form length (also software controllable), set the top of form, select/deselect (on-line with computer or under local control), form feed and line feed. Three LEDs indicate select/deselect, out of paper, and power on. If you keep the line feed switch depressed while turning on the power and then release the line feed, the printer generates a self-test.

Eight DIP switches on the front panel can select either parallel or serial interface, one of 10 character sets, an optional character set, and other printer control codes. In addition to the DIP switches there are the short plugs, a fuse, and a circuit breaker.

The nine-pin head generates a good correspondence print and can be software controlled to

print at 16.5, 10, 8.3, or 5 characters per inch. Line spacing is under software control and can be set at six or eight lines per inch. (Lowercase descenders are lost at eight lines per inch.)

Although TRS-80 graphics are supported, dot graphics are not. It is not possible to overstrike or proportionally print each character to fill in the dot matrix. Another characteristic of this printer is that you can use 10 and 5 cpi or 16.5 and 8.3 cpi on the same line. You cannot, however, use any other combinations on the same line.

This printer does have some limitations. The DIP switches could have been more accessible and I would have preferred a dotted j. The ability to use more combinations of the various print modes on any given line would also have been nice. A more serious problem is the manual knob for advancing the paper. I have been using only single sheets and have found that the knob tends to pull the paper unevenly.

Otherwise, I am delighted with my Microline 82A. It is fast, loaded with features, and economical. (Okidata Corporation, 111 Gaither Drive, Mt. Laurel, NJ 08054; \$649.)

Ed Thomas  
May 1982

★ ★ ★ 1/2

### **The Micromatic-80 Printer Micromatic Corp.**

The Micromatic-80 is an excellent output device for the TRS-80. It consists of a used, heavy-duty IBM 1980 terminal

system and an interface device. The typewriters are not new, but they are reconditioned and tested by the Micromatic Corporation before shipment.

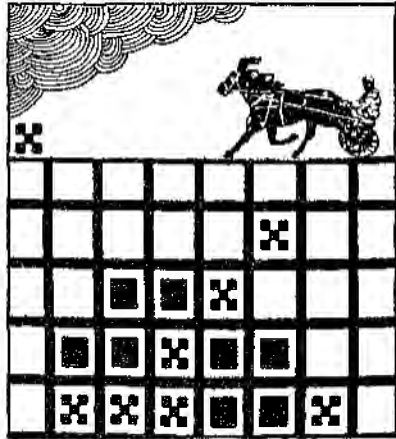
The major advantage of this system is letter-quality output. This feature is important for generating output equivalent in appearance to common business correspondence.

The system has certain disadvantages. Rate of output is especially slow—eight to nine characters per second. While this rate could be increased, the manufacturer feels the present rate is most appropriate for used equipment. Another potential disadvantage is possible service or repair requirements. An IBM Selectric is a complex device that will require periodic maintenance and adjustment.

The Micromatic-80 works beautifully with Radio Shack's Scripsit. The operating procedure is quite simple. The first step is to turn on the system, load Scripsit, and begin typing. No other software is required. The lowercase option, keyboard debounce, and keyboard reverse are all operational.

The Micromatic-80 system is an excellent means to economically produce letter-quality output. Many features do not compare to the daisy-wheel printers on the market, but the appearance of the output is equivalent. The use of the system off-line is a valuable feature that is unavailable with a dot-matrix or a daisy-wheel printer. This system is an excellent buy, and I recommend it for anyone seeking a moderately priced letter-quality printer.

*(The Micromatic-80 Printer was sold by Micromatic Corp., In-*



*dianapolis, IN, which has gone out of business.)*

David E. Clapp  
June 1981

★ ★ ★ 1/2  
**Model 800 Printer**  
**Base2 Inc.**

The Base-2 printer is considerably smaller and lighter than the standard Centronics 779 printer from Radio Shack. It prints 60 lines per minute bidirectionally. Its ribbon cartridge is supposedly good for 5 million characters. The printer uses an 8085 chip and two ROM chips, along with up to four RAM chips to achieve some sophisticated functions. Characters are printed in a 5-by-7 dot matrix.

Unlike many printers, this one is versatile with a capital V. It will take three options—a paper advance, a tractor feed, and extra memory. There is also a serial port handling baud rates up to 19,200 and accepting RS-232 or 20 mA current loop signals, along with an IEEE 488 port that should be compatible

with Pet systems. As an added bonus, the printer is completely silent when it is not printing. The Model 800 printer is an excellent product, especially if your printing workload is light to medium.

*(The Model 800 Printer is sold by Base2, P.O. Box 3458 Fullerton, CA 92631. It has been replaced by the Model 850, \$799.)*

Milan D. Chepko  
September 1980

### ★ ★ ★ 1/2 Novell Image 800 Printer Novell Data Systems

Locating the most cost-effective and reliable printer in the 150 cps range for your TRS-80 can be an arduous task.

I selected the Novell Image 800 Dot-Matrix Printer. Both the printer and the company are unknown to most people since they had been marketed primarily to the Fortune 1,000 corporations and other medium to large-scale data processing installations.

This printer comes with a Centronics parallel interface. I had the opportunity to use it with both the serial and parallel modes as I had a parallel to serial conversion device that operates at 300 baud. It worked well in the serial mode although this was not the configuration in which I had intended to operate the printer.

Installation of the parallel interface board required no electronics ability at all. The standard Radio Shack printer cables were 100 percent compatible. The Model I with the Expansion Interface and the Model III use

cable number 26-1401. The Model I without the Expansion Interface uses cable number 26-1411, while the Model II uses cable number 26-4401.

The print quality produced by the 9-by-9 dot matrix was excellent on forms ranging from stock, one-part, continuous-form to thick continuous-form mailing labels. This quality is maintained by an adjustable print head allowing reliable printing of one original and up to four copies.

Additional printer features simplify the programming and day-to-day operation of the printer and include automatic page advance, the ability to change character width and density via program control. The print types allowed by this feature are 10 cpi, 13 cpi, and 16.5 cpi in both single and double width. In addition is the availability of double-density printing at 10 cpi. This option reduces print speed by half since each character is overprinted once. The print quality in this mode is as good as that of a line printer using a print train.

As important as the ease of programming is the ease of operation. The step button, the form-feed button, and the paper-out detector are handy.

The ability to expand the buffer to 3,300 characters works in conjunction with the off-line indicator to some degree. The off-line indicator and the buffer are related since a printer-busy signal is not sent to the CPU in the off-line mode until the buffer is full.

The standard buffer for the Novell 800 is 300 characters. I requested the six IC sockets for the additional buffer memory be

installed (at no extra cost) allowing me to expand the buffer on my own without voiding the manufacturer's warranty.

By installing my own buffer memory I saved over \$250 on the price of the printer. The buffer memory consists of two 2114 static RAM chips for each 1K buffer increment to be installed. Novell charges \$100 per 1K of buffer memory. The two chips may be purchased at Radio Shack (part number 276-2504) for \$7.98 each.

Technical and repair service can be obtained in two forms: purchasing a service contract, or paying for necessary repairs on an hourly basis. Although I have not yet used the repair service, I have used their technical staff. The consultation was made via their toll-free number. The technical staff answered my questions in a prompt and professional manner.

*(Novell Data Systems, 1170 North Industrial Park Drive, Orem, UT 84057; \$1,395.)*

Ronald Beauchemin  
May 1982

★ ★ ★

**Quick Printer/P1  
Microprinter  
Tandy/Radio Shack  
Centronics Data Computer  
Corp.**

Centronics Data Computer Corporation of Hudson, NH, manufactures the P1 Microprinter, which is known as the Quick Printer when sold under the Radio Shack label. The printer is a 7-bit ASCII TTL printer with strobe and acknowledge pulse.

It employs nonimpact

discharge technology that requires only four moving parts to produce variable-pitch 5-by-8 dot-matrix characters at a rate of 150 lines per minute, with a vertical density of five lines per inch. The paper carries a conductive aluminized coating that is vaporized by a low-voltage discharge from the print head, and the printed characters are highly legible.

You can print either 5, 10, or 20 horizontal characters per inch. Underlining can be started and stopped by separate commands. An audio alarm provides a loud two-second tone. The full 96-character ASCII set, including both upper- and lowercase letters, can be printed.

The commands for print size can be given prior to listing a program or can be included in the body of the program. In addition, lowercase letters can be printed directly on the P1 by depressing the shift key when a program is typed on the keyboard.

Paper is available from Radio Shack or directly from Centronics in either shiny or matte aluminum finish. The printer is not supplied with a mating connector.

*(Centronics Data Computer Corp., 1 Wall St., Hudson, NH 03051; Tandy/Radio Shack, Fort Worth, TX 76102. The P1 and the Quick Printer are no longer available.)*

Henry G. Riekers  
March 1980

★ ★ ★

**Smith-Corona TP-1  
Smith-Corona**

Smith-Corona, best known for

their high-quality, low-price typewriters, will soon become well-known for their low-cost, high-quality computer printers. The TP-1 has a suggested retail price of \$895, but I've seen it advertised in *80 Micro* for as low as \$650, making it the cheapest letter-quality printer available.

You get a well-built, bare-bones, daisy-wheel printer. Most standard writing paper fits on the 13-inch paper carriage, but 15-inch computer paper won't. The actual writing line is 10.5 inches wide, giving you 105 printed characters with a 10-pitch printer, or 126 characters if you have the 12-pitch printer. The line spacing can be set to 3, 4.5, or 6 lines per vertical inch, giving you 33, 49.5, or 66 lines per standard 11-inch-high paper. You can set the impression strength of the character-striking hammer to any one of five levels.

The TP-1 has five special features: backspace, automatic underlining, programmable margins, programmable tab settings, and automatic forms control. It does not support proportional spacing, reverse line feeds, half line feeds, or reverse half line feeds. Incremental spacing isn't available; you can't advance or reverse the daisy wheel in increments of more than one character, nor can you change the pitch you're using.

Compared with the more expensive daisy-wheel printers, the TP-1 seems slow, averaging 12 characters per second. Typing characters rarely used in English slows the unit down to about 10 cps, but typing more common characters, such as vowels, or repeating the same character lets the unit hit about 16 cps.

What are its advantages? It's inexpensive. It can be attached to almost any computer since it's available with either a standard Centronics parallel port or with an RS-232 port.

Another advantage is that the inexpensive ribbons and print wheels are the same as those used on the Smith-Corona Typetronic typewriter and are available through stores selling Smith-Corona electronic typewriters. The TP-1 can handle up to four-part forms. If you use NCR carbonless paper, you can use six-part forms.

Last but not least, the TP-1 is simple to use. When I received mine, the print wheel and ribbon were already in place. They are easily accessible and simple to change, and the manual is easy to understand. The TP-1 even includes a simple diagnostic routine that checks the motor every time you turn the printer on.

At this moment there are two serious disadvantages to the TP-1. Because the print wheels were originally designed for the Typetronic typewriter, they don't have the greater-than and less-than signs. Since most Basic programs use the left and right carets in `If . . . Then` statements, this flaw practically eliminates the use of the TP-1 as a program-listing printer. I've been told by Smith-Corona that ASCII print wheels will be available later this year.

The second problem is the lack of a tractor-feed mechanism for the printer. When running form-feed paper, what starts out as a nicely centered column on page 1 ends up printing on the left edge of the paper by page 10. Trying to print mailing labels was a complete disaster. This



creeping also shows up in the vertical direction, but only if you're printing a few lines on each page with lots of repeating line feeds.

If you don't need proportional spacing or special features such as forward and reverse half line feeds that only more expensive printers can give you, consider the TP-1. It is an inexpensive letter-quality printer ideal for your personal correspondence or short documentation needs. (Smith-Corona, 65 Locust Ave., New Canaan, CT 06840; \$895.)

Terry Kepner  
November 1982

★ ★ ★ ★  
**Strobe 100 Digital Plotter**  
**Strobe Inc.**

A digital plotter is a device that draws pictures. It is specifically designed to make plots rather than print characters. The Strobe 100 is a drum plotter (paper is attached to a revolving drum that moves to position the pen vertically, while the pen moves horizontally) that is driven by computer.

The 100 is easily the most inexpensive drum plotter on the market, but that shouldn't fool you. It is accurate to .002 inch (about 50 microns). That means that the error made in drawings is considerably less than the thickness of the pen line! It uses a four-phase stepping motor to achieve this high resolution. The drum can take paper up to standard 8.5 by 11 inches, and the plotter can use a variety of pens, rather than the specific (usually expensive) brands required by some printers. Several different

pens and a high-quality plotting paper are supplied.

Connecting the Strobe 100 is easy if you get the Model I interface (\$110), which simply plugs into your expansion interface. It doesn't use the standard Centronics printer port or the RS-232 port, so you can have your printer(s) connected while running the Strobe 100. A new RS-232 interface (\$350) will place the routines in a PROM, and can be used with the Model III.

The Strobe 100 can digitize data. That is, you can place a predrawn graph, figure, or photograph on the plotter drum, enter the digitizing routine provided (or write your own), use the driver keys to position the pen over the appropriate point on the graph, and then get the coordinates of that point. In this manner, maps or pictures can be stored in the computer as an array of (x, y) coordinate pairs.

The manual is well produced and fairly well written. In addition to general information on specifications, connecting, and setting up the plotter, it includes a useful section on machine-language driver routines, with flowcharts and listings for Z80, 8080, and 6502 CPUs.

While I am quite enthusiastic about the hardware, I am a bit more subdued about Strobe's software package (\$70). It provides most of the things you want your plotter to perform, but does so at a basic level. It provides some commands that are available through USR routines from the machine-language driver, which can (with a little work) be used in many creative ways. Basic programs accompany the drivers, but they are

meant more to demonstrate the plotter's capabilities than for actual use. Nonprogrammers will find the Strobe 100 plotting software difficult to use.

Three plotter drivers are supplied on disk, one for each of the possible memory configurations on the Model I. There are also three versions of the utility subroutine that can be merged with your Basic programs. The other four programs are also Basic programs, but are called by USR routines rather than being saved in ASCII format. The USR approach is serviceable, but a bit more clumsy than necessary.

The driver supports a number of plotter functions: plot vector, seek, and alphanumeric character draw. The first USR function plots the points. This can be done with the pen down (draw the line) or pen up (don't). The second is used in digitizing the data, and the third draws characters of virtually any size. The plot routine requires three parameters: the x and y coordinates and pen state. This lets you move the pen to a new location on the paper without drawing a line between the new point and the previous one.

The seek routine must be called before you can digitize data. Once it is, you can move the pen to any position and enter that point with the enter key located on the plotter itself. This routine is also used in setting up the plotter initially for the graph.

The character plotting routine is the most useful. You can draw characters so small that you cannot read them all the way up to a single letter so big that it fills the paper. Characters may be drawn horizontally or vertically for labeling graphs and plots.

There is no character-generator routine for users to create their own characters; the character-drawing routine can produce some special characters, but no lowercase.

The most useful Basic program provided with the software package is DRAW8/BAS. This subroutine plots a number of curves on a common graph format, and allows you to draw and label axes, draw a grid, use various special symbols for graphing, and plot by connecting the symbols. You can choose the type of line to be drawn (solid with no symbols, symbols but no line, solid line and symbols, or dashed line and symbols). There is nothing to stop a user from generating his own programs using these subroutines; in fact, it is encouraged.

I am impressed with the Strobe 100 and highly recommend it to anyone who needs quality plots but has a limited budget.

*(Strobe Inc., 897 Independence Ave., Bldg. 5A, Mountain View, CA 94043; plotter \$785, Model I interface \$110, Model III RS-232 interface \$350, PASP software \$70.)*

Bruce Powel Douglass

September 1982

★ ★ ★ 1/2

### **Tandy Six-Pen Plotter Tandy/Radio Shack**

A plotter is a device designed to plot functions and data rather than print text. The Tandy Six-Pen Plotter is a flat-bed unit that can plot graphs of up to 7 by 10 inches. It can also plot 200 points per inch, which, although

not extremely fine resolution, is adequate for most purposes.

It requires special pens that can be purchased separately. As indicated by its name, the plotter lets you use up to six pens, allowing for the easy creation of multicolored graphs. With special felt-tipped pens, you can create overhead transparencies for those impressive professional presentations.

The Tandy plotter allows relative or absolute coordinate addressing, several line-type patterns for drawing circle and arc patterns, four rotational angles for plotting, and some special marker symbols for use in creating graphs.

Since it requires an RS-232C interface, most computers can drive this plotter, including the Models I, II, and III as well as the Color Computer. I reviewed the plotter using a Model I.

The plotter's buffer can hold 768 bytes of instructions. The plotter has its own Z80 microprocessor and operating system.

The Tandy Six-Pen Plotter is easy to program. It lacks some of the flexibility of the HP 7225B plotter, but its use is simpler. The plotter supports 10 types of lines and 93 ASCII characters, as well as four special marker symbols for use in graphs. It is limited to four angles (0, 90, 180, and 270). This makes for easy labeling of graph axes. That is impossible to duplicate with a normal printer!

Included with the plotter is Tandy-Graph, a software package that lets you draw simple line, bar, and pie charts. The Tandy-Graph disk contains a problem. The sample files provided for examples cannot be

read directly by the main program (called Tandy), as they have an extension not supported by the program. The Tandy program works simply and easily for the creation of good looking graphs and plots.

The software provided works well and is menu-driven for ease of use. It is not the most powerful software package imaginable, but it is serviceable and creates some very attractive graphs with a minimum of pain.

The plotter is simple to control, eliminating a great deal of the creative programming that normally accompanies the writing of software packages for specific purposes. The best commands available on the Tandy plotter are the rotational character-drawing command, 10 line types for plotting, and the circle- and arc-plotting commands.

The package is good and very useful. It is also competitively priced. If you need a multi-pen plotter that is simple to program and comes with software that will handle most of your graphics needs, then consider the Tandy Six-Pen Plotter. (Tandy/Radio Shack, Fort Worth, TX 76102; \$1,995.)

Bruce Powel Douglass  
October 1982

★ ★ ★

### **Trendcom 100 Printer Trendcom Inc.**

The Trendcom 100 is a microprocessor-controlled, bidirectional printer. It prints a 40-character line at 40 characters per second, on blue or black thermal paper. Trendcom pro-

jects the head life at 10 million characters.

The accompanying manual is short, concise, and clear. The warranty is satisfactory and repair service is available.

The Trendcom prints upper- and lowercase without a hardware modification by reversing the normal shift procedure. It also features a self-check routine and a ribbon cable complete with an interface card.

The drawbacks of this printer are few and minor. The thermal printer paper is expensive and narrow, measuring less than 4½ inches in width. Also, the paper cannot be exposed to sunlight.

I am completely satisfied with the Trendcom 100, especially considering that my cash outlay was over \$100 less than I would have spent on a Radio Shack Quick Printer.

*(Trendcom Inc., 480 Oakmead Parkway, Sunnyvale, CA 94086. The Trendcom 100 is still available and sells for \$295.)*

John D. Adams  
August 1980

### ★★★★

#### **TRS-80 Plotter/Printer Tandy/Radio Shack**

Radio Shack's Plotter/Printer exhibits features not found on commercial plotters costing several times its price. Features of the plotter include: functions controlled by Basic statements; uses replaceable Fisher Space Pen refills; can function in two modes of operation, Plotter and Printer; character size in the Print mode is software expandable to eight times the original

size; characters in the Print mode can be rotated, under software control, up to 360 degrees; arcs, dotted lines, and straight line routines are ROM resident within the Plotter and can be called in Basic; paper feed is bi-directional and under software control; paper width is 8½ inches with a 7½-inch printing area; and printing speed on the smallest character font is 10 characters per second.

The Plotter/Printer is a small computer that is microprocessor controlled with smart functions resident in ROM. The copy produced by the machine consists of thousands of .09 millimeter straight lines. Programming is straightforward and easy. Output to the Plotter/Printer is generated through two types of statements, the character string (CHRS) and the G-Codes.

The Plotter/Printer is a well-constructed piece of equipment. Connection to the TRS-80 is via the parallel printer port on the expansion interface, which presents a problem if you already have a printer attached to the port. Daisy-chaining the two devices does not work without buffering the cables.

Plotter/Printer output quality is excellent. The type rivals a letter-quality printer, with one major disadvantage: it is generated in uppercase only. The Plotter/Printer has the hardware necessary to become a multiple-font, letter-quality printer. Unfortunately, the firmware (ROM) to do this is lacking. Another area in which the Plotter/Printer is seriously lacking is the instruction manual. It is poorly produced and difficult to understand.

All in all, this is an excellent

plotter, and with a little work, an excellent printer as well.

(Tandy/Radio Shack, Fort Worth, TX 76102; \$995.)

Don Dejarnette  
January 1982

### —PRINTER SPOOLERS— BUFFERS/INTERFACES

★ ★ ★ 1/2

#### **Micro Cue Print Buffer MicroCompatible Inc.**

Available in four different models, the Micro Cue can help you solve your computer/printer interface problems.

The model tested here at *80 Micro* was the PB322P Print Buffer with two parallel transmit ports and two parallel receive ports. The unit contains 32K of memory and a Z80 microprocessor. A 1K ROM contains the software to control the system. All these electronics are driven by a 2.0MHz clock.

The on-board memory is the buffer's principal claim to fame. It allows a user to transfer files from the computer to the buffer for printing. This transfer operation takes only a few seconds. The computer is then free for other tasks while the buffer drives the printer. The 32K buffer will hold Scripsit files up to 18 pages long. It would be hard to write a Basic program that could fill the entire buffer.

Micro Cue's other notable feature is its multiple-port configuration. With the right model,

you can hook up two printers and two computers to the buffer and select any combination of computer and printer with a flick of a switch.

The Micro Cue recognizes all your computer's codes as well as messages from your printer. It lets you know if the printer is not ready and also recognizes out-of-paper signals. The buffer can receive data when the printer is not ready, but it will not send data until the printer is ready.

The instruction manual contains a limited amount of information; it is poorly laid out, and the diagrams are hard to read. MicroCompatible offers a detailed hardware description, including schematics, plus a programmer's guide containing the ROM's software listing for an additional \$25.

The Micro Cue is housed in a light gray cabinet. It is unobtrusive and quite useful. Depending on your needs and budget, it is a nifty peripheral for your favorite computer. (The Micro Cue is now called the Smart Buffer, and is sold by MicroCompatible Inc., P.O. Box 7624, Atlanta, GA 30357; \$335.)

G. Michael Vose  
March 1982

★ ★ ★

#### **P180C Parallel Printer Interface The Micro Works Color Computer**

The Micro Works must be working overtime to develop software and hardware for the Color Computer. Their P180C Parallel Printer Interface is designed so you can use a

parallel printer, such as the Centronics 730, through the serial port of your TRS-80C.

The printer interface is supplied in a ROM pack, though the instructions clearly state that it is not a ROM pack and it should not be plugged into the ROM slot of your Color Computer. A DIN plug connects to your computer's serial I/O jack. An edge connector connects to your parallel printer cable. A transformer plugs into a 120V ac outlet (the interface has its own power supply).

During operation, the interface converts your serial data to parallel before printing; this slows down the printing somewhat. You will notice this delay when listing a program.

Since the printer interface requires no software, there is no problem using the printer's functions which require control characters from the computer for activation.

This interface does cause slight radio frequency interference and snow in the TV screen. You can minimize this by moving the interface away from the TV. A longer cable

would help solve this problem.

The PI80C is a useful piece of hardware for the Color Computer. It can save you the expense of a serial printer if you already own a parallel printer. (*The Micro Works, P.O. Box 1110, Del Mar, CA 92014; \$69.95.*)

Howard Berenbon  
June/July 1982

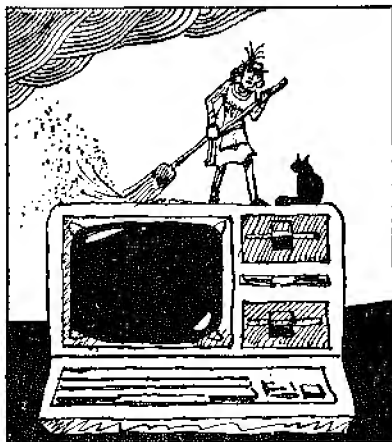
★ ★ ★ ★

### **Snapp Spooler Snappware Inc. Model I, II, and III**

Spooler stands for "simultaneous peripheral operations on-line," and in the case of this printer spooler it means that the computer and printer work at the same time.

If you're a hobbyist, a spooler is an extra nicety; if your plans involve business computing with printouts, a spooler is nearly essential. My wife and I use a production program to generate weekly reports. Using a spooler, we have gained 15-200 percent in execution time without making any program changes.

The Snapp Spooler is available for the Models I, II, and III. The Model I version comes with 42 different spooler versions geared to specific memory size/buffer sizes; each automatically initializes itself and sets memory size. Some versions allow high memory routines to remain unaffected. What buffer size you choose depends upon how much you use the printer and how much memory you can give up to the print buffer. With our production program running on a 48K machine, we started with a 4K buffer. As need for more



variables memory increased, we used a 2K buffer with no noticeable speed loss.

If you own a Line Printer III with the motor on/off hardware, Snapp includes a program that patches all 42 versions of the spooler to operate this feature. Early versions of NEWDOS didn't respect the top of memory pointers, so Snapp includes a program that applies Apparat's recommended patches to NEWDOS and corrects the problem.

The spooler incorporates the Radio Shack LPC driver, which is needed to run their printers. (This feature modifies the system to ignore LPRINT commands without print data following.) We use a Line Printer VI and wanted to use the graphics characters, but found that the LPC feature filtered them out. Bob Snapp quickly provided a patch to solve the problem and will supply it for other users with similar needs.

I've found this customer support typical of Snapp Inc., and refreshing in the software field. (Snappware Inc., 3719 Mantell, Cincinnati, OH 45236; \$100, hard-disk version \$150.)

Alan Moyer  
September 1982

### ★★★★ Scoopspooler Compulink

Other than the cute name, I like everything about Compulink's Model SS-1000 Intelligent Printer Interface.

This handy box is a hardware spooler. It quickly gobbles up data from the computer and passes it on to the printer at the printer's normal rate. By being

able to keep computing instead of waiting for the printer to finish an LLIST or document, I've increased my productivity by 20 percent or more.

The spooler's connectors and the cable included enabled a painless hook-up with my Model III and Line Printer VIII. (I suggest purchasers contact Compulink to make sure they get the right connectors.) The unit normally comes with 16K and is set up for a parallel printer; options include increasing the memory to 62K and adding RS-232 I/O.

The spooler accepts data from the computer at 3,000 characters per second. It doesn't take long to swallow a long listing and quickly return computer control to you. In addition, the interface lives up to its "intelligent" description. A display tells you how much data is in the buffer yet to be printed. A reset button allows cancelling a print-out. The space compression button lights a tally LED and then compresses consecutive spaces for storage. It expands them again when feeding the printer.

The pagination button lights a tally LED and limits the number of printed lines on a page to 62, and then does four line feeds before starting the next page. If you press the paging and space compression buttons at the same time, a self-test is performed, checking the ROM and RAM in the spooler and printing the test results.

The spooler can number pages and will put the heading of your choice on each. You can set the number of lines per page (default 66) and the number of printed lines per page (default 62). You can set the line width (default

80) by setting the left and right margin columns; overflow lines can be indented (default 5). The form-feed character (CHRS(12)) can be changed to another character from 0-31; in fact, the input character (from the computer) and the output character (to the printer) can be different.

You can also use the spooler to print single pages, pausing after each one so you can insert another sheet of paper. Finally, there is a software command to ignore any preset hardware or software commands and any subsequent software commands, storing and passing along all data precisely as received. This is good for printing graphics where some of the characters can be confused with spooler commands.

I received my spooler six days after my telephoned order and it was running perfectly 10 minutes after I unpacked it. The documentation is complete with plenty of technical data and is easy to read. The spooler is a handsome and useful addition to my system.

*(Compulink, 1215 Ravenwood Road, Boulder, CO 80303; \$349.)*

Ken Knecht  
September 1982

★ ★ ★ 1/2  
**UPI-3, UPI-4 Serial Printer Interface**  
**Speedway Electronics**  
**Model I or III**

The Speedway Electronics Serial Printer Interface plugs into the TRS-80 printer port and makes the TRS-80 think it has a parallel printer attached, so it

speeds up printing. There are two models of the Speedway Electronics Serial Printer Interface: the UPI-3, for the TRS-80 with an expansion interface; and the UPI-4, which runs directly from the Model I without the interface.

The unit plugs into my Model III, and it puts out either RS-232 or 20 ma current-loop signals. I used the 20 ma current loop, which is included in the RS-232 connector. I also used an optoisolator to electrically separate the two units.

The UPI-3 does exactly what it is advertised to do: It converts the parallel data to serial and puts in a line feed where needed at the end of the line. Mine worked the first time I plugged it in, and it has worked ever since. In addition, you can make the UPI-3 put in a delay to give the Model 33 Teletype time to complete the carriage return.

You can also select a number of other characteristics of serial data transmission. You can select, with a neat little DIP switch, handshaking, line feed after carriage return, nulls after carriage return, odd-even parity, number of bits per word, number of stop bits, and parity on or off.

The unit is nicely built. Construction seems excellent. The unit is built on a good glass-epoxy board. The parts and workmanship are first-rate.

The UPI-3 performs as advertised and can be depended upon to keep doing it. There are a few minor quibbles: The box is rather cheap (but serviceable), and the cable to the TRS-80 is too short for my taste. But in all other respects, it is an excellent unit.



*(Speedway Electronics, 11560 Timberlake Lane, Noblesville, IN 46060. The UPI-3 and -4 are now the UPI 3VD, \$149.95.)*

Jerry W. O'Dell  
February 1982

## SPEED-UP KITS

★ ★ ★ 1/2

### **Archbold High-Speed Modification Archbold Electronics Model I**

The latest Archbold High-Speed Modification permits some TRS-80s to operate at 5.36 MHz, three times faster than the original keyboard. Not only is the increase dramatic, but Archbold's newest circuitry monitors certain computer operations. Thus, errors during disk access do not occur because the board switches back to normal speed during this critical time.

Pushing the TRS-80 beyond a 100 percent speed increase can be fatal to programs using high memory. Archbold has solved this dilemma in several ways: the disk accessing is monitored, a few minor circuit changes increase memory stability, early units are stabilized with the Radio Shack buffered cable/twisted pair improvements, and a memory delay unit is available for use with the very highest speeds.

Archbold makes every effort to document each step of the process and assist the user. Unlike any other hardware change to the TRS-80, the Archbold board

provides a full-size photograph of the entire TRS-80 circuitry with the modification connections clearly numbered. Also, every revision of the computer is covered in the instructions. It is among the best hardware documentation ever prepared for computer add-ons.

Multiple speed options are available on this board, but call for some user intervention. Integrated circuits must be piggy-backed and holes drilled for switches. Even with the Z80B and memory delay line, there's no guarantee it will work reliably and consistently. But it can work, and if you need the additional speed, then the Archbold modification can provide a key to it.

There is one serious black mark against the Archbold modification. Archbold has joined manufacturers like Percom in taking what, from my point of view, is a ludicrous form of circuit protection—he has sanded the part numbers off the top of the integrated circuits. If anything goes wrong with the Archbold board, or if anything does not work upon first installing it, forget the diagnosis.

In summary, the Archbold speed-up kit is a fine product that can transform your pokey TRS-80 into a fast, more powerful computer. Calculations and sorting time are cut phenomenally, and graphics take on a remarkable new level of animation.

*(Archbold Electronics, 10708 Segovia Way, Rancho Cordova, CA 95670. The Model I speed-up kit is no longer available.)*

Dennis Bathory Kitz  
December 1981

★ ★ ★ 1/2

### **FCI-80**

**Personal Micro Computers  
Inc.**

★ ★ ★ ★

### **TC-8 Cassette System**

**JPC Products Inc.**

**Model I**

The FCI-80 and the TC-8 are hardware devices that speed up tape storage and make it more reliable.

The FCI-80 takes a novel approach to increasing cassette-load speed. Rather than increasing the density of tape data, the tape is moved faster. Programs are saved as usual at the normal speed, but when you want to load a program using the FCI-80, press fast-forward and play simultaneously on a modified CTR-41 recorder. This increases load speed to a minimum of 6,000 baud (about 12 times the conventional load speed).

The FCI-80 connects to the keyboard with a 40-conductor ribbon cable, connects to a 9V ac power adapter, and plugs into the earjack on the recorder. The operating system is in ROM at address 12288 (3000H), a reserved area just above Level II Basic.

Loading speed is almost unbelievable for an audio cassette. The speed is not fixed since the fast-forward speed of a tape increases as the size of the take-up reel increases. At its slowest, the FCI-80 loaded a 14K program in 16.5 seconds.

The program took 228 seconds with TRS-80 COS, a speed-up factor of more than 13. A 4.3K program loaded in about 5.7 seconds at the beginning of a tape, and it loaded at the

100-foot point in 3.9 seconds, which is nearly 9,000 baud (a speed-up factor of nearly 18).

I couldn't get the unit to work near the end of a 300-foot tape, but it worked fine from the beginning through at least the 100-foot point. You should be able to use tapes with at least 10 minutes per side.

The unit is ideal for someone who frequently uses large programs but rarely saves them. The unit does load programs rapidly, but I feel two very important features are missing. Programs must be saved at the same slow speed, and writing data to tape (one of the weakest features of TRS-80 COS) has not been changed.

One nice additional feature of FCI-80 is that a keyboard debounce program is also in ROM. This provides automatic key repeat when a key is held down and will beep when you press a key if the cassette output plug is connected to an amplifier.

The TC-8 saves and loads programs almost five times as fast as a conventional cassette, offers a significant improvement in raw data storage, and improves reliability.

The TC-8 connects to the keyboard with a 40-conductor ribbon cable and connects to a 9V ac power adapter and the recorder. You can purchase the TC-8 in kit form; the instructions are superb.

The operating system (called UTIL) is loaded from TRS-80 COS tape so the recorder is initially connected to the keyboard. Set memory size to 31400 when you turn on the computer, and load the operating system at the conventional speed. You can then plug

the recorder into the TC-8 and procede (save and load) at 2,400 baud. The results are what you would expect at this speed: A 4.3K program loads in 15 seconds, and a 14K program loads in 48 seconds.

The documentation is excellent, and the operating system is smooth. The syntax for saving raw data on the TC-8 is PRINT #1,X; the TRS-80 COS uses #1. Using that statement in a loop, the TRS-80 took 428 seconds to store 100 three-digit random numbers. The TC-8 performed the task in 22 seconds, almost 20 times faster.

The TC-8 software includes a monitor called Tiny that provides a means to find the beginning, end, and execution addresses of most machine-language programs. Tiny also lets you examine memory locations, relocate UTIL, write back-up copies of UTIL, and jump to any hex address.

The FCI-80 and TC-8 are certainly worth considering if you want to improve your cassette storage without making a major investment.

*(The FCI-80 is available from Personal Micro Computers Inc., 475 Ellis St., Mountain View, CA 94043; \$269. The TC-8 is available from JPC Products Inc., 12021 Patsano Court N.E., Albuquerque, NM 87112; \$120, \$90 kit.)*

Harley Dyk  
May 1981

★ ★ ★  
**High Speed Upgrade**  
**Simutek**  
**Model I**

Simutek supplies a high-speed

modification kit that can be installed by anyone with electronics experience in about an hour. The kit comes complete with generous lengths of wire, an instruction manual, and a push-button switch to enable or disable the speed-up. Your computer will run 50 percent faster with this modification, and the kit is covered by a 90-day warranty.

*(The High Speed Upgrade, manufactured by Simutek, 4877 East Speedway Blvd., Tucson, AZ 85712, has been discontinued.)*

Jake Commander  
November 1980

★ ★ ★ 1/2  
**Model I Speed-Up**  
**Archbold Electronics**  
**Model I**

The Archbold Model I Speed-up kit consists of a small circuit board that is extremely well made. A 20-wire ribbon cable extends from the speed-up board, and these wires are connected to various points on the computer logic board. In the kit box are a few extra parts: a diode, an LED, a capacitor, a connector/coupler, and lengths of wire and fine solder. You may or may not need these parts, depending on your board type and whether or not you have an expansion interface and disk drives. If you have an interface, you need a delay line for it.

The installation is not too difficult, if you're extremely careful when soldering or when you fold the keyboard unit away from the logic board. Remember that there is very little excess power

available; when you add accessories that use computer power, you are straining the limits of the computer's regulator. Don't make 300 modifications that all use computer power or the computer will crawl away and die. Use only those that are most important to you.

Once the kit is installed, you can power up in either high or low speed (a simple modification to the kit board) and then switch to the opposite speed with a simple OUT 254,1 (or OUT 254,0 depending on which way you want to go). The LED that is included changes color to indicate mode; red is normal, green is high speed, and yellow is 50 percent slower than normal (switch controlled). The board automatically reduces speed when cassette or disk I/O is performed, and returns to high speed (if that was selected) when the I/O is complete.

The exception is if manual switching to high speed is used; in that case, disk access will cause automatic slowdown, but cassette access requires an Out instruction or manual switching. This switching makes it possible to run 50 percent slower than normal or high speed. Switching can be done during program operation with no interference in the program's performance; however, the extra work for the minimum benefit seems to me to be a waste of time. The switch is not included in the kit but is readily available at your local supply store.

So far I have found only one operation problem in the speed-up modification: keyboard bounce. The problem is that debounce is accomplished by introducing a time delay in the

keyboard scan, so that if a key double strikes, the second strike will occur before the delay is complete and will be ignored. When you speed up the operation, you shorten the debounce delay. At my speed-up rate it is not bad, and cleaning the keys seems to have eliminated it altogether. I do see a problem for those who buy a Z80B and jump up to 6 MHz, because the delay will be so short as to be totally ineffectual.

I had one unexpected bit of good fortune: Though I have a disk system, I occasionally run the cassette for a few games and applications programs. With the speed-up kit, not only did *Invasaders* pose a real challenge, but it and my other cassettes loaded the first time with no errors. Even cassettes that I had given up for dead now load fine, and Basic and System tapes load at the same volume.

I'm very satisfied with the increased speed of operation, and even though the installation was a bit tricky, it was well worth it. If you would like that extra speed, then buy Archbold—it works and it works right! (*Archbold Electronics, 10708 Sequota Way, Rancho Cordova, CA 95670. This product is no longer available.*)

Richard C. McGarvey  
July 1981

★ ★ ★ 1/2

**SK-2**

**Mumford Micro Systems**

**Model I**

I decided to purchase a clock modification for my TRS-80 because the TRS-80's Z80 processor doesn't run at its best

clock speed of 2.5 megahertz.

I chose the SK-2 clock modification produced by Mumford Micro Systems. The options list of the SK-2 takes almost two pages. First, you have the choice of optional speeds. In its standard configuration, the SK-2 allows a 50-percent increase or decrease in selectable clock speed.

You can have only one speed as an option. The 100-percent speed increase option requires some rework of the SK-2 board and probable replacement of certain circuits on the CPU that can be substituted in place of the 50-percent speed reduction.

Actual installation is relatively simple, although I recommend you have a copy of the *Technical Reference Handbook for the TRS-80* (Radio Shack Catalog No. 26-2103) when you do the installation.

After installing the SK-2, I entered a small test program from the instructions. I could get the printout to speed up and slow down by flipping the switches correctly. Once in a great while the CPU hung up when I switched from normal to high speed. The problem never occurred when switching from normal to low speed or when the speed change was software selected, so it might be a contact bounce problem.

My first real problem came when I attached the expansion interface to the keyboard; I could not run reliably at high speed. I found a note on p. 4 of the instructions that said the cable from the keyboard to the new expansion interface is not buffered (there is no black box in the middle of the cable). A restrictive hardware timing ar-

angement in the interface must be undone before the memory in the interface will run at the higher speed. Once I made the changes, I no longer had a problem.

I am satisfied with the performance of the SK-2. I now use software corrections to allow disk operation at higher speeds. I obtained these corrections from Mumford; they also appeared in *80 Microcomputing*, April 1981.

The Mumford Micro Systems SK-2 clock modification lives up to all its advertising. The company is responsive to questions and inquiries, before and after the sale, and orders are rapidly processed.

Consider the SK-2 as a cost-effective clock modification alternative that can be implemented in the software-controlled version.

(Mumford Micro Systems, Box 435-C, Summerland, CA 93067; \$24.95.)

C. H. Ballard  
February 1982

### ★ ★ ★ 1/2 The Sprinter II Holmes Engineering Model I, II, and III

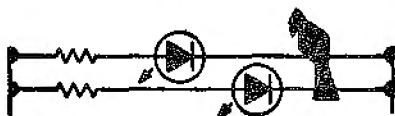
The Sprinter II is a speed-up modification that requires no damaging alterations. With a few exceptions, you can plug in the Sprinter II and run it on your TRS-80, instantly giving you two of eight operating speeds. One speed contains wait states (pauses) that allow slower ROMs to keep up. The other is without wait states. The price on the Sprinter II may seem a bit high but that is because the kit includes the 6 MHz Z80B CPU.

During operation an automatic slow-down occurs when you access disk or cassette. This is the default, but it can be overridden. You can also decide whether you want the unit to start in the high-speed or normal-speed mode. There are eight speeds, some slower than normal and others faster. Each has the option of operating with or without wait states to allow the slower ROMs to keep up. Top speed is 5.33 MHz without wait states, but a heavily modified computer is necessary to achieve this speed.

Sprinter II is well made. There are a few minor problems, however, that may not be immediately evident.

The first problem confronting most Sprinter II users is the need for high-speed memory. The 400-nanosecond Motorola memory present in most TRS-80s is not fast enough to handle the Sprinter II's acceleration. It is easy, however, to get 200-ns NEC chips that will rise to the challenge. If needed, the faster memory is available through Holmes Engineering for \$26 per 16K. Memory in the expansion interface is not usually a problem, but some obstacles emerge when you use an expansion interface.

In later versions of the expansion interface Radio Shack removed the buffered cable and installed a delay line (actually a chip). This chip causes problems when you try to speed up. You will most likely need to replace that chip in the expansion inter-



face with a modifying chip provided with the Sprinter II (at the additional cost of \$10). To find out if this modification is necessary, call Holmes Engineering.

A few other adjustments may be necessary. You may have to install resistors in the keyboard and in the expansion interface to help the memory keep up. If after all of this modification you still can't get high speed, you may have to cut a trace.

If you don't want to do the work yourself, you can send your system to Holmes and they will make the computer run at 5.33 MHz for a small fee. Their work is guaranteed. If you want to tackle the job yourself, you can depend on Holmes for support.

Holmes gives you a money-back guarantee. For this reason, they have an interest in your ability to get this little devil running to your satisfaction. (*Holmes Engineering, 3555 South 3200 West, Salt Lake City, UT 84119; \$99.50.*)

Richard C. McGarvey  
November 1982

---

## SYSTEMS

---

★ ★ ★

**EPS-80 Word Processor**

★ ★ 1/2

**ETF-80 Electric Typing  
Fingers  
Personal Micro Computers  
Inc.**

For people who do a lot of

writing. Personal Micro Computers is making an offer that's difficult to ignore: a word processor for under \$1,000.

The EPS-80 is a work-alike version of PMC's work-alike Model I, the PMC-80. However, the EPS system has 20K of ROM—12K containing Microsoft Basic and 8K with Michael Shrayner's Electric Pencil—and 48K of RAM.

The EPS-80 also contains a built-in high-speed cassette deck for tape saves and loads at 5,000 baud, more than 10 times faster than the TRS-80 Model I rate. The Zenith monitor with green phosphor screen can display 32 or 64 characters per line.

For printers, the EPS-80 is advertised with either PMC's DMP-85 dot-matrix printer (a work-alike C. Itoh 8510/NEC 8023) or the ETF-80 actuator, which works with IBM-like typewriters. I tested only the actuator with a Silver Reed 213C typewriter.

Loading a tape-based word processor at 300 baud is painfully slow. I found the EPS-80's built-in fidelity audio recorder as fast as advertised and experienced no bad loads during the three weeks I tested it. Another plus is a volume level meter, which lets you load tapes without hearing the ear-splitting whine of loading.

Having Electric Pencil in ROM beats waiting three minutes for a 16K word-processing program to load. When you turn on the EPS-80, you get the message: (B)ASIC OR (E)LECTRIC PENCIL? Hit E and you're in Pencil.

However, there is a disadvantage to a resident word-processing program: If you don't like it, you're stuck with it. The

EPS-80 contains one version of Pencil, while a newer version, 2.0, is already on the market. The version in the PMC machine uses the down arrow as the control key for numerous functions, an awkward arrangement; it has no centering and footer functions, loses letters if you type too fast on inserts and word wraps, and has an aggravating blinking cursor. All tabs are set eight spaces apart and cannot be altered.

On the hard output end of things, PMC's electric typing fingers required some patience to install. Two metal mounting plates with adhesive on their bottoms are placed on either side of the space bar. Two screws on the bottom of the actuator fit into the mounting pieces; two screws on top rest the metal ledge above the top row of typewriter keys. You must adjust the screws so the plastic typing fingers are .02-.08 inch above the keys.

After some trial and error, I got the actuator to work faithfully, pumping out 12 characters per second. This may seem slow for people used to fast dot-matrix printers, but once you compare the quality, you might find the wait worthwhile.

Left and right margins must be set manually. If you set the right margin too short, you will lose all characters from the margin to the point where the computer finds a carriage return or line feed. Also, the Silver Reed, like the IBM, will double space even when the single-space command is chosen. This is easily rectified by using the SX command, which suppresses line feeds following carriage returns, instead of S1 when for-

matting, or by manipulating switches on the ETF.

One of the biggest drawbacks of the EPS-80 is the amount of radio frequency interference it emits. To my surprise (and my neighbor's chagrin), the word processor completely blocked out TV channels 4, 5, and 7, and distorted the picture on channel 2. Wrapping the CPU in aluminum foil, then forming a cocoon around it of six aluminum roaster pans sewed together with number 20 bailing wire and grounded to a radiator, allowed me to get audio from the TV but the picture was still unrecognizable. And the neighbors still complained.

Despite the EPS-80's faults, PMC deserves kudos for releasing a "serious" micro aimed at the low end of the market—especially in light of recent trends, which seem to demand a low-end micro be a game machine first and computer second. So if you intend to get into bargain-basement word processing, the EPS-80 deserves your attention.

*(Personal Micro Computers Inc., 475 Ellis St., Mountain View, CA 94943. The EPS-80 currently sells for \$1,295 and includes an enhanced version of Electric Pencil, an electronic spreadsheet, a parallel printer interface, and a bus 40 card edge connector. The ETF-80 currently sells for \$595.)*

John P. Mello Jr.  
September 1982

★ ★ ★ 1/2

**LNW-80**

**LNW Research Corp.**

When I needed a hardware

upgrade, I shopped around and decided on the LNW. I ordered both circuit boards.

The circuit boards are silk-screened and solder-masked with plated-through holes. The processor board has a gold-plated edge connector.

The Z80A processor chip runs at 4 MHz. Most of the circuit design is a leap ahead of the TRS-80. The power supply has an off-board large filter capacitor. The design uses several IC regulators with overvoltage crowbars and fuses for protection. This circuitry is clustered in one section of the board to keep heat from sensitive circuits. Heat-sinks are mandatory on the regulators.

There are 90 ceramic and 18 tantalum capacitors on the board with only 0.1V of noise on the +5 line.

The memory design has Schmitt trigger devices on all lines; the address lines have series terminations resulting in a quiet memory. You must use 200nS memories with the 4 MHz clock.

The character-generator ROM can be a custom programmed 2716 or the TRS-80 character generator. The new Radio Shack ROM with lowercase descenders also works.

For the language ROMs, the two- or three-chip Radio Shack set or LNW six-chip set will do. For the six-chip set the ROMs are 2716s. The configuration is selected by a DIP switch of jumpers.

The processor speed is slowed to 1.77 MHz when the ROMs are accessed by inserting wait states in the Z80. When running the Basic interpreter, the speed increase over the TRS-80 is



around 40 percent (depending on the application). When running machine-language ROM-independent software, the LNW is more than twice the speed of the TRS-80.

The LNW-80 allows use of a Radio Shack keyboard or the LNW keyboard. There are two connectors on the circuit board. U30 and U34 must be removed to use the Radio Shack keyboard.

The foremost feature of the LNW is the on-board inclusion of high-density black-and-white and color graphics consisting of six additional 4116s with support circuitry. It is controlled by an Out command to port 254.

The graphics pixels use the same address space as the Basic ROMs, activated by an Out to port 254 with bit 3 set. The command deactivates Basic, making a machine-language driver necessary for graphics. This is fairly well documented in the manual. High- and low-resolution graphics and characters can be displayed at the same time.

The screen is divided into an inner and extended region. The inner region is 384 by 192 pixels. The extended region is another 96 pixels horizontally. Addressing for the two regions differs slightly.

For color graphics either a color monitor or a TV with on-board rfi modulator is used. Outputs for an RGB monitor can be wired in. There are two color modes: The first gives you 128 by 192 pixels in eight colors; the high-resolution mode gives 384 by 192 pixels. The bandwidth for this mode requires an RGB monitor. The color graphics requires a custom programmed

82S23 ROM, available from LNW. The ROM contents are detailed in the manual.

LNW lists a 24-by-80 character video screen in its advertising. This is a little deceptive. The LNW-80 has the capability of a 24-by-80 screen by using the high-density graphics, but to get it you have to write a driver program.

LNW's six-chip ROM set costs \$120. Considering that 2716s sell for about \$6 each, I decided to program my own. But which address blocks go in which ROM socket? I called LNW and was told the information was proprietary!

The expansion board is of the same high quality as the printed circuit board, except the gold plating costs extra. (The plating is worth it.) It has the same good memory design and Schmitt trigger drivers. There are 43 ceramic and 12 tantalum bypassing capacitors.

The board also has a few drawbacks. The printer port works well as long as you are only a few feet from the port. You never want to hook a long line to the output of a D-type flip-flop. My printer is 20 feet away and 74LS175 will not drive it. I had to add buffering.

They cut corners on the RS-232 port. Wire jumpers select the default RS-232 parameters. Why not use a DIP switch? The baud rate is also a wire jumper, and is not software-selectable. You can change soldered jumpers about three times before the pads fall off or the plating pulls out. LNW should have added the one extra chip to make baud rates selectable. Or they could have laid out the DIP switch on the circuit board.

The RS-232 connector pads are spaced for a printed-circuit mount, right-angle 25-pin connector. Do not solder one on the board. When the board is in place inside the LNW cabinet, there is no room to hook up to it with a plug. You cannot use a straight connector. Besides, the connector pads are 16 inches from the nearest opening in the case. Use a 25-wire extension cord.

The system expansion connector for outside devices is at right angles to the back of the case. You must remove the cover to use it.

Here is where the good designers at LNW went out to lunch en masse. The manual says the keyboard "is designed specifically for the LNW-80." The one I got was just a plain unencoded keyboard. A plastic bag with engraved keytops was included with instructions to replace certain keytops with the new ones. One of the new keytops was engraved sideways!

The key switches are the same wire-finger switches that gave Radio Shack grief for so long. With this system, if you don't have a keyboard debounce fix, you will get keybounce in spades at 4 MHz.

The circuit board is one-sided, with many wire jumpers on top of the board. The keyboard has to sit on them and is difficult to seat for soldering. Buy your keyboard somewhere else.

The LNW case is an attractive, low-profile cabinet made of heavy-gauge steel with room inside for both circuit boards, keyboard, power transformer, a muffin fan, and rfi filter. Unfortunately, it is overpriced.

The LNW is, for the most part,

a well designed, fast, effective computer. I use my LNW-80 with the Percom Doubler II and LDOS operating system. I have a time-of-day clock, plotter controller, music board, and 8-bit general-purpose port attached to my expansion bus. I designed and installed a piggyback board for selectable baud rates on the RS-232. All is working well.

Building an LNW-80 is not a project for a beginner. The manual is informative, but this is not a Heathkit.

*(LNW Research Corp., 2620 Walnut, Tustin, CA 92680; CPU \$89.95; System Expansion \$69.95 (\$84.95 with gold-plated contacts); Keyboard \$99.95; Computer Case \$84.95; System-Expansion Case \$59.95; CPU Hard-to-Find Parts Kit \$82; Video Hard-to-Find Parts Kit \$31; System-Expansion Hard-to-Find Parts Kit \$27.50; Level II ROM Six-Chip Set \$120.)*

Richard L. Brocaw  
May 1982

★ ★ ★ ★ ★

### **TRS-80 Color Computer Tandy/Radio Shack**

Tandy's Color Computer promises to be a powerful and expandable unit. The Color Computer has the same sleek silvery lines as its half-brother, the Model I, but unlike the early members of the Tandy family, the Color Computer abandons the Z80 microprocessor for the new Motorola MC6809E chip and will feature plug-in ROM (read-only memory) cartridges.

The keyboard has calculator-like buttons with a long throw

and tactile feedback. The power supply is totally self-contained. There are outlets for joysticks, cassettes, and a printer. The computer requires connection to a TV set.

The Color Computer incorporates several Model I features. The first is a Level I-like Basic, Color Basic. Whenever power is applied, or the reset button depressed, the computer displays a prompt of OK. Adding Extended Color Basic (Level II) will make the Color Computer much more powerful than its Z80 predecessor.

The second hand-me-down feature is a built-in cassette interface. The manual recommends the CTR-80, but we found our inexpensive, off-brand recorder worked just as well. However, if you don't buy Radio Shack's recorder, you will have to make the cables that lead from your recorder to the computer.

The Color Computer has a 600-baud serial printer port that is fully RS-232 compatible and interfaces to any Radio Shack serial printer. The serial interface responds whenever a LLIST or print-to-device command is given.

The permanent Color Basic memory of the computer is stored in a single 8K x 8 ROM. Extended Color Basic adds another 8K x 8. The basic Color Computer comes with 4K of dynamic RAM (random-access memory) that can be easily upgraded to 16K.

Many statements are left out of the manual. *Getting Started with Color Basic*. They are referenced on a "programming card," and this could be frustrating for the user who tries

something and continually gets an error.

Despite our unanswered questions, Tandy's Basic is capable of high-level computations with nine-digit precision. Tandy has also promised a new Extended Color Basic (Level II) with features including high-density color graphics (256 by 192), complex sound generation, save/load screen images, a real-time clock user-definable keys, and string arrays to 255 characters.

Several keys on the Color Computer have special or dedicated functions. The left arrow functions primarily as a back space. A shifted left arrow cancels the current line you are typing. This is similar to a control X command on other computer systems.

A break will interrupt the program in progress and return to the command level. It will break anything except a cassette routine, a print with no printer connected, or the sound command while it is executing.

The clear key will fill the screen with green blocks, effectively "clearing" the screen. The spacebar enters a blank character and moves the cursor one space forward.

During a List command or other data-display routine, shift/@ temporarily halts the program. Pressing any other key causes it to resume.

As the computer comes on, it is in an uppercase lock condition. Basic doesn't recognize lowercase characters, and the Color Computer can't display them. For text work (printing in upper- and lowercase), a shift/O should be depressed once to release the uppercase lock. The

same move will reinstate the up-percase.

The Color Computer in Color Basic uses letter combinations that most closely represent the error. There are 25 possible documented errors.

After months of playing, dissecting, and deciphering the Color Computer's hardware and software, we think it is a product with great potential and many applications from home to educational programs. A number of accessories are available for the Color Computer, including a cassette recorder, quick printer, modem, joysticks, and program packs. There is also a diagnostic ROM to help you locate any trouble spots in the Color Computer.

*(Tandy/Radio Shack, Fort Worth, TX 76102; \$399.95.)*

Tim Ahrens, Jack Browne, and  
Hunter Scales  
January 1981

★ ★ ★ ★ ★

### **The TRS-80 Model I**

**Tandy/Radio Shack**

Only one small computer system is designed for home consumption, assuming little or no background in either computer programming or electronics—the TRS-80 Model I.

Once it is set up, which takes only a few minutes, you are ready to begin with the Level I manual. The manual lives up to its reputation for being one of the best introductions to computers and their language. After you begin playing with the computer you enter a mental state known as manic-keypressive.

You won't want to stop until you get through the book as quickly as you can.

### **Flaws Along the Way**

While there are a few obvious errors with the instruction manual, I should have known there was going to be trouble when one of their examples referred to "Watts Law." You often have to refer to the answer pages to solve a problem and find that their solution requires information you were not given.

The graphics section is particularly troublesome. You are shown examples of graphics procedures, but are left to figure out exactly what is happening. You can imitate the manual, but you do not get a really firm grasp of what you are doing, and why.

### **The Next Plateau**

It will probably take you about two weeks to completely cover the manual; and that's where the next plateau begins. What can I do with the computer? This is where the Radio Shack manual lets you down. There is an information gap between learning the various mechanical computer functions, and developing working programs based upon your own information and needs.

There are a few sample programs in the back of the manual, but even if studied, they really only give you bits and pieces of information. Your learning depends on your ability to analyze.

An additional Level I Basic instruction course is available from Radio Shack for about \$13, which consists of eight taped lessons to be played on your computer. Each lesson contains

two to four parts. Unfortunately, for the material covered, I think that they could have done better with a small, inexpensive paperback.

There is no substitute for actual experience with the computer. Soon you will be thinking in computer terms and will find that you can build longer programs from your information.

This still leaves unanswered the question of what you can actually do with your computer.

There is potential in the Level I 4K, but it has its limitations. Much depends on how well you can take the individual language elements and combine them to do work. While you can store data or do record keeping with it, it takes computer memory that you don't really have.

What the machine does best is manipulate numbers repetitively. Let's take a simple example. Ohm's Law:  $E = IR$ . This elementary formula can solve many problems. If we assume 10V and 10 ohms we can solve for the current (I) by the use of  $I = E/R$ . For one value we can do it easily with a pocket calculator or even pencil and paper. In what Radio Shack calls the calculator mode, you can use your computer as a pocket calculator.

Now suppose you wanted to know the current as you varied both the voltage and the resistance, say 10-100V in 10V steps with the resistance changing at each voltage level from 1-10 ohms in 1-ohm steps. This takes 100 separate calculations. It would take some time to work this out with the pocket calculator or using the computer in its calculator mode. But you could write a short program to solve it. The computer takes off

on Run and you have the answers in just a few seconds.

Suppose that you now wanted to know the power for each set of values. We have 100 values, which means 100 more calculations for the power formula. This takes only a few extra lines added to the program to do this, and you end up with twice as much information.

Now, let's get real mean. Suppose you had a precision requirement and needed to know the values from 1-100V in 1V steps, and 1-100 ohms in 1-ohm steps. The program would be about the same length; just a few numbers would change. But to accomplish this you would be talking about 10,000 calculations, plus another 10,000 for the power formula. Try that on your pocket calculator or with pencil and paper!

While the computer can handle figures across the lines, Level I cannot add columns or handle titles in any convenient manner. This does not mean that the computer cannot be used for some form of data processing, but you will have to break the job up into smaller sections.

Even prepared 4K programs are limited in scope. An inventory program I recently saw allowed only 100 items with cost and inventory.

### **A Learning Tool**

By now it should be clear that what you really do with the Radio Shack TRS-80 is learn with it. It is an effective introduction to the computer. You will have to work at finding your own path into the computing world beyond, but help is out there: clubs, stores, magazines, and night schools. Good luck!

*(The TRS-80 Model I, Level I 4K, Tandy/Radio Shack, Fort Worth, TX 76102. The Model I is no longer being marketed by Tandy/Radio Shack, but has been replaced by its more sophisticated siblings.)*

Alexander Maclean  
August 1980

### —UPPER- LOWERCASE— MODIFICATIONS

★ ★ ★ 1/2

#### **CG-16 Symbol Generator G. P. Associates Model I**

The CG-16 Symbol Generator is an upper/lowercase and graphics generator board that is easily installed in any Model I configuration. It is compatible with both Electric Pencil and Scripsit.

In addition to lowercase with descenders and the standard TRS-80 graphics set, the CG-16

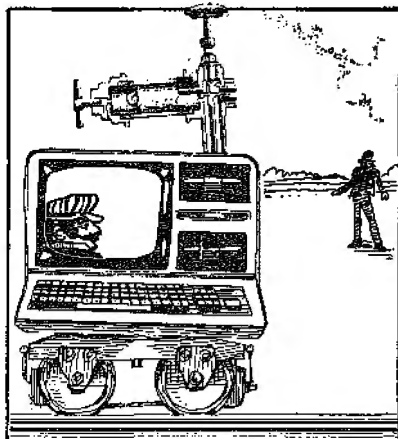
generates the complete ASCII character set and about 100 additional graphics symbols. Since many TRS-80 programs use the traditional coarse graphics, the board maintains compatibility and improves the appearance of existing graphics displays.

As for the lowercase video display with descenders—you have to see it to appreciate it. Reading the improved text display is definitely easier on the eyes. A selector switch lets you switch off the lowercase hardware if it is incompatible with the software you are running. This beats Radio Shack's permanently hard-wired mod.

To install the CG-16, the original TRS-80 graphics generator must be removed and retained for use in the new board. The job takes about 45 minutes—or less, if you are an experienced hardware hacker.

*(G.P. Associates, P.O. Box 22820, Sacramento, CA 95822, could not be reached.)*

Al Domuret  
March 1981



★ ★ ★

#### **LCA-47 Lowercase Adapter Micro Technical Products Inc. Color Computer**

The LCA-47 Lowercase Adapter improves the appearance of text on the Color Computer's video screen. It is a small printed circuit board that plugs into the socket normally occupied by the Color Computer's MC6847 video display generator IC; the IC, in turn, plugs into the LCA-47.

Installation of the LCA-47 re-

quires that the Color Computer's case and RF shield be opened, so all the usual precautions about the 90-day warranty apply. If you discover that the MC6847 is soldered into the motherboard rather than socketed, Micro Technical Products will refund your money. For the adventurous, they include instructions for sacrificing the IC and piggybacking the LCA-47 on the remains of the connecting pins. You must buy a new 6847 to make the computer operational, of course.

The LCA-47 board contains two DIP switches. One enables or disables the whole unit and the other reverses the foreground and background colors. The normal operation mode gives very readable dark characters on a light background. If you prefer the reverse, just flip the switch. The switches can be connected for remote operation instead of opening the computer's case each time.

The real value of this product lies in word processing. Users of Color Scripsit, C.C. Writer, Super Color Writer, and other color word-processing packages can now enjoy video output that looks like the printed page.

The Lowercase Adapter only solves the problem of character shape; line length and the number of lines per screen are unchanged from whatever is standard for a given program. The appearance of printed characters is unchanged as well, although you can get an accurate reproduction of the video display with a screen-print routine.

The LCA-47 is a well-made product that is supported well.

For the present, it makes the Color Computer a more finished product, and deserves the consideration of all Color Computer users engaged in text processing.

*(Micro Technical Products Inc., 814 W. Keating Ave., Mesa, AZ 85202; \$75, additional character generator ROMs \$25 each.)*

Scott L. Norman  
December 1982

### ★ ★ ★ **The Patch** **Cecdat Inc.** **Model I**

I've installed many dual-case mods, each one just a little different, but still basically the same. Then, along comes The Patch.

Ads for The Patch contained statements like "Electronic Module," "No software to load," "Operates like a standard typewriter," "Block cursor," "Special options available for five to ten dollars more." They convinced me to look into what The Patch could do.

The Patch allows you the option of either upper or lowercase characters. I tried it with TRSDOS 2.1, 2.2, 2.3, NEWDOS, and VTOS 3.0. Basic will accept lowercase but DOS will not. Fortunately, I had some zaps for NEWDOS to allow DOS to accept lowercase commands. CP/M accepted lowercase input but appeared on the screen in uppercase.

A problem occurred because one debounce in DOS plus one debounce in The Patch equals one key dead for 15 seconds. After I disabled the de-

bounce in NEWDOS, everything worked fine.

I highly recommend The Patch to anyone who is interested in having a reliable dual-case system with no software to worry about loading or losing. An extra plus is the one-year warranty you get with The Patch. I received mine in February 1980 and it has worked since without a hitch. (*The Patch 2.0 is sold by Cecdat Inc., P.O. Box 497, Hayden Lake, ID 83835; \$69.95.*)

Sal Navarro  
February 1980

## — MISCELLANEOUS —

### ★ ★ ★ **80-Graphix Board** **Programma International** **Model I**

The 80-Graphix board is offered by Programma International as a solution to the TRS-80's low-resolution graphics. In return for high resolution you get a programmable graphics generator complete with lowercase.

The board comes with easy-to-follow directions. Programming it is straightforward, but consumes a great deal of memory. The video memory is used to program the characters in the character generator.

The board offers three graphics modes: low resolution (normal graphics), high resolution, and program modes. These modes are accessed through the

cassette port, which has one major advantage: They can be easily accessed in either Basic or Assembly language.

Programma International's advertisements are slightly misleading. You can use their graphics and still access your normal graphics, but not both at the same time. When you send your graphics characters to the printer, it reverses upper and lowercase. Once you are in the program mode, the screen is blanked out, and your previously devised data statements POKE the desired characters into the board's RAM.

Also, the software shown in the ad displays the entire character set, and says which character you are on. The program you actually get only works with one character at a time, and does not display or construct an entire set. To use this generator in a programming situation, you must first define all your characters, one byte at a time, write down the data statements, and type them into your program.

If what you want is vector graphics, then I cannot recommend this board. However, if a graphics generator is your heart's desire, and you are willing to spend a rather large amount of time programming the board, the 80-Graphix board may be the ideal add-on for you. Once programmed, the switch between high resolution and normal graphics is easy in both Assembly code and in Basic. (*80-Graphix Board was sold by Programma International, Los Angeles, CA, which has gone out of business.*)

Bruce Douglass  
February 1981



★ ★ ★

**Acu-Data Tape Digitizer**  
**Alphanetics Mfg.**  
**Model I**

The Acu-Data is designed to facilitate CLOAD and tape duplication. It is a combination filter, rectifier, and pulse-shaping device. Placed at the recorder output, it conditions the tape data. Hum and other spurious noise is filtered out. Data pulses are then rectified and shaped to ensure that proper pulse amplitude and timing is achieved before the computer ever sees the pulse train.

The unit runs on 110V ac, so batteries are not necessary. Other features include an LED to indicate the presence and amplitude level of data, a polarity switch that allows the user to select either positive- or negative-going data pulses, and a copying digital output jack that allows duping of processed program tapes. There is also an optional switch that provides computer control of the recorder.

The Acu-Data is on line whenever it is plugged into a 110V ac outlet. No provision for removing ac from the primary of the line transformer is made and the transformer is forced to dissipate a respectable amount of hysteresis-generated heat.

The comprehensive user's manual is clearly written and includes many hints on using tapes, and there is an impressive warranty. The Acu-Data has proven to be a miracle worker when it comes to salvaging marginal tapes. Even lousy originals can generate good copies when the copying digital output is used for duplicating. Alphanetics has a winner; for

tape-oriented computerists who prefer to think in terms of black boxes and have no desire for breadboard projects, the Acu-Data is a worthwhile investment. (*Alphanetics Mfg., of Forestville, CA, is no longer in business.*)

Chris Brown  
 June 1980

★ ★ ★ ★

**AD-III Audio Amplifier**  
**Thomas Engineering/TCS**  
**Systems**  
**Model III**

I used T-Beep to provide sound with my Model I. However, when I hooked up to my new Model III, T-Beep locked on after CSAVE, requiring a keyboard entry of OUT255.0 to turn it off—a real bother. Then I saw a news release offering an internal audio amplifier for the Model III.

The TCS Systems Model AD-III 1-watt audio amplifier claims to have high-quality sound and is designed with a switching system for deactivating the amplifier during data-tape transfer. The price included the amplifier board, speaker, all wiring, instructions, and a 90-day replacement warranty.

Installation was described as "six quick connections to the CPU board, mounting the amplifier and speaker and you are up and running." I was not thrilled about wiring to the CPU board, however.

The AD-III arrived completely assembled and prewired. The step-by-step installation instructions included several perspective-view sheet drawings of the Model III disassembly, including screw locations. A

photocopy of the CPU board shows exactly where to solder each color-coded wire.

The assembled amplifier board includes an LM386 integrated-circuit amplifier, 5V relay, volume control, and four capacitors. The 2-inch diameter squareframe speaker is already wired to the amplifier board, and the six wires requiring connection to the CPU board are harnessed for easy handling—a first-class job.

After connecting the AD-III, press enter, and you should hear a short buzz. You are stuck with this one sound unless you use additional programming.

I am delighted with my AD-III. The price was right, the relay disables the amplifier during CSAVE, and installation was smooth and fast.

*(Thomas Engineering/TCS Systems, 22742 Cortes Drive, Novi, MI 48050; \$29.95.)*

Fred Blechman  
February 1982

★★★★

**Beta-80**  
**Meca**

★★★★

**Exatron Stringy Floppy**  
**Exatron**  
**Model I**

The Stringy Floppy by Exatron and the Beta-80 by Meca provide alternative investments to an expansion interface.

The Exatron Stringy Floppy (ESF) is designed for the nonhardware-type programmer. The wafers seem quite reliable. The speed and the data-handling

capabilities of the ESF are impressive. Machine-language programs can also be placed on the wafers.

Like the ESF, the Beta-80 plugs into the keyboard. With some sacrifice of speed, it offers you much of the convenience of a disk. The actual data transfer rate is slower than the ESF, although double density will be available that will increase its speed considerably.

Once the Beta-80 cassette is inserted, it is totally software-controlled. It can do a type of merge for programs, and it also can chain programs together using load statements. Data is saved only by saving arrays, but one must convert strings to numeric data in Basic first. The Beta-80 also saves machine-language programs.

Both the ESF and the Beta-80 have performed very well for me. Both units can give a powerful system without an expansion interface or additional memory. Both leave CLOAD and CSAVE totally functional, so your regular audio cassette can be used as a back-up. This also allows you to transfer existing audio cassette software to the new media.

The ESF requires frequent switching of wafers, but should be adequate for the hobbyist, average home computer enthusiast, and some small businesses. If you want the luxury of the directory, universal media (digital quality cassette), lots of on line storage, and the proven Phi-deck, then the Beta-80 may be for you. Most TRS-80 owners deserve more than audio cassette and such computerists should seriously consider the ESF and Beta-80.

*(The Exatron String Floppy is sold by Exatron, 181 Commercial St., Sunnyvale, CA 94086; \$99.50. The Beta-80 is sold by Meca, P.O. Box 696, Yucca Valley, CA 92284; it originally sold for \$699, but Meca could not be reached for an update.)*

Harley Dyk  
May 1980

★ ★ ★ ★

**Cryptext  
Cryptext Corp.  
Model I**

Cryptext is a combination of software and hardware that allows storage of almost any type of data, on either tape or disk, in an unrecognizable, unbreakable code.

The code can only be decoded if you use the exact original encoding key.

There is a built-in timer delay between keyboard input of the key and the negative response of the unit to prevent someone from trying to guess the right key.

Once Cryptext has its key, which isn't recorded in memory and can't be found by using PEEK or any other machine-language trick, it is simple to encode data. You send your data to the encoder 1 byte at a time, receive the altered, encrypted byte, and then save the byte. This is repeated as many times as is necessary for the data you want to store.

To decode your data, give Cryptext your 10-byte key, feed it the encoded data, and retrieve the decoded characters.

The 26-page manual is thorough, and includes a simple test program, command-struct-

ure summary, and pin-out diagram, as well as lessons on how to use the Cryptext commands in your Basic or machine-language programs.

Cryptext is a major advance in data protection, eliminating almost any possibility of someone stealing or altering your vital programs.

*(Cryptext Corp., P.O. Box 425, Northgate Station, Seattle, WA 98125; \$699.)*

Terry Kepner  
January 1981

★ ★ ★

**EPROM Programmer  
Model AN-551  
Design Solution Inc.  
Model I**

Need an economical method of programming your own EPROMs? The Model AN-551 EPROM programmer manufactured by Design Solution Inc. is a simple unit that may solve your problem.

The programmer comes assembled and tested with its own power supply and instruction manual. The unit will program the two most common EPROMs on the market today: the 2716, a 2K-byte device, and the 2732, a 4K device.

The instruction manual contains all the information on the concept, connection and programming. Two Basic programs let you program the EPROMs. Although the original advertisement for the programmer stated that information could be copied from ROM to EPROM or load TRS-80 memory from either ROM or EPROM, the software listings do not include this facility. However, since the programs

are in Basic, this is fairly easy to accomplish this yourself.

The unit is in a plastic case measuring 6 by 4 by 1 inches, and has a 40-pin edge-card connector protruding from one side. You must construct a special cable to connect the programmer from this bus to the system. The cable is simply a ribbon cable that has a female 40-pin edge-card connector on opposite sides of the cable. A standard extension cable will not work because of the configuration of the EPROM edge-card bus.

On the face of the unit are three switches that control the power on/off, write/read, and 2716/2732 mode options. Two LEDs indicate the status of these switches. There is a standard 24-pin socket for the EPROM. No zero-insertion force socket is provided, although you could easily install one. A jack on the top of the unit provides a connection for the power supply. The power supply is 12V ac, not dc as you might normally expect. (The programmer requires 26V during the programming mode and a dc adapter would not supply that.)

Once the EPROM is blank you have to enter the information you want to transfer into memory. The software listings will not load a program into that location for you. You will have to use a machine-language monitor or devise some other method of loading the program at 7000H (starting address of data transfer).

The software programs the EPROM and verifies it for you in about 300 seconds. If you obtain an error during the verification process all you can do is erase the EPROM and try again. If er-

rors are still present, you may have a bad EPROM.

This excellent device needs a better software support program. It should have been written in machine language and include the options mentioned in the advertisement.

*(Design Solution Inc., P.O. Box 1225, Fayetteville, AR 72701. The EPROM Programmer Model An-551 originally sold for \$99.95. Design Solution Inc. could not be reached for update information.)*

Mel Patrick  
June/July 1982

### **Exatron Stringy Floppy** (See Page 366)

★ ★ ★ ★ ★  
**Gold Plug 80**  
**E.A.P. Company**  
**Model I**

The Gold Plug 80 is a permanent cure for contact oxidation on Model I edge connectors. It is a well-made device consisting of an edge-card plug with gold-plated contacts, available with either 34 or 40 contacts. The rear of the plug has terminal tabs that fit exactly over the existing foil fingers on the TRS-80's connectors.

After installation, the original plugs have been extended about a half inch, meaning the plastic door covers no longer fit. E.A.P. cautions you about this in their advertising leaflet. They also have an excellent policy of permitting you to return any plugs for a refund.

An excellent set of instructions accompanies the plugs, and they

are shipped promptly.

Installation requires a soldering iron, rosin-core solder, a Phillips screwdriver, and your last pink pearl. I have some soldering experience, but it proved to be an easy, safe job. The contact is heated, a very small amount of solder applied, and then you go on to the next contact. It took about an hour to do all six plugs.

The Gold Plug 80 set I bought included all six plugs. The plugs are available individually for \$9.95, or you can get a pair for the keyboard-to-expansion interface cable for \$18.95.

As I said earlier, I did resolder every connector on the machine, and I haven't had a single unwanted reset since.

*(E.A.P. Company, P.O. Box 14, Keller, TX, 76248; \$9.95 each, \$18.95 set of two.)*

Robert Jacobs  
May 1982

### ★ ★ ★

#### **Home Computer Work Table Computer Roomers Inc.**

When microcomputing was "just a hobby," we were content to stick the computer in a back room of the house. But then we became computer-holics, only leaving the machine to go to work and eat, so we centrally located our TRS-80 in the living room.

We needed a table for it that was firm, well built, able to provide a home for peripherals, and able to provide a work space. The table had to look nice and be reasonably priced. This describes a computer table from Computer Roomers Inc., which has a nice walnut wood grain

look with black trim.

A shelf on the rear of the table runs along the entire width of the table and is the exact depth of our Shack disk drive and probably most 5-inch disk drives. An open space along the rear allows cables and wires to be routed neatly out the back.

The table is easily assembled, but while the instruction manual pictures the desk with two "wing" extension shelves, only one can be used at a time.

We thoroughly enjoy our new table, but some computerists might find its lack of drawers a slight disadvantage.

*(Computer Roomers Inc., 1250 Majesty, Dallas, TX 75247; \$179.95. Price varies depending on size.)*

Dan Keen and Dave Dischert  
October 1981

### ★ ★ ★

#### **LSI Soft-view CRT Langley-St. Clair Instrumentation Systems Inc.**

The LSI Soft-view CRT combats eye fatigue stemming from the cathode ray tube Tandy uses in the manufacture of the TRS-80. These replacement CRTs are available with either a slow-decay green phosphor or a medium-decay amber phosphor. The medium-decay phosphors fade at a rate nearly 100 times slower than the white phosphors.

The CRT comes with an eight page installation booklet that includes fully detailed installation procedures and a list of the tools needed if you plan to do your own installation. (The alternative is to rely on your local computer

## HARDWARE

---

center or TV repair shop for the installation.) This manual doesn't include any pictures or diagrams, but it is very easy to follow just the same.

The amber phosphors don't glow with the same intensity as the white. They are only 42 percent as efficient in their light-emitting properties. In addition to this, the new CRT features both the double dark leaded glass (standard) and the anti-glare surface (optional), which further reduces the screen brightness.

I became accustomed to the subdued amber screen very quickly, tending to be unconscious of the color and of the absence of flicker and glare until forced to use a standard machine. Having used the amber CRT, I found the black-and-white screen harsh and discomforting.

If you use your TRS-80 frequently or for extended periods at a single sitting, the improved viewing comfort and health features of these replacement CRTs are very much worth the price.

*(Langley-St. Clair Instrumentation Systems Inc., 132 W. 24th St., New York, NY 10011; \$80 - \$100.)*

Dave Smith  
December 1982

★ ★ ★ 1/2  
**Maxi-Disk**

★ ★ ★ 1/2  
**Shuffleboard**  
**Parasitic Engineering**  
**Model I**

Adding Parasitic's Maxi-Disk

gives the TRS-80 owner full-size floppy drives that more than triple the storage capacity per drive. Furthermore, installing Parasitic's Shuffleboard in a Maxi-Disk system converts your TRS-80 into a Z80 CP/M with access to either system through a simple command.

The Maxi-Disk package has everything you'll need for altering a TRS-80 to use 8-inch floppies. Photographs included with the instructions show you the correct installation procedure. Parasitic provides a disk program called DOSPATCH for the final software modification.

By adding this hardware to the TRS-80, the disk storage capacity per drive is tripled and you also have access to an unrivaled new software realm.

*(Maxi-Disk and Shuffleboard are sold by Parasitic Engineering, Berkeley, CA 94703; \$995 and \$249, respectively.)*

Kevin Cohan  
May 1980

★ ★ ★ 1/2  
**MAYDAY + S**  
**Sun-Research Inc.**  
**Model I and II**

Mayday is a fail-safe device that, in the event of a drop in line voltage, provides emergency power to a microcomputer system. Once on emergency power, you can terminate the system in an orderly fashion with no program crash or loss of data. Emergency operation time varies with the Mayday model and the size of your system, but the Mayday + S can generate up to 30 minutes of power.

The Mayday uses a modified,

120 Hz, square wave generator as a dc-to-ac converter. When power fails, this generator supplies power from a 12V battery to the computer. A specially designed isolation transformer allows the Mayday to maintain plus or minus 5 percent required computer power during switch over to internal power. Switch over time is about 5 milliseconds, so no memory loss occurs.

The computer monitor, interface, keyboard, and peripherals plug into a bank of outlets on Mayday's front panel. The entire system is controlled by one circuit breaker, while a light indicates system status. Also, most versions of the Mayday incorporate an MDS line surge protector.

One drawback of the MAYDAY + S is its inability to power 60 Hz ac fan motors used in some disk units and printers. To alleviate this problem, Sun-Research has developed a unit that supplies a 60 Hz sine-wave source of voltage. This new Mayday is suitable for use with Model II machines. In either version, a Mayday unit can be your first line of defense against system glitches.

*(MAYDAY + S is sold by Sun-Research Inc., P.O. Box 210, New Durham, NH 03855; \$325.)*

Chris Brown  
June 1980

★ ★ ★

**The MicroCompatible Inc.  
Plug Compatible Processor  
MicroCompatible Inc.  
Model I**

The MicroCompatible Inc.

Plug Compatible Processor (PCP) is just what the name implies—a plug compatible processor. It plugs into the keyboard expansion port and is complete with Z80A processor and support circuitry.

The 13-by-2-by-3-inch PCP sits on the desk behind the keyboard. A 40-pin connector plugs into the keyboard expansion port. The cable to the expansion interface plugs into the top of the PCP. All that remains is to plug the power cord into the wall.

The PCP contains a 4 MHz Z80A CPU. Support circuitry in the PCP generates special RAS, CAS, and MUX signals that allow reliable 4 MHz operation without inserting wait states to allow the CPU to wait on the RAM. (Merely speeding up the Z80A to 4 MHz will exceed the timing specifications on the 4116-2 RAM chips.) In fact, after simply changing a jumper plug inside the PCP, I've been running reliably at 4.6 MHz.

Several additional problems could arise with 4 MHz operation. The Model I uses software timing loops to generate certain timing signals. In order to read a tape which was written at normal speed, the processor must again be slowed to the original speed.

Also, floppies will not work at high speed with TRSDOS. When the TRS-80 is running at high speed, the control signals timed by the software loops are no longer compatible with the floppy drives. The software keyboard debounce routine also uses a timing loop. At high speed the time in the loop is more than cut in half. If the processor is not slowed down

during keyboard accesses, some keybounce may result. The PCP automatically returns to 1.77 MHz during disk accesses and keystrokes to avoid these problems. A high-speed disk switch is provided for NEWDOS 80 users.

Owners of early Model I's that haven't been modified receive an additional benefit. The PCP allows the use of the expansion interface. Early Model I's had a +5V lead brought out to the keyboard edge connector. This same pin was conveniently grounded in the expansion interface. Plug the early Model I's into an expansion interface and no more +5.

*(The Plug Compatible Processor once sold by MicroCompatible Inc., P.O. Box 7624, Atlanta, GA 30357, is no longer available.)*

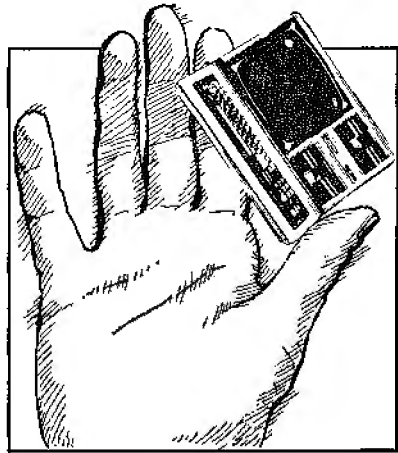
Paul Vonk  
June 1981

★ ★ ★

### **Plug 'n Power Controller Tandy/Radio Shack Model I**

The Plug 'n Power Controller connects to the cassette recorder jack on the Model I keyboard and plugs into an ordinary electrical outlet. The instructions cover use with the Model I, Level I and Level II, Model I TRSDOS, Model III, and the Color Computer.

Once loaded, the software prompts for the correct time to the nearest minute. It provides a real-time display of hours and minutes in the lower left of the screen. Times must be entered in the 24-hour system (where 17:00 equals 5 p.m.).



To control a light, a radio, or another small appliance, you need a BSR-type module for each appliance. The device to be controlled is plugged into the module. A one-letter house code and a one-number unit code are selected on the controller module. The module is plugged into any polarized electrical outlet.

A unit is turned on by entering the house code, the unit code, and the desired activity (on/off for appliance modules, on/off, or dim and bright for lamp modules). There is no feedback from the controlled module to the software. The status is only an indication of software-issued commands, not the actual condition of the module.

Commands are provided to insert, replace and delete command lines, page the display, jump through the list and exit the programming mode to the command mode.

Exiting to TRSDOS and using the DOS Dump command transfers the controller file to disk. Once the file is saved, the instructions don't tell you when or



how to get the file back into operation. The only clue is in the file name used in the example—DATA/CMD. You reboot the program by typing the file name used.

One disadvantage is the lack of any back-up in the event of power failure. The software has no automatic restart feature and the clock has to be reset each time the system is rebooted. If you are planning on using it for security lighting over weekends and vacation trips, be forewarned.

*(Tandy/Radio Shack, Fort Worth, TX 76102; \$39.95.)*

John W. Warne  
March 1982

★ ★ ★ ★ 1/2

**Rex 80 ROM Extender  
Personal Micro Computers  
Inc.  
Model I**

When I acquired a JPC cassette system, I looked for an answer to solve some of my software problems, particularly the JPC driver program that resides in its memory.

The Rex 80 ROM Extender is a hardware solution to my software problems. The Extender's advertisement boasted the use of 2,014 bytes set aside in a reserved area for your own programs. There was apparently no conflict between the Rex 80 and the standard Level II ROM. The advertisement also said that the board was compatible with most of the EPROMs on the market.

The board itself is approximately 24 inches square with an edge-card bus on two opposite sides. Four rubber feet on

3/4-inch standoffs (spacers) support the unit. The P.C. board is double-sided, has a 24-pin Tex-tool zero insertion force socket for your EPROM, and a 5V regulator for the power supply. An ac adapter is required to power the regulator chip on the PC board. It plugs into a female miniature phone jack on the board. Only the address lines are fully buffered on the board. This did create a problem but was easily fixed. Two pages of documentation accompany the Rex 80 for connection, operation and memory map reference.

Was it worth all the trouble of installing and EPROM programming? Yes! Having all of those routines on-line, never having to remember memory sizes and being able to insert different EPROMs for other tasks, definitely makes the TRS-80 a much more powerful computer system. *(The Model I, Level II, Rex 80 ROM Extender, sold by Personal Micro Computers Inc., 475 Ellis St., Mountain View, CA 94043, is no longer available.)*

Mel Patrick  
December 1981

**Shuffleboard**  
(See Page 370)

★ ★ ★  
**Silver-It  
Fuller Software  
Model I**

As with most TRS-80 users, I was slowly being driven crazy by the unreliable contacts between the keyboard and the expansion interface. My 80 had reached the point where the

well-known trick of cleaning the circuit board connections with an eraser was required before each session! Then a small Fuller Software ad caught my eye. They promised that a \$5 investment in Silver-It would put an end to spontaneous reboots. I ordered, and the kit arrived within one week.

Imagine my surprise when I opened the package and found no instructions for use. A disclaimer warned the purchaser that silver soldering is not for the beginner and that computer circuits are delicate; Fuller offers a full refund to anyone who does not feel up to the challenge, but I feel this warning should be included in the advertisement.

The kit consists of a small piece of solder wick and several inches of silver solder. You need a 25-watt soldering iron, absolute alcohol to clean the contacts (available at liquor stores under the name "Everclear"—don't try to substitute with rubbing alcohol), and liquid rosin flux. Be certain not to use the acid flux that is generally used for soldering.

Place the keyboard face down on a clean cloth. Remove the six screws from the bottom, keeping track of which screws belong in which holes, and place the base off to one side. Very gently lift the main circuit board; you will see spacers separating it from the keyboard circuit. Remove the spacers without flexing the cable connecting the two boards, then lift both boards out as a unit. Use the alcohol to clean both sides of the circuit board contacts and allow them to dry.

Radio Shack saved a few pennies by coating the contacts with regular solder instead of silver or

gold; this must be removed. Heat each contact with the soldering iron while using the solder wick to remove the old solder. Be careful not to cause a solder bridge between contacts. Next, clean the contacts again and immediately coat the surface with rosin flux.

Applying the silver solder is relatively simple. Start at the outer edge and heat the contacts with the iron. Touch the silver solder to the edge, at the same time starting to move the iron and solder toward the center of the board. Keep the solder just behind the iron in continuous contact with the board. When the inner edge is reached, lift both off together. You should achieve an even, shiny coating without much trouble. After all the contacts on both sides are soldered, use the alcohol to remove any residual flux.

After checking your work or solder bridges or lumps in the solder track, gently place the circuit boards back in the keyboard top. Very carefully reinstall the plastic separators on the posts and reassemble the base.

The interface procedure is exactly the same. I silver-soldered all the edge connectors on the interface board in hopes of avoiding any future problems. The entire procedure took about two hours.

Since reconnecting the computer six months ago, I have found no immediate problems. Everything still functions perfectly; no spontaneous reboots and no more eraser cleaning sessions!

This method, however, is not a cure to be applied by anyone unfamiliar with good soldering techniques or unaware of the

delicacy of computer circuits. With the keyboard wide open, stray static discharge could easily zap a chip. If you haven't soldered along edge connectors before, I suggest that you pick up a blank board with connectors at Radio Shack and practice first. There is more than enough silver solder to do this provided you don't go overboard.

*(Silver-It is still available from Fuller Software, 630 East Springdale, Grand Prairie, TX 75051.)*

Paul R. Prescott  
July 1981

★ ★ ★ ★

**Starbuck 8882**  
**Starbuck Data Company**  
**Model I and III**

The Starbuck 8882 is a data-acquisition and control system. The purpose of this black box unit is to relay data between the real world and the TRS-80. The TRS-80 can receive analog or digital data and can respond by setting bits in an output port in the 8882. Normally, the only input the TRS-80 ever sees is from character-oriented I/O such as the keyboard, cassette and possibly disk units. The Starbuck allows data to be collected in two main ways: A continuously varying analog voltage can be sampled from any one of eight input lines, and eight lines can be polled to detect open or closed switches connected to the unit.

The documentation has all the information you are ever likely to need. It is 75 pages in length and although the print is of only medium quality, its contents

should serve as a model to any manufacturer of peripheral hardware for the TRS-80. The soft-bound manual is arranged in such a way that engineers won't be disappointed: the more advanced information (including the circuit diagram of the 8882) can be skipped by the less-demanding user. It's refreshing to see such information made available rather than kept under a shroud of secrecy.

The possible applications of the 8882 are well covered. The circuit diagrams in the manual are simple op-amp designs with a few accompanying components, amounting to about \$5 worth of parts. One of the most exciting possibilities mentioned is interfacing to a Sears Roebuck BSR-X10 master control unit allowing ultrasonic control of household items such as lights and dimmers with your TRS-80.

The final section of the manual contains all the software listings required to use the 8882. This software is also available on disk as part of the complete system or separately if required.

The hardware is literally a black box and acts in a traditional black-box fashion: Signals are input through one side of it, and after being processed, emerge on the other side in a different form. The top of the box has three rows of eight screw-on terminal pairs; two rows are for digital I/O, and one row is for analog input. Two DIN sockets are provided for handling the joystick inputs. The box measures 8 by 7 by 3 inches, making it a reasonable size to have next to your TRS-80.

Power is supplied in the form of 12V ac at 200 mA from a small wall transformer. I have

reservations about this kind of power supply for a number of reasons. There is a small chance that by inserting or removing the 12V supply from the 8882 you could short out the mini-jack plug and damage the power-supply transformer. You are warned against this in the documentation, but the problem could have been entirely eliminated by having the supply integral within the box. It also means you can't switch the unit off by removing ac power to it from a single switch unless you resort to a switched multi-way ac outlet.

The unit is nothing short of excellent as regards its use in data acquisition and control on a micro. The uses for this type of hardware are as many and varied as the uses of the computer itself. The inputs available can be either digital or analog and are addressed from the TRS-80 via ports 160 and 161 (A0 and A1 hexadecimal) on the Model I version. The Model III version uses different ports, to be exact, 96 and 97 (60 and 61 hexadecimal), for the analog and digital inputs.

The only extra that could have been provided would have been an analog output. Even this can be achieved by using an electronic ladder network on the digital outputs to sum the voltages; up to 256 different voltage levels can be attained. The sky's the limit—you can now control the world from your TRS-80.  
*(Starbuck Data Company, P.O. Box 24, Newton Lower Falls, MA 02162; \$230.)*

Jake Commander  
August 1982

★ ★ ★ 1/2  
**TC-8 Cassette System**  
**JPC Products Inc.**  
**Model I**

The TC-8 Cassette System is a plug-in device that stores programs five times faster because it uses higher-density data. The computer outputs the information at a higher rate out of the rear keyboard connector.

JPC claimed anyone could build it and it's true. The manual is clear and concise. Important part placements are stressed (polarity markings on electrolytics, bands on diodes, etc.). The board quality is excellent; it is double-sided and parts locations are clearly marked on the components side. There are absolutely no adjustments or settings to bother with.

The documentation is written in the nicest format I've seen in a while. Each command or subject is covered on its own sheet in large type. All explanations are in easy-to-read English—not computerese.

Unlike the standard tape system, you can position your tape anywhere before the program you want and not have to look for a blank spot between programs. The TC-8 starts loading without getting confused by the portion of the previous program you just fed it. So far I've found no bad loads.  
*(The TC-8 Cassette System is sold by JPC Products Inc., 12021 Paisano Court, Albuquerque, NM 87112; kit \$89.95, assembled \$119.95. The TC-8 Cassette System will soon be out for the Color Computer.)*

Carl A. Kollar  
June 1980

**TC-8 Cassette System**

(See Page 350)

★ ★ ★

**Video I  
Archbold Electronics  
Model I**

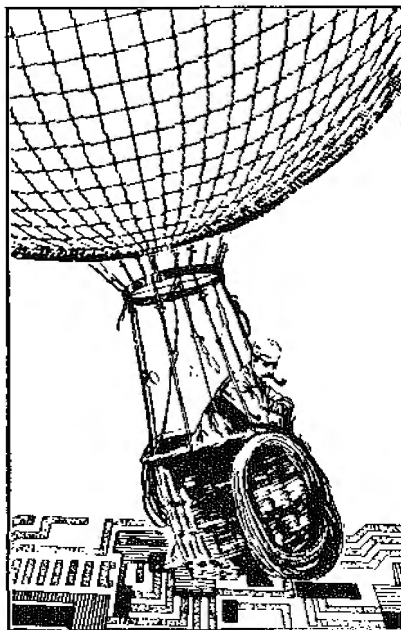
The Video I reverse video modification kit is a reasonable monitor enhancement for the price. The kit includes two pre-assembled PC boards, two resistors, one zener diode, a double-sided adhesive pad, hookup wire, and solder. One PC board mounts inside the keyboard housing and the other inside the video monitor.

**Assembling the Kit**

After hooking up the interconnecting cables, I crossed my fingers and turned on the computer. The video display came up normally, white on grey. I typed X = INP(2) and pressed Enter. The screen background lightened and the print became black. I then typed X = INP(1) and pressed Enter—the display returned to normal. I ran the test listing in the instruction to check for display linearity and found no distortion.

The address decoder scheme is nonabsolute: address bits A7 and A5-A2 2 are not used in the decoding scheme. Any 8-bit address with bit 7(A6) low and bit 2(A1) high causes a reverse video condition. Whenever bit 7(A6) goes high, the decoder circuit logic remains unchanged. A high on bit 1(A0) and a low on bit 7(A6) will decode and switch to normal video.

In the reverse video mode, dot data from Z30 in the video missing circuit is fed into the reverse video board, inverted, and fed in-



to Z41. Z41 then inverts the dot data (it goes high). This results in a light background and black character dots.

**One Little Problem**

The kit was easy to assemble and install. The only problem I found was in the operation section of the instructions. Pressing keys 7, 8, and 1 simultaneously is supposed to produce normal video, while keys 7, 8, and 2 should give reverse video. This function does not work on my unit. I suspect it works in the Level I unit—the instructions show a keyboard without the satellite ROM board, the obvious modification in Level II.

(Archbold Electronics, 10708 Sequoia Way, Rancho Cordova, CA 95670. Video I is no longer available.)

Howard Walker  
May 1982



# BOOKS







★ 1/2

**55 Color Computer Programs  
for the Home, School &  
Office**

**Ron Clark**

**ARCsoft Publishers**

This is one of the few much-needed books devoted exclusively to the Color Computer. Unfortunately, few of the programs it contains go beyond the level of the exercises presented in the Color Basic and Extended Color Basic manuals.

The book is divided into six major sections: Business Programs, Programs for Students, Programs for Teachers, Music Programs, Programs for the Home, and Graphics Art. An appendix essentially duplicates the operator's information cards for the Color Computer.

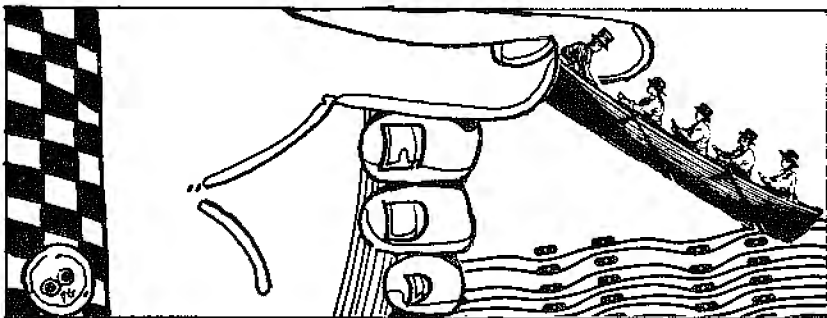
The 14 business programs have a professional air to them (except for a cutesy "Executive Go/No Go," which returns a yes or no answer depending on the value of a random number), but suffer from a generally simplistic approach. They seem intended for a 16K cassette system, though many would probably run in 4K; few people would choose such a system for business purposes. There is nothing

here about file handling, although even a cassette recorder gives you the ability to work with sequential files. Instead, all information is meant to be entered from the keyboard at run time. Students reading this section can get a feel for the elements of business computer applications, but they will learn only the elementary use of Extended Color Basic.

Many of the student programs fare little better. Four are devoted to simple quizzes in which an item and its match are read from a data statement. If you answer incorrectly, the computer reveals the answer; there is no storage of missed questions for subsequent review. Two of the four teacher's programs deal with examination score evaluations and perform useful functions. Once again everything must be entered from the keyboard.

Roughly half the book is devoted to recreational programs that generate music, draw pictures, and perform miscellaneous domestic calculations. These are moderately amusing, but offer the purchaser little of real value.

The Color Computer is becoming relatively sophisticated (extended memory and disk drives



are readily available), and there is a need for literature that will help owners make full use of the machine's capabilities. Unless you feel a need for additional programs much like those in the manuals, there is little to be learned from this particular collection.

(ARCsoft Publishers, P.O. Box 132 Woodboro, MD 21798; Softcover, 127 pp., \$9.95.)

Scott L. Norman  
September 1982

★ ★ ★

**57 Practical Programs and Games in Basic**  
**Ken Tracton**  
**Tandy/Radio Shack**

*57 Practical Programs and Games in Basic* is well organized and written. Each chapter features a formula, a flowchart or diagram, and a program. A sample run of the program lets you see what it does.

Although there is a good selection of programs, the author fails to organize by subject in his index. Also, the book only offers two rather simple financial programs, a compound-interest program, and a depreciation program. I would have preferred the inclusion of more comprehensive programs; however, the ones included can serve as guides to writing your own.

I found this book useful. I recommend it to anyone new to microcomputing.

(Tandy/Radio Shack, Fort Worth, TX 76102. This book is no longer available.)

Merrilyn Ferris  
April 1981

★ 1/2

**57 Practical Programs & Games in Basic**  
**Ken Tracton**  
**Tandy/Radio Shack**

This book contains a wide variety of programs including Moments, Skewness, Kurtosis, and Space Wars, but many of them do not work. The errors contained in the programs range from the obvious, like a missing parenthesis, to very obscure ones such as reversed variables in assignment lines.

One good point is that the programs are flowcharted adequately. This can help you learn the relationship between program logic and flowcharting principles.

This book does have many special-interest programs, but be aware that it may take some debugging to get them to work on the 80.

(57 Practical Programs & Games in Basic, marketed by Tandy/Radio Shack, Fort Worth, TX 76102, is no longer available.)

James R. Fatz  
May 1980

★ ★ ★ ★

**1001 Things to Do with Your Personal Computer**  
**Mark Sawusch**  
**Tab Books Inc.**

Probably the most annoying question aimed at computerists by non-computerists is "OK, so what can you do with it?" Mark Sawusch addressed this question and came up with over 1000 answers!

Sawusch's 355-page book is an amazing collection of prac-

tical ideas and programs divided into 12 broad categories, and includes a glossary of programming terminology, an appendix of financial formulas and metric conversions, and an index. Each category contains at least several, and as many as dozens of potential applications. A run through the chapter titles indicates the enormous scope of this book: Business and Financial Applications, Technical and Scientific Applications, Educational Applications, Games and Recreation Applications, Artificial Intelligence and The Future Computer, Utility Programs, Miscellaneous Applications, and A Compendium of Additional Applications.

This book is not intended as a how-to book as much as a what-to-do book. More than 1000 ideas and 75 programs are offered to the reader, and are covered in sufficient detail to provide a basis for a virtually unlimited number of spin-off ideas.

I was particularly impressed with the supporting information provided in many of the chapters. Simple formulas are used throughout, so you can easily develop your own programs by building on or altering the example programs provided. All the programs are Level II Basic, with some programs designed for disk use. This means they can be adapted to the majority of microcomputers that use Microsoft Basic. There are no machine language or Assembly programs included.

My only criticisms are with the programs and listings the book contains. Some programs are very long, yet no indication of memory requirements is given.

I would like to see each program with a REM line indicating memory needed, and whether the program can be used without disk. This, together with the fact that the author has made no attempt to explain the line-by-line operation of the programs, and has not listed the variables and their usage, make this book too advanced for beginners. The main thrust of this book is to stimulate ideas for those already familiar with Basic programming.

(Tab Books Inc., Blue Ridge Summit, PA 17214. Hardcover, 355 pp., \$13.95, Softcover, \$8.95.)

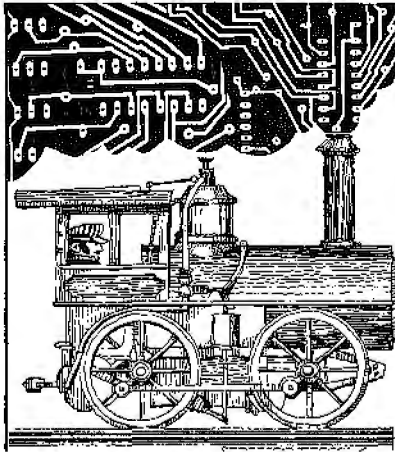
Fred Blechman  
December 1980

★ ★ ★ ★

**6809 Assembly Language Programming**  
**Lance Leventhal**  
**Osborne/McGraw-Hill**

*6809 Assembly Language Programming* is the fifth in Lance Leventhal's series on Assembly-language programming. Other books in the series concern programming the 280 and the 6502.

Leventhal assumes the reader has a basic knowledge of microcomputing principles. The book is divided into five basic sections. The first, Fundamental Concepts, gives an introduction to Assembly language. It describes the functions of an assembler, including special features such as macros and conditional assembly. It goes on to describe the architecture of the 6809, and then gives the syntax rules of the standard Motorola assembler.



The second section, Introductory Problems, starts you programming. It begins with a statement of the purpose of the program, gives a sample program, the source code (with comments) and the object code. They are followed by an explanation of the instructions and programming methods used in the program.

Advanced Topics, the third section, covers advanced programming methods and gives notes on their use. This section also shows some peripherals for the 6809. An analog-to-digital converter is shown for the 6809. Since this is a full 8-bit device, it could extend joystick range.

The fourth section, Software Development, deals with software design: structured programming, modular programming, and flowcharting. These chapters also have a thorough discussion of coding, debugging, testing, and documentation.

Finally, the 6809 instruction set gives details of how each instruction affects memory locations and registers, often shown

in several addressing modes. Included here are mnemonics, allowable modes, object code, and the number of bytes needed.

The book is an excellent value and contains the vital information necessary for programming the 6809.

(Osborne/McGraw-Hill, 2600 Tenth St., Berkeley, CA 94710; \$16.99.)

Phil Saunders  
March 1982

★★★★

**8080/Z80 Assembly Language—Techniques for Improved Programming**  
Alan R. Miller  
John Wiley & Sons Inc.

The examples in *8080/Z80 Assembly Language—Techniques for Improved Programming* are in Z80 op-codes because most assemblers use them. The examples are applicable to the 8080 chip if you limit the instructions to those compatible to both.

The appendix gives the 8080 instruction mnemonics and a cross-assembler listing. The book discusses the octal base of the 8080 instructions. This makes the instructions easier to remember if you must program without a chart.

The book briefly explains gates and uses them to represent Boolean logic. The book discusses magnetic and paper-tape I/O routines in detail, and it also discusses memory-mapped I/O, a topic usually neglected in such books.

Three sections of the book make it worthy to include in my reference library, even though I have been programming in

Assembly language for many years. A 50-plus-page chapter explains how to develop a monitor for your particular 8080 or Z80 system. Another chapter explains how to link your own routines to CP/M.

An appendix details the undocumented Z80 instructions, how to add them to existing assemblers, and how to use them. These three items alone would be worth the price of the book. Bravo, Alan Miller!  
(John Wiley & Sons Inc., 605 3rd Ave., New York, NY 10158; \$10.95.)

Dennis Thurlow  
February 1982

★ ★ ★

***A Bibliography of Computer Music:***

***A Reference for Composers*  
Sandra J. Tjepkema  
University of Iowa Press**

This bibliography includes only what might be thought of as the academic approach to computer music. Although much research is carried out at the university level, a significant proportion of computer musical activity is also occurring less formally. By limiting her listing primarily to official computer, computer music, and professional music publications, Tjepkema has missed a wealth of important work.

In coverage, major sources of computer music have not been included, among them the *Alternative Music Press*, composer notes from record jackets, and reprints from seminars and gallery performances.

The underrepresentation of the writings of major composers

is also unfortunate. And finally, the absence of a bibliography cuts off the bibliography from its source: the music itself.

These drawbacks, though important to working composers, do not diminish the heart of the collection itself. The volume includes over 20 years of writing (1956-1979) and about 1,000 entries and cross-references covering all phases of computer music. Included are interviews with composers, technical reports, and even user manuals for electronic music studios. Analog synthesis is specifically not covered.

(University of Iowa Press, Iowa City, IA 52242; Hardcover, 294 pp., \$17.50.)

Dennis Bathory Kitz  
May 1982

★ ★ ★ ★

***The Alternate Source Magazine***

**The Alternate Source**

Not long ago, the only sources one had for reliable information on the TRS-80 were either of the novice—"I love my computer"—variety, or were in the form of arcane treatises on the advantages of memory-mapping re-starts to ROM.

We have since witnessed the birth of *80 Micro* with its glossy, eclectic approach. Less heralded was the simultaneous appearance of *The Alternate Source (TAS)*.

TAS is not a pretty publication. Favoring the modest journal approach rather than the popular one, it is dutifully prepared on an IBM Selectric with a TRS-80-based text editor resulting in a plain, neatly typed document.

TAS is oriented toward disk owners; nearly all of the first issue and fully half of the fourth issue offered disk information exclusively.

Machine- or Assembly-language programs, particularly utilities, are another TAS emphasis. Technical questions from readers are answered by Jesse Bob Overholt from the Circle J Software Ranch on "180,000 microacs," and letters from readers are regularly published.

Each issue of TAS is available on tape or disk as an Electric Pencil file. Programs need not be entered by hand; TAS can either be read or run.

TAS also contains a bulletin board for new information, includes software reviews, covers information about the Model II, and has no advertising except for its own software library. A special update sheet called Between the Issues serves as a free-form newsletter/editorial page.

The approach of the magazine is somewhat "old school" in that it views the TRS-80 not as a departure from, but rather as a logical new member of, the historical data-processing family. It treats data processing as a professional rather than a hobbyist or industrial concept. You will not find hardware articles, other than reviews, nor will you find games or pastimes, photographs or diagrams, or programming related to electronics or process control.

*The Alternate Source* presents literate and detailed applications articles. Its subscription rate is easily paid back in the high quality of the programs it publishes.

*(The Alternate Source, 1806 Ada St., Lansing, MI 48910; \$24*

*for 18 months.)*

Dennis Bathory Kitzz  
January 1981

★ ★

**An Introduction  
To Computer Music  
Wayne Bateman  
John Wiley & Sons Inc.**

The appearance of *An Introduction to Computer Music* by Wayne Bateman heralds a renewed interest in the genre.

The production of music generation peripherals for the TRS-80 and other personal computers also demonstrates that many composers other than academics are now enthused about sound and music created with the aid of digital circuits.

Bateman's book deals exclusively with computer music, rather than concrete and synthesized music. Computer music is not merely electronic music, though, but rather a very versatile technique of composing and orchestrating sound and structure beyond that normally available to humans. The book is neither academically thorough nor popular, occupying a dangerous middle ground in which Bateman is not entirely comfortable. Bateman is a lucid writer, but the topic is too big.

The question of the computer's validity as a musical device is briefly discussed in the first chapter. Bateman believes in that validity, and presents the physical and mathematical fundamentals of its tones and their harmonics. Frequency spectra, additive and subtractive synthesis of complex tones, sampling intervals, and phase relationships are presented.

These topics are complex, but vital to computer composition, so Bateman includes a formidable but inevitable helping of mathematics.

The unsatisfying chapters on computer operation and languages follow. These chapters present flowcharts and theoretical programs in Fortran, Pascal, and English. Basic program samples are in the appendix. The author does not tell what machinery to use to test his theoretical programs, on the assumption that the hardware might become outmoded. This leaves the reader unclear on how to "plug in" to the computer.

Successive topics include modulation (not musical, but sonic), dynamics and waveform analysis. The chapters contain a great number of graphs representing sonic events. The waveform-analysis chapter is Bateman at his best. It is here that Bateman describes the computer's synthesis of complex tones—sounds that can't be created by sounding objects, but are the results of waveforms, manipulated and reformed, within the composer's mind.

The text discusses recorded and natural sounds, proposing a difference in approach between the more common analog processing and the difficult but potentially more accurate and reproducible method of sound generation with a computer.

Finally comes the art of the electronic medium: Scales and tonality are presented with lucidity and depth of understanding surprising and gratifying. Obviously, Bateman is at home with contemporary Western music and its long history, and his tone and

selected musical examples are both to the point and refreshing.

Bateman has included a probing discussion of the dilemmas of the computer in modern society, "Machines and Human Creativity." Bateman speaks of the personification of machines in this way: "Anyone who programs a computer quickly becomes accustomed to its cold, mechanistic responses to every instruction, and to its banal incapability of humanistic interaction." Bravo for Bateman.

*Introduction to Computer Music* remains an unsatisfying work. It is because personal computer users are given no hint on how to begin the task of composition. Because it seems mathematically detailed, the book can be overwhelming. Also it assumes some knowledge of music theory, and is not directed to the growth of the extemporizing composer/performer. But, the book does present a topic returned from public banishment, and deserves the attention of composers and other musicians, as well as computer hobbyists.

(John Wiley & Sons Inc., 605 3rd Ave., New York, NY 10158; Hardcover, 314 pp., \$24.95.)

Dennis Bathory Kitz  
December 1980

★ ★ ★ 1/2  
**Basic Adventures**  
**Bob Liddil**  
**SO-Northwest Publishing Inc.**

Bob Liddil's *Basic Adventures* is a collection of 18 adventure games by such well-known authors as Scott Adams, Greg Hasset, Lance Miklus, and Teri Li. The games are all available in

software form, and include titles such as *Dog Star Adventure*, *Atlantean Odyssey*, and *Lost Dutchman's Gold*.

These adventures are excellent representations of the genre, although not all are identical to the software versions. *Atlantean Odyssey* is written in Basic in the book, but the software version is in Assembly language. It appears virtually the same on the screen, but undoubtedly runs slower in Basic—a minor nuisance. Other adventures have been mildly refined; for instance, many spelling errors appearing in the originals have been corrected.

Adventure games require mental acumen rather than the digital dexterity needed for arcade-type games. The program sets a scenario and describes the situations you get yourself into. You must decide where to go, what to do, and how to do it. You could be a moonshiner running hootch: do you run the roadblock or turn around? How do you fix your souped-up '57 Chevy? How is it broken down? Can you avoid the "revenuers" long enough to make your delivery? If moonshining doesn't enthuse you, there are space adventures, too. If that bores you, you could be a CIA agent on a deadly mission, or a prospector searching for a lost cache of gold, or a . . . . The list goes on.

This book also gives advice on how to write and sell adventure games. An adventure-generator program provides the bare bones for creating your own adventure; you just plug in the plot and its various twists and turns. Liddil gives hints on how to sell your creations and lists potential markets.

Liddil says he wants to expose more people to adventure games. He appears to have found a good way of doing just that. For the price of one software adventure, *Basic Adventures* provides 18 (of course, you have to do a lot of typing). Not only is this a financially painless introduction to adventure, but the listings are valuable as tutorials on how adventure programs work. *Basic Adventures* is a good effort and should be of interest to anyone who plays adventure games or thinks he might want to.

*(80-Northwest Publishing Inc., 3838 South Warner St., Tacoma, WA 98409; Softcover, 252 pp., \$19.95.)*

Michael E. Nadeau  
August 1982

★ ★ ★ ★

***The Basic Conversions Handbook for Apple, TRS-80, and PET Users***

**David A. Brain, et al.  
Hayden Book Company Inc.**

Here's a book that will be appreciated by anyone who likes to pick and choose from programs written for a variety of micros. The idea is simple: provide a series of glossaries permitting the owner of one of the three computers mentioned in the title to translate programs written for either of the others. In particular, the book is concerned with the Apple II and the TRS-80 Model I and III. As a Color Computer owner, I still found it very useful because of the amount of information included.

The book comprises four chapters and three appendices.



The first chapter is a short introduction to the ideas of translation. It is the second chapter that will get the heaviest workout from TRS-80 owners. It deals with the details of translating Applesoft and PET Basic programs into Tandy dialect. A separate table for each conversion is provided, and commands are listed alphabetically.

The great joy of the chapter is that Apple commands are explained in detail; an informative table explains Apple PEEKs, POKEs, and CALLs so you can decide how (or whether) to approximate them if they come up in a translation. As the Color Computer can approximate Apple commands better than the other TRS-80s, the book effectively serves for direct translation from Applesoft II into Extended Color Basic; there need be no intermediate Level II translation.

The remaining chapters follow the same format, but in less detail: TRS-80 and PET into Apple II, and TRS-80 and Apple into PET. Again, the authors have done a thorough job, including just enough detail to let the would-be translator make intelligent decisions about reworking published programs.

The appendices cover subroutines that allow a (non-Color) TRS-80 to simulate specific Apple and PET commands. Also included are examples of complete program conversions, and useful charts concerning video screen coordinates and the PET graphics set.

This is a good, nonflashy example of a book that can add much to the enjoyment of your computer. I regret only that the limited market has made it ne-

cessary for the publisher to charge \$7.95 for what amounts to a paperback dictionary.

(Hayden Book Company Inc., 50 Essex St., Rochelle Park, NJ 07662; Softcover, 79 pp., \$7.95.)

Scott L. Norman  
September 1982

★ ★ ★ 1/2

**Basic Faster and Better &  
Other Mysteries**  
**Louis Rosenfelder**  
**IJG Inc.**

*Basic Faster and Better & Other Mysteries* is a book worth its price. The introduction answers the question "What is faster and better?" Rosenfelder explains how sacrificing in one area leads to benefits in another. The programmer must decide on the desired results and then make the proper decisions (and sacrifices, if necessary) to attain the best operating program possible.

Rosenfelder's point is that the programmer must evaluate the needs of the program and the prospective user. Next he must write a program that is as easy and as fast as possible while using a minimum of memory. The resulting program is sure to be Basic that is both faster and better.

Rosenfelder points out many of the errors made by programmers, and then shows you how, based on his own experience, you can beat these problems and actually write better Basic. He covers possible trade-offs that you can make, as well as the appropriate times for their use.

His table of contents is so extensive it looks like the index to most other books. Rosenfelder

covers every trick of the trade in detail, and supplies an invaluable library of subroutines that you can use over and over to reduce memory usage, speed execution and make a user-oriented program.

If you don't like typing a library of subroutines and demos to keep for future use, you can buy two disks with the information already on them and ready to merge into your own programs for \$19.95 each.

Although any programmer can gain valuable information from this book, it is intended for disk users. The book also assumes the reader has a NEWDOS or NEWDOS80 operating system. There may be some incompatibilities with operating systems other than Apparat systems such as in the I/O routines. If you are in doubt, call or write IJG, and ask them about possible problems with your operating system.

(IJG Inc., 1953 W. 11th St., Upland, CA 91786; Softcover, 300 pp., \$29.95.)

Richard C. McGarvey  
May 1982

★ ★ ★ 1/2

**Basic for Home Computers  
Basic Second Edition  
Data File Programming in  
Basic  
Introduction to 8080/8085  
Assembly Language  
Programming  
Using CP/M  
The Wiley Self-Teaching  
Guide Series  
John Wiley & Sons Inc.**

Each of these five books in  
The Wiley Self-Teaching Guide

Series covers the fundamentals of its topic, using some standard version for reference. The chapters are logically arranged for ease of learning; knowledge builds on knowledge.

Quizzes follow each chapter, and each paragraph has fill-the-blank self-check questions following it. The beginning of each chapter outlines what you should learn from it.

These books are good self-teachers, but their beginnings cover many concepts at once, and you can get confused unless you are a careful reader.

*Basic for Home Computers* is a continuation of Bob Albrecht's effort to prove that anyone can learn to compute using Basic. The book is so well illustrated that you don't need a computer to learn the language, and it teaches a simple version of Microsoft Basic that should run on any machine.

*Basic Second Edition* reviews the entire command set and focuses on three things: debugging programs; using Basic to solve real-world problems; and expanding on earlier concepts, such as doubly subscripted arrays. The real-world applications section encourages lazy business programming. The book does not teach how to use the commands in programming, but a chapter on files at the end answers questions about what Basic words like Open and Close mean.

*Data File Programming in Basic* uses Microsoft Basic as the reference, and it deals with data files—the workhorses of business programming. It covers disk and tape files, both random and sequential, and shows you how to write utilities to convert from

one set of files to another (the real problem in many business applications).

CP/M is an operating system for 8080, 8085, and Z80 systems that is considered a standard for business software. CP/M is revised frequently, so many versions of it exist. *Using CP/M* uses the Digital Research Version 2.0 as the reference and notes the differences from earlier versions. You should have a computer to learn how to use CP/M. On the whole, the book adequately explains the fundamental use of CP/M.

*Introduction to 8080/8085 Assembly Language Programming* is a standard book that teaches number systems, hex, and register layout. The self-tests make Assembly-language programming easy for the do-it-yourselfer.

You do not need a computer to learn simple Assembly-language programming, but be aware that you are learning the Intel abbreviations of the instruction set. The book should teach machine code in octal instead of hex. If you already know the 8080 or Z80 and want to learn about the 8085, skip this book. It only briefly notes one difference.

As an approach to learning, the series is good, but I wouldn't use it as my only learning source. I recommend the first Basic book for teaching a beginning class on a low budget. I would definitely use the CP/M book to learn CP/M—there is no other simple CP/M book on the market. I recommend the Assembly-language programming book only if you have CP/M-based assembler.  
(John Wiley & Sons Inc., 605 3rd Ave., New York, NY 10158.

Basic for Home Computers costs \$9.50; Basic Second Edition costs \$12.95; Data File Programming in Basic costs \$12.95; Introduction to 8080/8085 Assembly Language Programming costs \$10.95; and Using CP/M costs \$12.95.)

Dennis Thurlow  
February 1982

★ ★ ★ ★

**Basic Fortran**

**James S. Coan**

**Hayden Book Company Inc.**

James Coan has packed a wealth of information into his text that can be of value to programmers in any of the high-level languages. The text is designed for an elementary Fortran course, but lends itself equally well to the individual who wishes to learn Fortran on his own and as a supplemental text for a precalculus mathematics course. All you'll need is a knowledge of first-year algebra. Coan develops all additional algorithms required in the text.

There are two premises upon which *Basic Fortran* has been written. The first is that the reader is essentially a newcomer to computer programming; the second, that for a rewarding learning experience, the student should begin meaningful programming immediately.

Responding to premise one, the author clearly defines his method for familiarizing the student with Fortran while simultaneously inculcating good programming habits. Not without humor, he draws the analogy between learning to program and learning to drive a car:

Although each process can be accomplished from a book, both are facilitated by the availability of a machine on which to exercise the knowledge.

The author presents the student with his first program on page two. Logically enough, this three-liner is a program to generate output from the computer printer, thereby giving immediate reinforcement to the novice programmer. Each line of the program is analyzed, and Coan gives the student only enough information to foster understanding without inundating him with related but nonessential detail. By increments of one and two lines, this initial program is expanded throughout chapter one until the three-liner has been developed into an 18-line program to compute and write paycheck amounts.

Chapter one closes with problems and a summary. Each type of statement introduced in the chapter is reviewed, as is each structural or programmatic concept. The problems exercise the material presented in the chapter, and an appendix provides answers and sample solutions to every other problem.

While the chapter-one format is basically that of the entire book, the author departs from this format by introducing problems and exercises more frequently chapter by chapter. He is careful to divide the material into readily digested elements and encourages the reader to become fully familiar with each new concept before continuing. The self-instructed programmer can easily establish his own pace for absorbing the chapters of the book and the features of Fortran.

Of the 11 chapters in the book, the first five are devoted primarily to the language of Fortran, while the remainder emphasize applications. There are more than 80 programs developed throughout the text. Applications range from simple business and finance to more complex data processing, graphing, quadratics, trigonometrics, polynomials, probability, and random simulations.

One of the many attractive features of this book is that all sample programs are listed in an index, which facilitates review and permits the book to serve as a handy reference. Other appendices provide a table of Fortran-supplied functions and a Z80 random number function (since most Fortrans do not incorporate random-number generation).

As with *Basic Basic* and *Advanced Basic*, Mr. Coan has filled a small number of pages with a sizable quantity of information, and has presented it so as to capture the interest of the broadest spectrum of students of Fortran.

(Basic Fortran is sold by Hayden Book Company Inc., 50 Essex St., Rochelle Park, NJ 07662; Softcover, 248 pp., \$10.95.)

Dave Smith  
June 1981

★ ★ ★ ★ 1/2  
***The Basic Handbook: An Encyclopedia of the Basic Computer Language***  
**David A. Lien**  
**CompuSoft Publishing**

David Lien's *Basic Handbook*, a standard reference for both the

neophyte and experienced Basic programmer, has been an excellent solution to problems such as finding substitutes for Basic words you used in old versions of Basic, but can't use in a newer version. Those familiar with the book will be glad to hear it has been thoroughly updated.

New features include special sections on dialects of Basic. There are more cross-references and information throughout the text. The vocabulary has been expanded from about 250 words to over 500. Seventy-eight computers are covered.

Extensive coverage is given on each Basic word. It is easy to flip through the book and quickly find a particular word. Each word is explained clearly and concisely. The *Handbook* also notes which systems support a particular command.

Short test programs provided with each word quickly demonstrate if the word is in your computer's vocabulary. Included are hints on usage, alternate spellings, and suggestions about what to do if your computer cannot handle a word. The book's cross-references are invaluable when converting programs from one dialect to another. The book refers you to several words to perform the same function, making it easy to find a usable substitute.

Special sections are provided on four computers: the Acorn Atom, the Atari, the Tektronix, and the TRS-80 Color Computer. These sections note Basic words unique to each vocabulary, special features of each dialect, and have liberal cross-references to the main body of the text.

The Index and Scorecard section is a key element of the

handbook. All 500-plus Basic words described are listed, with space for marking the results of each word's test program.

*The Basic Handbook* is one of those books you will refer to again and again. It is an excellent book. Whatever your programming interests, *The Handbook* will be a useful addition to your library.

(CorpuSoft Publishing, P.O. Box 19669, Dep. F, San Diego, CA 92119; Softcover, \$19.95.)

Alyson Grupp  
February 1982

***Basic Scientific Subroutines, Vol. 1***  
**F.R. Ruckdeschel**  
**Byte Publications**

*Basic Scientific Subroutines, Vol. 1* is a new book covering algorithms and Basic routines for plotting, complex number arithmetic, vector and matrix manipulation, series approximations of functions, and more.

This book is particularly worthwhile because the author spends time describing the physical interpretation and meaning of the techniques, along with the algorithms. The text is useful for students well versed in science but not in programming, or for programmers who lack some math or scientific background. As a supplementary text, the book is excellent.

The routines are listed in two appendices, with driver programs both in North Star and Microsoft Basic. The book is not meant as a Basic primer, yet it does present methods for changing scientific algorithms into

Basic routines.

Areas covered in this book include: plotting routines, complex variables, routines for converting rectilinear complex numbers to polar complex numbers, vectors and matrices, basic series approximations, and routines for eigenvalues, matrix arithmetic, and matrix inversions. A useful topic nicely dealt with is the use of matrices for the rotation, transposition, and scaling of vertices of figures. Another is the routines to determine character polynomials and eigenvalues of matrices. These routines are primarily useful for scientists and engineers.

This book is useful for scientists, engineers, and students. With a few minor exceptions, the algorithms are excellent and the Basic routines well designed. Since the book gives the routines in a commentated form, as well as compacted form in North Star and Microsoft Basic, the book can serve not only as a supplement to a computer text, but also as a sourcebook of scientific subroutines.

*Basic Scientific Subroutines, Vol. 1* is well worth the investment.

(Byte Publications, Peterborough, NH 03458; Hardcover, 316 pp., \$19.95.)

Bruce Powel Douglass  
September 1981

★ ★ ★ 1/2

**CLOAD**  
**CLOAD Magazine**  
**Monthly**

CLOAD differs from all the other publications discussed; it is supplied on a C-30 cassette

rather than being printed on paper. CLOAD is strictly software-oriented and contains half a dozen programs each month. You have no long listing to enter from the keyboard; just insert the cassette and CLOAD. There is a variety of programs, both for games and more practical purposes.

Included with each month's tape are a few printed pages of editorial material that contain comments on CLOAD programs, TRS-80 software in general, and announcements. CLOAD does not contain any advertisements as yet. (CLOAD Magazine, P.O. Box 1267, Goleta, CA 93107, costs \$50 per year on cassette, \$90 on disk.)

Rod Hallen  
January 1980

★ ★ ★

**Computer Games for  
Businesses,  
Schools and Homes**  
**J. Victor Nahigian and  
William S. Hodges**  
**Winthrop Publishers Inc.**

Are you looking for a good computer games book that will give you a solid core of coding from which to develop your own personalized computer games? The programs contained in this 157-page book are for the personal computerist. However, the particular version of Basic, which runs on PDP-type machines, used in these programs is not within the grasp of the average personal computerist. If you don't have a DEC-type Basic or DEC machine, you must make some changes

before you power up.

A listing of the variable names and their uses within the programs follows a short description and history of each game. This is a nice touch, especially if you are the type who likes to tear apart a program. A typical sample run and a program listing in an easy-to-read 5-by-7 dot matrix round out the games' introductions.

As with any collection, some of the games are losers. What remains, however, are 21 winners that can be divided into five general categories: calculation games, casino games, dice games, grid/math games, and miscellaneous. The book is a good buy for anyone willing to accept the challenge of converting and debugging programs to expand his game library. (Winthrop Publishers Inc., Cambridge, MA 02114; Softcover, 157 pp., \$10.95.)

Len Gorney  
September 1980

★ ★ ★ 1/2

**Computer Programming in Basic:  
A Self-Instruction Course in 4 Volumes**  
Ian Williamson, Rodney Dale,  
and Tim Elloart  
Cambridge Learning Enterprises

This series of four magazine-sized books from England is a great place to begin Basic programming.

The course of 60 lessons, designed for the complete rookie, is arranged sequentially from introductory math functions through subroutines and For . . . Next loops. At least one

lesson in each booklet is devoted to a broader concept such as computer problem solving, high- and low-level languages, and compilers and interpreters.

The first book deals with getting the machine rolling with Read, Data, and Print statements. It also introduces the Let statement early in the sequence. The second book places important stress on documentation, covering initial concepts such as flowcharting and REM statements. Book three moves into loops, arrays, and random generation. The fourth volume, titled *Advanced Basic*, covers subroutines, files, and strings, among other topics.

The booklets are liberally spiced with examples and immediate answers. The clarity of the explanations was impressive.

Since the books can be used without a computer at hand, they are ideal for classroom work where machines may not always be available. They are not geared to a specific system, giving an introduction to Basic without regard to the quirks of various hardware.

Each book contains the same summary of Basic with statements and commands listed along with math functions and operators. These are referenced with the lessons that deal with those topics. The table of contents and lesson abstracts are complete and useful.

A subtle yet helpful aspect is the format—not only are lessons broken into small concepts with complete explanations and extra examples on difficult topics, but the series' division into four booklets gives a sense of accomplishment when each is finished, as well as letting the

books easily stay open and flat on the table beside a computer.

Unlike most books dealing with beginner's Basic, these feature lessons on debugging and error statements. The language and spelling is British—playing noughts and crosses for tic-tac-toe is a kind of behaviour—but very straightforward; the authors have accurately anticipated questions and answered them honestly.

We found only two minor problems. Some lessons in the first booklets deal with math concepts that may cause trouble for those short on math background, and metric units may bother those unfamiliar with metric conversions. The authors carefully avoid using the British monetary system.

These books are a pleasant surprise. They are not typeset or overly illustrated, but they are durable, practically bound, and impressive in their completeness, clarity, organization, and emphasis. For a beginner, they provide instruction in Basic and a useful future reference. (Cambridge Learning Enter-

pises, could not be reached for an update.)

Kenneth Delp and  
Bruce Eldredge  
April 1982

★ ★ ★ 1/2  
**Computer World**  
**Kraftwerk**  
**Warner Brothers Records**

It is not surprising that Kraftwerk, a German band known for anticipating pop music trends, has been seduced by the microcomputer. This progressive combo has not been able to resist the simple dualism of all those ones and zeros or the antiseptic allure of the computer room. Kraftwerk has discovered the melody of the micro-processor.

Their latest album *Computer World* has a microcomputer on its cover. Some of the song titles are: "Computer World," "Pocket Calculator," "Numbers," "Computer Love," "Home Computer," and "It's More Fun to Compute."

At first the music sounds mechanistic and the lyrics simple-minded. Repetitive rhythms played through heavily synthesized keyboards accompany synthesized voices chanting one or two-sentence refrains—for instance, "I am the operator of my pocket calculator. I am adding, now subtracting. I am the operator of my pocket calculator. . . ." Or, "I program my home computer—beam myself into the future."

Lyrics aside, the songs will hypnotize you with their haunting, melodic refrains. The depth and range of the synthesized keyboards and the intricate syn-





copation of the rhythms in Computer World are fascinating. I found myself listening again and again to the title cut whenever I felt the need for some easy listening, 2001 style.

Computer World's rich harmonies and interactive rhythms reflect Kraftwerk's view of the microprocessor age. Listening to the music, it is easy to visualize the inner regions of the processor where signals control and consort amongst layers of circuit etch and gate logic. Listening to Computer World is like taking a swing on the crystal lattice jungle gym of an LSI chip. It is exhilarating!

The album's theme, however, is less lighthearted. It seems to be a confused lament against technology (the same technology that made this album possible), especially as used by government and big business. Another disturbing aspect is the album's back cover. The group's four members are pictured around the console of an antiquated electrical panel. Holding mechanical poses, the closely shorn members of the band are clad in the brown shirts and ties of the Hitler Youth. It may not be fair to assume that Kraftwerk has equated technology and the age of the computer with totalitarianism, but they do seem confused about it!

Computer World is not likely to be one of the platinum albums of the 80s, but the price of innovation is often anonymity. The album's portrayal of the computer, especially the microcomputer, in synthesized melody and syncopated rhythm is a clever approximation of what goes on inside a microprocessor. If you have ever fanta-

sized about the domain of the microworld within your computer, Computer World is for you. The music of Kraftwerk is the backbeat of the computing age.

*(Computer World is sold by Warner Bros. Records, \$7.99 record, \$8.95 tape.)*

Chris Brown  
December 1981

★ ★ ★

***The CP/M Handbook with MP/M***  
**Rodnay Zaks**  
**Sybx Inc.**

Sybx has published books on how to build systems from chips, how to program in high-level languages, and how a beginner can learn machine language. Their manual on the Z80 chip is the Bible for Z80 users. Knowing this, I thought *The CP/M Handbook* might be an in-depth machine-language study of CP/M's internal workings. Instead, what we get is a simple instruction book for any level user.

This book could be a manual for the businessman. It explains how to determine if a system is adequate for an application, and how to recover from errors a typist might make. The difference in file protection between a multi-user MP/M and single-user CP/M is discussed, as well as how much space it takes to save a certain amount of data on disk. In the section explaining the uses of each command, the author anticipates any error messages a command might generate, tells how to recover, and then helps the reader understand why the error occurred

and why the recovery works.

Examples are taken from commercially available software such as Micropro International's WordStar and Structured Systems Group's NAD mailing list. This means the chances are good that the examples will be relevant to the system's actual use, which is a big plus!

The book is also ideal for the serious business programmer, since CP/M is in such wide usage that it is considered the standard for business software.

The standard features of CP/M up to revision 1.4 are covered, and followed by changes in other revisions such as CP/M 2.2, MP/M 2.1, and Cromenco's CDOS. Use of Basic and CBasic is discussed, even to how the running of these may differ from one machine to another.

You won't be riding the crest of a new wave by reading this, and the author won't make you chuckle, but if your future in computing goes beyond playing games, you'd better have a copy of this book on your shelf. (*Sybex Inc., 2344 6th St., Berkeley, CA 94710; Softcover, 321 pp., \$14.95.*)

Dennis Thurlow  
March 1981

★ ★ ★ ★

### **CP/M Primer**

**Stephen M. Murtha and  
Mitchell Waite  
Howard W. Sams & Co. Inc.**

At last there is a definitive book on CP/M (Control Program for Microcomputers). The book introduces the CP/M operating system (including version 2.0) in a clear, concise fashion. The

book is illustrated using a video monitor outline to show how the user's screen will look.

If the reader is already familiar with microcomputers, skip the first two chapters. But be sure not to skip any of the excellent drawings by Bob Gumpertz; they add much to the enjoyment of the book.

(*Howard W. Sams & Co. Inc., 4300 West 62nd St., P.O. Box 7092, Indianapolis, IN 46206; \$14.95.*)

J. C. Hassall  
March 1982

### **The Creative Kid's Guide to Home Computers** **Fred D'Ignazio** **Doubleday & Co. Inc.**

*The Creative Kid's Guide to Home Computers* is a good starting point if you're interested in computing. D'Ignazio attempts to go beyond the fun-and-games uses of a computer and inspires you to become more than just a user. This is not a how-to book. It is, as D'Ignazio says, a "What if?" book—a creative guide.

The book is divided into four sections. The introduction develops a cursory understanding of how computers work. It is, as D'Ignazio calls it, a "Whirlwind Tour." This is where the book proves itself. We are not inundated with technical information. We learn how and why the machine does what it does.

The next section is entitled "Games." Here we learn about the many computer games available. While we read about all of these fun and exciting games, we stumble upon a

chapter entitled "You're Trapped."

And we are. D'Ignazio led us right into it. The program Trapped! is a simple Basic program listed opposite a flowchart. The program demonstrates commands such as Random, Let, Input, Run, GOSUB, and so on. The listing is set up with guidelines that describe what each section of the program does. D'Ignazio uses a game to help us learn Basic.

Projects is the next section, and it discusses other possible uses for the microcomputer, from robotics to aiding the handicapped. None of the projects is particularly simple. However, even some of the most complex are projects that have been done by teenagers and high-school students.

The final section, the appendices, lists a limited number of books, magazines and catalogs, a checklist for the first-time computer buyer, an ASCII coding chart, a decimal to binary chart, and a glossary.

I'm not entirely certain of what grade level D'Ignazio is trying to reach with this book. It's too complex for grammar-school students, and too simple for adults. I think that it could be used on the junior-high and high-school levels, as well as for beginning adults attempting to understand the microcomputer.

*The Creative Kid's Guide to Home Computers* is not the be-all and end-all microcomputer resource guide. It is, as I said, a starting point—and a good one. (Doubleday & Co. Inc. Garden City, NY; Hardcover, 130 pp., \$9.95.)

Chris Crocker  
April 1981

★ ★ ★ ★

***The Custom TRS-80 & Other Mysteries*  
Dennis Bathory Kitsz  
IJG Inc.**

If you have a TRS-80 Model I and you want to know what makes it work (or not work), and what you can do to make it do things it normally can't do, then this book will fascinate you.

A regular columnist for several microcomputing magazines, Dennis Kitsz has combined his software and hardware expertise with his ability to write in readable (and sometimes humorous) form.

This book is almost a TRS-80 Model I encyclopedia, and it contains many photos, schematics, tables, and Assembly-language listings. Subjects include converting binary to decimal, reading the pins, color codes, and power supplies, creating Basic tokens, and many more.

The overall coverage is very broad. There will be few who pick up this book and don't find within its pages something new and interesting. However, much of the book may be well beyond their present abilities or interests; it is not for the beginner.

I reviewed the first printing of the first edition, February 1982, and some problems were evident, mainly in the captioning of the figures and text references, such as "see Fig. (?)"—obviously meant to be identified before printing. Figures 2-11 and 2-39 appear to be missing, but they aren't; there's just a big gap in the numbers.

This book is destined to become one of the reference classics among TRS-80 Model I owners.

(IJG Inc., 1953 W. 11th St.,  
Upland, CA 91786; Softcover,  
\$29.95.)

Fred Blechman  
October 1982

★ ★ ★ ★  
**Don't**  
**Rodnay Zaks**  
**Sybex Inc.**

Is your software unreliable? Does your computer behave erratically? Perhaps the problem is not with the program or the hardware. Maybe the difficulty lies with you, the user. Starting with the premise that computers are reliable, *Don't* explores how the human element can contribute to their successful operation.

Author Rodnay Zaks makes a good point with his unusual title choice—most of the secret to reliable computer operation can be expressed in the form of don'ts. The book also delves into some of the positive aspects of computer housekeeping, but in the end, *Don't* is a sobering tale, and one that may save you a lot of time and money if you heed its lessons.

Much of *Don't* is devoted to computers in business settings. Discussions about computer security and the design of a computer room are not going to appeal to a hacker. Hobbyists, however, will find familiar advice in the chapters on floppy disks and electrical problems. As a further gesture, each chapter includes a box that lists specific tips for the home user.

Instead of supplying you with checklists, Zaks wants his readers to know why they shouldn't smoke near their computer or leave a disk out of its

jacket. A knowledge of computer jargon is helpful but not necessary; with the exception of a few problems that are left unexplained, *Don't* is intelligent, yet easy reading. Cartoons and photos help to break up the pessimistic text.

Salesmen and manufacturer's literature will provide you with plenty of assurances, but in the end it's your computer and your problem. If you are going to spend \$10,000 to \$20,000 for a business system, *Don't* is required reading. Hobbyists who are plagued with mysterious failures just might find some answers in it, too. In any case the \$11.95 cover price is an inexpensive way to peace of mind. (Sybex Inc., 2344 6th St., Berkeley, CA 94710; Softcover, 200 pp., \$11.95.)

Tim Daniel  
September 1982

★ ★ 1/2  
**Elementary Basic: Learning to Program Your Computer in Basic with Sherlock Holmes.**  
**Elementary Pascal: Learning to Program Your Computer in Pascal with Sherlock Holmes.**  
**Chronicled by John H. Watson**  
**Edited with Commentaries by Henry Ledgard and Andrew Singer**  
**Random House Inc.**  
**Vintage Books Division**

The premise these books are based upon is that Sherlock Holmes was an ace programmer. Even if apocryphal, Ledyard's and Singer's idea is intriguing. It is also entertaining because the

authors take pains to capture the flavor and proper Victorian prose of the Holmes stories.

John Watson opens each chapter describing an encounter with Holmes and his "Analytical Machine." Holmes uses the magical machine to solve a murder, unlock the mystery of a socialite's disappearance, identify cigars by their ashes, find a rare Bible, break up a burglary ring, create a cipher for the Foreign Office, and much more. Each problem is analyzed by the master of Baker Street, displayed first as an "algorithm"—a shorthand form of the program—then as an actual program.

At the end of each chronicle, there is a didactic commentary by the authors detailing the concepts used by Holmes to solve the problem. As you might expect, the commentaries lack the stylistic charm of the chronicles, but they are invaluable in giving a detailed explanation of Holmes' methods. They also serve as counterweights making the books easier to read.

If you don't have prior knowledge of either one of these languages, you could find these books—despite their clever approach to the subject—heavy sledding. Before you pick them up, you should have been puttering in the languages for awhile or taken a short course in the fundamentals, preferably the latter. However, reading the books together can give you some insights into how Pascal stacks up against Basic.

*(Random House Inc., Vintage Books Division, 201 E. 50 St., New York, NY 10022; Softcover, 265 pp., each \$12.95.)*

John P. Mello Jr.  
December 1982

★ ★ ★

### ***Fast Basic***

**George and Thomas Gratzner  
John Wiley & Sons Inc.**

*Fast Basic* is a book on using machine-language subroutines to enhance TRS-80 Basic on the Models I and III. If you understand Basic, integers, floating point, and strings, this book should be easy to follow.

Each chapter of *Fast Basic* has a summary and self-help questions. The first chapter discusses the representation of data and programs in memory, including binary and hexadecimal numbers. Chapter 2 takes a closer look at what's in memory and the various formats of Basic programs—how to PEEK and POKE to high addresses and the like. Chapter 3 discusses the organization of the TRS-80, as well as where and how Basic keeps the data tables, work areas, and address tables.

The book goes on to discuss how Basic operates, such as executing GOTOs, as well as recovering lost programs. The variable tables are discussed in detail, so you can get a variable out when you need it for Fast Basic. Chapter 5 talks about hardware, including devices, CPUs, and buses. Chapter 6 begins the introduction to Fast Basic with the Z80 and its registers and how to perform a load of the registers or memory.

USR, ROM, and DOS calls are discussed along with software accumulators for integer, single- and double-precision numbers, and their addresses. An in-depth look at ROM subroutines for the math routines is given along with an explanation of various functions in machine language,

mostly using the ROM calls.

The approach taken in writing Basic programs into Fast Basic is to first write them in a simplified form of Basic called Simple Basic. Then, translate this form into USR calls and machine code. The book gives three methods of storing the machine code in memory, including POKEs into reserved memory, packing into arrays or strings, and packing into remark lines.

The last chapter tells how to enhance Fast Basic to be even faster. It gives the Block Search and Move commands of the Z80 instruction set along with applications, and it demonstrates simple ways to set and reset the video graphics in machine language, as well as using the keyword jump table and how to take over the syntax-error routine to add commands directly to Basic.

This book is well-written and provides many useful programming techniques. It contains much useful information on the organization of Basic, variable and work tables in the Basic environment, and how to properly enter and exit ROM subroutines. *Fast Basic* is well worth \$14.95. (John Wiley & Sons Inc., 605 3rd Ave., New York, NY 10158; Softcover, 278 pp., \$14.95.)

Bruce Powel Douglass  
November 1982

★ ★ ★ ★

***Free-Lance Software Publishing***  
**B.J. Korites, Ph.D.**  
**Kern Publications**

If you are serious about programming, sooner or later you'll

want to make it pay, and *Free-Lance Software Publishing* might be the most comprehensive book available on the subject.

The book explores the publishing process and most aspects of software salesmanship. Topics include direct sales of software, service bureaus, selling through intermediaries, user groups, case studies (winners and losers), selling techniques, writing the user manual, contracts, pricing, theft protection, and tax considerations.

The book is intended for savvy software authors who are after the big money. Korites thinks the way to do this is to sell to small- and medium-sized corporations or affluent user groups that will buy in small volume but pay a high price. He thinks little money can be made in mass-marketed software unless it is sold in large volume, but he does devote a chapter to this facet of the business.

This might be true, but many authors find their game or business software does sell in quantity once published by a large marketing organization. The book gives little attention to this phenomenon, which is unfortunate since large marketing organizations are often the first outlets fledgling software authors try in the rough and tumble world of freelancing. At this stage of their careers, authors most need guidance and usually can't get it.

The skills necessary to write a saleable program are entirely different from the skills necessary to sell that program. Once the distinction is made, the value of a good publishing reference becomes obvious.

(Kern Publications, P.O. Box H211, Littleton, MA 01460; Softcover, 133 pp., \$36.)

Chris Brown  
February 1980

★ ★ ★ ★

**Getting Started with TRS-80 Basic**

★ ★ ★ ★

**Going Ahead with Extended Color Basic  
Tandy/Radio Shack  
Model I and III**

Hot off the presses from Radio Shack are two books designed for the new computer enthusiast. *Getting Started with TRS-80 Basic* and *Going Ahead with Extended Color Basic* are not only complete and easy to understand introductions to your microcomputer, but are also the kind of books that make learning fun.

Most importantly, they are tools to help you use your computer effectively. As it says at the beginning of *Getting Started*, both are do-it-now books.

*Getting Started* comes complete with do-it-yourself exercises, sample programs, tables, and worksheets. The book utilizes cartoons that make learning fun. This book provides a good foundation along with giving confidence to all beginners to get started on bigger and better projects. *Getting Started* begins with the basics from setting up your TRS-80 and turning it on, to writing your own simple programs and debugging them.

*Going Ahead* is for those of you who have some knowledge of TRS-80 Basic, and want to try

out new and colorful things. It is split into three sections, making it easy to skip those parts you are already familiar with. If you're in an artsy mood, start with the first section. If you're ready for more complicated problems and want to play around with a few figures, the middle section is for you. The back section is for everyone; it has answers to the exercises, which appear throughout the book, along with worksheets and useful tables.

Both of these books are very enjoyable, and make learning easy. If you're not a computer enthusiast to begin with, there is the possibility that one or both of them will turn you into one. I guarantee they will prove to be valuable companions.

(*Getting Started with TRS-80 Basic* and *Going Ahead with Extended Color Basic* are sold by Tandy/Radio Shack, Fort Worth, TX 76102; Softcover, \$15.95 each.)

Lynda Stratton  
June 1981

★ ★ ★ 1/2

**Graphic Software for  
Microcomputers  
B.J. Korites, Ph.D.  
Kern Publications**

*Graphic Software for Microcomputers* is a good buy. It not only discusses the theory of operation of the simple points (drawing a line, circle, density filling, and so on), but also covers complicated graphic maneuvers (scaling, rotating, motion translating, and the like). The book even provides Basic programs that do each of these activities.

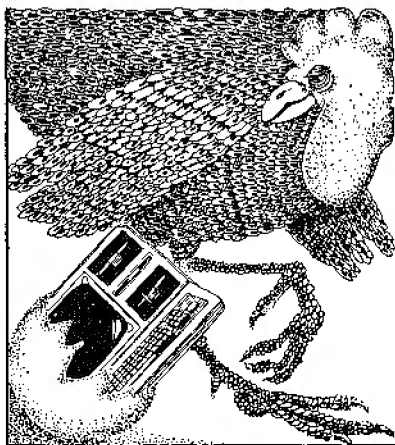
The book was originally written for the Apple computer, but is easy to use with the TRS-80 Color Computer. All you have to do is change all the references to HPCOLOR to PSET or LINE, and HCOLOR to COLOR.

Dr. Korites knows how to teach students and how to prepare a reference manual. Each concept is given its own page, which is split vertically between the explanation and the Basic program. You should know basic math and have at least a nodding familiarity with the concept of X-Y Cartesian coordinate systems.

Dr. Korites assumes you have no knowledge of the next level of math required, which includes vector mathematics and matrix manipulation. He includes sufficient instructions in the text to enable you to understand these ideas well enough to use the graphics techniques explained in the book.

(Kern Publications, P.O. Box H211, Littleton, MA 01460; \$21.95.)

Terry Kepner  
May 1982



★ ★ ★ 1/2  
***How to Make Money with Your Microcomputer***  
**Carl Townsend and Merl Miller**  
**Dilithium Press**

*How to Make Money with Your Microcomputer* is an excellent starting point for cashing in on the money-making potential of your computer. The book is a guide to developing ideas into profitable business.

The breadth of opportunities appeals to a wide variety of computer enthusiasts with varying levels of expertise. The book gives an inside look at the details of operating many different kinds of microcomputer businesses, and the reader can select a feasible business that matches his skills and interests.

Most of the book deals with the business of selling microcomputer products or services, but individual chapters cover a specific type of business such as how to operate a service bureau or how to develop and sell software or hardware.

For each business, the authors state the minimum experience required and possible monetary rewards. Most of the opportunities discussed require experience or specialized knowledge, but some require little computer knowledge.

*How to Make Money with Your Microcomputer* is not exhaustive, but it does serve as a good starting point for both the novice and the experienced microcomputerist. It is particularly valuable for the experienced microcomputerist interested in starting his first business and for the professional looking to publish. The thorough



coverage of article writing is justification enough for the price of the book.

(Dilithium Press, 11000 S.W. 11th St., Beaverton, OR 97005; Softcover, 154 pp., \$12.95.)

Silvia Burnes  
October 1982

★★★★

***Inside Basic Games***  
**Richard Mateosian**  
**Sybex Inc.**

*Inside Basic Games* teaches interactive Basic programming through the use of games in a clear and concise style. Each chapter begins with the objective to be tackled, along with the steps of progression to be used in attaining this goal, and ends with a summary.

One of the most interesting and useful concepts is the use of Free Basic in the programs. Its principal goal is to free the programmer from having to think about line numbers, although it does support structural program techniques. Actually, Free Basic is a form of pseudocode, but is much more precise in that it uses Basic instruction rather than verbal instruction. Once you have constructed your program using Free Basic, Mateosian explains the techniques used in translating it into Basic.

Each game is described per se and then in terms of their Basic and Free Basic instructions. The effects and advantages of certain routines in regard to their "cannibalization" and "generalization" abilities are discussed. Your main objective is to learn how to write routines that can easily be integrated into other programs.

General programming techniques such as the top-down development theory are explained, along with more specific techniques like space saving and file maintenance functions for a certain program.

The games are written in Microsoft Basic and can be run on the TRS-80, Apple II, and PET/CBM. He identifies the small changes in each program that must be made when converting from one system to another.

I recommend this book to anyone interested in writing their own interactive programs. (*Inside Basic Games is marketed by Sybex Inc., 2344 6th St., Berkeley, CA 94710; Softcover, 325 pp., \$14.95.*)

Bonnie Welsh  
September 1981

★★★★

***Inside Level II***  
**John Blattner and**  
**Bryan Mumford**  
**Mumford Micro Systems**

In a field where we accept crude screen imagery and squawks and bleeps rather than fine-grained photography and music, it's refreshing to find *Inside Level II*—18 chapters of detailed descriptions of and applications for ROM subroutines and entry points, in a volume that manages to be immensely thoughtful, literate, and cleanly designed. Mumford Micro Systems may be a small company, but it isn't a member of the "type it, copy it, bind it in the attic" crowd.

Blattner and Mumford explain in their introduction the appeal

of joining Basic's writing ease and input/output power to Assembly language's speed: "It is not atypical for 25 percent of a given (Basic) program. . . to require 95 percent of the operating time. If this time-critical part can be rewritten in Assembly language and efficiently linked to the remainder of the program, one can enjoy the best of both worlds."

Part One describes the way Basic is organized throughout ROM and RAM. ROM utilities (registers, buffers, and variables) are covered, as well as the manipulation of numerical data.

Significant I/O routines are carefully explained, and a pot-pourri of miscellaneous Level II subroutines is also covered, as are the easiest ways to send text to the screen, get a string of characters from the keyboard, and produce text on a printer. Routines linked to the disk system are boldly marked.

Part Two of *Inside Level II* applies the material from Part One, encompassing assemblers and monitors, relocating Basic programs, VARPTR use, Basic-assembler program linking, linking multiple program segments, and tape load/save. The book concludes with sample composite programs and tape utilities to CSAVE/CLOAD programs at speeds higher than 500 baud. The program samples include preliminary remarks, fully annotated source code, line-by-line comments, and descriptive notes.

Notwithstanding the detail and careful illustration, *Inside* is by no means an easy book. Blattner and Mumford say it bluntly: "To take full advantage of the information in this book requires a knowledge of Z80 Assembly-lan-

guage programming."

(Mumford Micro Systems, P.O. Box 435, Summerland, CA 93067; Softcover, 65 pp., \$15.95.)

Dennis Bathory Kitsz  
October 1980

★ ★ ★

### **Introduction to T-BUG** **Don and Kurt Inman** **Dilithium Press**

The TRS-80 has a great deal of appeal for users with no machine-language coding experience, but Radio Shack's documentation of their Editor/Assembler program and T-BUG Z80 monitor and debugging aid is very inadequate. *Introduction to T-BUG* explains each command with short programs, tables, and sketches, and adds greater detail on the use of certain ROM subroutines.

A section on machine-language graphics is later expanded into a game program. Another chapter includes one of the most lucid explanations of flag bits that this reviewer can recall.

In the final chapter, errors are deliberately left in the game program, so that the reader can follow as the authors demonstrate and explain debugging.

While not a complete text on Z80 machine-language programming, this book provides the novice with a "go ahead" text full of useful tricks, enabling him to begin a more detailed study of the subject, as well as have a little fun.

(Dilithium Press, 11000 S.W. 11th St., Beaverton, OR 97005; Softcover, 120 pp., \$9.95.)

James H. Sheats  
July 1980

★ ★ ★ 1/2

***Introduction to TRS-80  
Graphics***  
**Don Inman**  
**Dilithium Press**

The TRS-80's graphics aren't as impressive as some other microcomputers', but a well-understood TRS-80 can draw rings around a more sophisticated unit that is a mystery to its operator. *Introduction to TRS-80 Graphics* makes this in-depth understanding possible.

The author, Don Inman, takes the reader through basic shapes, such as rectangles, to the more sophisticated shapes involving curved lines. The reader not only understands how he produced his designs but also knows how to modify them.

Also included are sections on bar graphs, animated word graphics, and even video games, all of which can be especially useful for teaching programs.

The book is well-written, and it's laced with examples and exercises. Graphics are a dream for the TRS-80 owner.  
 (Dilithium Press, 11000 S.W.  
 11th St., Beaverton, OR 97005;  
 Softcover, \$11.95.)

Stephen F. Nowak  
 April 1980

★ ★ ★ 1/2

***Introduction to TRS-80 Level  
II Basic and Computer  
Programming***  
**Michael P. Zabinski**  
**Prentice-Hall Inc.**  
**Model I**

The teaching of computer programming becomes tricky because there is a fine line between catching a student's at-

tention and overwhelming him with jargon and detail. Some authors of computer instruction books have tried to overcome this problem by taking a light-hearted, humorous approach to their subject matter. Others have presented the material factually in a straightforward manner, letting the student learn if he/she can.

Michael Zabinski takes a scholarly, textbook approach in his new book. This book would be suitable for use in a junior high or high school course in computer programming and would also be appropriate for college students or adult education students.

What makes Zabinski's book so acceptable is its liberal use of examples and a set of extremely challenging problems to solve using a computer. Even experienced programmers enjoy the pure challenge of solving good problems and this book has dozens. It has challenges for the quick learner but also is accessible by students at all levels.

This book covers all the Level II Basic commands and functions. Starting with the elementary stuff like turning on the system and getting it to print "Hello," the text moves naturally and easily into a well-written introduction on how the TRS-80 uses numbers. Zabinski introduces the subject of using numbers early, but gently for those who question their math skills. Once introduced, the text moves on to define numeric and string variables and the TRS-80 arithmetic functions.

At this point, the student has learned very little actual programming. The book takes advantage of the student's still

uncluttered memory and discusses program logic and the use of line numbers. From there the discussion proceeds through input statements, program line editing, debugging, flowcharts, If . . . Then, For . . . Next, Read . . . Data, subscripted variables and all the rest of the TRS-80 commands and functions.

Of course, the only way to learn anything is to do it, so Zabinski makes liberal use of examples. The examples are set off from the text and are annotated, line by line. They are well-chosen and illustrative without being too simple.

This is a well-organized book, put together by a knowledgeable programmer and educator. The writing style is a little stiff, typical of a textbook, and some graphics would have brightened up its appearance. The publisher chose to use the standard gimmick of a dot-matrix printer to simulate TRS-80 screen displays on the printed page (which I personally find annoying). However, the text is complete and informative.

*(Prentice-Hall Inc., Englewood Cliffs, NJ 07632; Softcover, 162 pp., \$9.95.)*

G. Michael Vose  
June 1981

★ ★ ★ 1/2

**Learning Level II**

David Lien

CompuSoft Publishing

David Lien's *Learning Level II* includes corrections and a double set of cut-and-paste pages to convert the Level I manual to Level II. The material is about the same as that in Radio Shack's Level II reference

manual, but it's arranged for teaching, not reference. Particularly well covered are the editor, Print Using, strings, and error messages.

The book doesn't cover everything. The Delete command is omitted, as are Random, STRING\$, CDBL, CINT, and CSNG. The rest are covered to my satisfaction, except for USR. Also, I'd like more on communicating between machine language and Basic for those who have some familiarity with both.

My only real complaint is the number of typographical errors. However, a few days after I got the book I received an errata sheet from CompuSoft with corrections. Another problem is Lien's inaccuracy in stating that the lack of abbreviations in Level II will lengthen programs and may make individual lines too long. Actually, there is ample space for conversion.

With all these shortcomings, I recommend the book without hesitation. Whether you are upgrading from Level I or starting on Level II with no previous computer experience, this book is valuable. Lien is an excellent teacher and writer.

*(Learning Level II has been replaced by The Basic Handbook, published by CompuSoft Publishing, P.O. Box 19669, Dept. F, San Diego, CA 92119; Softcover, 352 pp., \$19.95.)*

Everett Ogdan  
May 1980

★ ★ ★ ★

**Learning TRS-80 Basic for Models I, II/16 and III**

David A. Lien

CompuSoft Publishing

This book will give the begin-

ner an easy start into the Basic language. Dr. David A. Lien is one of the best documentation writers in the business, and *Learning TRS-80 Basic* may be the best beginner's book for the TRS-80.

The book is written in an informal, humorous style and is liberally sprinkled with cartoons. It manages to be readable without being cute, yet it is thorough and precise without overdoing any one subject. The complex problems that might throw beginners are either treated very lightly or not gone over.

The first section is the meat of the book—420 pages of instruction on the Basic keywords and programming techniques. The second section contains answers to the exercises given in the first section. The third is 16 ready-to-use programs, each less than two pages long. The fourth is a series of appendices with information specific to the Model I, III, or II/16. And the fifth section is an index that cross-references the Basic commands and terms used in the book.

The last 30 pages of the tutorial are devoted to program control. Lien covers flowcharting and debugging programs. Included is a short memory test for times when the hardware seems to be at fault. These chapters alone are worth the price of the book. The answers to the 79 questions in the tutorial tend to be a little short. An English explanation of what is happening in the program could have been used more often than it was.

To some the appendices may be of more use than the tutorial. Appendices A-H include separate ASCII code tables for the Model

I/III and II/16, and explained listings of the error messages for the I/III and II/16. The important parts of the appendices are tutorials on using disks and on setting up the hardware. The appendix on setting up the Model I Expansion Interface is interesting for its thoroughness, as is the section on using the real-time clock without Disk Basic, information that is hard to find. Finally, a section on using two cassette recorders with the Model I is invaluable.

Also included is a section reviewing the Basic keywords that are peculiar to the Model II/16, essential to any book that purports to explain Basic for all TRS-80s, Color Computer excluded.

The book's biggest flaw is that in trying to ease the beginner into computing it sometimes neglects the advanced programmer. Keywords like PEEK, POKE, USR, and VARPTR are covered too briefly.

Lien states that the book was written for the Model III, but that it can be used equally well on the Model I, II, and 16. I think it was written for the Model I and III with the II/16 thrown in as an afterthought. To his credit, Lien thoroughly documents the multitude of small differences between the machines in the margins of the book. I came away impressed with the book's completeness, but felt that the overall effect was a little busy.

The only Disk Basic commands documented are CMD "S" (return to the disk operating system), Save and Load. However, anyone with disk drives who is just learning Basic would still benefit greatly from this book.

*Learning TRS-80 Basic* is one of the best introductions to Basic you can buy. In readability and thoroughness it far surpasses the Model I, III, or II/16 Basic manuals.

(CompuSoft Publishing, P.O. Box 19669, Dept. F, San Diego, CA 92119; Softcover, 544 pp., \$19.95.)

Art Huston  
November 1982

★ ★ ★  
***Learning with Computers***  
**Alfred Bork**  
**Digital Press**

Alfred Bork is a pioneer in computer-assisted learning (CAL). He has made significant contributions to the field through materials and strategies developed for physics classes at University of California-Irvine's Educational Technology Center, his work for the National Science Foundation, and by offering courses on computer learning.

Bork has assembled his papers, some coauthored, over the past 10 years into seven chapters on learning with computers. The key word here is "with." He views the computer as a tool that educators can use to enhance learning.

The reader learns how to use computers to manage classroom records, provide testing applications, instruct, remediate, and motivate students. The text is not about a computer's architecture or acquiring a programming language.

Bork thinks educational materials should be prepared by teams of teachers, programmers,

and designers. Material preparation takes time, effort, money, an understanding of software/hardware capabilities, and good review and evaluation.

The approach should include a dialog interaction between student and program while using graphics, personalization, problem solving, and immediate feedback to motivate students and enhance education. The text provides a realistic view of the place of computers in education.

Several things will bother you about this book. The price is steep for what amounts to a collection of reprinted articles. While the material is very readable, interesting, and exciting, the social scientist or educational-material evaluator will find no data supporting claims about the efficacy of graphics, dialogs, or the computer learning environment.

However, the lack of assessment and evaluation is typical of the lag between instruction and critical analysis. There is a critical need for task analyses of learning procedures and statistical tests of the value of CAL materials and procedures. I agree with Bork's projections, but such speculation must eventually be supported by proof.

The physics-course examples might be difficult for the non-scientific reader to understand, and the redundant examples and demonstrations will bother the cautious reader.

Two years ago I was captured by Bork's innovative and enthusiastic approach to education at his NSF Chautauqua course. I hope educators will enjoy a similar fate by reading *Learning with Computers*. The CAL field is ready for many critical,

analytic, and creative minds.  
(*Digital Press, 12 Crosby Dr.,  
Bedford, MA 01730: Hardcover,  
\$28.*)

Timothy B. Jay  
February 1982

★ ★ ★ 1/2

**Microcomputer Data-Base  
Management**

**E.G. Brooner**

**Howard W. Sams & Co. Inc.**

*Microcomputer Data-Base Management* is an introduction, definition, and explanation of microcomputer data-base systems written for all experience levels. Mr. Brooner writes in a practical, down-to-earth manner that conveys complicated ideas without intimidating the reader.

The book describes and analyzes several commercial data-base software packages. It reviews data bases and their distinctive characteristics and introduces sorting techniques and the methods used to search files. A discussion of sequential, random-access, and binary search methods is thorough and well-written. An introduction and explanation of hashing is also given.

Other topics include sequential reading and searching of unsorted files and how to put keys in order. The book also provides an in-depth discussion of three different commercial data bases: Whatsit, Profile, and Selector.

For the real-estate executive, a sophisticated, custom appraisal program adapted for the Selector package is examined. A multiple-file software package (not a data-base system), two applica-

tions for Profile, and an overview of QSORT and sort utilities are also given.

This is an excellent tutorial book packed with information on data-base systems and their software applications. It is helpful for everyone from beginning computer-science students to those who already own or use computers.

(*Howard W. Sams & Co. Inc.,  
4300 W. 62nd St., P.O. Box  
7092, Indianapolis, IN 46206;  
Softcover, \$12.95.*)

Suzanne L. Foster  
October 1982

★ ★ ★ 1/2

**Microcomputer Math**

**William Barden, Jr.**

**Micro Applications  
Publishing**

*Microcomputer Math* deals with the structure of computer mathematics on an elementary level. Although specifically designed for the TRS-80 and the Color Computer, most of the information applies to all binary coded arithmetic. This book is CPU-dependent: no CPU instructions are mentioned: Z80, 6809, or otherwise.

Barden's loose style and anecdotal writing approach lessen the fears of readers intimidated by the subject. It makes for wordy reading, but for many, a painless approach is worth the verbosity.

The book is divided into 10 chapters, a glossary, and two appendices. The topics range from an introduction to binary numbers to a brief discussion of floating point. Each chapter ends with a few exercises to test

understanding, and the uncommented answers can be found at the end of the book.

Big Ed, the book's main character, is introduced in the first chapter. Throughout the book, Ed learns more and more about numbers through the help of customers from microcomputer component manufacturers.

This book is an excellent introduction to the elementary aspects of computer mathematics, provided you enjoy Barden's writing style. It fails in that it doesn't cover any topics of interest for the reader who has a good grasp of the elementary concepts and wishes to know more.

*(Micro Applications Publishing, 28182 Palmeda, Mission Viejo, CA 92692; \$9.95.)*

Bruce Powel Douglass  
March 1982

★  
***Microcomputing Techniques with Sample Programs***  
Stanley J. Evans  
Resten Publishing Co.

Everyone has preconceptions about a book before reading it, but any I had about this one were changed by the time I finished page five. Evans' book is not only a waste of money, it is a waste of time.

The bulk of the book is about computer languages. Basic gets 30 pages, PL/M gets 77, and APL is covered in nine. The Basic section has so many errors that anyone trying to learn from it would probably give up on computing altogether.

Part II consists of reference tables (hex to decimal conver-

sions and hex arithmetic) and two "Useful Programs," a number base conversion program with no documentation and a Dow Jones forecaster written in Radio Shack's Level I shorthand. The latter is from the TRS-80 Level I manual and has a couple of errors in it—one from the manual, the other compliments of Dr. Evans.

Computer arithmetic is well described, but mnemonic coding and ASCII code are given only cursory mention. More important, while Evans covers the concepts of flowcharting and programming, his programs contain mistakes and do not do what they are supposed to according to the text. Anyone trying to learn from this book is going to be confused by all the errors. Anyone who catches the errors is experienced enough not to require this book.

*(Resten Publishing Co., 11480 Sunset Hills Road, Reston, VA 22090; \$14.95. We were unable to contact this company for updated information.)*

Douglas Stewart  
March 1981

★★★ 1/2  
***The Micro Millennium***  
Christopher Evans  
Viking Press

At least one of Christopher Evan's predictions from his book *The Micro Millennium* has come true: The anti-computer movement of the early phase of the Computer Revolution has begun.

*The Micro Millennium* claims that we are in the early phase of a sweeping cultural revolution spurred by the advent of



microprocessors. The book is a concise and personable history of the people and scientific advancements that have helped nurture the computer to its present development.

In Evan's definition, the revolution did not begin until the invention of the transistor. He compares the Computer Revolution to the Industrial Revolution, which caused changes in all levels of society through "emancipation of the power of muscles."

Evans argues that the Computer Revolution will bring "the emancipation of the power of the brain." He then cogently predicts the computer's role in the short-term future, and some of the changes that will result from the computer's information storage and retrieval functions. Professions, such as law, medicine, and teaching will be a few of the jobs most severely affected by the spread of computers since they barter information.

The author also writes about the possibility of Ultra Intelligent Machines (UIMs) far surpassing the highest human IQ, and gives credence to the belief that computers are already capable of creative and original thought, as opposed to programmed logic.

Unfortunately, *The Micro Millennium* does not explore the ramifications of UIMs as cogently as it presents the history, present and short-term future of computers. Evans gets bogged down with straw dogs, and the book begins to lose its interest.

In the last chapters, the focus is lost altogether. The book rambles to an unintelligent close. But before he runs out of steam, Evans is able to convince

us that the Revolution is inevitable.

(*The Micro Millennium*, Viking Press, 40 W. 23 St., New York, NY 10010; \$10.95.)

Nancy Robertson  
August 1980

★ ★ ★

***Microsoft Basic Decoded & Other Mysteries***  
**James Farvour**  
**IJG Inc.**

In the foreword to this book, Harv Pennington says that "This book will delight the professional and the beginner." For the professional, this is undoubtedly true, but the beginner will find the book intimidating at first glance, and formidable upon the second. When you read it you'll discover it isn't quite as lucid as promised in the foreword.

Despite the unevenness and omissions of the first six chapters, the book is well worth its price because of the completeness and accuracy of the seventh and eighth chapters. Beginners and professionals will find it a welcome addition to their library, although the beginner may have a rough time reading and understanding it. Without a doubt, it is the most comprehensive book on Level II Basic ROM to be published so far. It is also the most comprehensive book on the interactions of Level II Basic ROM and TRSDOS.

(*IJG Inc.*, 1953 W. 11th St., Upland, CA 91786; Softcover, 312 pp., \$29.95.)

Terry Kepner  
January 1982

★ ★ ★ ★

### **More TRS-80 Basic**

**Inman, Zamora, & Albrecht**  
**John Wiley & Sons Inc.**

*More TRS-80 Basic*, a new addition to the Wiley Self-Teaching Guides, is an introduction to some of the more esoteric aspects of using and programming your TRS-80. This book covers such diverse and interesting subjects as graphics, animation, disk and cassette I/O, and the use of sound with Basic programs.

The overall composition of the book is excellent. It has an open layout liberally sprinkled with illustrations, allowing the reader to quickly locate any particular piece of information. Clear explanations are followed by reiterations, ensuring that the reader fully grasps new concepts before moving on. It's easy to skim the repetition if you already understand the subject, and you'll appreciate the reinforcement when you're still struggling with something new.

Excellent summaries are found at the end of each chapter. Self-tests appear throughout the book, generally at the end of each chapter, but are also embedded within those chapters that cover especially difficult or complex material. In keeping with the advanced emphasis of the book, the more common Basic commands are dispensed within the first few pages of the book. They are listed with concise, lucid definitions, and serve as a handy reference when the meaning of a particular term slips your mind.

Chapters dealing with graphics and animation make terribly obscure points seem

simple and logical. Best of all, you learn sophisticated techniques that work. You are given opportunities to try things out and experiment with your newfound knowledge.

Your understanding of what your hardware does and how it does it will also grow with this book. Both cassette and disk storage are covered in some detail, and a sample data file is set up using each medium. Sufficient cautions are given on the danger of indiscriminate POKING and other potentially destructive acts.

The inclusion of a short sort routine would be extremely helpful. The authors skirt this issue in the sections on data files by having the data entered in alphabetical order. One other annoying, although minor point deserves mention. There are too many Radio Shack ads in this book!

All in all, the book is a pleasure to work through. It is both entertaining and educational, and will be valuable to anyone who is interested in getting beyond the obvious and who wants to use more of the TRS-80's capabilities.

*(John Wiley & Sons Inc., 605  
3rd Ave., New York, NY 10158;  
Softcover, 280 pp., \$12.95.)*

Alyson Grupp  
December 1981

★ ★

### **Murder in the Mansion and Other Computer Adventures in Pocket Basic for the TRS-80**

**Jim Cole**  
**ARCsoft Publishers**  
**Pocket Computer**

The Pocket Computer isn't

meant to be a game machine. It has a limited memory and text display area and no graphics. Given this, Jim Cole might have attempted the impossible in this book of 10 game programs. It is, therefore, hard for me to decide whether to congratulate Cole on his technical achievement or criticize him for his basic failure in game design.

All the games are based on pseudo-random-number generation, and they perform exactly as promised in the sample runs. In each, the player is asked to put in a number at the start of the game. The program uses this number in a formula to generate pseudo-randomness to vary the games. Cole gives two methods for doing this in an appendix.

It is amazing that Cole managed to get so much variety out of such a simple idea. The title game, *Murder*, is a mystery that asks you to discover the place of the murder, weapon, and the murderer. If you guess wrong, it gives you a transparent clue.

*Lost Safari* is a rescue game of a hunting party in which the player guesses what the party is hunting, what African country they are in, and how many days it will take to reach them. *Saturn Death* is a space opera in which you must deal with asteroids and an "alien death probe." You can miss, damage, or destroy the target, and the asteroid can hit and destroy you. This is undoubtedly the showiest of the games.

*Buried Treasure* is a treasure hunt, similar to *Lost Safari*. The *Black Pearl*, *Barrel of Apples*, and *Wood Chuck Chuck* are thinly disguised number-guessing games that quickly approach maximum boredom.

*Memory Test* scores you on how well you remember a list of words and a secret number. *Scrambled Eggs* presents you with mixed-up letters you must sort out to spell words. *Fractured Descriptions*, the last program, is a party game that scrambles similes to inform you "Bob is as smart as a song" or "Joyce is as tall as an apple."

The book includes a sample run of each program.

Unfortunately, the programs, while elegant as exercises in Pocket Basic, are totally unsatisfactory as games.

I do not mean this as a criticism of Cole. He did what he could with the machine he had. It isn't his fault that the machine can't support a game.

Whether or not you should buy the book depends upon your interest. If you are looking for an evening's fun with a new game, forget it. On the other hand, if you want to study some finely thought-out programs—if you've just bought your Pocket Computer and are learning to program it—then this book might be just the thing for you. (ARCsoft Publishers, P.O. Box 132, Woodsboro, MD 21798, \$6.95.)

Bert Latamore  
February 1982

★ ★ 1/2  
***Musical Applications  
of Microcomputing***  
**Hal Chamberlin**  
**Hayden Book Company Inc.**

The true musical application of microprocessors is not the topic of Hal Chamberlin's book. Instead, the contents of this book cover the many methods of producing sound through electronic

means. Both traditional analog and developmental digital concepts are presented; uses of microprocessors are introduced; and the analysis, development, and reproduction of natural and original conceived sounds are discussed. In many ways this is a professional's book; cursory attention is given to the basics of acoustics and electronics, but these seem included by obligation rather than enthusiasm.

The author is recognized as one of the leading and most imaginative figures in the field of digital processing of sound, and one expects that this book would clearly demonstrate his wide-ranging skills. Perhaps it is an inevitable consequence that this wealth of knowledge can't be satisfactorily compiled in one volume which has both popular and professional pretenses.

Nevertheless, this book is destined to become one of the definitive books on digital creation and sound reproduction, and digital interfacing of analog synthesizers. It contains sample circuitry, Assembly programs, waveform diagrams, and drawings

The book is divided into three sections: Background, Computer-Controlled Analog Synthesis, and Digital Synthesis and Sound Modification. Chamberlin reveals himself to be an expert engineer/programmer, but the sections on the goals of music synthesis and its history are superficial. By contrast, the section describing the parameters of sound is clear and thorough.

Certainly this book is a valuable contribution to the field, and will be used by anyone seriously interested in the electronic production of sound who

is also highly qualified in electronic theory and application. But, just as certainly, the obvious significance of its contents will be a frustration to musicians and experimenters who wish to put its ideas to use. Chamberlin's work, impressive though it is, remains unbendingly technical, with neither recognition of nor concern about the musical art at even its rudimentary levels.

(Hayden Book Company Inc., 50 Essex St., Rochelle Park, NJ 07662; Hardcover, 650 pp., \$29.75.)

Dennis Bathory Kitsz  
September 1981

★ ★ ★ 1/2

***The Nature of Computation:  
An Introduction to Computer  
Science***

**Ira Pohl and Alan Shaws  
Computer Science Press**

*The Nature of Computation: An Introduction to Computer Science* is a rigorous introductory textbook to computer science and is for anyone just beginning in the field.

The authors claim their objective is to provide the following: a survey of the field; initial literacy of the language and methods found in computer science; and a historical, philosophical, and social perspective. The authors have succeeded admirably in achieving these goals.

The book is organized logically. First, it presents the reader with a taste of what programming is like with the use of a machine-independent, high-level language known as Algolic. Then, the book provides an interesting account of the develop-

ment of computational techniques from the abacus of the intricacies of the structure of a modern computer.

Third, the book explores programming languages and operating systems, before turning to an introduction of the theory of Turing Machines, computability, and algorithms. The book ends with a discussion of the social and ethical questions raised by the various ways in which our society uses computers.

A set of comprehensive exercises and questions at the end of each chapter, along with the comprehensive nature of the text itself, make *The Nature of Computation* an ideal textbook for an introductory college course. But it can also be used by an individual who wants a painless and clear introduction to computer science.

(*Computer Science Press, 11 Taft Court, Rockville, MD 20850; \$22.95.*)

Joel Benjamin  
October 1981

★ ★ ★

**On-line Magazine**

**Dave Beetle**

**Every three weeks**

This is strictly a classified-ad magazine. It has a very short lead time, so ads received by the publisher up to four days before publication can appear. A long list of computer club meetings held all over the country is a feature of each issue. As space allows, the publisher includes new hardware and software announcements.

(*On-line is no longer available.*)

Rod Hallen  
January 1980

★ ★

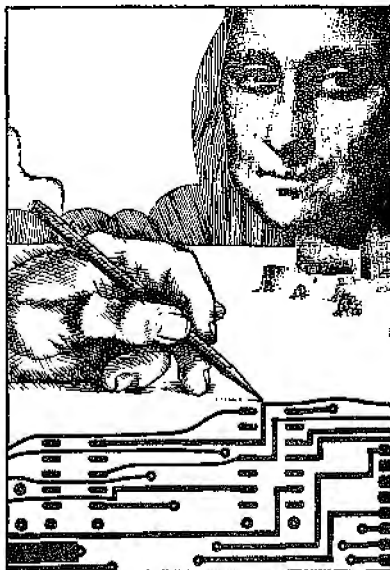
**Owning Your Home  
Computer: The Complete  
Illustrated Guide**  
**Robert L. Perry**  
**Everest House**

It's truly amazing how the total knowledge on a subject can be reduced to 224 pages (counting 24 pages of appendix, glossary, bibliography, and index). This book has an incredibly technical and involved first chapter on microcomputer networks (specifically, computers connected by telephone to other computers). It would have made a good appendix after you know something about computers, not before.

Nontechnicians, do not fret. Just skip over to chapter two, which is written just for you: "What is a Home Computer?" Did you know that the computer is a "mind appliance?"—an egg-beater for the brain.

Perry does settle down to talk about the parts of a microcomputer and the common terms: input, data, CTR, etc. He also devotes many pages to the various systems on the market, emphasizing those with exciting sounds, colors, music, and graphics—things that whirr, beep, and go bump in the night. However, his information is vague and incomplete. He doesn't mention Radio Shack's Model III, which is replacing the Model I, or their Color Computer. He confuses hardware and software advantages, implying that some systems are user-oriented when that often depends on the software.

Perry talks about computer uses in education, especially home education, in aiding the



handcapped, and in business. He gives some useful suggestions for finding a good microcomputer for your business and even gives a lesson in Basic computer programming, but even here there are errors in his information. If you want to learn programming, get a book written just for programming.

One of the main thrusts is the future applications of the computer. Specifically, they are the computers that can tie in through modems and telephones to larger computers and networks of computers for communications and sharing information and programs, and computers that will run the house—doing everything from controlling the thermostat and locking the doors to watching the kids and feeding the dog.

Some of Perry's information is useful and helpful for laymen, some is interesting for technical people, and some is inaccurate and confusing. The book at-

tempts too much; it tries to speak to all audiences on all aspects of microcomputer development and use, and it reaches none. It is poorly organized and difficult to read. As a "complete" book, it has something for everyone, but, unfortunately, it does not have much for anyone.

(Owning Your Home Computer: The Complete Illustrated Guide, sold by Everest House, 33 W. 60th St., New York, NY 10023 is no longer available.)

Mary Shooshan  
June 1981

★★★  
**Pascal**

**Paul M. Chirlian**  
**Matrix Publishers Inc.**  
**Interscholastic Book Service**

Professor Chirlian has made every effort to keep this textbook on Pascal as simple as possible. He explains how to use a flow chart, and the differences between working from a terminal and preparing a program for a batch operation. Debugging is well covered, as well as the idiosyncrasies in the language as it is used in different makes of computers.

Once the basics of programming in any language are grasped, understanding a new language involves only recognizing limitations and syntax.

The book is loaded with syntax flow charts so the experienced programmer can quickly skim through them. The language limitations are stated in the text, but an index makes the job easier.

The teacher's edition includes answers and example solutions

to problems provided at the end of each chapter.

(*Matrix Publishers Inc., Inter-scholastic Book Service, 11000 S.W. 11th St., Beaverton, OR 97005; Softcover, 211 pp., \$12.95.*)

Dennis Thurlow  
April 1981

★ ★ ★ ★

### **Pascal**

**David L. Heiserman**  
**Tab Books Inc.**

Heiserman has written the definitive programmer's and user's guide to Tiny Pascal—350 pages of examples and exercises ranging from loading the tape to writing sophisticated programs. None of the pedantry that seems to infect many Pascal books is apparent, either. No "Integration by Simpson's Rule" here; just practical hobbyist material. The first-time user can sit down and get a program running within a few minutes.

The book begins with operating instructions and proceeds to introduce syntax diagrams. Learning to read these is critical to learning Pascal well. From there, Heiserman dives into I/O with the Write statement, including detail on its graphics and screen control capabilities. Two chapters introduce variables and what to do with them; special care is taken not to confuse Basic programmers with the difference between an equal sign (=) and the Pascal assignment statement (: =).

Syntax diagrams for all parts of Tiny Pascal are presented, from If . . . Then . . . Else to looping features such as Repeat . . . Until and For . . . Do. The

when and why of using these statements are developed with clear examples that are meant to be typed in and run, not just read.

"A Miscellany of Pascal" collects most of the features you'll need when writing programs, most importantly the Case-Of, which is similar to Basic's On . . . GOTO. Plot and Inkey, for graphics and "on-the-fly" keyboard input, are also introduced.

The later chapters form a good introduction to structured design and programming. Numerous examples show the development of several programs from idea to completion. The final chapter is one that I would have welcomed when learning the language: translating Basic into Pascal. Each statement is covered in good detail, though there is really no easy way to make direct translation because of the highly structured nature of Pascal and the unstructured nature of most Basic programs. Nevertheless, the chapter is useful because it helps the beginner put things into a more familiar light.

An appendix containing tables of cursor control codes, TRS-80 graphics characters, and Pascal syntax diagrams is also helpful.

*Pascal* is an excellent book, especially for the beginner. For less than \$30 (the cost of this book plus Tiny Pascal) anyone with a TRS-80 can begin to learn the techniques of structured programming. It'll be the best 30 bucks you ever spent on your computer.

(*Tab Books Inc., Blue Ridge Summit, PA 17214; 350 pp., Softcover \$10.95, Hardcover, \$16.95.*)

William L. Colsher  
July 1981

★ ★ ★ ★

**Pascal**

**David L. Heiserman**

**Tab Books Inc.**

The content of Heiserman's book is excellent. The features of Tiny Pascal are exhaustively described. There's no chance of getting lost if you start at the beginning and work your way through. The first chapter tells you how to load the Tiny Pascal tape, and describes most of the monitor and editor commands available, complete with a print-out of the screen at each step.

The book then introduces Pascal's Write and Read statements so that you can immediately start to see things happen. Write and Read are not direct translations of Print and Output from Basic, nor are they quite the same as the statements in standard Pascal. Heiserman goes well beyond the incomplete description provided in the user's manual.

Thoroughly described and documented game programs in the back of the book include Graphic Dice, Graphic Slot Machine, Missile Shoot, Real-Time Animation, Hangman, Screwball Golf, and a whole chapter devoted to a space ranger mission game. There's also a chapter on structured programming, comparing top-down and bottom-up program design, and another on translating between Basic and Tiny Pascal.

Do you need Pascal by David Heiserman? I'd say yes, if you have the Tiny Pascal tape and little or no previous knowledge of Pascal. If you're an old hand at standard Pascal, you'll find it useful if the user's manual and your own experiments don't suf-

ficiently clarify the differences between Standard and Tiny. (Tab Books Inc. Blue Ridge Summit, PA 17214; 350 pp., Hardcover, \$16.95, Softcover, \$10.95.)

Ken Webb  
January 1982

★ ★

**Pascal: An Introduction to Methodical Programming**  
**William Findlay and David Watt**  
**Computer Science Press**

The main purpose of this book is to teach you the techniques of methodical programming; the achievement of fluency in Pascal is a secondary objective. The style is businesslike because the book is intended for classroom use, so the format may not be the most efficient for the somewhat experienced user.

Some of the features of the book include exercises, several appendices, and material that is specially marked if not essential to understanding. A significant drawback is the cross-cultural confusion in terminology resulting from its European approach; the authors are Scottish. For example, the authors make the assumption that Pascal is the common language of computing. In American computing, if a common tongue exists at all, it is Basic.

The book's cost effectiveness for the independent hobbyist is questionable. One could do better in an introductory Pascal edition than this college text. (Pascal: An Introduction to Methodical Programming is sold by the Computer Science Press,



11 Taft Ct., Rockville, MD  
20850: Softcover, 306 pp.,  
\$16.95.)

Chris Brown  
May 1980

★ ★ ★ ★

**The Pascal Handbook**  
Jacques Tiberghien  
Sybex Inc.

*The Pascal Handbook* is different. It is a handbook that makes no attempt to teach, reflect, or elucidate. Rather, it contains hundreds of pages of words used by and about Pascal, with a straightforward and comprehensive outline of the what's and where's of each word.

Tiberghien refers to these words and concepts collectively as "features." Since Pascal features are used differently from version to version, descriptions are provided for each feature as it exists in alternative versions of the language. These variants include the original Jensen and Wirth definition for the Control Data 6000, the proposed ISO standard, the University of California at San Diego (UCSD) Pascal, Hewlett-Packard 1000, OMSI Pascal-1 for the PDP-11, and Pascal/Z for the Z80 compiler. Each word is categorized either as a symbol, an identifier, or a concept.

Instead of circumlocutory explanations in elaborate English, each Pascal feature is described under four terse headings: syntax, description, implementation-dependent features, and examples. Each feature is presented with extraordinary clarity; even typeface plays an important role. Boldface headers indicate reserved words, upper-

case is used for identifiers, and lowercase refers to concepts. Although putting the keywords in alphabetical order doesn't parallel any logical relationship in Pascal, it certainly simplifies using the handbook.

I have only one minor suggestion: The number of examples for each feature could be increased. The Pascal Handbook would then serve not only as a programming reference, but also as a powerful tool for evaluating and selecting systems using various implementations of Pascal.

(Sybex Inc., 2344 6th St., Berkeley, CA 94710; Softcover, 473 pp. *The Pascal Handbook* is no longer available.)

Dennis Bathory Kitz  
January 1982

★

**Pascal Primer**  
David Fox and Mitchell Waite  
Howard W. Sams and Co. Inc.

*Pascal Primer* despite its strengths, may not be the book of choice for TRS-80 users. The experience of the authors and their obvious care and planning are impressive. Nevertheless, it has critical flaws.

The book is about programming, not about the use and characteristics of compilers and editors; the student will have to learn about those elsewhere. While this limitation makes sense for a book intended to interest all Pascal users, it causes some problems for a real beginner. If you know nothing about Pascal, you will need to learn about your compiler and editor before you can do any programming at all.

What about the advanced beginner? The book does a very nice job of explaining the advantages of Pascal over Basic, and the principles of structured programming. Some Pascal concepts that are not explained clearly in other beginning texts, are done very well here.

Another problem with this book is that it is based on UCSD standard Pascal, which causes trouble for the users of Tiny Pascal. Also, the authors used an Apple II computer and Apple UCSD Pascal for the development of the programs included within the text. Despite their claim that the book can be effectively used with any version of Pascal, it may be that the Apple II user stands to gain more from the book than the TRS-80 user. (*Howard W. Sams and Co. Inc., 4300 West 62nd St., P.O. Box 7092, Indianapolis, IN 46206; Softcover, 206 pp., \$16.95.*)

Margaret M. Grohman  
January 1982

★★★  
***Problem Solving and Structured Programming in Basic***  
**Elliot B. Koffman and Frank L. Friedman**  
**Addison-Wesley**

As an introduction to both computing and Basic, this book is a reference worth having on your shelf. It is readable and well-printed, with sharp graphic displays and examples that are extremely important to the beginner.

By treating the general operations that any Basic is designed to perform, the authors ensure that the book is often language independent. Also, they dis-



cuss three dialects: minimal Basic, Dartmouth Basic, and Basic-Plus.

This leaves out some features, but prepares you to read your system manual. With *Problem Solving* by Koffman and Friedman, your Basic can become fluent.

(*Problem Solving and Structured Programming in Basic is sold by Addison-Wesley, Jacob Way, Reading, MA 01867; Softcover, 306 pp., \$17.95.*)

Mikel Aickin  
May 1980

★★½  
***Programmers' Software Exchange Catalog***  
**Programmers' Software Exchange**  
**Quarterly**

This is a catalog of programs, which are equally divided between the TRS-80 and the Apple, with a few for the PET thrown in. Programmers with software for sale can get their creations listed without charge. Also listed are public-domain programs that are available for a \$5 copying charge each, or \$49.95 for 100. The majority of software listed is games, but there are some interesting applications programs here too. (*Programmers' Software Exchange, 2110 N. Second St., Cabot, AR 72023, sells for \$1 per copy.*)

Rod Hallen  
January 1982

★ ★  
***Programming in Basic  
 for Personal Computers***  
**David L. Heiserman**  
**Prentice-Hall Inc.**

If you just bought a microcomputer this book may be a worthwhile investment, but for anyone who has written a few Basic programs it's not worth the price. There are no special techniques or innovative programming ideas introduced that are not covered in any elementary programming text.

The reader is taken in a very deliberate fashion from powering up the computer through the common Basic statements, which are illustrated by examples as they are introduced. Some, however, are flogged unmercifully. For example, in chapter 4 I counted nine illustrations of For . . . Next loops for timing delays.

There are enough typographical errors scattered through the book to discourage the novice, though most should be picked up by the alert reader. An additional trouble spot is the author's inconsistency when referring to rows and columns in two-dimensional arrays. In one instance, the first subscript refers to a row while in the next it refers to a column.

Chapter 5 refers to a nonexistent figure; chapter 12 promises that better ways to input data will be presented in chapters 13 through 15, but there is no chapter 15. The author seems to think that simply keeping listings of a program, as it is developed, constitutes documentation. There is no real emphasis on variable listings, what variables represent, or how pro-

gram segments or subroutines work. Also, in a text designed for the novice programmer it would seem that the concept of an algorithm would be introduced. The word is never used.

In summary, I cannot recommend this inadequate, poorly edited book to anyone with even a moderate amount of programming experience. The novice could, however, learn enough from making corrections to justify its expense.

*(Prentice-Hall Inc., Englewood Cliffs, NJ 07632; Softcover, 333 pp., \$9.95.)*

Bryan Moran  
 July 1981

★ ★ ★ 1/2  
***Programs for Beginners on  
 the TRS-80***  
**Fred Blechman**  
**Hayden Book Co. Inc.**

This book fits in after you have gone through the Level I beginner's manual. You can use it before or after conversion. It contains 21 programs, all run with 4K of memory and all but two can be used in either Level I or Level II. The programs apply to both Model I and Model III.

(Model III conversion information is not included in the book, but the author will provide the instruction when sent an SASE.) No printers, disks, or interfaces are required.

The author admits that the programs are simple; they have little practical value regardless of whether your TRS-80 is used for business or pleasure. The value of this book is the accompanying discussion that goes with each program.

Each program begins by the

author telling you what you will learn. Several subjects are covered by each program. A description of the program is given next. Then a detailed explanation of each line of the program. This is the purpose of the book. You can learn and get ideas on how to write your own programs.

This is an excellent book for the beginner.  
(Hayden Book Co. Inc., 50 Essex St., Rochelle Park, NJ 07662; \$9.95.)

Douglas Stewart  
February 1982

★ ★ ★ ★

### **Programming Techniques**

#### **For Level II Basic**

**William Barden, Jr.**

**Tandy/Radio Shack**

Radio Shack's latest venture into the world of publishing is a book that teaches TRS-80 users advanced programming techniques. Covering everything from simple debugging to complicated sorts and tape I/O, this book can enhance the work of both amateurs and experts.

The book contains 12 chapters and is arranged in the author's self-teaching style, which mixes wit and humor with instruction. It is assumed that the reader has at least a nodding acquaintance with Level II Basic, but the text is most certainly not aimed at the experienced programmer. Those who are experienced, however, would not be unwise to read this guide—many chapters would be of interest to any programmer concerned with conservation of time and memory.

Barden's intent is to uncover

some unusual applications for Level II. Such applications certainly can't be found in the Level II Reference Manual. In fact, the book is less a handbook of technique than it is a handbook of well-documented applications.

The first chapter lays the groundwork for the rest of the book by giving the reader more explicit descriptions of Level II functions than presented in the reference manual. This chapter also covers the basic-programming methods that are used in later chapters.

Perhaps the greatest attention in this book is paid to increased programming efficiency, including methods to achieve faster graphics, searching, and sorting.

The reader is taken through a series of applications, each more involved than the one before it, and all related in one way or another. Most of the text is followed closely by actual examples, many of which combine several advanced methods providing the reader with a better understanding of not only the material at hand, but also how the current concept can be used with others.

One of the best chapters in the book deals with high-speed searching and sorting of data. Barden outlines several methods of accomplishing these fearful tasks, finally leading up to the fastest and most efficient. His descriptions are clear, and his examples are excellent. Imagine a 1000-element mailing list that doesn't take all night to sort.

Although this book doesn't have it all, the techniques it contains are helpful to all programmers. A special invitation goes out to professional programmers—you would do well to use

the methods in this book, especially if it's business oriented. I have seen several high-priced software packages that might come near being worth their price, had they made better use of the Level II's features.

*(Tandy/Radio Shack, Fort Worth, TX 76102; Softcover, 142 pp., \$4.95.)*

Tom Yager  
November 1980

★ ★ ★ ★

**Purser's List Magazine**  
**Robert Elliot Purser**  
**Quarterly**

This is a list of software available for the TRS-80, PET, and Apple computers. The printing is very small, so an enormous amount of software is listed. Individuals who have programs to trade are included. There is no charge to vendors or individuals who want to be listed. Advertising space is available at a reasonable cost. A unique feature is a photo section showing screen displays of various programs. This is the best and most complete list that I have seen. Every owner of the TRS-80, PET, or Apple computers should have a copy. *(Purser's List is available from Robert Elliot Purser, P.O. Box 466, El Dorado, CA 95623.)*

Rod Hallen  
January 1980

★ ★ ★

**Radio Shack Model III**  
**Service Manual**  
**Tandy/Radio Shack**

To use the *Radio Shack Ser-*

*vice Manual* you need a basic knowledge of digital electronics and microprocessor technology.

Since the writers expect the reader to possess a working knowledge of the computer, most of the manual consists of schematics, circuit traces, waveforms, and parts lists.

The manual falls short in the video monitor screen. Although it contains a schematic, it doesn't provide the kind of detailed service usually found with commercial televisions and monitors. The monitor is a modified RCA 12-inch black-and-white television, similar to the KCS-206. Anyone seriously considering servicing the video section should obtain KCS-206 service data, file 1981, B-1.

Another shortcoming is the omission of three important tables: a pin-out of the 50-connector bus I/O connector, a pin-out of the 34-connector parallel printer port, and a pin-out of the RS-232 port. Although the information is available on the appropriate schematics, a table listing each would be helpful. The manual provides advanced experimenters with the needed information for interfacing an unlimited variety of devices.

*(Tandy/Radio Shack, Fort Worth, TX 76102; \$9.95.)*

Jerry P. Turpin  
March 1982

★ ★ ★

**Robot Intelligence—With**  
**Experiments**  
**David L. Heiserman**  
**Tab Books Inc.**

What, another book on building a robot? After reading

the back cover and taking a quick thumb through, I realized *Robot Intelligence* is different from the average guide. It is not a book about building robots; it is a book about simulating robot behavior on a TRS-80.

The book's theme is the computer simulation of machine intelligence developed by a technique the author calls Evolutionary Adaptive Machine Intelligence (EAMI). These simulations are not very spectacular, but they do illustrate behavior that will prove thought-provoking if you have an interest in machine intelligence. The reader must study the explanations of how the programs work to gain anything from the simulations, the author warns. Typing the programs in and watching them run will not be enough.

The programs are straightforward and well-documented; the author includes hints on debugging them. He makes extensive use of subprograms, easing the task of entering the increasingly larger programs. For each step up in intelligence only the short main program must be revised and the new subroutines added.

Although the book is a source of much fun and interest, there are a few faults. The book lacks sufficient editing, as evidenced by the large number of typos. There are typos in the program listings, though the printout composite listings at the end of the chapters are generally error free.

The lengthy discussion of the philosophy of machine intelligence in the early part of the book may bother some who are anxious to get to the experiments.

These small criticisms do not detract from the fascination of this book and the creature it creates.

(Tab Books Inc., Blue Ridge Summit, PA 17214; Softcover, 322 pp., \$9.95.)

Don Stauffer  
June/July 1982

★ ★ ★ 1/2

### ***Running Wild—The Next Industrial Revolution***

**Adam Osborne  
Osborne/McGraw-Hill**

*Running Wild—The Next Industrial Revolution* is an entertaining and thought-provoking book. It contains history and conjecture. Although approaching futurism from a technological perspective, the author, Adam Osborne, remains skeptical of its benefits.

Osborne's premise is based on a simple dictum of physics that states that any ordered system undergoing change is an unpredictable one. He claims the microelectronic revolution has changed our lifestyle to the point that serious questions of cultural balance are being raised. This theory isn't new, but Osborne's treatment of it from a microcomputing viewpoint is unique.

He examines the recent history of technology in general and microelectronics in particular. Spawned in the space program, microminiature components and LSI have, he feels, created a revolution in information management.

Large corporations, such as IBM, have been caught off guard, and, unable to respond to the needs of the new science, have

relinquished the role of innovator to small, newly established companies.

Osborne also makes some interesting predictions such as fully one half of all current blue- and white-collar jobs will disappear in the next 25 years and programming will be taken over by computers.

The book is worthwhile not because of the accuracy of the analysis, but because it makes us aware of the social impact of today's innovations on tomorrow's reality. For this reason, it is a welcome addition to the nuts and bolts world of microcomputer literature.

(Osborne/McGraw-Hill, 2600 10th St., Berkeley, CA 94710; Softcover, \$3.95.)

Chris Brown  
April 1980

★ ★ 1/2

**Software Exchange Magazine**  
**The Software Exchange Bimonthly**

This is another software-oriented magazine. The main thrust is to review software and software-related books and periodicals. In addition, the editor and other insiders talk about software marketing, availability, and implementation. Classified advertising space is provided and letters from readers are published.

(*The Software Exchange*, P.O. Box 55056, Valencia, CA 91355, could not be reached for update information.)

Rod Hallen  
January 1980

★ ★ ★ 1/2

**The Soul Of A New Machine**  
**Tracy Kidder**  
**Atlantic Little, Brown**

As narrative history, Tracy Kidder's book *The Soul Of A New Machine* fails. By the last sentence the reader has no more insight into the principal characters and their dilemma than at the book's beginning. I hesitate to say this because I like so much about Kidder's book: it is tempting to allow his shortfall simply because he is the first contemporary journalist with the good sense to take the leaders of the computer revolution seriously. Kidder writes with style and skill but the reader demands more than timeliness.

The book is a first-person narrative describing the events surrounding the development of Data General Corporation's Eclipse NV/8000, 32-bit super minicomputer. The story is rife with accounts of corporate infighting, personality conflicts, the exploitation of young engineers and the survival savvy of grizzled project managers who have seen it all before. In fact, the tale is the stuff of a computer adventure game.

Kidder's attention to detail and journalistic style make the book a joy to read and, by contemporary standards of technical literature, a masterpiece. The book's most glaring fault is Kidder's character development. The people in his book never become more than shadowy images. Like the forces that drive them, the personae of Kidder's engineers lie buried below the surface of the narrative. Kidder's own role in the whole project and his relationship with Data

General is never clarified.

Kidder's portrayal of the computer engineer of the 80s is not flattering. The engineers who people his book are one-dimensional—technical. Ultimately, they are like other mortals. They search for meaning in what they do, invent some where there is none, and make the compromises that life requires just to stay in business. In spite of the heroics, all they have done is design a computer.

*The Soul Of A New Machine* may not be the definitive work of narrative history, but it is certainly the first serious journalistic effort to deal with computers and the men and women who build them. Tracy Kidder removes some mystique from computer technology and provides insight into the creators of these machines. His fondness for things technical and his ability to communicate the beauty and excitement of computer building are laudable. Despite its shortcomings, *The Soul Of A New Machine* is well worth reading. (Atlantic Little, Brown, 34 Beacon St., Boston, MA 02106; Hardcover, 293 pp., \$13.95.)

Chris Brown  
December 1981

★★★★

### **So You Are Thinking About a Small Business Computer** Canning Publications

Here is a highly organized, easy-to-follow manual designed to help the small business owner, who has a limited knowledge of computers, make an intelligent choice when considering purchase of a system.

The primary message is clear:

to avoid time-consuming and costly pitfalls when acquiring a first computer system. This requires that the business person do his/her homework; you will need to assimilate a fair amount of information on the subject before seeing a newly installed computer perform its first task. This manual contains most of that needed information, and lists references for all the rest.

The authors discuss many of the benefits to be gained by computerization, giving a highly realistic analysis rather than a pie-in-the-sky outlook. Warnings and cautions are given to help dispel common myths concerning computers.

The book is thorough in covering important basic information—how a computer works; the differences among maxi-, mini-, and microcomputers and their manufacturers and sales procedures; and common computer terms. There is also an excellent discussion on how to use a computer consultant effectively. Photos and captions describe representative system's components, capabilities, and prices, and an appendix lists leading computer suppliers.

All facets of the subject appear to have been examined. In helping the business person assess his/her computer needs, the authors even discuss the possibility of avoiding an in-house system by farming out the work to a computer service. Heavily emphasized, however, is the goals-oriented approach to choosing a system, beginning with software. The authors repeatedly suggest searching out existing combinations of software and hardware that are already performing the desired



job, and in the same line of business.

Finally, the text suggests that, no matter what reason has prompted a business to investigate computerization, odds are that a successful acquisition will open up more avenues of use for the system. The recommendation is that future expenses can be reduced or eliminated by buying or leasing a somewhat larger system than is required for the immediate job at hand.

*(Canning Publications, Vista, CA; \$12.50. This company could not be reached for update information.)*

David W. Smith  
July 1981

### **Starting Forth**

(See Page 154)

★★★

**Supermap**  
**Fuller Software**  
**Model I**

*Supermap* is a detailed map showing what functions are performed by different areas of TRS-80 memory, especially ROM. It doesn't contain a printout of Radio Shack's copyrighted ROM code, but describes the purpose of the code in various locations. You'll need your own disassembled listing of ROM to make the best use of the book.

*Supermap* documents about a thousand memory locations, in addition to revealing many interesting secrets such as how the TRS-80 obtains a random seed or responds to a default on the "MEMSIZE?" question. It also examines the Basic and machine-language tape formats and the

editor/assembler, an alphabetized table of entry points for Level II commands, and the RAM storage format for Basic programs.

This information is invaluable to anyone who wants to program a TRS-80 in machine language. It's also useful to the Basic programmer who wants to find out what some PEEK or POKE in a published program does.

According to a note from Fuller Software, the next edition will contain extra material, including a table format summary of the most useful subroutines, and will sell for \$8.95 instead of the current \$18.95.

*(Fuller Software, 630 East Springdale, Grand Prairie, TX 75051; Supermap is still available in limited quantities.)*

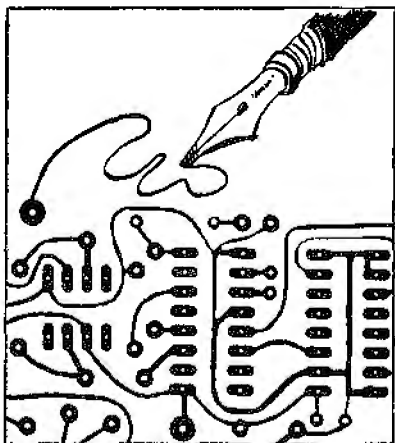
Paul Wiener  
July 1980

★★★

**TRS-80 Assembly Language**  
**Herbert S. Howe, Jr.**  
**Prentice-Hall Inc.**

Learning to write TRS-80 Assembly-language programs can be a very difficult task. Unlike Basic, in which the instructions bear a logical relationship to the operation intended, Assembly-language instructions usually resemble the names of exercises rather than computer operations.

Hubert S. Howe, Jr., a columnist for the *TRS-80 Monthly News Magazine*, offers his book as a guide to understanding this complex yet elegant language. Except for the few reservations noted below, this book is a good reference for both beginning and experienced Assembly-language programmers.



Howe divides his book into two sections. Part I introduces and explains the basic concepts of Assembly-language programming. In addition to describing the operation of the Z80, the heart of the TRS-80, it explains the organization of the TRS-80's memory and how to use the various permanent subroutines that reside in it.

Howe assumes the reader has access to a 16K Level II Model I and has Radio Shack's Editor/Assembler or an equivalent assembler (such as the Apparat EDTASM that comes with NEWDOS+).

Part II of the book illustrates the practical tasks that you can perform using Assembly-language programs. Three chapters are particularly outstanding. Chapter 15 describes how to use Assembly-language subroutines in conjunction with Basic programs. Chapters 16 and 17 discuss disk input/output and disk files.

On the negative side, a glossary and index are notably absent from the book. Both of these are important in an in-

troductory guide, since beginners can't be expected to retain everything they read the first time through.

Nevertheless, this book should prove valuable to TRS-80 owners. It is written specifically for the TRS-80; it takes advantage of the short-cuts and features designed into this computer, and it provides helpful hints of practical importance to programmers.

(Prentice-Hall Inc., Englewood Cliffs, NJ 07632; Softcover, \$9.95.)

Edward D. Young III  
October 1981

★ ★ ★

**TRS-80 Computing  
Newsletter**

★ ★ ★

**TRS-80 Bulletin  
Computer Information  
Exchange Inc.  
Monthly**

This is a true newsletter in that the bulk of the material published comes from its readers. It seems to lean toward the hardware side of computing, but there is a lot of software material, including some tutorial articles. Several of the articles saved me both money and time; they discuss how the TRS-80 works, how to make the computerist's life easier, and the latest news from Radio Shack on what is coming in the near and distant future. At the present time *TRS-80 Computing* does not contain many ads, but that may change as it is discovered by the small-computer business world.

The *TRS-80 Bulletin* is a com-

panion publication to TRS-80 Computing. In fact, most of the material presented comes from that magazine. However, *TRS-80 Bulletin* is given away free at computer stores. Free subscription applications are also available.

*(TRS-80 Computing and TRS-80 Bulletin are available from Computer Information Exchange Inc., P.O. Box 158, San Luis Rey, CA 92068.)*

Rod Hallen  
January 1980

★ ★ 1/2

**TRS-80 Disassembled Handbook**

**Robert Richardson  
Richcraft Engineering Ltd.**

The premise of the *TRS-80 Disassembled Handbook* is that once a user has struggled through machine code, "it is ridiculous to 'reinvent the wheel' . . . when these routines already exist in Level II ROM." Much of this premise is sound, and author Richardson, with some questionable generalizations about the primacy of his book and the ease of his method, takes the reader through the many useful subroutines available in Microsoft's Basic.

Richardson's technique stands out as singularly effective, because most of the speed of original machine-language software is maintained, and memory is conserved through generous use of calls to Level II ROM. On the other hand, Richardson is sometimes overbearing and self-righteous, and often useful points are obscured by his tone. The author is also a bit overeager in

self-appreciation when he states that his work is the first revelation of the inner workings of Level II ROM.

Even worse, the organization of the book is weak. Richardson does not help us learn; he presents a technique, some examples of its use, and plenty of oratory. Is the *TRS-80 Disassembled Handbook* worth its cover price? On its own, it provides only a modest return in exchange for your investment. *(The TRS-80 Disassembled Handbook is sold by Richcraft Engineering Ltd., Chautauqua, NY; Vol I \$10, Vol. II \$15.)*

Dennis Bathory Kitz  
June 1980

★ ★ ★

**TRS-80 Graphics  
Robertson and Grillo  
Tandy/Radio Shack**

There aren't many books that deal with a variety of graphics techniques for the TRS-80. *TRS-80 Graphics* is a good, inexpensive introduction for the beginner to this fascinating topic.

The book is in a large format and is easy to read. It doesn't even mention machine language, and so all the graphics techniques used are implemented in Basic.

This book is a good elementary introduction to TRS-80 graphics. One valuable aspect of this book is the preponderance of example Basic routines, along with example outputs for each routine. For the beginner, this approach should smooth over any fears of dealing with a complex subject.

The book includes a valuable

discussion of histograms, or bar-graphs, with routines to generate examples. While certainly a simple topic, it is useful for the end-user.

Since this is an elementary approach to graphics, pixel graphics aren't even mentioned until the seventh chapter, and not very much is done with them after that.

For the neophyte learning to use his computer, this book may well provide a valuable introduction to the topic of computer graphics.

*(Tandy/Radio Shack, Fort Worth, TX 76102; \$7.95.)*

Bruce Powel Douglass  
March 1982

★ ★ ★ ★

### **TRS-80 Interfacing**

**Jonathan A. Titus**

**Blacksburg Continuing Education Series**

**Howard W. Sams and Co. Inc.**

*TRS-80 Interfacing* is a basic book that should be in the library of all hardware designers who wish to work with the TRS-80 system for input/output control. It is written for a general reader who has a fairly good understanding of Level II Basic. All of Titus's tests and experiments are designed to operate on a Level II 4K machine with read/write memory and data storage capability.

Titus introduces the reader to the control signals one by one, and helps the reader grasp the purpose for this control system and its use. He then goes into the operational control of these signals by using Level II instructions in computer programming

steps. Titus leads the reader from the initial concept to a point where the reader can write the programs and generate the control signals he desires. For the size of the book, much information is presented.

The reader can obtain an overview of the subject in a quick reading. Next the breadboard circuit can be assembled, and the TRS-80 can be put to use. The book gives the reader practical experience and guidance with program examples and actual hardware. He should develop a good understanding of I/O devices and interfacing. *(TRS-80 Interfacing is sold by Blacksburg Continuing Education Series, Howard W. Sams and Co. Inc., 4300 West 62nd St., P.O. Box 7092, Indianapolis, IN 46206. \$10.95.)*

Ed Neister  
September 1980

★ ★ ★ ★ ★

### **TRS-80 Interfacing Book 2**

**Jonathan A. Titus,**

**Christopher A. Titus, and**

**David G. Larsen**

**Howard W. Sams and Co. Inc.**

The first volume of the Blacksburg Group's series on interfacing the TRS-80 covered the available input/output signals and software commands and the construction of simple I/O ports, address decoders, and interfacing A/D and D/A converters. This second volume deals with more advanced and sophisticated interfacing techniques such as data acquisition, signal processing, remote control, interrupts, and using D/A converters for graphics.

The first chapter is dedicated

in front of you to get the most out of the material. If you are upgrading your system from Level I to Level II, you will find it a necessary supplement to the owner's manual. (TRS-80 Level II Basic is sold by Tandy/Radio Shack, Fort Worth, TX 76102; Softcover, 250 pp., \$9.95.)

Dan Keen and Dave Dischert  
January 1981

★ ★  
**The TRS-80 Means Business**  
Ted Lewis  
John Wiley & Sons Inc.

Ted Lewis, author of *The TRS-80 Means Business*, outlines three goals for his book. According to the preface, the reader will learn how suitable Radio Shack's Model II is for his or her business application, develop Basic programs, and select the necessary hardware and software for a complete system. The reader, however, is in for a big surprise.

This book only assures the first-time buyer of a small-business computer about one thing—computers have helped businesses with similar needs. Real-life or practical examples sell computers, but the book provides few of these.

After addressing why you should buy a computer, the author tackles the job of programming. This seems useless; most business folks I've met could care less about direct access, hash functions, or binary search trees. But, even if you do want to try programming, the water gets deep quickly and there are pitfalls like missing

program listings. However, the programs are available on disk.

Hardware selection hints consist of a thumbnail sketch of the Model II technical parameters and brief mention of printers. The most useful feature for the would-be computer buyer is a two-page list of questions concerning software purchases.

The TRS-80 Model II does mean business, but this book remains trapped in the generalities common to the old school of computing.

(John Wiley & Sons Inc., 605 3rd Ave., New York, NY 10158; Softcover, 195 pp., \$12.95.)

Tim Daniel  
October 1982

★ ★ ★ 1/2  
**TRS-80 Microcomputer  
Technical Reference  
Handbook**  
Tandy/Radio Shack

The *TRS-80 Handbook* is written clearly and simply; anyone who has an understanding of hardware nomenclature will be able to keep pace with the book's technical data. It offers the inquisitive 80 owner information on testing and analyzing.

The book begins by displaying a system block diagram and a memory map. The first section contains a detailed analysis of what occurs during system operation. The second section, "Adjustments and Troubleshooting," is technical but useful. The remaining sections include information on TRS-80 interface with analog devices, the expansion-port edge-card pin connections, a complete parts listing, and schematics for the entire system.

to the circuitry required to use the microcomputer to drive devices such as lamps and AC motors, and to control devices powered by the AC line. All discussions are clearly illustrated by ample use of diagrams and design examples using common integrated circuits.

Chapter 2 explains how to use the TRS-80 to generate a signal that represents a physical measurement. A D/A converter is required to generate an external voltage; the chapter covers 8- and 10-bit converters' construction and design and their application in computer graphics. Two data acquisition projects are described.

These projects lead into a discussion of the more practical aspects of the subject, such as using data averaging and digital filtering to counter the noise a TRS-80 generates. Another chapter covers serial communication and remote control techniques used to measure signals at a location remote from the computer.

The final chapter explains interrupts, a confusing topic to a novice, with many visual aids and useful diagrams. It's difficult to discuss interrupts while limiting the subject to only Level II Basic. The authors have developed Basic programs that POKE machine-language programs into memory and then call them from the Basic program, to illustrate certain principles of interrupt operation.

The writing style, like other books in the Blacksburg series, is clear and unpretentious. Liberal use of diagrams, sample programs, and tables make complicated concepts understandable, though some of the larger

programs were short on documentation, and the book's interfacing projects are general enough to be used for a wide variety of applications. The authors have tackled a complex subject and produced a book that is an excellent addition to the library of anyone interested in using the TRS-80 as a control system.

(Howard W. Sams and Co. Inc.,  
4300 West 62nd St., P.O. Box  
7092, Indianapolis, IN 46206;  
\$9.95.)

George D. Dooley  
July 1981

★ ★ ★ ★

**TRS-80 Level II Basic,  
A Self-Teaching Guide  
Albrecht, Inman, Zamora  
Tandy/Radio Shack**

We wish this book had been available years ago as we struggled to learn Basic programming on a Level II. It assumes you know nothing about the Level II machine or programming.

The book is clearly written and illustrated with plenty of examples and periodic quizzes. To help you through the somewhat dry, technical process of learning computer programming, it has frequent cartoon drawings that add humor and provide a break in the text.

The book contains a comprehensive chapter on graphics, and the appendices cover a range of topics from setting up the TRS-80 to ASCII codes. While it covers much material, machine language and such techniques as string packing are not dealt with.

Unless you know programming, you'll need the computer

Particularly useful is an exact copy of circuit board layout, which is not in the user's manual. This handbook is an invaluable aid to the do-it-yourselfer.

(TRS-80 Micro Computer Technical Reference Handbook is sold by Tandy/Radio Shack, Fort Worth, TX 76102; \$9.95.)

James R. Fatz  
May 1980

★ ★

**TRS Yellow Pages  
Micro Architect  
Five times per year**

It is hard to tell what direction this paper will take. Classified ads will be accepted, while the bulk of the November-December 1978 issue is made up of advertisements for the publisher's software. Subscriptions are free. (TRS Yellow Pages is published by Micro Architect, 96 Dothan St., Arlington, MA 02174.)

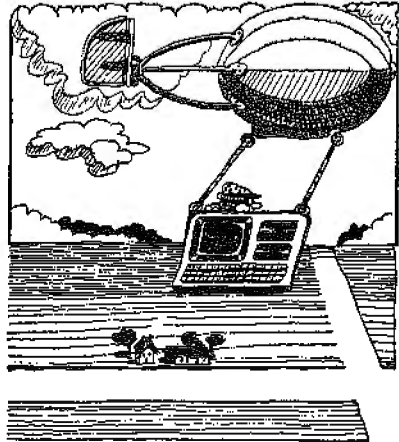
Rod Hallen  
January 1980

★ ★ 1/2

**Videotext: The Coming  
Revolution in Home/Office  
Information Retrieval**  
Efrem Sigel, et al.  
Knowledge Industry  
Publications Inc.

Videotext, "printed information transmitted to a tv or computer terminal video display," has attracted the interest of major economic forces. The governments of Britain, France, Germany, and Japan have each invested millions to research videotext's potential markets

and to establish videotext technology. According to the authors of this book, these groups are betting on "the willingness of business (and some individuals) to pay high costs for information, provided the information is tailored to their needs and can be delivered nearly in-



stantaneously." The book defines the terminology of the new technology, looks at its history, and gives an overview of the present.

Videotext is differentiated in the book as teletext and viewdata.

This book considers British videotext, CEEFAX and Prestel, thoroughly, and then goes on to discuss videotext developments in other countries. The last chapters, dealing with videotext in Canada, France, the United States, and other countries, are the most impressive. American readers may be disappointed to see so little coverage given to Qube, CompuServe, The Source, and other American developments, but the book provides the only comprehensive overview of

## BOOKS

---

international videotext developments.

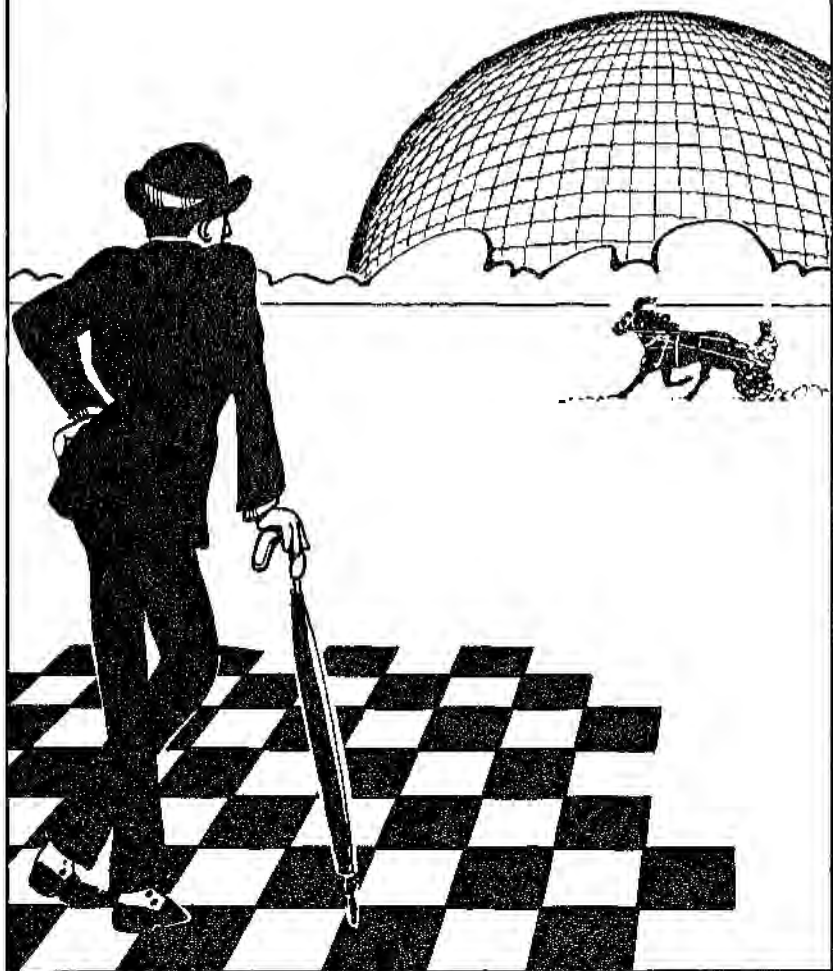
The estimated cost of developing the different systems discussed, along with the names of the companies who provided the hardware and software, is provided. An appendix lists the addresses of companies involved with videotext businesses and technology. This book, written by people who are well-

established in the young field, is clearly addressed to business readers, and for them it is an excellent source. But *Videotext* is not "essential reading for everyone involved in the communication of information." (*Knowledge Industry Publications Inc., New York, NY; Soft-cover, 152 pp, \$8.95.*)

Nancy Robertson  
January 1982



# INDEXES





# SOFTWARE PRODUCT INDEX

Accel2 . . . . .	16
Adventure . . . . .	82
Aids II . . . . .	26
Aids III . . . . .	26, 27
Air Flight Simulation . . . . .	82
Aircraft Instrument Approach Simulator . . . . .	266
Alcatraz II . . . . .	82
Alien Defense . . . . .	83
Android Nim . . . . .	84
Ann Rose—Our Accounts Receivable Clerk . . . . .	1
Ants . . . . .	85
Arcade 80 . . . . .	86
Armored Patrol . . . . .	87
Asptch 3.2 . . . . .	205
Astro Mines . . . . .	86
Astro-Scope . . . . .	88
Astrology . . . . .	267
Asylum . . . . .	89
Attack Force . . . . .	90
AutoBasic . . . . .	267
AUTOK . . . . .	206
Automated Communications Exchange . . . . .	301
B-1 Bomber . . . . .	90
B17 . . . . .	207
Bascom Basic Compiler . . . . .	18
Basic Compiler . . . . .	16, 19
Basic-1P . . . . .	149
BasicPro . . . . .	207
Bootstrap . . . . .	208
Boss . . . . .	179
Bounceoids . . . . .	91
Business Mail System/The . . . . .	30
C-Bug . . . . .	180
C.C. Writer 1.0 . . . . .	238
Calcs III . . . . .	27
Cassette Comm . . . . .	195
Cave Hunter . . . . .	92
CBasic . . . . .	149

# INDEX

---

CC EDTASM +	63
CCA Data Manager	27
CCForth	151
Cheaptalk	191
Check Register Plus	2
Checkwriter-80	2
Chextext	239
Cocobug Debugging Monitor	181
Color Berserk	92
Color Diagnostics, Editor and Assembler	184
Color Pac Attack	93
Color Pascal	152
Color Scripsit	242
Color Space Invaders	93
ColorForth	154
Colorterm	195
Combat	94
Compiler Basic	158
Compressor 1.1	208
CompuServe Software Package/The	197
Computer Acquire	95
Computer Downs	269
Computer Major League	95
Conflict 2500	96
Cross-Reference Utility	209
Crush, Crumble, and Chomp	97
Dancing Demon	141
Data Ace	32
Datahandler/The	33
Date-O-Base Calendar	39
Datestones Of Ryn/The	98
dBase II	40
Deathmaze 5000	98
Defense Command	99
Deluxe Personal Finance	146
Dialog Information Retrieval Service	198
Dilithium Tapes	74
Discat	209
Disk Directory	210
Disk Doctor/The	211
Disk Drive Timer (DDT)	211
Disk Instruction Course	75
Disk Mailing List	41
Disk*Mod	212
DISKNAME/BAS	213
Doctor and Fetch	76
DOSPLUS	279
DOSPLUS 3.3	52
DOSPLUS 3.4	54, 283
Dragons of Hong Kong	100
Draw5	102

DSMBLR . . . . .	213
Dynamic Report Generator . . . . .	6
EDAS . . . . .	65
EDAS 4.0 . . . . .	66
Edit . . . . .	214
EDTASM+ . . . . .	66
Electrasketch . . . . .	141
Electric Pencil 1.0 . . . . .	243
Electric Spreadsheet . . . . .	7
Eliminator . . . . .	102
Eliza . . . . .	76
ENHBAS . . . . .	141
ESP-1 . . . . .	215
Everest Explorer . . . . .	103
Extended Color Basic . . . . .	159
Extended Mumath . . . . .	161
Falling Bricks . . . . .	86
Faster 1.2 . . . . .	215
FED File Editor for LDOS . . . . .	217
File Management System . . . . .	42
Finance—Loans and Investments . . . . .	7
Flextex . . . . .	244
FLOPPYCAT/BAS . . . . .	213
Forbidden City . . . . .	104
FORTRANSiator . . . . .	217
Fractional Sound . . . . .	77
G.E.A.P. 1.0 . . . . .	142
G.E.A.P. 1.3 and 2.0 . . . . .	143
Galactic Empire . . . . .	104
Galaxy Invasion . . . . .	105
Gambiet . . . . .	106
General Accounting Package . . . . .	8
Generalized Subroutine Facility . . . . .	219
Gomoku . . . . .	107
GRBasic 3.0 . . . . .	162
H-O-R-K-S . . . . .	146
Haunted House . . . . .	108
Heathkit Assembly-Language Programming Course . . . . .	77
Hellfire Warrior . . . . .	108
Hexspell . . . . .	239
Hypergate Centurian . . . . .	109
Hypergate Patrol . . . . .	109
ICL (Interactive Control Language) . . . . .	163
IDM-IV . . . . .	43
IDM-V . . . . .	33
In Memory Information 3.0 . . . . .	9
Individual Study Center/The . . . . .	78
Infinite Basic . . . . .	219
Information Storage and Retrieval (ISAR) . . . . .	44
Institute/The . . . . .	110
INTASM 2.0 . . . . .	68

# INDEX

---

Interlude: The Ultimate Experience . . . . .	111
Invaders From Space . . . . .	113
Invaders Plus . . . . .	105, 113
Invasion From Outer Space . . . . .	105
ISAM . . . . .	45
KEEPIT 2.0 . . . . .	220
Keynote . . . . .	192
Kim's Game . . . . .	114
KWICOS . . . . .	56
Lablnakr . . . . .	245
Labyrinth . . . . .	114
Laser Defense . . . . .	115
LDOS . . . . .	57
LDOS 5.1 . . . . .	58, 283
Level I in Clevel II RAM . . . . .	221
Level III . . . . .	164
Life List . . . . .	269
Line Renumbering . . . . .	222
Lisp . . . . .	165
Litigation Support System . . . . .	9
Local Call for Death . . . . .	100
Maces & Magic . . . . .	116
Macro Mon—The Shadow . . . . .	182
Mail/File List . . . . .	30
Maillist . . . . .	45
Master Control . . . . .	223
Master Directory 1.1 . . . . .	223
Master Reversl . . . . .	117
Maxi Check Register . . . . .	2
Maxi Manager A.3.1 . . . . .	33
Maxi Micro Manager . . . . .	27
Maxprint . . . . .	224
Meteoroids . . . . .	93
Micro Music . . . . .	193
Microcosm III . . . . .	118
Micro-poly . . . . .	119
Microproof . . . . .	239
Microsoft Basic Compiler . . . . .	20
Microstat . . . . .	10
Midway Campaign . . . . .	119
Mind Thrust . . . . .	120
Miz'spell . . . . .	239
MMSFORTH . . . . .	166
MMSFORTH 2.0 . . . . .	167
Mod II Games . . . . .	121
Model II Compiler Basic . . . . .	170
Model II Scipsit . . . . .	246, 250
MON-2 . . . . .	183
Morse Code Transmit and Receive Program . . . . .	270
Moving Signboard . . . . .	271
MULTIDOS . . . . .	283

Music Composer/Editor . . . . .	193
Music Master . . . . .	192
Mychess . . . . .	106
MZAL . . . . .	69
Name and Address System . . . . .	30
New Starship Voyages . . . . .	85
NEWDOS80 2.0 . . . . .	61, 283
Newsript 6.0 . . . . .	248
Newsript 7.0 . . . . .	249
Newtalk . . . . .	224
Newtrieve . . . . .	46
North Atlantic Convoy Raider . . . . .	90
Nukewar . . . . .	90
OBJREL . . . . .	225
Odin . . . . .	117
Omniterm . . . . .	199
On His Majesty's Ship Impetuous . . . . .	100
Orchestra-80 . . . . .	194
Orchestra-85 . . . . .	310
Orchestra-90 . . . . .	310
Packer 1.5 . . . . .	225
Paddle Pinball . . . . .	121
Palantir Word Processing . . . . .	250
Parsector 8 . . . . .	85
Pascal I and II . . . . .	172
Pascal-80 . . . . .	173
Penetrator . . . . .	122
Pensawrite Word Processor . . . . .	252
Personal Finance . . . . .	147
Pinball . . . . .	113
PL/I-80 . . . . .	175
Planet Miners . . . . .	123
Planetary Lander . . . . .	123
Poor Man's Text Editor . . . . .	253
Power Pack . . . . .	184
Powerdraw . . . . .	145
Profile II . . . . .	47
Profile III + . . . . .	48
Profile Plus . . . . .	47
Programmer . . . . .	226
Proofreader . . . . .	239
Pyramid . . . . .	124
QEDIT . . . . .	206
QSD Utility Disk #1—LDOS . . . . .	227
Quest . . . . .	124
Quill Driver: A Text Formatting Program . . . . .	228
Qwerty . . . . .	253
Radex-10 . . . . .	49
Radio Shack Basic Compiler . . . . .	20
Rat's Revenge . . . . .	125
Refware Thesaurus . . . . .	254

# INDEX

---

Regression II.....	79
Remasscm-1.....	80
Remodel-Proload.....	219, 228
Renumber.....	229
Robot Attack.....	125
RPM.....	215
RSM-2 Monitor.....	186
Sargon II.....	106
Scarfman.....	126
Scripnt.....	50
Scrip232 (Scripmod).....	256
Scripsit 1.0.....	243
Scripsit 2.0.....	246, 250
SDS80C Editor/Assembler/Monitor.....	70
Search and Sort.....	229
SECS (Screen Edit Control System).....	145
Select Word Processing.....	250
Sex-O-Scope.....	88
SFinks.....	106
SFinks 3.0.....	127
Sigmon.....	145
Small Home-Business Software.....	148
Smart Terminal Program.....	201
Smart80D.....	202
Snapp III.....	230
Snapp IV Auto Map.....	230
Snapp V Extended File Mapping Support.....	232
Sole.....	233
Sort-II.....	51
Southeastern Textan.....	234
Space Ace 21.....	85
Space War.....	93
Space Waste Race.....	80
Special Delivery.....	51
Spectaculator.....	13
Spell 'N Fix.....	256
ST80-III.....	202
Star Blaster.....	128
Star Run.....	86
Star Warrior.....	129
Starfighter.....	129
Statistical Analysis.....	10
Step-By-Step.....	81
Stud5.....	102
Subscript 5.0.....	244
Super Color Writer 1.0.....	258
Super Micro Pro Football.....	130
Super Nova.....	131
Superscript.....	260
Superstep.....	187
Swamp Wars.....	131



T80-FS1 Flight Simulator . . . . .	133
Tanktics . . . . .	133
Tasmon . . . . .	188
TDS/DFT . . . . .	204
Telewriter . . . . .	261
Temple of Apshai . . . . .	134
Tiny Pascal . . . . .	177
Tower of Orlandor . . . . .	135
Trakcess . . . . .	234
TRS-232 Formatter . . . . .	271
TRS-80 Disk Editor and Assembler . . . . .	71
TRS-Opera . . . . .	135
Tshort . . . . .	235
Two Heads of the Coin . . . . .	100
Ultra-Mon . . . . .	190
Uniterm . . . . .	204
UOLISP . . . . .	165
UCSD Pascal Compiler . . . . .	22
Utility Pack . . . . .	235
VARKEEP . . . . .	236
Video Checkers . . . . .	136
Visigraph . . . . .	15
Voyage of the Valkyrie . . . . .	136
Win21 . . . . .	137
Winged Samurai . . . . .	138
Wizard/the . . . . .	138
Word . . . . .	264
Word-IV . . . . .	43
Word Challenge . . . . .	135
Wordo . . . . .	139
Wordsmith . . . . .	265
XBasic . . . . .	178
XBE . . . . .	237
Xbug . . . . .	190
XTEND40 . . . . .	215
Z80ZAP/CMD . . . . .	238
ZBasic 2.0 . . . . .	25
ZBasic Compiler . . . . .	22
Zork . . . . .	140



# HARDWARE PRODUCT INDEX

80-Graphix Board.....	364
Acu-Data Tape Digitizer.....	365
AD-III Audio Amplifier.....	365
Anadex DP-8000 Printer.....	313
Anderson-Jacobson Selectronic 841.....	313
Archbold High-Speed Modification.....	349
Base2.....	314
Beta-80.....	366
C. Itoh Prowriter 8510.....	316
C. Itoh Daisy Wheel Printer.....	318
Centronics 730.....	319
Centronics 737.....	314
Centronics 739.....	320
CG-16 Symbol Generator.....	362
Cognivox.....	305
Color Computer Disk System.....	275, 275
Comprint 912.....	321
Coosol 101B-80E Printer.....	321
Cyrptext.....	367
Daisy Wheel II.....	322
Dip-81.....	323
Disk-80.....	288
Doubler.....	279
DP-9500 Line Printer.....	323
Drive Control Unit.....	281
Dynatyper Keyboard Actuator.....	323
Eprom Programmer Model AN-551.....	367
EPS-80 Word Processor.....	354
Epson MX-80.....	314
Escon Selectric Interface.....	324
ETF-80 Electric Typing Fingers.....	354
Exatron Stringy Floppy.....	366
FCI-80.....	350
Floppy Disk Controller Kit.....	282
Gold Plug 80.....	368
Hayes Stack Smartmodem.....	301
High Speed Upgrade.....	351
Home Computer Work Table.....	369

# INDEX

---

IDS 440 . . . . .	325
IDS 445 . . . . .	314
IDS 460 Printer . . . . .	326
Internal Memory/The . . . . .	289
Isolated Word Speaker Trained System . . . . .	307
KGS-80 Keyboard Actuator . . . . .	327
LCA-47 Lowercase Adapter . . . . .	362
Line Printer III . . . . .	329
Line Printer IV . . . . .	329
Line Printer VI . . . . .	330
Line Printer VII . . . . .	332
Line Printer VIII . . . . .	333
LN Doubler 5/8 . . . . .	283
LNW Expansion Board . . . . .	290
LNW System Expansion Board . . . . .	291
LNW-80 . . . . .	356
LSI Soft-view CRT . . . . .	369
LX-80 Expansion Interface . . . . .	293
Maxi-Disk . . . . .	370
Mayday +S . . . . .	370
MDX-2 Complete Kit of Parts . . . . .	295
MDX-2 PC Board . . . . .	295
MDX-3 Interface PC Board . . . . .	296
Memory Box/The . . . . .	297
Micro Cue Print Buffer . . . . .	345
Micro Mainframe Disk Controller . . . . .	287
MicroCompatible Inc./The Plug Compatible Processor . . . . .	371
Microconnection/The . . . . .	303
Microline 80 . . . . .	314, 335
Microline 82A . . . . .	335
Micromatic-80 Printer . . . . .	336
Micromouth . . . . .	308
MM+ (Memory Expansion) . . . . .	298
Model 800 Printer . . . . .	337
Model I Speed-Up . . . . .	351
Modem I . . . . .	304
Modem 80 . . . . .	304
MT-32 Printer/Interface Module . . . . .	299
Music Box/The . . . . .	309
Novell Image 800 Printer . . . . .	338
Orchestra-80 . . . . .	194
Orchestra-85 . . . . .	310
Orchestra-90 . . . . .	310
P1 Microprinter . . . . .	339
Patch/The . . . . .	363
Percom Disk Drives . . . . .	277
Percom Doubler II . . . . .	283
PI80C Parallel Printer Interface . . . . .	345
Plug 'N Power Controller . . . . .	372
PMC SFD-51-A . . . . .	283
QuCeS Hard Disk Subsystem . . . . .	278

Quick Printer . . . . .	339
Rex 80 ROM Extender . . . . .	373
RS-232C Board . . . . .	299
SA800 Eight-Inch Disk Drive . . . . .	293
Shuffleboard . . . . .	370
Silver-It . . . . .	373
SK-2 . . . . .	352
Smith-Corona TP-1 . . . . .	339
Snapp Spooler . . . . .	346
ScoperSpooler . . . . .	347
Speak-2-Me-2 . . . . .	311
Sprinter II . . . . .	353
Starbuck 8882 . . . . .	375
Sirobe 100 Digital Plotter . . . . .	341
Tandy Six-Pen Plotter . . . . .	342
TC-8 Cassette System . . . . .	350, 376
TFD 100 Percom Disk Drives . . . . .	277
TFD 100-1 Percom Disk Drives . . . . .	283
Trendcom 100 Printer . . . . .	343
TRS-80 Color Computer . . . . .	358
TRS-80 Plotter/Printer . . . . .	344
TRS-80 Serial I/O Board . . . . .	300
TRS-80 Voice Synthesizer . . . . .	312
TRS-80/The . . . . .	360
UPI-3 Serial Printer Interface . . . . .	348
UPI-4 Serial Printer Interface . . . . .	348
Video 1 . . . . .	377
Wolfbug 64K Color Computer Upgrade . . . . .	300



# BOOKS INDEX

<i>55 Color Computer Programs for the Home, School, &amp; Office</i> . . . . .	381
<i>57 Practical Programs and Games in Basic</i> . . . . .	382
<i>1001 Things to Do with Your Personal Computer</i> . . . . .	382
<i>6809 Assembly Language Programming</i> . . . . .	383
<i>8080/8085 Assembly Language—Techniques for Improved Programming</i> . . . . .	384
<i>A Bibliography of Computer Music: A Reference for Composers</i> . . . . .	385
<i>The Alternate Source</i> . . . . .	385
<i>An Introduction to Computer Music</i> . . . . .	386
<i>Basic Adventures</i> . . . . .	387
<i>Basic Conversions Handbook for Apple, TRS-80, and PET Users</i> . . . . .	388
<i>Basic Faster and Better &amp; Other Mysteries</i> . . . . .	389
<i>Basic for Home Computers</i> . . . . .	390
<i>Basic Fortran</i> . . . . .	391
<i>Basic Handbook: An Encyclopedia of the Basic Computer Language/The</i> . . . . .	392
<i>Basic Scientific Subroutines Vol. 1</i> . . . . .	393
<i>Basic Second Edition</i> . . . . .	390
<i>CLOAD</i> . . . . .	394
<i>Computer Games for Businesses, Schools and Homes</i> . . . . .	394
<i>Computer Programming in Basic: A Self-Instruction Course in 4 volumes</i> . . . . .	395
<i>Computer World</i> . . . . .	396
<i>CP/M Handbook with MP/M</i> . . . . .	397
<i>CP/M Primer</i> . . . . .	398
<i>Creative Kids Guide to Home Computers/The</i> . . . . .	398
<i>Custom TRS-80 &amp; Other Mysteries</i> . . . . .	399
<i>Data File Programming in Basic</i> . . . . .	390
<i>Don't</i> . . . . .	400
<i>Elementary Basic: Learning to Program Your Computer in Basic with Sherlock Holmes</i> . . . . .	400
<i>Elementary Pascal: Learning to Program Your Computer in Pascal with Sherlock Holmes</i> . . . . .	400
<i>Fast Basic</i> . . . . .	401
<i>Free-Lance Software Publishing</i> . . . . .	402
<i>Getting Started with TRS-80 Basic</i> . . . . .	403
<i>Going Ahead with Extended Color Basic</i> . . . . .	403
<i>Graphic Software for Microcomputers</i> . . . . .	403
<i>How to Make Money with Your Microcomputer</i> . . . . .	404
<i>Inside Basic Games</i> . . . . .	405
<i>Inside Level II</i> . . . . .	405
<i>Introduction to 8080/8085 Assembly Language Programming</i> . . . . .	390

# INDEX

---

<i>Introduction to T-Bug</i> . . . . .	406
<i>Introduction to TRS-80 Graphics</i> . . . . .	407
<i>Introduction to TRS-80 Level II Basic and Computer Programming</i> . . . . .	407
<i>Learning Level II</i> . . . . .	408
<i>Learning TRS-80 Basic for Models I, II/16 and III</i> . . . . .	408
<i>Learning with Computers</i> . . . . .	410
<i>Micro Millennium/The</i> . . . . .	412
<i>Microcomputer Data-Base Management</i> . . . . .	411
<i>Microcomputer Math</i> . . . . .	411
<i>Microcomputing Techniques with Sample Programs</i> . . . . .	412
<i>Microsoft Basic Decoded &amp; Other Mysteries</i> . . . . .	413
<i>More TRS-80 Basic</i> . . . . .	414
<i>Murder in the Mansion and Other Computer Adventures</i> . . . . .	414
<i>Musical Applications of Microcomputing</i> . . . . .	415
<i>Nature of Computation: An Introduction to Computer Science/The</i> . . . . .	416
<i>On-Line</i> . . . . .	417
<i>Owning Your Home Computer: The Complete Illustrated Guide</i> . . . . .	417
<i>Pascal</i> . . . . .	418, 419, 420
<i>Pascal: An Introduction to Methodical Programming</i> . . . . .	420
<i>Pascal Handbook/The</i> . . . . .	421
<i>Pascal Primer</i> . . . . .	421
<i>Problem Solving and Structured Programming in Basic</i> . . . . .	422
<i>Programmers' Software Exchange</i> . . . . .	422
<i>Programming in Basic for Personal Computers</i> . . . . .	423
<i>Programs for Beginners on the TRS-80</i> . . . . .	423
<i>Programming Techniques for Level II Basic</i> . . . . .	424
<i>Purser's List</i> . . . . .	425
<i>Radio Shack Model III Service Manual</i> . . . . .	425
<i>Robot Intelligence—With Experiments</i> . . . . .	425
<i>Running Wild—The Next Industrial Revolution</i> . . . . .	426
<i>Software Exchange</i> . . . . .	427
<i>Soul of a New Machine/The</i> . . . . .	427
<i>So You are Thinking About a Small Business Computer</i> . . . . .	428
<i>Starting Forth</i> . . . . .	154
<i>Supermap</i> . . . . .	429
<i>TRS-80 Assembly Language Programming</i> . . . . .	429
<i>TRS-80 Computing</i> . . . . .	430
<i>TRS-80 Bulletin Computer Information Exchange</i> . . . . .	430
<i>TRS-80 Disassembled Handbook</i> . . . . .	431
<i>TRS-80 Graphics</i> . . . . .	431
<i>TRS-80 Interfacing</i> . . . . .	432
<i>TRS-80 Interfacing Book 2</i> . . . . .	432
<i>TRS-80 Level II Basic, A Self Teaching Guide</i> . . . . .	433
<i>TRS-80 Means Business/The</i> . . . . .	434
<i>TRS-80 Microcomputer Technical Reference Handbook</i> . . . . .	434
<i>TRS-80 Yellow Pages</i> . . . . .	435
<i>Using CPM</i> . . . . .	390
<i>Videotext: The Coming Revolution in Home/Office Information Retrieval</i> . . . . .	435



# COLOR COMPUTER SOFTWARE INDEX

Alcatraz II.....	82
C-Bug.....	180
C.C. Writer 1.0.....	238
Cave Hunter.....	92
CC EDTASM+.....	63
CCForth.....	151
Cocobug Debugging Monitor.....	181
Color Berserk.....	92
Color Diagnostics, Editor and Assembler.....	184
Color Pac Attack.....	93
Color Pascal.....	152
Color Scripsit.....	242
Color Space Invaders.....	93
ColorForth.....	154
Colorterm.....	195
CompuServe Software Package/The.....	197
Date-O-Base Calendar.....	39
Disk Doctor/The.....	211
Extended Color Basic.....	159
Finance—Loans and Investments.....	7
Meteoroids.....	93
Newtalk.....	224
Power Pack.....	184
Quest.....	124
SDS80C Editor/Assembler/Monitor.....	70
Secs (Screen Edit Control System).....	145
Signon.....	145
Space War.....	93
Spectaculator.....	13
Spell 'N Fix.....	256
Star Blaster.....	128
Super Color Writer 1.0.....	258
Telewriter.....	261



# MODEL II SOFTWARE INDEX

Basic Compiler . . . . .	16, 19
Dancing Demon . . . . .	141
dBase II . . . . .	40
EDAS 4.0 . . . . .	66
General Accounting Package . . . . .	8
Litigation Support System . . . . .	9
Microstat . . . . .	10
Mod II Games . . . . .	121
Model II Compiler Basic . . . . .	170
Model II Scripsit . . . . .	246, 250
Palantir Word Processing . . . . .	250
Profile II . . . . .	47
Profile Plus . . . . .	47
Scripsit 2.0 . . . . .	246, 250
Select Word Processing . . . . .	250
Statistical Analysis . . . . .	10
XBasic . . . . .	178
Snapp III . . . . .	230
Snapp IV Auto Map . . . . .	230
Snapp V Extended File Mapping Support . . . . .	232



# SOFTWARE MANUFACTURERS ADDRESS INDEX

**80-N.W. Publishing Co.**  
3838 S. Warner St.  
Tacoma, WA 98409

**Aardvark-80**  
2253 South Commerce  
Walled Lake, MI 48088

**ABS Suppliers**  
P.O. Box 8297  
Ann Arbor, MI 48107

**Ace Computer Products**  
1640 N.W. 3rd St.  
Deerfield Beach, FL 33441

**Acorn Software Products Inc.**  
634 North Carolina Ave. S.E.  
Washington, DC 20003

**Advanced Operating Systems**  
450 St. John Road., Suite 792  
Michigan City, IN 46360

**Adventure International**  
P.O. Box 3435  
Longwood, FL 32750

**AGS Software**  
Orleans, MA 02653

**Alan Seville**  
San Diego, CA

**Allen Gelder Software**  
Box 11721 Main Post Office  
San Francisco, CA 94101

**Alpha Products**  
79-04 Jamaica Ave.  
Woodhaven, NY 11421

**Apparat Inc.**  
4401 South Tamarac Parkway  
Denver, CO 80237

**Archbold Electronics**  
10708 Sequoia Way  
Rancho Cordova, CA 95670

**Armadillo International Software**  
Austin, TX 78712

**Ashton-Tate**  
10150 West Jefferson Blvd.  
Culver City, CA 90230

**Aspen Software**  
P.O. Box 339-M  
Tijeras, NM 87059

**Automated Simulations**  
P.O. Box 4247  
Mountain View, CA 94040

**Avalon Hill**  
Dept. C-10, 4517 Harford Road  
Baltimore, MD 21214

**B.T. Enterprises**  
171 Hawkins Road  
Centereach, NY 11720

**Basics and Beyond Inc.**  
Amawalk, NY 10501

## INDEX

---

**Big Five Software**

P.O. Box 9078-185  
Van Nuys, CA 91409

**Big Systems Software**

27574 Via Rosalie  
Mount Clemens, MI 48043

**Byte Miser Software**

720 West Haven Blvd.  
Rocky Mount, NC 27801

**Chameleon Software**

Box 3435  
Longwood, FL 32750

**Circle Enterprises**

P.O. Box 546  
Groton, CT 06340

**Circle J. Software Ranch**

Carollton, TX

**Cognitec**

704 Nob Ave.  
Del Mar, CA 92014

**Compu-Things**

Suite #2, 270 Broadway  
Revere, MA 02151

**Compuquote**

Canoga Park, CA 91305

**Computer Applications Unlimited**

P.O. Box 214, Dept. ABM  
Rye, NY 10580

**Computer Information Exchange**

San Luis Rey, CA 92068

**Computer Software Design**

Anaheim, CA 92855

**Computerware**

Box 668  
Encinitas, CA 92024

**Cornsoft Group**

6008 North Keystone Ave.  
Indianapolis, IN 46220

**Cornucopia Software**

P.O. Box 5028  
Walnut Creek, CA 94596

**Cosmopolitan Electronics Corp.**

P.O. Box 234  
Plymouth, MI 48170

**Cottage Software**

614 North Harding  
Wichita, KS 67208

**Custom Software Engineering**

807 Minutemen Causeway  
Cocoa Beach, FL 32931

**Dan G. Haney and Associates**

San Mateo, CA 94401

**DAR Sales**

209 Kenroy Ave.  
Roseville, CA 95678

**Datasoft Inc.**

19519 Business Center Drive  
Northridge, CA 91324

**David C. Whitney Associates Inc.**

P.O. Box 451  
Chappaqua, NY 10514

**Designer Software**

3400 Montrose Blvd., Suite 718  
Houston, TX 77006

**Digital Research**

P.O. Box 579  
Pacific Grove, CA 93950

**Dilithium Press**

P.O. Box 92  
Forest Grove, OR 97116

**Disco-Tech Microcomputer Products**

P.O. Box 1659  
Santa Rosa, CA 95402

**Discovery Bay Software**

P.O. Box 464  
Port Townsend, WA 98368

**Discovery Games**

936 West Highway 36  
St. Paul, MN 55113

**Don Coon**

1228 Alpine  
Dewitt, MI 48820

## SOFTWARE MANUFACTURERS

---

**Dynacomp**

Pittsford, NY 14534

**Dynamic Software**

58-04 208 St.  
Bayside, NY 11364

**Ecosoft**

Indianapolis, IN 46206

**Edu-Ware East**

P.O. Box 336  
Maynard, MA 01754

**Electronic Display Technology**

3200 Polaris, Suite 3  
Las Vegas, NV 89102

**Epyx/Automated Simulations**

P.O. Box 4247  
Mountain View, CA 94040

**Ernest H. Fellows**

9230 Bingham  
San Antonio, TX 78230

**ETS Center**

Box 651 35026-A Turtle Trail  
Willoughby, OH 44094

**Exador Inc.**

P.O. Box 3435  
Longwood, FL 32750

**Exatron Corp.**

181 Commercial St.  
Sunnyvale, CA 94086

**Fantastic Software**

P.O. Box 27734  
Las Vegas, NV 89126

**Far West Systems and Software**

P.O. Box 3301  
Eugene, OR 97403

**FMG Corporation**

P.O. Box 16020  
Fort Worth, TX 76133

**Frank Hogg Laboratory Inc.**

130 Midtown Plaza  
Syracuse, NY 13210

**Galactic Software**

11520 North Port Washington  
Mequon, WI 53092

**General Computer Co.**

4873 Langer Lane  
Woodbridge, VA 22193

**Hayden Software Division**

50 Essex St.  
Rochelle Park, NJ 07662

**Hexagon Systems**

P.O. Box 397, Station A  
Vancouver, BC V6C 2N2  
Canada

**Howe Software**

14 Lexington Road  
New City, NY 10956

**IJG**

1953 West 11th St.  
Upland, CA 91786

**Instant Software Inc.**

Elm St.  
Peterborough, NH 03458

**Interpro Corp.**

P.O. Box 4211  
Manchester, NH 03108

**J.C. Sprott**

Madison, WI

**J.F. Consulting**

74355 Buttonwood  
Palm Desert, CA 92260

**Johnson Associates**

Redding, CA

**Kwik Software**

P.O. Box 328  
Bolivar, MO 65613

**Level IV Products**

32429 Schoolcraft Road  
Livonia, MI 48150

**Lindbergh Systems**

41 Fairhill Road  
Holden, MA 01520

**Lobo Drives International**

358 South Fairview Ave.  
Goleta, CA 93117

# INDEX

---

**Lockheed Corp.**

3460 Hillside Ave.  
Palo Alto, CA 94304

**Logical Systems Inc.**

11520 North Port Washington Road  
Mequon, WA 53092

**Macrotronics Inc.**

Turlock, CA

**Manhattan Software**

P.O. Box 1063  
Woodland Hills, CA 91365

**Mark Data Products**

23802 Barquilla  
Mission Viejo, CA 92691

**Martin Consulting**

94 Macalester Bay  
Winnipeg, Manitoba R3T 2X5,  
Canada

**Marvin Plunkett**

Rosburg, OR

**Med Systems Software**

P.O. Box 3558  
Chapel Hill, NC 27514

**Melbourne House Software**

Dept. CS, 347 Reedwood Drive  
Nashville, TN 37217

**Meta Technologies**

26111 Brush Ave.  
Euclid, OH 44132

**Micro Architect**

96 Dothan St.  
Arlington, MA 02174

**Micro Pro Systems**

Route #2, Box 533  
Cumming, GA 30130

**Micro Software Systems**

1815 Smokewood Ave.  
Fullerton, CA 92631

**Micro-Fantastic Programming**

New York, NY 10001

**Micro-Systems Software Inc.**

4301-18 Oak Circle  
Boca Raton, FL 33431

**Microed**

3910 Bandini St.  
San Diego, CA 92103

**Micromint Inc.**

917 Midway  
Woodmere, NY 11598

**Microsoft**

10700 Northup Way  
Bellevue, WA 98004

**Microtrend, U.S.A.**

Louisville, KY 40202

**Miller Microcomputer Services**

61 Lake Shore Road  
Natick, MA 01760

**Misosys**

5904 Edgehill Road  
Alexandria, VA 22303

**Mumford Micro Systems**

P.O. Box 435  
Summerland, CA 93067

**Myatt & Smith**

Tustin, CA 92680

**Nelson Software**

P.O. Box 19096  
Minneapolis, MN 55419

**New Classics Software**

239 Foxhill Road  
Denville, NJ 07834

**Northeast Microwave**

P.O. Box 6153  
Syracuse, NY 13217

**Odesta Publishing**

930 Pitner  
Evanston, IL 60202

**Omicron Software**

P.O. Box 2547  
Sepulveda, CA 91343

**Org-Tex Industries**

Lewisville, TX

**Peggytronics**

381 First St., Suite 5147  
Los Altos, CA 94022



**Pensadyne Computer Services**  
4441 West First St.  
Vancouver, BC V6R 4H9  
Canada

**Personal Software (VisiCorp)**  
2895 Zanker Road  
San Jose, CA 95134

**PFDC Software**  
784 Goucher St.  
Gretna, LA 70053

**Powersoft**  
11500 Stemmons Expressway  
Suite 125  
Dallas, TX 75229

**Practical Applications**  
San Carlos, CA 94070

**Program Design Inc.**  
11 Idar Court  
Greenwich, CT 16830

**Programma International**  
Canoga Park, CA 91315

**Programs Unlimited**  
20A Jericho Turnpike  
Jericho, NY 11753

**ProSoft**  
Dept. C, Box 560  
North Hollywood, CA 91603

**Quality Software Distributors  
(Powersoft)**  
11500 Stemmons Expressway  
Suite 125  
Dallas, TX 75229

**Racet Computes**  
702 Palmdale  
Orange, CA 92665

**Rational Software**  
Pasadena, CA

**RCM Computers**  
221 Hirschfield Drive  
Williamsville, NY 14221

**Remsoft Inc.**  
571 East St.  
Euclid, OH 44119

**Richcraft Engineering**  
Drawer 1065  
Chataugus, NY 14722

**Robert M. Chambers**  
Napeon, Ontario, Canada

**Schneider Enterprises**  
1252 North Brownslake Road  
Burlington, WI 53105

**Select Information Systems**  
680 Beach St., Suite 396  
San Francisco, CA 94109

**Simutek**  
4897 E. Speedway Blvd.  
Tucson, AZ 85712

**Singular Systems**  
810 Stratford  
Sidney, OH 45365

**Small Business Systems Group**  
6 Carlisle Road  
Westford, MA 01886

**Small Computer Co.**  
230 W. 41st St. Suite 1203  
New York, NY 10036

**Small Systems Software**  
P.O. Box 366  
Newbury, CA 91320

**Snappware**  
3719 Mantell  
Cincinnati, OH 45236

**Soft Sector Marketing**  
P.O. Box 340  
Garden City, MI 48135

**Softco**  
Boise, ID

**Software Affair**  
858 Rubis Drive  
Sunnyvale, CA 94087

**Software Associates**  
P.O. Box 2248  
Springfield, VA 22152

**Software Concepts**  
105 Preston Valley Shopping Center  
Dallas, TX 75230

## INDEX

---

**Softwin Associates (Microsoft)**

10700 Northup Way  
Bellevue, WA 98004

**Softworx Inc.**

P.O. Box 9080  
Seattle, WA 98109

**Southeastern Software**

512 Conway Lane  
Birmingham, AL 35210

**Spectral Associates**

145 Harvard Ave.  
Tacoma, WA 98466

**Star-Kits**

P.O. Box 209  
Mt. Kisco, NY 10549

**Storybooks of the Future**

527 41st Ave.  
San Francisco, CA 94121

**Sturdivant & Dunn Inc.**

Box 277, 124 Washington St.  
Conway, NH 03818

**subLogic Communications Corp.**

713 Edgebrook Drive  
Champaign, IL 61820

**Superior Graphic Software**

406 Little Mountain Road  
Waynesville, NC 28786

**Supersoft**

P.O. Box 1628  
Champaign, IL 61820

**Synergistic Solar Inc.**

P.O. Box 580595  
Miami, FL 33156

**Syntonic Software Corp.**

Houston, TX

**Tandy/Radio Shack**

One Tandy Center  
Fort Worth, TX 76102

**The Alternate Source**

704 North Pennsylvania Ave.  
Lansing, MI 48906

**The Bottom Shelf**

P.O. Box 49104  
Atlanta, GA 31359

**The Heath Co.**

Benton Harbor, MI 49022

**The Innovative Penguin**

Harvey, LA 70059

**The Management**

P.O. Box T  
Aledo, TX 76008

**The Micro Works**

P.O. Box 1110  
Del Mar, CA 92014

**The Microperipheral Corp.**

2643 151 Place N.E.  
Redmond, WA 98052

**The Software Exchange**

Milford, NH 03055

**Transformation Technologies**

194 Lockwood Lane  
Bloomington, IL 60108

**TYC Software**

Genesco, NY 14454

**Unique Printing and Stationery Co., Inc.**

11 Maiden Lane  
New York, NY 10038

**Web Associates**

Monrovia, CA 91016

**William Fink**

Suite 24B, 1105 N. Main St.  
Gainesville, FL 32601

**Wilson Software Division**

539 Springhouse Lane  
Camp Hill, PA 17011

**XYZT Computer Dimensions Inc.**

2 Penn Plaza, Suite 1500  
New York, NY 10121

# HARDWARE MANUFACTURERS ADDRESS INDEX

**A.M. Electronics**

3366 Washtenaw Ave.  
Ann Arbor, MI 48104

**Alphanetics Mfg.**

Forestville, CA

**Anadex Inc.**

Chatsworth, CA 91311

**Anderson-Jacobson Inc.**

521 Charcot Ave.  
San Jose, CA 95131

**Archbold Electronics**

10708 Sequoia Way  
Rancho Cordova, CA 95670

**Atomic City Electronics**

3195 Arizona Ave.  
Los Alamos, NM 87544

**Base2 Inc.**

P.O. Box 3458  
Fullerton, CA 92631

**C. Itoh Electronics Inc.**

5301 Beethoven St.  
Los Angeles, CA 90066

**Cecdat Inc.**

P.O. Box 497  
Hayden Lake, ID 83835

**Centronics Data Computer Corp.**

1 Wall St.  
Hudson, NH 03051

**Compulink**

1215 Ravenwood Road  
Boulder, CO 80303

**Computer Printers International**

Mountain View, CA 94043

**Computer Roomers Inc.**

1250 Majesty  
Dallas, TX 75247

**Computex**

17321 El Camino Real  
Houston, TX 77058

**Coosol Inc.**

Anaheim, CA 92803

**Cryptext Corp.**

P.O. Box 425  
Northgate Station  
Seattle, WA 98125

**Data Impact Products Inc.**

745 Atlantic Ave.  
Boston, MA 02111

**Design Solution Inc.**

P.O. Box 1225  
Fayetteville, AR 72701

**Displayed Video**

111 Marshall St.  
Litchfield, MI 49252

## INDEX

---

**E.A.P. Company**

P.O. Box 14  
Keller, TX 76248

**Electronic Systems**

San Jose, CA

**Epson America Inc.**

2384 Hawthorne Blvd.  
Torrance, CA 90505

**Escon Products Inc.**

12919 Alrosta Blvd.  
San Ramon, CA 94583

**Exatron**

181 Commercial St.  
Sunnyvale, CA 94086

**Fuller Software**

630 East Springdale  
Grand Prairie, TX 75051

**G.P. Associates**

P.O. Box 22822  
Sacramento, CA 95822

**Hayes Microcomputer Products Inc.**

5835 Peachtree Corners East  
Norcross, GA 30092

**Holmes Engineering**

3555 South 3200 West  
Salt Lake City, UT 84119

**Icrom Enterprises Ltd.**

1240 Bay St., Suite 205  
Toronto, Ontario, Canada M5R 2A7

**Integral Data Systems**

Milford, NJ 03055

**JPC Products Inc.**

12021 Paisano Court North East  
Albuquerque, NM 87112

**Kogyosha Co./Nik International Trading**

Personal Micro Computers,  
475 Ellis St.  
Mt. View, CA 94043

**Langley-St. Claire**

132 West 24th St.  
New York, NY 10011

**LNW Research**

2620 Walnut  
Tustin, CA 92680

**Lobo Drives International**

354 South Fairview  
Goleta, CA 93117

**Meca**

56677 Sunset Ave.  
Yucca Valley, CA 92284

**Micro Mainframe**

11325 Sunrise Gold Circle  
Building E  
Rancho Cordova, CA 95670

**Micro Technical Products Inc.**

814 West Keating Ave.  
Mesa, AZ 85202

**Micro-Design**

6301 Manchaca Road  
Austin, TX 78652

**Microcompatible Inc.**

P.O. Box 7624  
Atlanta, GA 30357

**Micromatic Corp.**

Indianapolis, IN

**Micromint Inc.**

917 Midway  
Woodmere, NY 11598

**Microtek Inc.**

9514 Chesapeake  
San Diego, CA 92123

**Mumford Micro Systems**

Box 435-C  
Summerland, CA 93067

**Newtech Computer Systems Inc.**

Brooklyn, NY 11201

**Novell Data Systems**

1170 North Industrial Park Drive  
Orem, UT 84057

**Okidata Corp.**

111 Gaither Drive  
Mt. Laurel, NJ 08054

## HARDWARE MANUFACTURERS

---

**Optronics Technology**  
2990 Altantic Ave.  
Penfield, NY 14526

**Parasitic Engineering**  
Berkeley, CA

**Percom Data Co.**  
11220 Pagemill Road  
Dallas, TX 75243

**Personal Micro Computers Inc.**  
475 Ellis St.  
Mountain View, CA 94943

**Programma International**  
Los Angeles, CA

**Quality Computer Services**  
178 Main St.  
Metuchen, NJ 08840

**Rochester Data Inc.**  
Rochester, NY

**Simutek**  
4877 E. Speedway Blvd.  
Tucson, AZ 85712

**Smith-Corona**  
65 Locust Ave.  
New Canaan, CT 06840

**Snappware Inc.**  
3719 Mantell Ave.  
Cincinnati, OH 45236

**Software Affair**  
858 Rubis Drive  
Sunnyvale, CA 94087

**Speedway Electronics**  
11560 Timberlake Lane  
Noblesville, IN 46060

**Starbuck Data Company**  
P.O. Box 24  
Newton Lower Falls, MA 02162

**Strobe Inc.**  
897 Independence Ave. Bldg. 5A  
Mountain View, CA 94043

**Sun-Research Inc.**  
P.O. Box 210  
New Durham, NH 03855

**Tandy/Radio Shack**  
One Tandy Center  
Fort Worth, TX 76102

**The Micro Works**  
Del Mar, CA 92014

**The Microperipheral Corp.**  
2643 151 Place North East  
Redmond, WA 98052

**Thomas Engineering/TCS  
Systems**  
22742 Cortes Drive  
Novi, MI 48050

**Trendcom Inc.**  
480 Oakmead Parkway  
Sunnyvale, CA 94086

**Voicetek**  
P.O. Box 388  
Goleta, CA 93116

If you own a TRS-80, you need this reference guide. Included are over 500 concise reviews of hardware, software, and books. Each review is star-rated, and is updated to include current information on price and availability. The products covered include:

- Utilities
- Games
- Business Applications
- Operating Systems
- Word Processors
- Languages
- Data Bases
- Editor/Assemblers
- Monitors
- Educational Applications
- Graphics Packages
- Communication Software
- Printers
- Plotters
- Modems
- Disk Drives
- Music and Voice Synthesizers
- and more.

The reviews are taken from the pages of *80 Micro* magazine, the leading source of information in the TRS-80 world. Authors include some of the most knowledgeable people in microcomputing, including **William Barden, Dan Robinson, Dennis Kitsz, Terry Kepner and Jake Commander**. All reviews are cross-referenced and arranged by category in alphabetical order for easy access and readability.

You will find this reference indispensable.

ISBN 0-913531-00-6

