80-U.S.

THE TRS-80 USERS JOURNAL.

Vol. VI., No. 1 $3 per copy January, 1983

Getting Started in Computing: Case Histories

0 60959 01
0 318585

STRIKE GOLD

with the most sophisticated TRS-80 operating system available!

LDOS can't really turn your TRS-80 gold, but you'll think you've made the DOS strike of the decade when you turn your micro on to LDOS. You'll find a bonanza of features like full keyboard type-ahead; a true background spooler; file backup by date, class, and between different drive types; hard disk support; data transportability between Model I and III; and a complete communications utility including disk file send and receive. Support for Radio Shack's Doubler and selected others is also provided. With our Job Control Language, you get true "hands off" running of your application programs - give a single command and then walk away. The 400 page manual includes examples of all commands and utilities. The Operator's Guide gives you step by step instructions on how to use LDOS with your applications. Stop running with only "half" a computer! Let LDOS provide the missing features to speed up and simplify your TRS-80 computer system! Visit a dealer or contact LSI for more information on the most popular sophisticated operating system for your TRS-80.

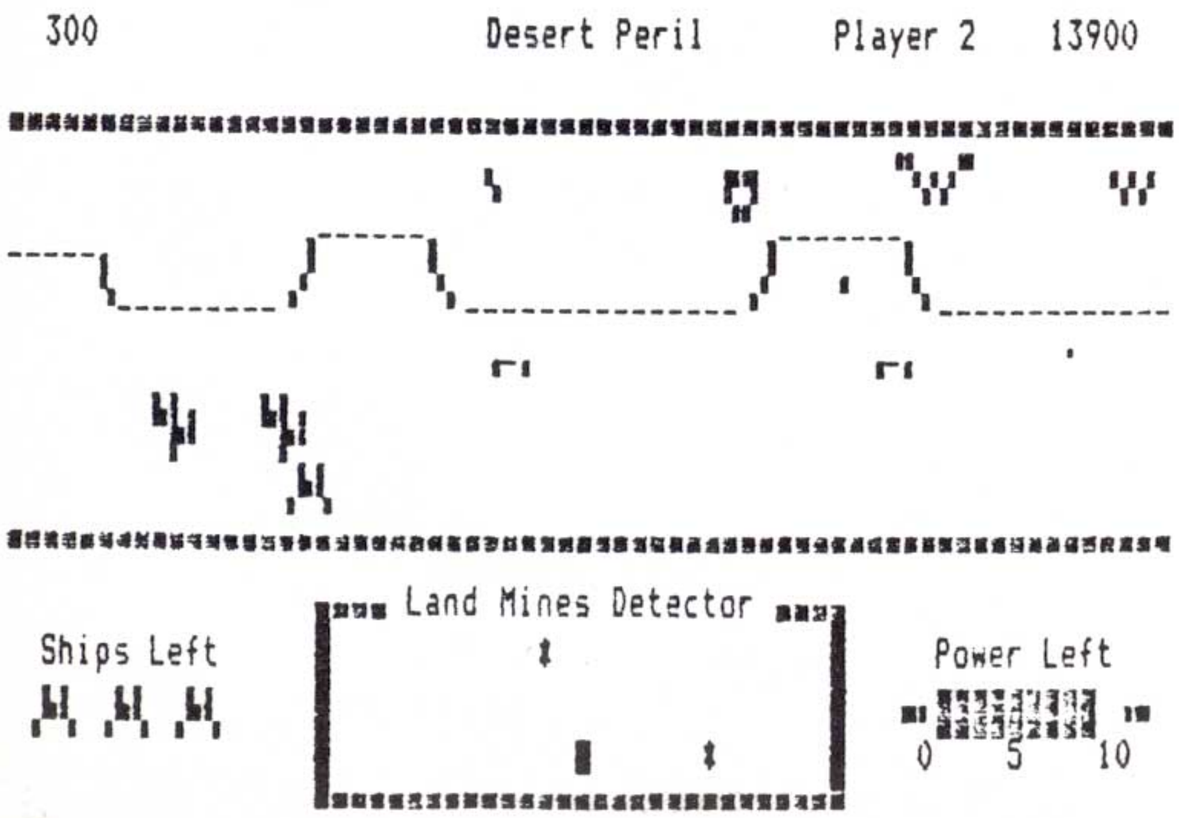
LDOS is available worldwide through thousands of dealers for just $129.

TRS-80 is a trademark of Tandy Corporation. LDOS is available for the TRS-80 Model-I and Model-III. Prices and specifications subject to change without notice. LDOS is a product of Logical Systems, Inc.

Logical Systems, Inc., 11520 N. Port Washington Rd., Mequon, Wis. 53092 (414) 241-3066

LOGICAL SYSTEMS INC.

Desert Peril
(By John Olsen, Jr.)



2

 Please address correspondence to: **80-U.S. Journal, 3838 South Warner Street, Tacoma, Washington 98409, telephone (206) 475-2219.**

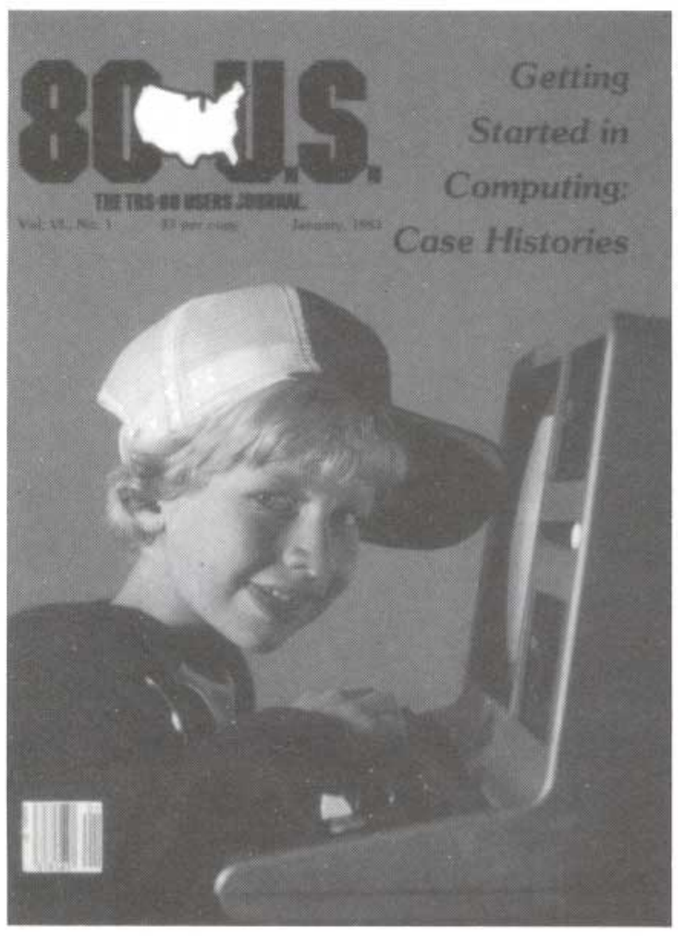
Advertisers: 80-U.S. Journal will accept relevant commercial advertising which pertains to, or is for use on, the Tandy Corporation microcomputers. Write for a current rate schedule.

Authors: We constantly seek material from contributors. Send your material (double spaced, upper/lower case, please) and allow approximately 4 to 6 weeks for review. Programs must be supplied in machine readable form on diskette or tape, clearly marked as to model and operating system. Text files may be on diskette. Media will be returned if return postage is provided. Cartoons and photographs are welcome. Generous compensation will be made for non-trivial works which are accepted for publication. 80-U.S. Journal pays upon acceptance rather than on publication.

SUBSCRIPTION PRICE: U.S.: $16 for one year, $31 for two years and $45 for three years. **Canada and Mexico:** $25 per year, no two or three year subscriptions are offered. **All other:** $30 per year via surface mail, $72 per year via airmail. Two and three year subscriptions are not offered.

ISSN Publication #0199-1035. The 80-U.S. Journal is published monthly by 80-Northwest Publishing, Inc., 3838 S. Warner St., Tacoma WA 98409-4698. Printed in the United States of America.

POSTMASTER: Please send change of address form 3579 to 80-U.S. Journal, 5615 West Cermak Road, Cicero, Illinois 60650. **Second Class postage PAID at Tacoma, WA and additional entry points.**



Our cover's enthusiastic model is Rhett Branin of Tacoma, WA and the photographer was Fred Johnsen, also from Tacoma, WA.

80-U.S.

THE TRS·80 USERS JOURNAL

TRS-80 is a trademark of the Tandy Corp.

Vol. VI, No. 1 — January, 1983

REVIEWS

DEPARTMENTS

80-U.S.

Publisher/Editor-in-Chief
I. Mike Schmidt

Editor
Cameron C. Brown

Associate Editors
Terry R. Dettmann
Spencer Hall
Jim Klaproth

Contributing Editor
Robert W. Liddil

Advertising Sales
Donald Scarberry

Promotion/Circulation
Robert P. Perez

Production
Catherine D. Doud
Sherry A. Miksell

Editorial Secretary
Eva R. Jones

Accounting
Helen Dalton

RENEWING?

Check your label to be sure it's correct. For uninterrupted service include your label with your order.

MOVING?

Please enclose your label or write your name and address as it appears on your label.

Name _____

Address _____

City/State _____

Write in new address:

Name _____

Address _____

City/State _____

80-U.S. Journal
Subscription Department
5615 West Cermak Road
Cicero, IL 60650

Editorial

Cameron C. Brown

1982 has seen a lot of changes. Software has become more professional and much of it looks as though Madison Avenue was involved in its packaging. Hardware is getting more reliable, lower priced, and we are now able to purchase machines that are pretty to look at. Who would have ever thought that the selection of a computer would be based on the color of its casing?

It appears to me that we will be looking back on 1982 as the year in which we all grew up just a little bit more. As consumers, we have learned what to look for and, more importantly, what to avoid. I hate the phrase "user-friendly", but we are now getting material that truly is kind to the operator. No longer is the primary criteria "Does it work?", but we now want to know "Does it work well enough to meet my needs"? We are being told to select computers based upon the software available for it. Just last year, the key question was "How reliable is the hardware, and how many bugs does it have?"

New Year's is resolution and prediction time. This is an excellent time to look forward and take a guess as to what we can expect for 1983. If Jeanne Dixon can do it, so can *80-U.S.*, though we don't claim to match her record of success.

Tandy Corporation will release new models and discontinue some others. At the same time, they will be attacked for not caring and that the consumer has been taken. Tandy stock will rise another 20% even though trade journals will continue to comment that they are losing ground in the marketplace.

Compatibility will not be any better and we will still be under the mistaken notion that the problem has a solution.

The court system will make contradictory rulings regarding copyright of software and all lawsuits currently in place will be settled out of court since no one really does know what is going on.

A new computer will be released that has a built in modem, hard disk, floppy backup that connects to your TV. It will be marketed as a home information center, just like the home entertainment centers of today.

A portable, hand-held computer that has the capability of a full-blown Model III will come on the market. Model III owners will be able to install one megabyte of RAM in their machines.

Another mainframe manufacturer will enter the micro market and lay an egg. Their machine will not come anywhere close to the expectations that consumers will hold for it. We will hear of yet another new method for data storage that will supposedly cut costs by 100 percent. Of course it will be at least one year late in getting on the market, if at all.

The secret will finally come out that over 90% of the software on the market is actually written by teenagers who do the writing during their algebra classes.

Forth user's groups will be granted tax-exempt status by a California court since they are really a religious cult.

Somebody will finally start a company that does not have the word micro, super, computer, or an acronym in its title.

Owners of an original 4K, Level I, Model I will band together and form a club similar to antique car clubs. They will begin to appear in mass at conventions and shows and be revered in numerous "Remember when?" articles and discussions.

All in all, 1983 promises to be another exciting year. It should be full of growth, improvement and progress. The one prediction I can make with complete confidence is that *80-U.S. Journal* will be there reporting, commenting, and letting you know what's new while still giving you programs that entertain, teach and enlighten.

Happy New Year.

Special: SAVE $15
NEWSCRIPT + Mailing Labels
$139.95

"IF YOUR PRINTER

— NEWSCRIPT COMPATIBLE PRINTERS —

• Anadex 9500 (Parallel/Serial) • Anadex 9501 (Parallel/Serial) • Anadex DP-9000 (Parallel/Serial) • Anadex DP-9001 (Parallel/Serial) • Atari 825 (F
Atari 825 (Serial) • C. Itoh 8300P (Parallel) • C. Itoh 8300P (Serial) • C. Itoh Printmaster F10 (Parallel) • C. Itoh Printmaster F10 (Serial) • C. Itoh
8510 (Parallel) • C. Itoh Prowriter 8510 (Serial) • C. Itoh Prowriter II 8550 (Parallel) • C. Itoh Prowriter II 8550 (Serial) • C. Itoh Starwriter FP 150(
• C. Itoh Starwriter FP 1500 (Serial) • C. Itoh Starwriter F10 (Parallel) • C. Itoh Starwriter F10 (Serial) • C. Itoh II 8510A (Parallel) • C. Itoh II 8510A
Centronics 737 (Parallel) • Centronics 737 (Serial) • Centronics 739 (Parallel) • Centronics 739 (Serial) • Decwriter II • Diablo 1610 • Diablo 1620 • D
• Diablo 1650 • Diablo 630 • Epson MX-70 (Parallel) • Epson MX-70 (Serial) • Epson MX-80 (Parallel) • Epson MX-80 (Serial) • Epson
FT (Parallel) • Epson MX-80 with FT (Serial) • Epson MX-80 with Graftrax 80 (Parallel) • Epson MX-80 with Graftrax 80 (Serial) • Epson MX-80 w
80 and FT (Parallel) • Epson MX-80 with Graftrax Plus and FT (Parallel) • Epson MX-80 with Graftrax Plus and FT (Serial) • Epson MX-80 with Graftrax Plus (Parallel) • Epson MX-80 with Graftrax
• Epson MX-80 with Graftrax (Serial) • Epson MX-100 with Graftrax Plus (Parallel) • Epson MX-100 with Graftrax Plus (Serial) •
MX-100 with Graftrax
IDS Paper Tiger 440 • IDS Paper Tiger 460 • Kogyosha (All typewriters) • NEC 8023A (Parallel) •
• NEC Spinwriter 5515 • NEC Spinwriter 5520 • NEC Spinwriter 5525 • NEC Spinwriter 5530 • NEC
NEC 3510 • NEC 3515 • NEC 3520 • NEC 3525 • NEC 3530 • Olivetti 121 • Quime Sprint 5 • Quime
Line Printer I • Radio Shack Line Printer II • Radio Shack Line Printer IV • Radio Shack Line Printer V • Radio
VII • Smith Corona TP1 (Parallel) • Smith Corona TP1 (Serial) • Teletye 33/35 • Teletype 43 • TI 725 • TI 745 •
printers.

IS ON THIS LIST, NEWSCRIPT™ SHOULD BE ON YOUR COMPUTER."

NEWSCRIPT'S exclusive print processor gives you total printer control.

NEWSCRIPT's exclusive print processor takes over where formatting leaves off with over 200 print processing and support features. NEWSCRIPT takes you beyond text formatting — beyond just printing text on paper. With NEWSCRIPT's Print processor you take full advantage of your printer's hardware and software features, creating an unsurpassed printed image. Text editing (the part you see), is only part of the job, printing is the other part — the part others will see.

NEWSCRIPT controls over 80 popular printers.

You don't even have to know how your printer works — only what it can do. NEWSCRIPT will do the rest with such major features as underlining, right justified proportional printing, top and bottom titles, top/bottom left-right-center page numbering, chaining and embedded printing of any length file with disk spanning, fully formatted merging of form letters with selective editing, boldface, sub/super scripts, character substitution/translation, table of contents, indexing, hanging indents, paragraph numbering, line numbering, double width characters, italics, hard and soft hyphens, in-memory spooling, and many other features.*

NEWSCRIPT'S text editor has sophistication to match its print processing.

Buffered key entry rates to 450 characters per second — you never drop characters. Windowing to 240 characters, block move and copy within and between files, definable auto save, "HELP" and "WHOOPS" commands, repeat and query last command, search and replace within column and line limits as well as globally, and an automatic interface to the ELECTRIC WEBSTER spelling checker (sold separately).

NEWSCRIPT'S 277 page manual contains an introductory tutorial with explanations of the beginner's most common needs, a "How to Section" to help when you're stuck, a fully alphabetized description of the commands with literally hundreds of examples, a topical index with over 1,300 entries, and a handy quick-reference card (naturally!).

Start getting the printed results only NEWSCRIPT can give you for $124.95. Requires TRS-80 Model I or III with 48K and 1 disk (minimum — 2 recommended)

Newscript™

is available at computer stores, selected B. Dalton Book Sellers, and selected independent book dealers. If your dealer is out of stock order direct. Include $3.00 (domestic), $6.00 (Canada) for shipping and handling. Foreign residents add $15.00 plus purchase price, in U.S. funds.

TO ORDER, CALL NOW, TOLL-FREE:
(800) 824-7888, Operator 422
Calif: (800) 852-7777, Oper. 422
Alaska/Hawaii: (800) 824-7919
For technical information call:
(213) 764-3131, or write us.

PROSOFT®

Dept. C, Box 560, No. Hollywood, CA 91603

NEWSCRIPT companion programs (sold separately): MAILING LABELS $29.95, DAISY WHEEL PROPORTIONAL $49.95 (not required for Daisy Wheel II), PENCIL & SCRIPSIT FILE CONVERSION $24.95, ELECTRIC WEBSTER (spelling checker and automatic correction) $149.50, GEAP (TRS-80 graphics — requires Epson MX-80) $49.95, DOTWRITER (Hi-res graphics — requires Epson MX-80/100 with Graftrax) $69.95, GEAP/DOTWRITER combination (requires Epson MX-80/100 with Graftrax) $99.95

Dealers: NEWSCRIPT is distributed by IJG, Inc. (714) 946-5805

****Some features work only if your printer has the mechanical capability.***

Directions

I. Mike Schmidt, Publisher

Our old ball of mud has made it around the sun one more time and it's time to look at the past year and into what the next holds for us. The past year has been an exciting one for us here at *80-U.S.Journal*. We have more than doubled our staff and space and our circulation has more than doubled too. It's wonderful, and we can't go on without thanking all of you who made that possible.

The one change, apparent with this issue, is that we have gone to an all slick format. As I write this, of course, we haven't yet seen it so I hope it turns out to be what we expected. We have every reason to think it will.

IS RADIO SHACK IN TROUBLE?

With Captain Kirk pushing Commodore, Bill Cosby pushing Texas Instruments and a host of others pushing yet another host of others on television one wonders about Radio Shack and their place in the sun. I recently had the opportunity to study in detail their annual report to stockholders. It was interesting, to say the least. If there is any doubt about who will be on top of the microcomputer business in 1983 you should study this report. There will probably be a shake-out of computer manufacturers starting this year, and from what I can see, Tandy will still be there when it's over.

Last year Tandy had total sales of over 2 billion dollars. About 624 million of that came from computer hardware, software and supporting peripherals. That's about 31% of their business from computers. Not too shabby, is it?

They haven't stopped there either. You can bet that in 1983 there will be announcements of even more new hardware and software.

This year will also bring more VisiCalc type software packages. Let's face it, there are many microcomputer users who simply don't want to learn how to program. There will definitely be a place for those types of software packages. 1983 promises to be quite a year and I wouldn't miss it for anything.

I also predict that if *Byte* and *80-Micro* don't quit their "catalog size" war, they will have to engage some form of motor express to deliver their publications in 1983 because of a postal carrier revolt.

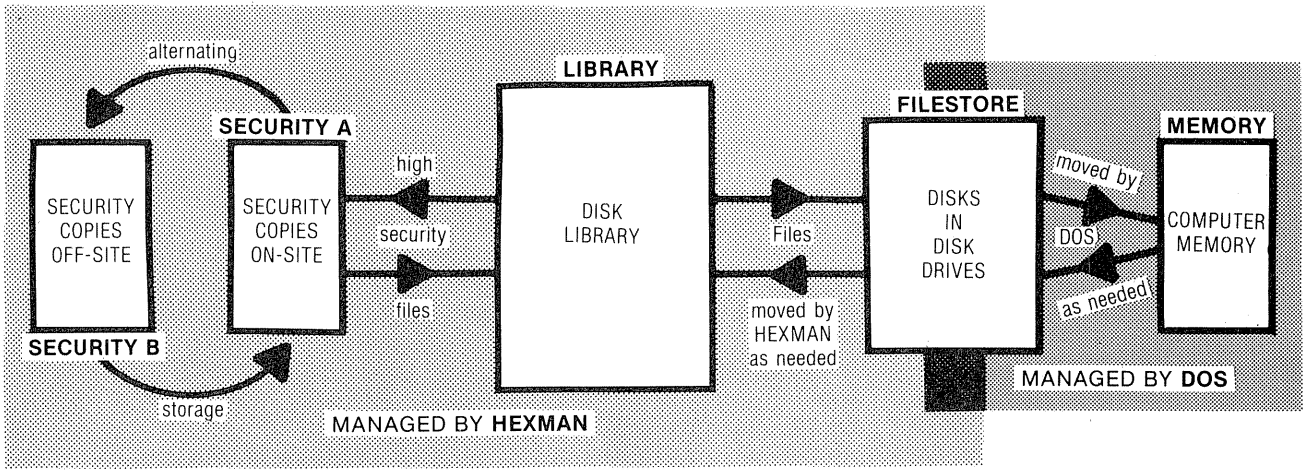
Closer to home we have just installed an RS-232 on an AM Varityper. We can now send text and program listings directly from a Model II to the typesetter. Text can be imbedded with typesetting codes so that the typesetter simply runs out the copy almost without human intervention. In fact, I am writing this on my Model 16 using a text editor. We send text and listings from the Model I, III and Color Computer to the Model II via RS-232, and from the Model II to the typesetter. We can also send Scripsit files for typesetting, and when authors submit articles on Scripsit, we can edit them on the screen and then send them across to be set, in which case the only person who actually keyed in the text was the original author! Isn't technology wonderful?

All in all, it has been an interesting year and we look forward to the next being all of that and more. Make happy holidays, enjoy yourselves, and remember that good years, like good days, are made, not had. Happy holidays!

HEXMAN D.M.S.

DISK MANAGEMENT SYSTEM

Computerized file storage and retrieval for your whole disk library.



Your Disk Operating System makes a fine job of managing the files that are in your disk drives. No need to worry about where your old files are kept or where to store new files, your DOS takes care of it. Now HEXMAN takes over where your DOS leaves off, and extends worry free file storage and retrieval to your whole disk library.

Storage Management

Only a small portion of your disks can be in your disk drives at any one time. These disks form the "Filestore" to which your computer has immediate access, and the remaining disks form you disk "Library". Ideally the "Filestore" should contain the most active files, while inactive files are kept on "Library" disks stored near the computer.

HEXMAN comes as close to this ideal as it can by monitoring file activity and keeping the most frequently used files in the Filestore. If previously inactive files are needed, they are moved in from the Library. If the Filestore becomes full, the least active files are moved out to the Library. The net result is the files you are most likely to use are ready and waiting.

HEXMAN also performs other storage management chores such as daily backups of modified files, on-site and off-site storage of security copies, and file growth monitoring.

File Retrieval

HEXMAN manages up to 8000 files (2000 originals with up to 4 copies of each). If remembering that many eight letter files names gives you a headache — don't worry. HEXMAN gives you two easy ways to find the files you need. With HEXMAN you find files by function e.g. "Get all the files I need for the end of month Sales Reports" or by description e.g. "Find the letter I sent to Radio Shack about my disk drives". HEXMAN finds your files in seconds. Once you have found them you may Review them, Load them or Delete them as needed.

Requirements

Model III — 48K, 2 drives.

Model I — 48K, 2 drives. Double Density adaptor. Lower case modification.

LDOS operating System. If you do not have LDOS, please add $35 for a copy of the smal-LDOS operating system.

$169 US

Introductory offer: $20 off. Valid until Jan.31/83

VISA master charge

90-day trial - just $20.

We are sure HEXMAN will become a "must have" program for all serious TRS-80 users. But check it out for yourself. Purchase HEXMAN now and try it out. If you feel you can live without it (unlikely!), return it for a prompt refund less a $20 evaluation charge.

Offer valid for direct sales only - dealer terms may vary

WARNING — TIMELOCKED SOFTWARE

This product is timelocked. Unregistered copies will expire at a future date. Registered owners will receive an extension patch when needed.

A NOTE ON CANADIAN MAIL — please allow 5 to 10 days in each direction for mail delivery. For faster service leave your orders or enquiries on our 24-hour answering machine.

HEXAGON SYSTEMS

P.O. Box 397, Station A
Vancouver, B.C. Canada V6C 2N2
Telephone (604) 682-7646
Electronic Mail-Micronet 70235,1376

5

Notes, etc.

We recently received a letter from Mr. Fritz L. Schweitzer, Jr. of Darien, CT. He gave an excellent discussion on how to correct a "Direct Statement in File" error that can occur in a Model III.

When a program line is close to the magic 256 bytes in length and is then saved in ASCII format, a "Direct Statement in File" error may result when you attempt to load the program. All you do is list the program and locate the offending line. It would be wise to print it out for reference. Now, go into the DEBUG utility and enter key F. Respond to the filespec question with the name of your program. Page through the program until you find the line that caused the problem; it's not too hard since the ASCII file is easy to read along the right-hand side of the display.

The entire line is shown and you will notice that some of the last bytes of the display are missing from the printed listing. Key in the letter M in DEBUG mode, move the cursor over the display and change all characters that are not on the paper listing to zeros. Now press ENTER and the change is put onto the disk file of the program.

If you now reload the program it will still not run, but it can be loaded. Just change the offending line by splitting it up so it is shorter, and add any characters that you may have had to zero out when you were in DEBUG. Save the "corrected" program and you are done.

Color Computer disk users

Be careful when going from disk to tape and back again. When Color Computer programs are saved to tape and reloaded without the disk pak installed, troubles will occur. All Disk BASIC commands are defaulted to an exclamation point (!) token. The Color Computer ROM cannot interpret the commands and a listing will not show the correct commands. So, be sure that your disk pak is installed whenever you have a program that uses Disk BASIC commands, even if you don't plan on running it. This leads us to the following correction.

Color Computer Text Editor Oct. '82

Believe us, the program works, but not as we listed. Tokens appeared in the place of the Disk BASIC commands and here are what they really are. In line 110, !2,2048 should be 110 FILES 2,2048. In line 150, ! is VERIFY; in lines 430 and 1180, ! is KILL; and in lines 1190 and 1200 the ! stands for RENAME.

Downloader, Nov. '82

We dropped a line in Listing 1, page 32. Be sure to add:

```
50040 DEFUSR2 = &HF01F:CLS:
BD = 85
```

@ News

This issue marks the end of our @ News column, but don't fear, Spencer will be with us in a new and exciting role starting next month. We will still be publishing appropriate articles for ESF users and owners, so stick with us. ESF readers have been with *80-U.S. Journal* since the early days and we don't plan on forgetting those who helped us grow into what we are now.

Puzzler

We might be asking for something that may not be possible. We want to be able to input a whole number X, and have the computer either GOSUB or GOTO line number X. For example, if we input 120 for X, we want to have the BASIC program branch control to line number 120. So, this month's puzzler is to figure out how to GOTO X. Don't confuse this with the ON X GOTO (or GOSUB) where the X can be a variable.

We now have a simple solution to the November puzzler and our winner is Mr. Leonard Zucker of Charleston, SC. His code generates a bridge hand and uses Model III graphics to display the suits. The program does not sort and refers to face cards by number (Ace = 1, King = 13, etc.) but it is very fast and quite short. The winning code is:

```
1 ' *** BRIDGE DEAL ***
10 GOSUB500
20 CLS:PRINTCHR$(23):DEFINT
A-Z
30 NR=52
40 DIM X(NR)
50 FOR I=1 TO NR:X(I)=I:NEXTI
60 A=RND(NR):GOSUB100
70 X(A)=X(NR)
80 NR=NR-1:IF NR<1 THEN 80
90 GOTO60
100 CD=X(A)
110 ST=(CD-1)/13:VL=CD-1-13*
ST
120 PRINTUSING"## !     ";VL+
1,S$(ST),
130 RETURN
500 POKE16420,1:FORI=0TO3:S$
(I)=CHR$(192+I):NEXT
510 RETURN
```

Be sure to send your solution for this month's puzzler to: Puzzler, c/o *80-U.S. Journal*, 3838 South Warner, Tacoma, WA 98409.

Our listings

We are working feverishly to get

our programs into a form that our Comp/Set typesetter can handle automatically. A computer is being hard-wired to send the programs over an RS-232 line. This will give you clear, crisp listings that should be much easier to read and use when typing them into your machine. Don't worry, the listings will still be a direct feed from a working program to the typesetter. They will not be retyped or in any way modified.

The typesetter does justify spacing, so be careful with very long program lines so that you don't exceed the 255-byte per line limit. If everything checks out, you may find some typeset programs in this issue.

In this issue

This month signifies a hallmark for us. We are on enamel paper and have the capability for lots of color. It is like starting anew, and that is exactly what this issue is all about. We can all remember when it was all so new and confusing. In this issue we have a few articles, some humorous, which should spark those memories. For those of you who are just jumping into the micro world, perhaps we can help prevent some problems.

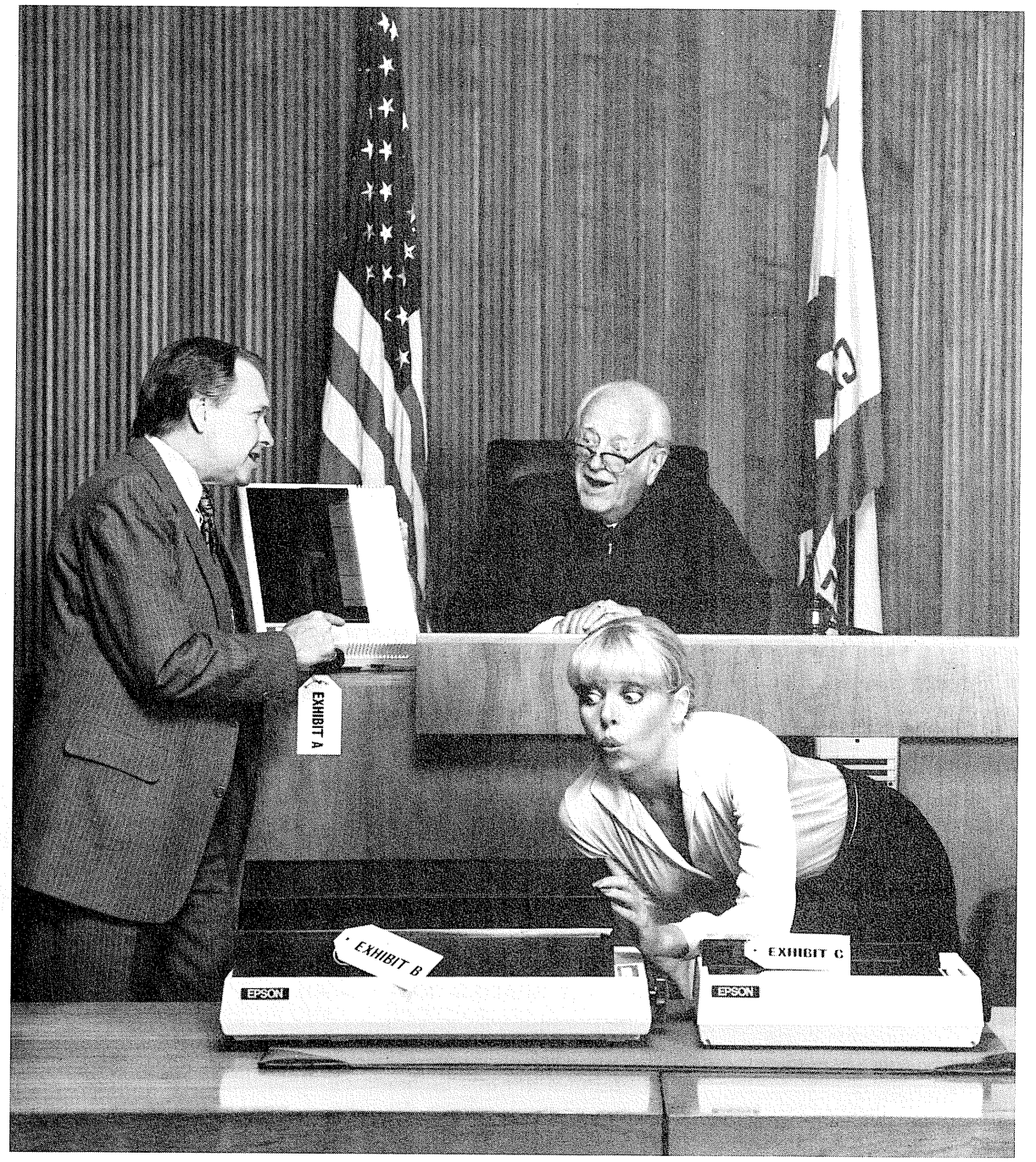
T.G. Melatis begins by raising some interesting questions regarding our "progress" through the ages. Gary Shade introduces us to his family for a light-hearted look at computing, and T.R. Dettmann follows the trials and tribulations of one man as he went about computerizing his business. Dr. Simpson completes our tour of people starting out in computing by giving us a look at what they do in his church.

Tasha Taylor dispells fears about having to know mathematics while at the same time making a nice comparison between structured writing and structured programming. For those with a math interest we have five different ways to compute pi by Dr. Alan Mandell.

There is an excellent graphics program that almost makes animation a reality on your Model III by Gary and Bill Grout, and Jerry Latham gets us around the printer not ready problem on the Model I.

Model II owners rejoice. We asked Terry Dettmann to do an article on Supervisor Calls and he went all out. In this first part of a three-part series, he shows how to scroll protect the screen and gives a program that also works as a function grapher. Parts two and three will take us into flipping video displays and culminate in a disk directory program. Now the Model II owner can also get organized. Waldo Boyd points out a decidedly simple way to obtain powerful editing for your Model II BASIC programs.

No matter what machine you own, don't miss Files and foibles, Basically BASIC, BASIC bits, or our other regular columns and features. They are there to entertain and inform. May you find 1983 to be full of wonderment and fun. -Cam ■



The facts.
And nothing but the facts.

Epson.

Our MX Series printers are the best-selling dot matrix printers in the entire world. This is why.

Fact One: We've got your type.
Epson invented correspondence quality printing. And we still do it better than anybody.

But more importantly, we do *more* of it than anybody. On every Epson MX printer, you get a choice of *sixty-six* different type styles, including italics, a handy subscript and superscript for scientific notation and small print, enough international symbols to print most Western languages, and one-pass underscore. All in a tack-sharp, easy-to-read 9x9 matrix with full descenders.

If print quality is important to you, your next printer is an Epson.

Fact Two: Ultra-high resolution graphics.
There simply is no higher resolution graphics than GRAFTRAX-PLUS. And it comes standard on every MX.

You get bit image graphics free of wander, walk and jitter; graphics with blacker blacks and better definition because you can precisely place a dot at any of 25,920 positions inside a square inch (that's 120 DPI h x 216 DPI v); graphics that make our printers rival plotters.

If graphics is what you want, your next printer is an Epson.

Fact Three: Feature shock.
The Epson MX Series is packed with the features printer fanatics want most. A bidirectional print head with logical seeking of the shortest line. Programmable tab, form length and right margin. Top of forms recognition. Skip over perf. Software printer reset. True backspace. Self test. The world's only disposable print head. And more. Lots more. On our MX-100 and MX-80 F/T, you even get both friction *and* tractor paper feed.

If you want performance, your next printer is an Epson.

Fact Four: Incredible reliability.
Epson MX Series printers are the most reliable printers ever made. And the reason is precision.

Take a look inside one and you'll understand. Notice the simplicity, the fit, the finish. That's why Epson has an out-of-box reliability rate others can only envy: 98%. And why our mean cycles between failures (MCBF) is over five million lines.

The whole package is neatly wrapped in a rugged plastic injection-molded case designed to look good — and stay looking good.

If you want a printer that works, and keeps on working, your next printer is an Epson.

Fact Five: World's best seller. Period.
Within months after we introduced the Epson MX-80, it was the best selling dot matrix printer in the world. It still is. Not because of anything *we* said, but because tens of thousands of computer enthusiasts went out and did some serious comparison shopping. *They* decided it was the best printer for the money they could buy. And who are we to argue?

If you want the world's best seller, your next printer is an Epson.

Fact Six: Quality doesn't cost more.
Don't take our word for it. Check it out. Look at what they give you. Then look at what we give you. Then look at the price.

There is only one logical choice. Epson.

And that's the truth.

"In my judgment, this is the best printer you can buy."

Your next printer.

EPSON

EPSON AMERICA, INC.

COMPUTER PRODUCTS DIVISION

3415 Kashiwa Street
Torrance, California 90505
(213) 539-9140

Letters

Reference *80-U.S.* for November 1982, "Notes" (p. 16).

You are in error when you say one cannot 'PRINT@ 20, "L";' without erasing what's at position 30. What you have in your listing is a special case occasioned by the "END" statement. The END statement will start the cursor at the semicolon, run it to the beginning of the next line and print O.K. But try, '40 GOTO 40' and it will print both characters just fine. Your special case should be watched for, however, in any program, to keep from losing data.

As long as I'm troubling to drop you a line, let me mention that I enjoy your magazine very much. I would like to see more Color Computer items, but I understand your limitations in that regard. One thing I find is that a number of program listings for the other TRS-80s work quite well on the Color with only minor changes. It would be nice if you printed more alternate listings — even if extensive changes are required. Like a lot of people, I suspect, my exposure to computing is through a single machine. It is frustrating to have a program listing which you want to use, but are unable to figure out the necessary conversions, or don't have the time to do so. These, in any case, are minor things, and don't detract from the quality of your magazine. Keep up the good work.

Glenn J. Christensen
Coos Bay, OR

You are right, but the problem lies in the way the Color Computer interprets the semicolon. The cursor is held at that position until another PRINT command is given. Then it will erase to the end of line, no matter what. By printing a " " at some position past the line in question, you can move the cursor to a location in which an erasure to end of line does not affect the output. POKEing graphics into video memory is one method we found to work. You can expect to see more Color Computer programs and we agree that many of our programs can be put on the Color Computer. Space does not allow us to give different listings for every model TRS-80. —Ed.

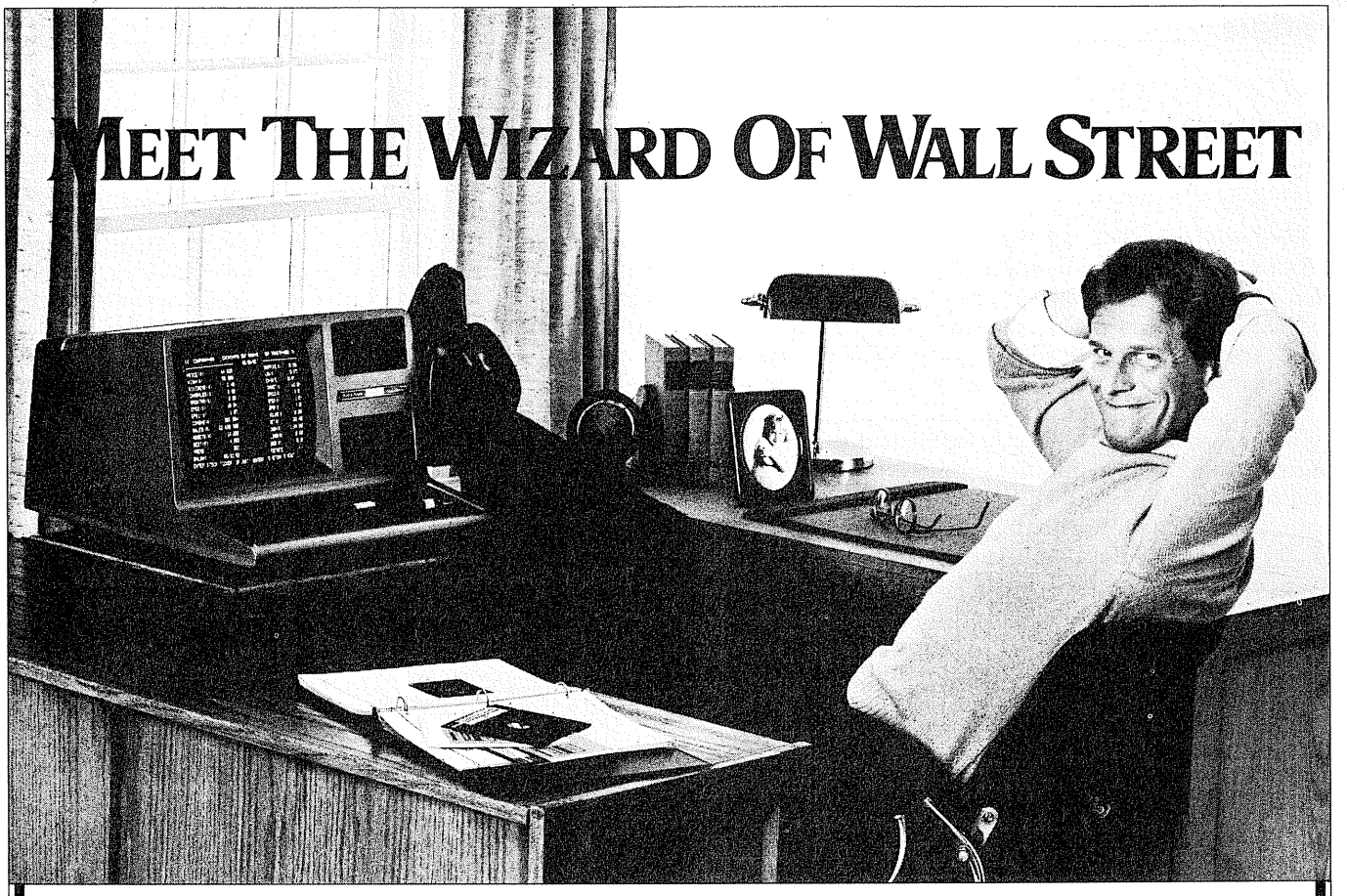
I was amused by Ron Goodman's attempts to recover a program he accidentally "newed," in your October 1982 issue.

If he is using Level II BASIC, he could have saved a lot of effort by entering the following: POKE 17130, 1 then SYSTEM/11395 and finally LIST.

You recover the entire program this way with a lot less effort. For the Model III, use POKE 17386,1 instead of 17130,1.

Don't bother trying this with Disk BASIC, it will work

MEET THE WIZARD OF WALL STREET



Standard & Poor's STOCKPAK system can turn any investor into a whiz

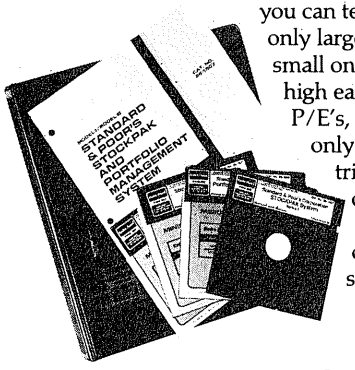
Whether you're an active investor, or just considering the best ways to invest, STOCKPAK can turn you into a Wall Street Wizard.

Sophisticated software that's easy to use

STOCKPAK is a sophisticated software system that offers you an easy and economical way to obtain and evaluate important investment information to help you make better investment decisions. In fact, in virtually no time at all, you can put STOCKPAK to work for you performing a wide range of investment functions.

Screening power is at your command

Our Screen and Select system lets you apply your own investment criteria to a 900-company database. For example, you can tell STOCKPAK to find only large companies, or only small ones...only those with high earnings growth, or low P/E's, or high dividends... only those in certain industries, or on specific exchanges...only those above or below a certain price... or only those with high sales, or low debt, and so on.



In fact, you can draw from over 200 specific information items to compose a single screen. And, you can establish your own targets and cut-off points in each of these 200 areas. STOCKPAK will oblige by searching through the entire database to find only those companies that satisfy your criteria.

More feats of wizardry with STOCKPAK

Standard & Poor's time-tested software also lets you evaluate and manage actual or hypothetical portfolios of up to 100 stocks each. And, you can create your own customized reports—displayed the way you want them.

Whether you use STOCKPAK to help improve your investment decisions, or just for the fun of it, it's good to know the source of this wizardry: Standard & Poor's, one of the world's leading financial publishers with a 120-year reputation of service to the investment community.

Visit your local Radio Shack Computer Store today

STOCKPAK is designed exclusively for use on the TRS-80 Models III and I computers with 32K business systems and two disk drives. The basic software and sample database plus complete documentation is available from your Radio Shack computer store for only $49.95. The STOCKPAK monthly updating service can be ordered directly from Standard & Poor's for only $200 annually. Visit your local Radio Shack today for additional information and a demonstration.

9

STANDARD & POOR'S CORPORATION

25 BROADWAY, NEW YORK, N.Y. 10004

only in Level II.

N.B. Parrish
Marathon, TX

We found you can LLIST but not RUN, EDIT or CSAVE a program recovered under this method. Notice that even with this approach the first program line is listed correctly but the pointers are still flawed. If you are desperate for at least a listing of a NEWed program, this method is acceptable. —Ed.

In the April 1982 issue, you published an excellent utility (Short leader for data tapes) by Kenneth R. Smith. It was just what the doctor ordered for my long data tapes. But unfortunately, it will not work on a Model III. In the July '82 issue, Letters to the Editor, (page 6), James A. Sladek sent in a patch to the routine for later Model I ROMs but unfortunately this also does not work on a Model III.

I am writing this letter in hopes that someone has adapted this utility to the Model III and can help me out.

Finally, I would be remiss if I closed this without congratulating you on your fine magazine. Keep up the good work.

Don Bazzurro
26 Winona Drive
West Springfield, Mass. 01089

Can anyone help? Regarding your last comment, we certainly intend to. —Ed.

I have found your latest issue (October 1982) of *80-U.S.* one of the most enjoyable I have received so far. As one of the earlier advocates of personal computing (TRS-80 Model I — 1978) I thought I had picked up on most of the useful tips/subroutines/utilities, etc., that have appeared in microcomputer magazines. That issue of *80-U.S.*, however, proved me wrong.

Tim Bowman's article on Automatic Program Dating was one example. Having written many programs since 1978 — I thought to myself — how dumb you have been — a simple idea — easy to implement — but so very useful!

Thomas Quindry's article on Restoring Your Data Pointer was another. In one demo package of twelve programs I had incorporated into a single package — I had used the brute force method of setting up dummy reads according to the relative placement of each program within the composite series.

Recovering dead programs and DECIPOKE are several others I haven't had time to thoroughly review yet — but they and others in this issue really made my day.

I had to take time out from my reading to get this note off to let you know that articles such as those in this

issue are well received by people like me — and I'm hoping future issues will continue this idea.

Richard E. Robinson
Neptune Beach, FL

The October issue was a collection of a number of very brief articles. We do plan to do more of it and are pleased you found it to be worthwhile. —Ed.

I have recently purchased a Line Printer VII from our local Radio Shack dealer. I have enjoyed the printer very much, as it does exactly what I bought it to do. I could, however, think of several things I would like it to do but I won't get into that in this letter.

The documentation supplied with the printer leaves an awful amount to the imagination. One of the most interesting features about the Line Printer VII is the dot addressable graphics. The manual briefly brushes over the commands and gives only one poor example of a program.

I believe that many of your readers could benefit from an in-depth article covering the graphic capabilities of the Line Printer VII. I for one would be most appreciative to have such an article in my reference library.

Thank you for your time and cooperation.

Robert E. Wesley
Plattsburgh, NY

Any writers out there who wish to help? —Ed.

Kenneth R. Smith addressed a needed subject in the September 1982 issue on the Epson and Graftrax-80. He deserves our thanks for jumping right in and challenging the dot-by-dot straight on.

Well, some of us non-Epson dot matrix owners are left stranded by this program because of differences in printer codes. Not willing to be left out of the fun I reached for the NEC 8023A-C printer manual for a few clues to the differences.

First I grouped a few lines together where possible, then proceeded to compare the printer codes. Two lines, 110 and 130, were changed to make the default setting right. Next, the graphics dots mode was changed in line 260. The NEC printer returns a '63' (line 280) for 'printer ready' so change this. We are almost there now.

The major difference was in the method of addressing the needle firing positions. The NEC calls the bits from top to bottom which is just opposite from the MX-80.

Top	Bit 0	Needle 1	1
	Bit 1	Needle 2	2
	Bit 2	Needle 3	4
	Bit 3	Needle 4	8
	Bit 4	Needle 5	16
	Bit 5	Needle 6	32
	Bit 6	Needle 7	64
Bottom	Bit 7	Needle 8	128

The top needle is fired when the least significant byte is sent to the printer. The bottom needle is fired when the MSB (most significant byte) is set. This method of dot handling is reversed in line 5010.

Now for the final checkout. Type 'RUN.' Slowly but surely the image begins to develop on the printer paper. Moving forward at the same pace as the Epson, the printer turns out the sine wave, first moving right then transversing to the left.

This author can take credit only for the changes. With these guidelines in hand any owner of a dot addressing bit image printer can probably adapt the program to his printer. Why not try it!

Program Listing for
Low-Resolution Graphics
for the NEC 8023A-C Printer

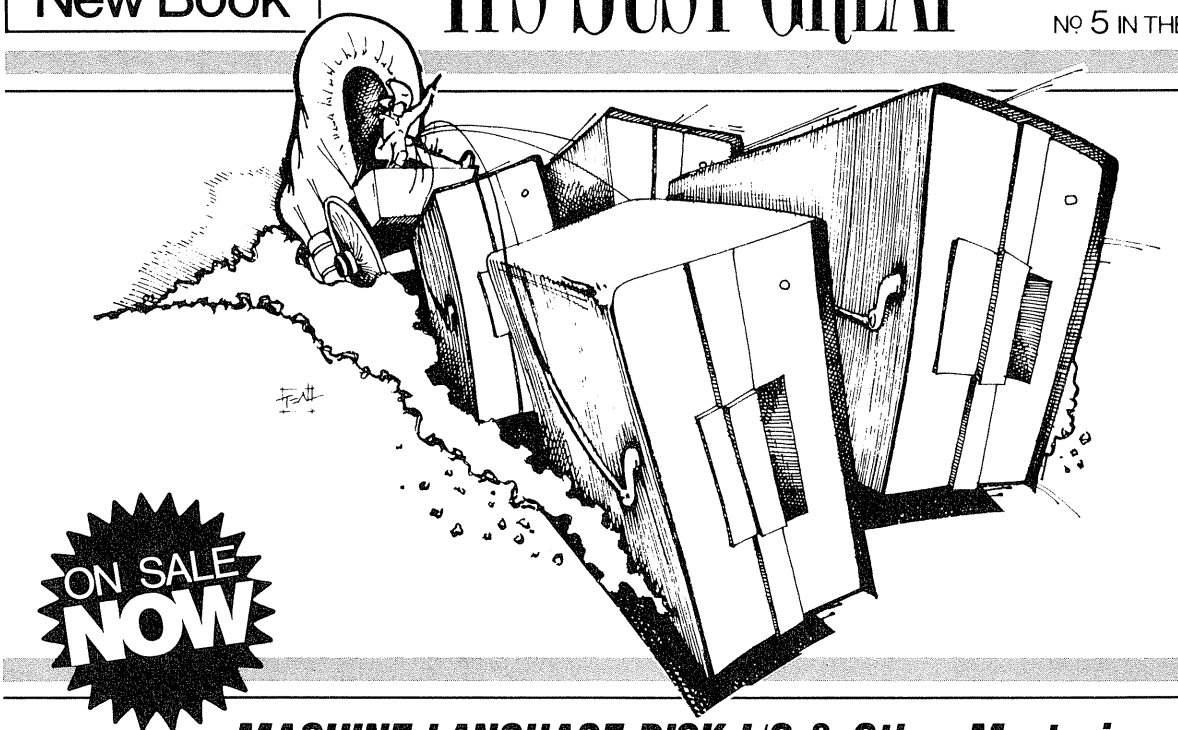
```
100 DEFINTA-W:DIMA(479):' LOW-RES
OLUTION
110 LPRINTCHR$(27);CHR$(78);
130 LPRINTCHR$(27);"B";
140 FORL=0TO712 STEP8:A(240)=170
160 FOR D=0TO7:X=(L+D)/100
180 IFX=0THENGOSUB1000
190 IFX=INT(X)THENGOSUB2000
200 IFX*2=INT(X*2)THENGOSUB3000
210 IF(L+D)/10=INT((L+D)/10)THENG
OSUB4000
220 Y=SIN(X):Y=INT(Y*80+240.5)
240 GOSUB5000:NEXTD
260 LPRINTCHR$(27);CHR$(83);"0480
";
270 FORI=0TO479
280 IFPEEK(14312)<>63THEN280
290 IFPEEK(293)=73THEN OUT 248,A(
I)
300 IFPEEK(293)<>73THEN POKE14312
,A(I)
310 A(I)=0:NEXTI
330 LPRINT:NEXTL:END
1000 FORY=0TO479:GOSUB5000:NEXTY
1030 FORY=0TO479 STEP80
1040 A(Y)=A(Y)OR248
1050 NEXTY:FORY=0TO479 STEP40
1060 A(Y)=A(Y) OR 240
1070 NEXTY:FORY=0TO479 STEP8
1100 A(Y)=A(Y) OR 224:NEXTY:RETUR
N
2000 FORY=235TO245:GOSUB5000:NEXT
Y:RETURN
3000 FORY=240TO244:GOSUB5000:NEXT
Y:RETURN
4000 FORY=240TO242:GOSUB5000:NEXT
Y:RETURN
5000 IFY<0 OR Y>479 THENRETURN
5010 A(Y)=A(Y) OR 2[(D):RETURN
```

D.D. Freeman
Garland, TX

New Book

IT'S JUST GREAT

№ 5 IN THE SERIES



ON SALE NOW

MACHINE LANGUAGE DISK I/O & Other Mysteries.
The most complete book on TRS-80 Model I and III disk I/O available!

Dateline: California, November 1982.

A hushed stillness engulfed the captive hearts of thousands here as the news continued to spread of a fantastic new book from the publishers at IJG.

Young and old alike could be seen gathering together around their microcomputers in anticipation of what some here had called only, "The Power".

Then, out of the West a cloud of dust could be seen on the horizon, coming fast . . .

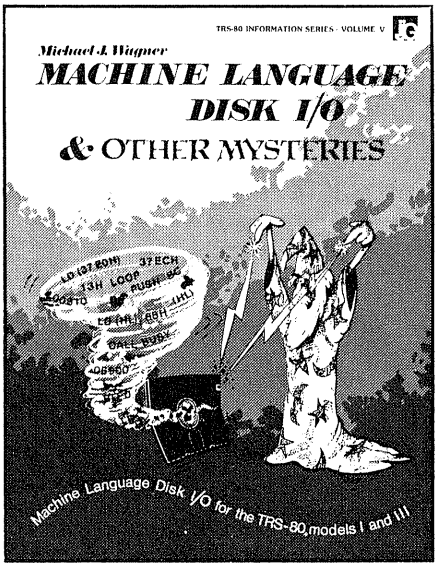
Machine Language Disk I/O & Other Mysteries.

270 pages of powerful information including the source code for a small disk operating system and two complete disk I/O driver routines for the TRS-80 Model I and III, with flowcharts for all the functions.

On TRSDOS File Organization

MACHINE LANGUAGE DISK I/O & Other Mysteries explains what constitutes a disk file, how records are stored on disk, creating a file, reading & writing single bytes and complete records, closing & killing files, error processing, TRSDOS error codes and much more.

Plus a TRSDOS error message display, a disk formatter program, a program to calculate the password for a given file, a full screen file editor, and a complete smart terminal program.



On Machine Language Disk I/O

MACHINE LANGUAGE DISK I/O & Other Mysteries explains what the floppy drive system is all about, the Western Digital 1771 and 1793 Floppy Disk Controllers, read/write access, head step and seek commands, the force-interrupt-of-function command, post-non-maskable-interrupt processing for the TRS-80 Model III, and still much, much more.

Harness The Power!

MACHINE LANGUAGE DISK I/O & Other Mysteries is available for $29.95 at computer stores, B. Dalton Booksellers and independent book dealers around the world. If your dealer is out of stock, order direct from IJG.

Include $4.00 for shipping and handling. Foreign residents add $11.00 plus purchase price. U.S. funds only please.

IJG, Inc. 1953 West 11th Street
Upland, California 91786
Phone: 714/946-5805

Helping you help yourself.

12

From stone to computer: 200,000 years

T. G. Melatis, Agoura, CA

I can't claim to be a recent convert to microcomputing. I was fortunate enough to earn a degree in Electrical Engineering. Upon graduation, I landed my first job with the U.S. Naval Oceanographic Office, as a civilian. They taught me how to install and maintain PDP-8 and PDP-9 minicomputers as well as a full range of electronic instruments and navigation equipment. Keeping those beasties alive on a pitching, rolling ship, 2000 miles from the nearest telephone, makes one quite resourceful.

I graduated to the PDP-11 at my current job. From desktop models to rooms full of large minicomputer systems, I've had to keep up on the latest hardware and software systems. My hobby, computing, is basically a busman's holiday.

My other hobby is to work as a volunteer at the Calico Early Man Site near Yermo, California. As a volunteer, I dig for stone tools that were made by early man. How early? Drs. J. L. Bischoff and T. L. Ku have dated chemicals that were deposited from the soils onto some of the tools using a Uranium-Thorium dating technique at 200,000 years, plus or minus 20,000 years, before present. The tools are considered primitive. They consist of choppers, scrapers, chisel-like tools called burins, gravers, borers, small blades, and a host of others that have very technical differences. The site is too old to find anything that looks like spear or arrow points. What is important is the age. It is at least 150000 years older than most contemporary archaeologists would like to believe that man was on the North American Continent.

So, what is the point? I wish I could say that my TRS-80 was instrumental in the efforts at the site, but it wasn't. The point is that both the stone tools and the computer have a lot in common, physically, as well as a measure of our heritage and our future. They both serve as examples of what I believe is a basic and erroneous measure of man's capabilities, as we commonly try to measure ourselves against our past, or to plot our future. What follows are my comparisons of the stone tools and the computer — my own analysis and conclusions about their importance. I hope that they will provide a sobering, if not challenging, outlook at technology.

Comparisons

What do a microcomputer and a stone tool have in common? The most obvious is that they are both tools. While it may take a trained eye to tell the difference between a naturally broken rock and a rock broken by man, the second is still something constructed by man and used to make his life easier or more successful.

The second thing they have in common is the basic material from which they are both made. A lot of primitive cutting or scraping tools are made from flint, chert, obsidian, jasper, or chalcedony (kal-'sed-knee). The bulk of the tools at Calico are made from chalcedony. Early man used these materials because he could knock off flakes that had very sharp edges. In fact, obsidian edges can be sharper than any steel edge. The electron microscope has shown these edges to be sharp down to several molecules in thickness. These minerals are all composed of the same basic material as the microcomputer chip — silicon. The impurities in the mineral silicon, from which the tools are made, are as important to tool making as other impurities are important to the electronic processes in the silicon-based microcomputer chip, even if for different reasons.

Unquestionably, our ancient ancestors had no concept of silicon, much less the Periodic Table of Elements or crystalline structures. However, I find it ironic, if not downright mystical, that two of mankind's major accomplishments at opposite ends of a 200,000-year sample are based on the same material. (The reader should be aware that man and his ancestors have been making stone tools for a lot longer than 200,000 years — more like 3,500,000 years.)

The third similarity, one which all tools exhibit, is that tools are generally developed to improve productivity. The computer has evolved from a desire to calculate answers with greater precision and accuracy, manipulate large amounts of data or do repetitive tasks with relatively little chance of error. It has proved itself as a tool, a device that allows man to do more work, or do it easier. Calico's stone tools were developed for just the same reason — to do more work or do it easier.

Why? A modern man analyzing the stock market with a computer or an ancient man using a stone blade to cut meat away from bones so that he might carry more meat from his hunt are both seeking the same thing — some little gain in productivity that will give him a slightly better chance of survival. The medium of exchange (dollars, or pounds of meat) is irrelevant.

Lest we get too big for our britches, we should be reminded that man is not the only creature that uses tools.

DOSPLUS 4.0, the perfect score.

The DOSPLUS 4.0 hard drive system is here!

A hard drive without the dynamic new DOSPLUS 4.0 is like an eggshell without the egg. The new DOSPLUS 4.0 is the leading edge—the latest in the line of advanced disk operating systems from MICRO-SYSTEMS SOFTWARE, INC. It's the only current operating system written from the ground up for hard disk operation. Not just a driver but a fully developed system. What that means to you is more bang for your buck! Finally an unlimited TRS-80 for small business. For only $1899 you get the DOSPLUS "PLUS."

THE DOSPLUS 4.0 FEATURES

- Single volume addressing/Double sided floppies seen as one drive-one file can expand to limit of the hard drive
- Hard Disk—disk editing utilities
- Incredible I/O speed
- Runs any combination of densities or tracks
- Also operates 8'' drives with special hardware—comes with expanded users guide and complete DOS technical section on I/O calls and DCB organization
- Ability to use hard drive as the "system" drive.

"PLUS" MANY OF THE SENSATIONAL NEW DOSPLUS 3.4 FEATURES

- BASIC array sort—multi key, multi array
- Tape/Disk—Disk/Tape utility (with relocator)
- Input (controlled screen input)
- Random access and ASCII modification on Diskdump
- BASIC checks for active "DO"
- Backup and Format from a "DO" file
- Much improved Backup (More reliable)

$1899 TAKE YOUR MICRO TO THE MAX. ORDER NOW!

COMPLETE WITH 5 MEG SYSTEM.

- I/O package much faster (disk access time reduced)
- Repeat last DOS command with ''/'' [ENTER]
- Short directory (filename and extension) available
- Short directory of Model III TRSDOS disks
- Single file convert from Model III TRSDOS
- COMPLETE device routing supported (DOS and BASIC)
- Ability to save BASIC programs directly to another machines' memory (if equipped with DOSPLUS 3.4)

NOTE: The final versions of 3.4 and 4.0 will have almost identical features and documentation.

THE COMPLETE SYSTEM

- Smooth, silent, swift
- Error-free disk I/O
- *Add on up to 4, 10 meg units for a total of 40 megabytes!
- Plugs on the 50 pin data bus.—no loss of floppy drives
- Completely self-contained—just plug it in and go

*10 meg units available Soon.

NOTE: Specify 40 or 80 track when ordering DOS diskette. After initial bootup, user can create any DOS desired.

The first in the industry backed by a lifetime warranty**

**Lifetime warranty on original media

DOSPLUS

DOSPLUS first in quality!
First in the industry!

MICRO-SYSTEMS SOFTWARE, INC.

4301-18 Oak Circle
Boca Raton, FL 33431
Telephone: (305) 983-3390
(800) 327-8724

MS INC.

13

The Darwin's Finch has been observed to use a cactus needle to dig grubs. Chimpanzees have been seen uprooting small saplings to use to knock down a piece of fruit that was out of reach. They will pick up a stick to use as a club to defend themselves from a predator. Termites farm fungus. Ants cultivate aphids to milk like miniature dairy herds. Tool making, or use, is no longer a sufficient definition of intelligence. A better test of intelligence has been suggested: "Does the subject use a tool to make another tool?" One of the advantages of being a man is that we are still the only creatures writing the rules. The stone tools and the computer pass both tests.

Some readers might be surprised to see my definition of tools to include farming as an example. What is a tool? I define it to be any thing or physical process that is inserted between a living thing and an objective that it is pursuing in a quest for survival. The termites use the fungus as a source of enzymes to help digest forms of cellulose that their systems could not otherwise digest. The ants don't eat the aphids, only the byproducts of their digestion. Are these any different than early man's use of stone tools to score a bone so that it may be cracked and the marrow eaten (or the bone used to make something) and man's current use of engineered genes to make microbes to eat oil spills or make insulin? More simply, the man that eats an apple is surviving. The man who then plants the seeds is using the apple tree as a tool to get more apples — an increase in productivity.

Erroneous Measure

Man is fond of measuring himself against his past. Unfortunately, he uses a materialistic yardstick with which to make his measurements. He uses his tools and their productivity as a measure. I believe this is one of our greatest errors. We consider early man's tool kit primitive largely because it was not very productive compared to our more modern tools. A computer can manage the use of a lot of tools to increase their productivity. Is this a fair measure? Early man needed only a handful of simple tools to help his small hunter-gatherer band survive. My company needs dozens of computers and thousands of other machines to keep a competitive edge for several hundred people to survive. The tools of modern man are more productive, but they have to be in order to support larger populations. I can't help but wonder if early man's tools were not more productive per person simply because there were so few early men to support.

If we can't measure our tools by their productivity, what measure should we use? If our new measure of intelligence is "the use of a tool to make a tool," how should we view the stone tool and the computer at opposite ends of the 200000-year sample? I would suggest that we first remember that we are not looking at two isolated points on the time scale. The time scale should be viewed as a continuum of little events, from a stone tool, to bronze, to iron, to the array of man-made and man-modified materials that we currently use for making tools.

The same continuum applies to the knowledge we apply to making tools. This idea is not new. What is seldom mentioned in this context is that as man progressed up the time and knowledge scale, his numbers increased. More thinking heads at each point must surely lead to a higher probability that man will create more tools as well as tools with higher productivity. Should modern man expect anything less than a computer or a trip to the moon when he has the benefit of all that history and an increasing number of ideas from which to choose?

The Chevrolet Syndrome

This leads to a form of human behavior that I call the "Chevrolet Syndrome." Basically, this means that man does not need to fully understand all the principles needed to make a tool to be able to effectively use the tool. The more knowledge that one has about the tool, the more productivity one can expect from the tool. Yet, few people really understand all the principles that underly the tools that they use each day. How many men have within their experience all the knowledge that is necessary to build an automobile from raw materials? Yet, with a few hours of instruction, an individual can be licensed and drive coast to coast. Similarly, one can buy a package of tomato seeds, plant them, and harvest tomatoes worth at least ten times the cost of the seeds. Does one need to be a farmer to do this? No, just follow the directions on the seed package. The single greatest contribution of the home or hobby computerist will undoubtedly be to hasten the onset of the "Chevrolet Syndrome" as applied to the computer. Yet, I am amazed by the number of people who are still befuddled by, or fearful of, using anything that has the word "computer" attached to it.

Another aspect of the "Chevrolet Syndrome" is applying a tool to a new job — using it in a new way. Closely related to this is the innovation that is required to form a new tool that is based on an old tool. An example would be combining the concept of a stone blade with the process of working metal to make a blade. The new blade could not be as sharp, but it wasn't as brittle. Thus, it would have a longer life than the stone blade. It would represent an increase in productivity because a single blade might last a lifetime. The time spent making new blades could now be spent making some other tool or process.

Mental Measure

I've spent a lot of words leveling out the current end of our time scale. What about the other end of the 200000-year sample? How does a continuous process of development, of little steps, reflect on innovation? A lot of our progress is a series of little steps that modify or add to a previous tool. There are, however, identifiable points of innovation along the sample. How are they identified? The introduction of a new tool, process, or method that did not previously exist, is an adequate identification for an innovative tool. The examples should come readily to mind.

But what of the quality of mind that created them? Shouldn't the mind that first looked into a stone, saw the yet unformed stone tool, picked up another stone and broke it free, be considered on the same level

as the mind that invented the transistor? What of the mind that first formed clay into a bowl or plate? What about the mind that first formed a bow from a shaft and some fiber? Aren't we short-changing our ancestors by looking at their tools as being primitive just because they were simpler, or less powerful, than our tools? What examples did they have from which to work?

I believe that speculations such as those of Erich von Daniken in *Chariots of the Gods* have done our ancestors and ourselves an even greater injustice. His premise is that ancient man was jolted into creativity by the intercession of an outside force, visits by extraterrestrials. While I would like to believe that beings from other worlds exist and might even visit Earth once in a while, I don't believe his premise or his evidence. In any event, this is not the forum for such as argument. What is important is some of the evidence of ancient man's intelligence that has been found as a result of the controversy caused by von Daniken's publications.

One of the most noted mysteries was that of Stonehenge. Mr. von Daniken suggested that it could not have been erected without extraterrestrial help — that the stones were too massive to be moved by such a primitive people. The stone used for some, if not all, of Stonehenge is unique. The closest source is many miles away. Unfortunately, an English boys' school moved a stone the size of one of those used to build Stonehenge the whole distance in about two weeks. They used nothing more than logs for rollers, ropes for pulling and some flatboats for crossing water.

A boys' school could do it, but why would our ancestors want to expend that much effort? Researchers found that it was a calendar. One of the things that it indicates by alignment of stones and the sun are the two equinoxes — those times of the year when the days are made of equal light and dark periods. These two points in the solar year are the times to safely plant, and later, finish the harvest. Stonehenge may have had all sorts of religious and and ceremonial uses, but it was first

a tool. Armed with this knowledge about our more recent ancestors, researchers are evaluating numerous structures and finding evidence that several have astronomical significance, mostly as calendars.

Another example of our ancestors' intelligence can be found in the tools that he made from stone. While we might assume that heat-treating tools is a modern process developed for working metal tools, there is evidence that our ancestors heat-treated stone tools. (There is no evidence of this at Calico.) The silicon minerals that man used are not regular crystals like diamond (carbon based) or quartz (silicon based). They are amorphous. A regular crystal would be like building a wall with bricks mortared together in a regular pattern. An amorphous crystal is like building a wall by clumping the bricks together with mortar in random patterns. The random order of the bricks and mortar leaves many imperfections in the wall, allowing water to get into the wall, reducing its strength. The amorphous nature of the silicon-based minerals, like flint or chalcedony, also let in water molecules which reduces the crystal's strength and the chance of breaking off flakes with usable shapes or sharp edges. While our ancestors probably discovered heat treating by accident and didn't know why it worked, they did experiment and develop the use of fire to drive off the water before shaping the stones into tools.

Again, the question, "How should we view the quality of the minds of our ancient ancestors?" The more that researchers look at ancestral artifacts and constructions, the more they find that our ancestors were smarter, and smarter sooner, than they had previously thought. And why shouldn't they find them smarter? A single moment of insight, of innovation, is a brief look, not at what is, but at what might be. It is a distinctly human trait that creates something in the mind that doesn't exist in the tangible world — something that begs for a reality. For each individual, it is a unique moment in time that may lead to something that is timeless. Each innovator is isolated from time at the moment of insight. It should make no difference how potent the idea is. The idea may prove materialistically to be unproductive, or impossible to implement at the time. But this is irrelevant at the moment of insight. An insight is a goal, not a thing or a result. Shouldn't the stone innovator be considered at the same insight level as the men who brought us Electric Pencil or VisiCalc?

We like to believe that the scientific method is a fairly recent advance. Is it? To be sure, ritualization and institutionalization are fairly recent in man's history. But, the practice has to be as old as the first man. Adam and Eve may have been evicted from the garden of Eden, but God did not send them out unarmed.

The Calico site is a work site — a place that early man came to make his tools. If he didn't live there, why would we find his tools if he used them elsewhere? First, not all of them were successful. This one does not have the right shape. That one is too dull. The third looks right and the edges show microscopic wear patterns. Why did he leave it behind?

Perhaps we judge our ancient ancestors by their discards, not their successes. The used tool may have been the model for a replacement tool or the new Mark 38 hide scraper. Early man must have had an idea, tried it, evaluated the results, modified the idea and tried again. At Calico, I've seen many examples of what I believe is hypothesis, experiment, analysis, and revised hypothesis — what we call the scientific method. Early man may not have had a name for it, but I believe he practiced it.

Conclusions

I remember the first tool that I unearthed at Calico. I had been digging for about three days, finding nothing but rocks, dirt and a few tiny chalcedony chips. Kneeling down for hours at a time, digging a layer one inch deep with nothing larger than a dentist's probe is not very exciting. Finally, a wedge-shaped stone was free. I thought, "What a curious shape for a stone." It was about an inch and a half long and about a half inch wide. As I handed it to the dig supervisor, I had unconsciously held the blunt edge against my index finger, between my thumb and middle finger. I had been holding a finger blade exactly as its maker had intended. At that instant, the 200,000-year gap from maker to finder closed with a bang, not a whimper.

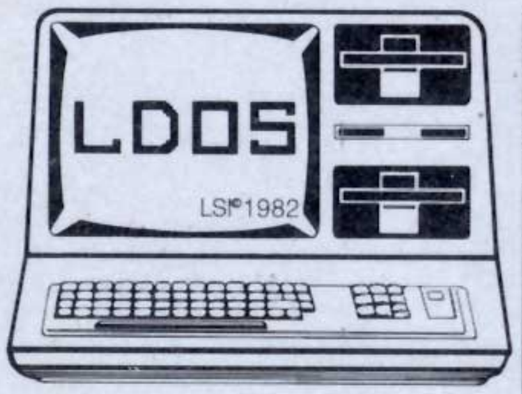
In his book, *Future Shock*, Alvin Toffler invites us to look about our world. He claims that just about everything that we see has been made within our lifetime. He's right. Modern man is the most powerful force in modern nature. His tools are so powerful that many of them could end the world (man is the only creature that has wars) or subject large numbers of his own kind to fear and deprivation by the misapplication of his tools. But I can't fault the tools. To quote Roger Rosenblatt, "When a computer can smile at an enemy, cheat at cards and pray in church all in the same day, then, perhaps, man will know his like." (*Time*, Vol. 119, No. 18, May 3, 1982, "The Mind in the Machine," page 59.)

Whether I'm chipping at the soil to free an ancestral artifact, or debugging a computer program, I can't help but feel a sense of oneness with my ancestors and my contemporaries — a common thread that binds both ends of our 200,000-year sample. In spite of the risks, man's history is one of innovation for productivity. It is a record of co-operation — a rational intelligence for survival. ■

Author note: The Calico Early Man Site is just off Interstate 15 (Mineola Rd. off-ramp), near Yermo, California. The site is managed by Ruth D. Simpson, County Archaeologist and Staff Member of the San Bernardino County Museum. The site curator, Fred Budinger, gives a one-hour tour that is free. It is one of the best tours in California, and will take the public into an active archaeological dig. If you are ever on the road between Los Angeles and Las Vegas, Wednesday through Sunday, from 8:00 a.m. to 5:00 p.m., stop in for the tour. Tours start each hour on the half hour. The last tour is at 3:30 p.m. There is no 12:30 p.m. tour.

LDOS MICRO REVIEW

Volume 1 No. 1 ★ SPECIAL EDITION ★★★ January 1, 1983



The BASIC Answer

You'll think you've made the DOS strike of the decade when you turn your micro on to LDOS. You'll find a bonanza of features like full keyboard type-ahead; a true background spooler; file backup by date, class, and between different drive types; hard disk support; data transportability between Model I and III; and a complete communications utility including disk file send and receive. Support for Radio Shack's Doubler and selected others is also provided.

With our Job Control Language, you get true "hands off" running of your application programs — give a single command and then walk away. The 400 page manual includes examples of all commands and utilities. The Operator's Guide gives you step by step instructions on how to use LDOS with your applications. Stop running with only "half" a computer! Let LDOS provide the missing features to speed up and simplify your TRS-80 computer system! Visit a dealer or contact LSI for more information on the most popular sophisticated operating system for your TRS-80.

LDOS is available worldwide through thousands of dealers for just $129.

The BASIC Answer is a BASIC text processing utility. It is designed to allow the BASIC programmer to build code in a structured manner. "Source" code is written with a word processor or text editor which allows the user to exploit the powerful editing and movement features characteristic to those types of editors. Source code can even be created by your own BASIC interpreter. **The BASIC Answer** is then used to process these files into normal interpretive BASIC code.

Free Yourself from Line Numbers

The BASIC Answer allows substitution of labels for **line numbers!** This means that your BASIC code now can read like a novel. Instead of the typically undescriptive "GOSUB 1000", a label such as "GOSUB @Search.Name" is used. Imagine yourself reading code filled with such descriptive branches and understanding it at a glance, even years later. This feature even allows totally relocatable BASIC routines without the renumbering problems.

TRS-80 is a trademark of Tandy Corporation. LDOS is available for the TRS-80 Model-I and Model-III. Prices and specifications subject to change without notice. LDOS and The BASIC ANSWER are products of Logical Systems, Inc.

```
ORLP!=2TOHA!PRINT@32,"primes found
FHA!/LP!=INT(HA!/LP!)THENGOTO48"CH
EXTLP!:IFVAL(FA$)=LO!THENFA$="* Pr
R!(CO%)=LO! on this scan"USING"##,
R$(CO%)=FA$LEN(FA$)-1)FORLO!=ST!T
O%=CO%:PS%=PS%+1ELSEFA$=LEFT$(FA$,
ORLP!=ØTOlØPRINT@Ø,"factoring "US
RINT@64*LP!+192,PR!(LO%),PR$(LO%):
O%=LO%-INPUT"ORIGIN OF SCAN";IN$Ø
FLO%=-1IFVAL(IN$)<2THENllNG"###,##
EXTLP! ST!=INT(VAL(IN$))##";PS%;R
O%=CO%+INPUT"   END OF SCAN";IN$(
FCO%=llEN!=INT(VAL(IN$))IMPR!(1Ø),
FHA!/LP!=INT(HA!/LP!)THENGOTO48"CH
EXTLP!:IFVAL(FA$)=LO!THENFA$="* Pr
R!(CO%)=LO! on this scan"USING"##,
R$(CO%)=FA$LEN(FA$)-1)FORLO!=ST!T
O%=CO%:PS%=PS%+1ELSEFA$=LEFT$(FA$,
ORLP!=ØTOlØPRINT@Ø,"factoring "US
RINT@64*LP!+192,PR!(LO%),PR$(LO%):
O%=LO%-INPUT"ORIGIN OF SCAN";IN$Ø
FLO%=-1IFVAL(IN$)<2THENllNG"###,##
EXTLP! ST!=INT(VAL(IN$))##";PS%;R
O%=CO%+INPUT"   END OF SCAN";IN$(
ORLP!=2TOHA!PRINT@32,"primes found
FHA!/LP!=INT(HA!/LP!)THENGOTO48"CH
```

A New Concept in Variable Usage

The BASIC Answer allows variable names to be as long as 14 characters and ALL 14 are significant. Imagine reading:

"IF ACCNT.OVERDUE #> 0 THEN GOSUB @PRINT.DUN"

rather than

"IFAO#>0THEN GOSUB52130"

Which would you rather read? It also introduces to BASIC the concept of Global and Local variables. This feature circumvents the tedious problem of variable tracking because a Local variable is only viable in its own subroutine!

NOW AVAILABLE
LDOS 5.1 Quick
Reference Card
$5.95.

End the Multiple Machine Hassle

The BASIC Answer introduces the concept of **"Conditional Translation."** This feature allows the programmer to place different "machine dependent" code simultaneously into the same Source Code. **The BASIC Answer** can be "switched" when processing to ignore the unwanted or include extra code! No more multiple master programs to confuse maintenance. All the masters could now be rolled into the same program. Modify the one master and you've modified them all. Process the same code with different switches set, and get two or more versions from the same source.

The BASIC Answer combines the self-documenting power of COBOL with the relative ease of BASIC together with the power of a word processor.

The BASIC Answer is available for just $69.00.

LOGICAL SYSTEMS INC.

11520 N. Port Washington Rd.
Mequon, WI 53092
(414) 241-3066

15

The computing family

A humorous profile

Gary Shade, Arlington Heights, IL

With great anticipation, you pull up outside of your house. If you've gotten prior approval, your wife and kids are watching from the window. You try to appear casual as you remove the oversized container from your car and bring it into the house. In your head, the theme from the movie 2001 plays over and over again . . . da-da-da-daaa.

Your fingers tremble as you open the container. The "banker" and the rest of the family are anxiously standing by. Then, suddenly, you can see it — the top of your new Model III computer. All the months of saving, dreaming about it and justifying the need for it, are over. It's home. It's yours, but your wife is still asking, "Well, what can it do?"

She asks this question, even though you've cut out hundreds of articles showing what the heck it *can* do. Hasn't she read any of them? How can she ask such a question? So, you ignore her. She begins to tap her foot.

You plug the computer in and ready yourself to turn it on. Then, panic strikes! Where's the on/off switch? The salesman didn't show you how to turn it on, did he? A quick call to the salesman solves the problem, however. As you put down the phone and manage a sheepish grin, you notice your wife is tapping her foot a little louder.

You turn the computer on and the disk drive springs to life. The motor whirs and the light on the front of the drive glows red. Then, almost as suddenly, the drive is quiet — nothingness fills the room. You panic again. Your kids are yawning, your wife is shaking her head, and now the dog has decided to claim the shipping carton as her own.

Quickly, you grab the DOS manual that came with the system. You frantically search for a clue

USER PROFILE

Name: You

Age: Doesn't matter

Background: Who cares

Hobbies: CB turned computerist, or electronic gadget man

Reasons for buying a computer (in order of importance):

1) Familiarity
You use one at work to check on part prices. You also think you know BASIC. You further know that computers will never replace man. After all, you still have a job, right?

2) Automate the household (with a computer, there's nothing you can't do)

a) Automate your wife's recipes

b) Your bank account reconciliation

c) Control the lights and appliances (the salesman sold you on that)

d) A burglar alarm

e) Maybe have it figure out how to cheat better than you did last year on the taxes

3) To teach the kids something

4) To win your wife over and have her start computing

5) Start your own business
You're not sure what, yet, but you are sure that by having a computer, you are on the leading edge of technology.

Name: Your wife

Age: She won't even tell you

Background: Not a career woman, but worked before having children

Reasons for resisting the computer:

1) She thinks it's just an expensive toy

2) She can't see one good reason for owning one

3) She has no intention of putting her recipes on the computer

4) She knows better than to have the computer reconcile the household checking account. She's argued with the bank enough to know what computers like to do once they start working with someone else's money.

near the beginning of the manual. Perspiration begins to bead on your brow as you notice your kids are now asleep, and your wife is gazing at the ceiling. You try to figure out where all the confetti came from, and you remember the dog and the carton.

Ah ha! You found it. "Insert the diskette in drive zero and press reset." Which is drive zero? And, where's reset? The answers must be on another page. Alas, however, you've run out of time, your wife's run out of patience, and the kids have run out of energy. Even the dog's asleep . . . some best friend.

Your enthusiasm has worn off, and you feel discouraged. Then, as if to add insult to injury, your wife asks, "Where do you intend to keep that thing, dear?" You never gave the "computer room" much thought, but there's nothing wrong with where it's at now. "What's wrong with right here?" you ask your wife. "I think it looks just fine next to the toaster. After all, you let me keep the CB on the kitchen table for nearly two years." ■

5) As for controlling lights, her answer has always been, "Get up off your lazy butt and turn the thing off yourself !"

6) She laughs sarcastically when you say "burglar alarm." "A $3000 burglar alarm? Great idea!"

7) As for her getting into computing, she has better things to do than sit in front of a tube all night. Besides, she just might miss Dallas.

Name: The kids

Ages: Both under 15

Background: Like all kids under 15 years old

Reasons for being interested in the computer:

1) Games

2) Games

3) Games

Name: The dog

Age: 50 human years

Background: The backyard

Reasons for being interested in the computer:

1) She likes big cardboard boxes computers are shipped in. Besides, she understands they're the "in thing."

2) Other than that, she agrees with your wife.

Microstrokes for microfolks

Fast graphics and multiple images

Model III, PMC-80, LNW80

Bill Grout, Gary Grout, San Rafael, CA

Ever wished you could plot a graphic image for the computer screen in a few swift strokes rather than fool with a lot of X and Y coordinates?

Microstrokes is for you!

Microstrokes is a BASIC program which you can adapt to your need for graphics. It contains an assembly language routine which will draw a picture in the time it takes to crack a pretzel, and erase it just as fast.

But the real benefit of the routine is that it eliminates the painstaking part of plotting an image. Using microstrokes you choose a point on the screen where you want your image to appear and then use eight basic microstrokes to create it. You don't consider a single X or Y. You merely type a data statement containing a string of microstrokes that tells the computer what to draw. Each piece of data in the data statement instructs the computer to execute one of the eight strokes, and the computer then draws a pixel horizontally, vertically or diagonally as directed. With a little familiarity with the eight strokes, you can draw graphics images freehand without resorting to paper at all. Even the novice who has never created a computer graphic will find the microstrokes program surprisingly easy to master.

BASIC Microstrokes

Microstrokes, a BASIC program, demonstrates an assembly language routine which plots graphics. The advantage of this routine over other methods of graphic display is that only the origin and size of the figure must be specified to plot it. The figure is drawn independently of the X and Y coordinates of the screen after you have chosen the origin (starting point) of the design.

Several advantages are gained this way. First, the initial plotting of the image is simplified greatly. Secondly, the images can be manipulated easily to change their position and size on the screen. Once you have interfaced the assembly language routine to your BASIC program you can call up images by using USR statements. There are no limits as to the complexity of the image drawn other than those of memory size and your computer's normal graphics capabilities.

You will find that the BASIC program will display nine different images on the screen. These images are controlled by the keyboard numbers one through nine. By pressing one of the digit keys the corresponding image will appear and the previous image will be erased. Pressing the arrow keys will move the image on the screen. If you wish to change the rate of movement, you can use the F key to make the image move faster, or the S key to slow it down. Is the image too small? You can increase the scale of the image by pressing the U key. The size doubles in one jump! To decrease the size of the image you press D. Down it goes just like Alice in Wonderland! Holding down any of the keys causes that instruction to be repeated continually.

Plotting an image

But how is the image drawn? If you examine lines 640 to 730 in the BASIC program (we'll get to the assembly language routine later), you'll see the data statements which compose the image. These are made up of a string of numbers, and each number we have called a "microstroke." Each microstroke tells the computer to move and set a pixel in a certain direction. Basically there are eight directions to move in. Here is a list of what the microstroke numbers mean: 1 — up, 2 — down, 4 — left, 5 — up and left, 6 — down and left, 8 — right, 9 up and right, 10 — down and right.

If you want a line that goes straight up, you type a data statement using a number of microstroke 1's separated by commas. Line 640 is an example of a straight line drawn to the left, each "4" setting a pixel one move left. Notice the minus one at the end of the data statement. This tells the computer when the end of the image has been reached. By repeating and alternating the numbers of the different microstrokes, you control what will be drawn on the screen. For example, a data statement like line 730 creates a small box on the screen. (See Figure 1.)

By now you're probably starting to "get the picture." The size and complexity of your drawing is not limited by the length of your data statement. You can use multiple lines to build your image and the computer will keep building until it finds the minus one.

You can also move in any direction and not set a pixel. You do this by adding 128 to the number of the microstroke. A data statement that reads: DATA 1,1,1,129,1,-1 will look like a dotted I. (See Figure 2.)

Here are some sample images you might like to try:

Rocket

```
Data1,1,1,1,1,1,1,9,9,10,10,2,2,2,2,2,2,2,10,10,10,4,4,5,
5,4,4,4,6,6,4,4,9,9,8,2,8,8,1,2,2,2,-1
```

Castle

```
Data1,1,1,8,2,8,8,1,8,2,2,2,2,2,10,10,10,9,9,1,1,1,8,2,8,2,
8,4,4,4,4,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,2,6,6,6,9,9,
8, 8,-1
```

Submarine

```
Data8,8,8,8,8,8,8,9,10,8,8,8,9,10,8,8,8,9,10,8,8,8,9,10,8,
8,8,9,10,8,8,8,9,10,8,8,8,130,132,132,132,132,132,132,132,
132,132,132,132,132,132,1,1,1,1,8,4,2,2,2,2,2,2,2,2,6,8,8,8,
2,2,2,8,8,2,8,8,8,8,8,8,8,2,8,8,2,2,4,2,4,4,2,4,4,4,4,4,4,4,4,4,
4,4,4,4,
Data4,4,4,4,4,4,4,4,4,4,4,4,4,4,5,4,4,4,2,4,1,1,1,1,8,8,2,
8,8,8,8,1,8,8,8,8,8,8,8,1,8,8,8,8,8,8,8,1,1,1,1,8,8,8,-1
```

Shark

```
Data9,9,9,9,9,2,2,2,2,8,8,8,8,8,8,8,8,10,8,8,8,8,8,8,8,10,
8,8,8,8,10,8,8,8,8,8,8,8,8,8,8,8,9,8,1,9,9,9,2,6,6,2,6,2,2,10,10,
4,1,1,4,4,4,4,4,4,4,4,4,4,4,4,6,4,4,4,4,4,4,4,4,4,4,4,4,4,6,4,
4,4,4,4,4,4,4,4
Data5,4,4,4,4,5,10,2,2,4,5,5,5,2,4,4,4,4,4,4,9,9,4,4,4,4,4,
4,5,4,5,4,5,8,8,8,8,8,8,8,8,8,9,8,8,130,2,129,1,8,8,8,8,8,8,8,8,
8,8,8,8,8,8,8,8,8,-1
```

The BASIC Program

Let's look a little closer at the BASIC program.

When the program is run, it loads the assembly language routine and the nine images into memory. It also creates a table which keeps track of the starting address of each image. Line 20 assigns the starting address of the plot routine. The program then branches to lines 420 to 460. These lines poke the routine into high memory starting at &HF000 and continuing until a minus one is read in a data statement. The program then returns to lines 40 to 130. These lines set up the addresses of the parameters needed to plot an image. These parameters are:

IM — The number of the image.
X — The X coordinate of the start of the image.
Y — The Y coordinate of the start of the image.
SC — The scale of the image.
SW — A variable which determines if the image is to be plotted or erased.

Line 140 pokes the starting address of the first image into an image table beginning at location &HF200. Following this, lines 160 through 240 poke each image's data string into memory and add its starting address to the image table. Next, the first image is set for display in lines 250 to 290, and lines 300 to 390 actually plot the image on the screen.

The machine language routine is called in line 300 with the statement A=USR1(0). The subsequent line, 310, strobes the keyboard using the INKEY$ function searching for an instruction. The program allows the automatic repeating of instructions by holding down a key as a result of lines 320 to 330 which clear the

INKEY$ buffer.

To achieve the effect of movement on the screen, an image must be drawn and erased in quick sequence. This toggling of the image takes place in lines 340 to 370. The variable SW controls the setting and resetting of the image. By poking a zero into the SW location the image is set; poking a one in then resets it. At line 350 the program branches briefly to line 410 where the logical NOT function is used to change quickly from zero to one or vice versa. The value of TG in line 410 is then poked into SW effectively switching the image on and off.

From line 380 the program jumps to a subroutine in lines 740 to 870. These lines interpret information input through the keyboard. If the arrow keys are pressed the origin coordinates of the image are altered. If the U or D keys are pressed the scale of the image is changed. Similarly the S and F keys change the rate of movement of the figure. The INKEY$ buffer is then cleared keeping this section of the program continuously awaiting information from the keyboard.

The number of images allowed in this BASIC program is determined by the IN variable in line 150. Here it has been limited to nine, but you may alter the number of images to suit your programming needs.

When you run this BASIC program, the computer pokes all the information for the plot routine and for your images into memory at the initialization of the program. If you have a lot of images or your images are long and complicated, there may be a little delay as the routine pokes them in. But after this is done you can draw any image almost instantly with the statement A=USR1(0).

We hope this method of plotting graphics will be useful to you. Microstrokes is actually the "paint by number" version of graphics plotting, yet the results can be as sophisticated as your imagination will allow. With a little practice, you can even develop a facility for creating images without first using paper. All you have to do is keep in mind the direction of the next point you want to plot, then type in the microstroke. We can see some immediate applications for dressing up BASIC game programs, as well as for programming graphic presentations for business applications. We hope microstrokes will bring the microartist out in you!

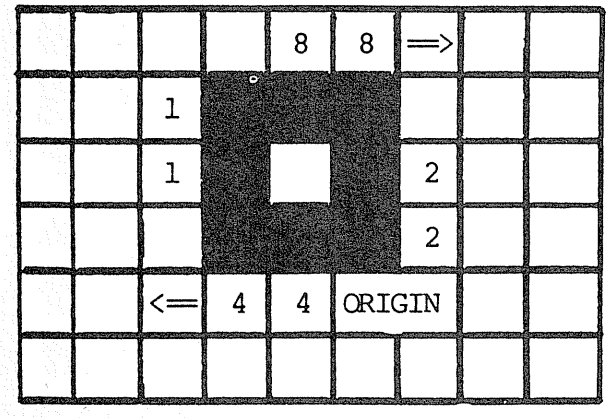


Figure 1
Line 740 Data 4,4,1,1,8,8,2,2,-1

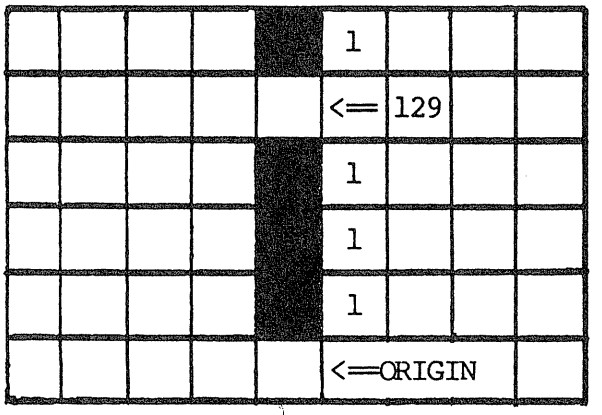


Figure 2
Data 1,1,1,129,-1

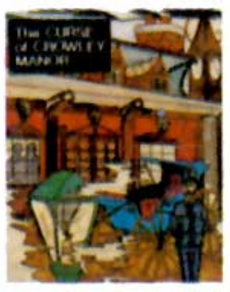
Listing 1

```
1 CLS: PRINT" MICROSTROKES GRAPHICS PROGRAM"
2 PRINT " This program demonstrates an assembly
language routine which simplifies encoding graphic
images. "
5 REM KEY FUNCTIONS U---> UP SCALE D--->DOWN
SCALE ARROW KEYS MOVE IMAGE F--->FASTER
S--->SLOWER
10 CLEAR 100
20 X=&HF000
30 GOSUB 420 : 'POKE USR CALL INTO MEMORY
40 CLS
45 PRINT "The images are now being poked into
memory."
46 PRINT "This is done once at the beginning of the
program."
47 PRINT "When the images have been loaded, you
may call them by pressing any of the digit keys."
50 DEFUSR1=&HF00C
60 'POKE 16526,12 AND POKE 16527,240 FOR LEVEL 11
USR ROUTINE
70 IM=&HF005: '-4091 DECIMAL
80 X=&HF000: '-4096 DECIMAL
90 Y=&HF001: '-4095 DECIMAL
100 SW=&HF002
110 TS=&HF0F8: '-3850 DECIMAL THIS ADDRESS IS THE
START OF WHICH CONTAINS TWO ENTRY ADDRESSES FOR
EACH IMAGE
120 SC=&HF007: '-4089 DECIMAL POKE SCALE SIZE HERE
130 CC=&HF200: '-3584 DECIMAL START OF IMAGE
BYTES
140 POKE TS,00: POKETS+1,242: 'PUT START OF FIRST
IMAGE INTO THE TABLE THE NEXT START OF IMAGE
ADDRESS USED WOULD BE TS+2 AND TS+3 IF TWO
IMAGES WERE TO BE USED
150 IN=9:CO=0:F=1
160 READ BB
170 IF BB=-1 POKE CC,0: CC=CC+1: CO=CO+1:
GOTO 210
180 POKE CC,BB: ' MOVE IMAGE BYTE INTO MEMORY
190 CC=CC+1 : ' INCREMENT MEMORY ADDRESS
200 GOTO 160 : ' REPEAT UNTIL DONE
```

Continued on page 112

Adventures

FROM THE COMPANY THAT STARTED IT ALL!



CURSE OF CROWLEY MANOR

A mysterious adventure into the depths of the occult. This program uses many unique plot-twists and strange devices to guide you either through to victory, or on a certain trip to the gates of hell. Will you be able to solve the CURSE OF CROWLEY MANOR?

TRS-80 16K TAPE $19.95
TRS-80 32K DISK $20.95

Skill Level: Moderate.

STONE OF SISYPHUS

Hideous monsters or lavish treasures beyond all price? You may not have a choice in this rather unfriendly subterranean world where survival will depend largely upon hard intellect and a little luck. Sure to provide hours of frustrating enjoyment, STONE OF SISYPHUS features the kind of action you've come to expect from the incredible MACES & MAGIC Series!

TRS-80 32K DISK (Mod 1) $29.95
TRS-80 48K DISK (Mod 3) $29.95

Skill Level: Moderate.

KID VENTURES

These make-believe adventures are designed for little folk between the ages of 3 and 7, and will allow the child to actually interact with the story. All three KID VENTURES are based upon popular children's stories, poems or songs. All feature graphics and sounds, too!

Kid Venture #1 - Red Riding Hood
TRS-80 16K TAPE (Specify Mod 1 or 3) . $14.95
Kid Venture #2 - Night Before Christmas
TRS-80 16K TAPE (Specify Mod 1 or 3) . $14.95
Kid Venture #3 - Old McDonald's Farm
TRS-80 16K TAPE (Both Mod 1 & 3) ... $14.95

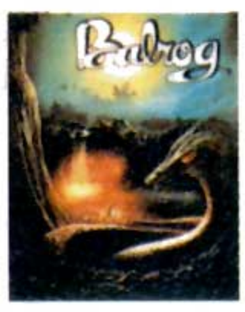
Skill Level: Beginner.

ESCAPE FROM TRAAM

You survived the initial crash on a bizarre, distant world, but you may soon wish that you hadn't! You must escape this dangerous place — but be forewarned that if your decisions are not tempered with intelligence and caution, you may not see home again!

TRS-80 16K TAPE $19.95
TRS-80 32K DISK $20.95

Skill Level: Advanced.

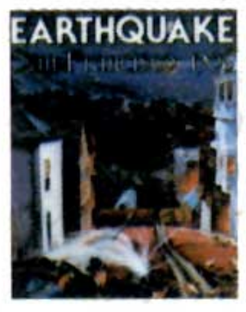


BALROG

This is the first chapter in our popular MACES & MAGIC Series. You equip a character of your own creation and journey into a weird world of dank dungeons and fiery dragons. This fantasy adventure features one of the largest data bases ever created for a microcomputer role-playing game. A real challenge!

Requires two disk drives
TRS-80 32K DISK (Mod 1) $29.95
TRS-80 48K DISK (Mod 3) $29.95

Skill Level: Moderate.

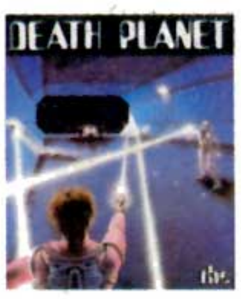


EARTHQUAKE - San Francisco 1906

The title says it all! You're in the middle of one of the most violent earthquakes of modern times — and your survival hinges on wits alone! EARTHQUAKE — SAN FRANCISCO 1906 features one of the most powerful data bases available for a game of this type. Hours of exciting fun guaranteed!

TRS-80 16K TAPE $19.95
TRS-80 32K DISK $20.95

Skill Level: Moderate.

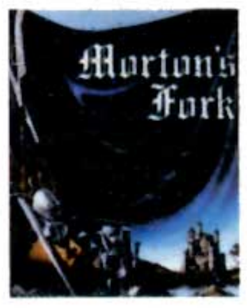


DEATH PLANET: The Dog Star Adventure

The evil General Doom and his Roche soldiers are gearing up for a galactic onslaught — will you be able to rescue the kidnapped princess and save the royal treasury in time? An adventure from Lance Micklus, author of STAR FLITE.

TRS-80 16K TAPE $19.95
TRS-80 **COLOR COMPUTER** 16K TAPE $19.95

Skill Level: Beginner.



MORTON'S FORK

The MACES & MAGIC Series continues! Magical beings and strange creatures populate this mystical realm. Equip your warrior with the necessary accouterments, then plunge into the awesome world of MORTON'S FORK!

TRS-80 32K DISK (Mod 1) $29.95
TRS-80 48K DISK (Mod 3) $29.95

Skill Level: Moderate.

HIDDEN VALLEY

Sure, you're an exalted, courageous, robust warrior — but can even your sturdy constitution save you from the terrors of HIDDEN VALLEY? This dungeon-type program features 99 separate levels of play. This is the second entry in the new DEMON VENTURE Series.

TRS-80 16K TAPE $19.95
TRS-80 32K DISK $24.95

Skill Level: Moderate

REIGN OF THE RED DRAGON

First in the new DEMON VENTURE Series, REIGN OF THE RED DRAGON features dungeon-esque excitement laced with real time graphics. Brace yourself for the incredible!

TRS-80 48K DISK $24.95

Skill Level: Moderate.

SKILL LEVEL KEY

BEGINNER
Excellent introductory adventure for younger players. Minimal problem-solving skills required.

MODERATE
Moderate problem-solving skills required.

ADVANCED
Advanced problem-solving skills required.

TO ORDER

See your local dealer. If he does not have the program you want, then call 1-800-327-7172 (orders only) or write for our free catalog.

Published by

Adventure INTERNATIONAL

a subsidiary of Scott Adams, Inc.
BOX 3435 • LONGWOOD, FL 32750
(305) 830-8194 # 18

Prices subject to change without notice

Recipes for making pi

Five ways to generate 3.14159 . . .

Models I/II/III, PMC-80, LNW80

Alan Mandell, Portsmouth, VA

Pi, and the identification of its complete value, has intrigued mathematicians from the time of its discovery in early Greece to the present day. The school-taught value of 3.1415... represents the beginning of an infinite series of non-repeating digits.

Large computers have spent countless hours crunching numbers to develop longer lists of the decimal digits pi contains. In the 1960s, an ENIAC computed pi to 2035 decimal places. Since then, even longer lists have been produced. Pi is defined in terms of the circle. It is the ratio of the circumference of a circle to its diameter: PI = C/D.

The programs provided in Listings 1-4 are based on different infinite series that were discovered in the history of mathematics research. In theory, all of them will, if continued far enough, produce the value of pi. Because pi is infinite in length, the effort will eventually fail. On the small microcomputer, the effort fails sooner either through lack of memory space or exceeding the capacity of the machine's number-handling capacity. It is interesting, however, to run these programs and watch the successive approximations as the computer examines large sample sizes.

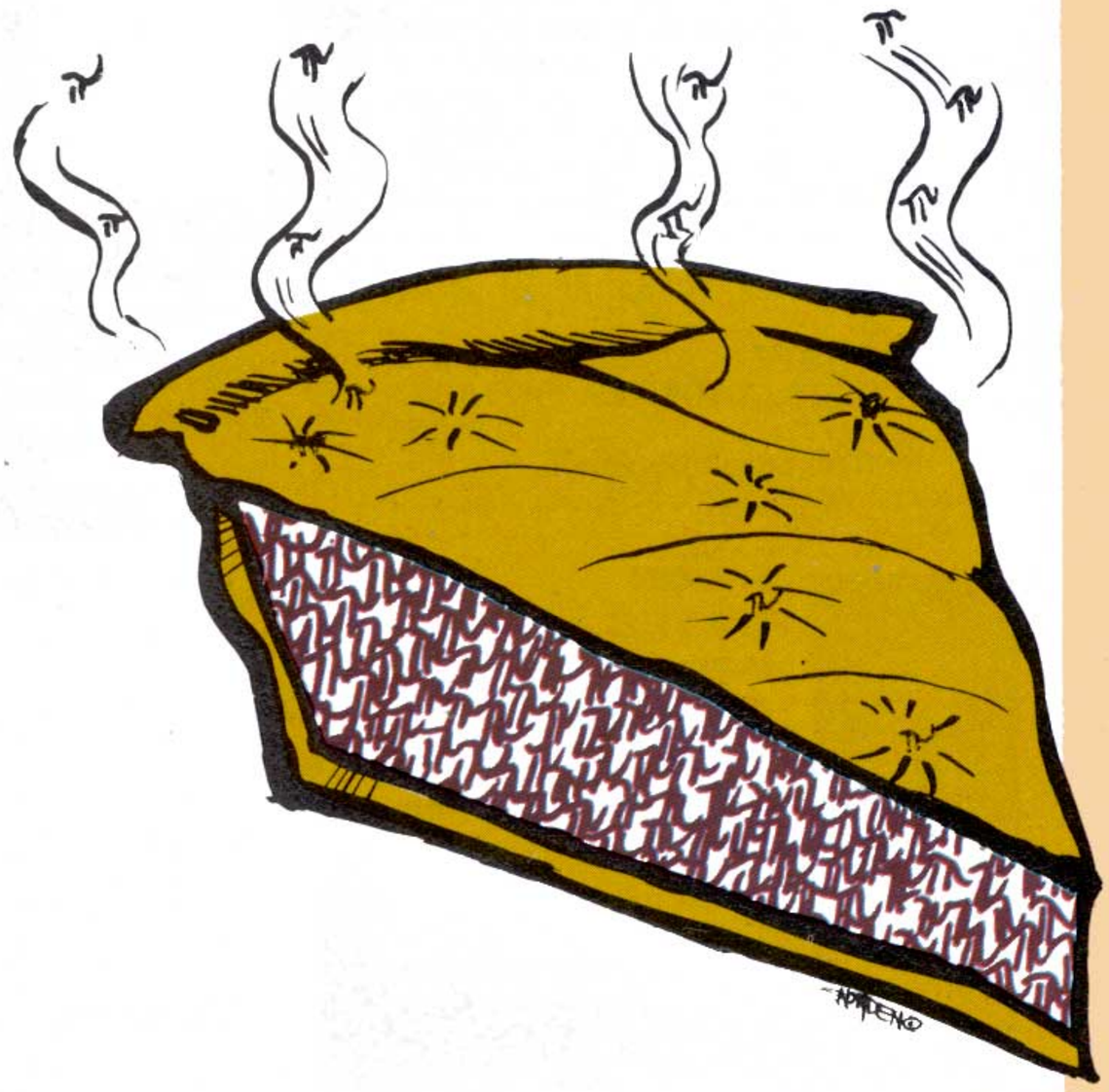
Listing 5 lets the computer replicate Buffon's "needle approximation" of pi. Georges L. L. Buffon (1707-1778), a French biologist, reported the peculiarity that dropping a needle randomly on a ruled plank, and keeping track of the number of times the needle landed on a ruled line, produced an approximation of the value for pi. He later went on to develop a mathematical description (proof) for the oddity. The proof is discussed below. The program permits the computer to randomly drop a needle, display it, and compute the resultant value for pi. The number of trials that you run is limited only by the amount of time you want to spend.

Historically, several famous mathematical operations were identified which, if carried out indefinitely, would produce closer and closer approximations of the value of pi.

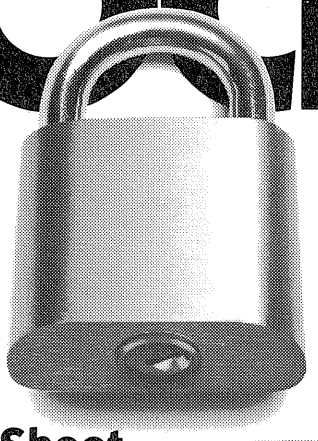
Around 240 B.C., Archimedes used a method involving the perimeters of regular polygons inscribed within, and circumscribed outside, a circle of known diameter to determine successive approximations of pi. Consider, for example, a circle of diameter 1 (no specific dimensional units). The circumference (perimeter) of the circle will be equal to pi times the diameter (C = PI·D). If we inscribe a regular hexagon inside the circle, its perimeter (the distance around the figure) will be equal to the number of sides (6) times the length of a side (all sides are equal in length in a *regular* polygon), of P_i=6s.

Because one side of a regular hexagon inscribed in a circle is equal to the radius (R) of the circle, $P_i = 6R$. If we circumscribe a regular hexagon outside of the circle, its perimeter can be found in the same way. $P_o = 6s$. Using simple trigonometry, side s is determined to be (2/3)R·SQR(3) in length. The perimeter of the circle (its circumference) lies somewhere between the perimeters of the two polygons which enclose it.

If we successively increase the number of sides in the two polygons, their perimeters will get closer and closer to the circle, and we can find more accurate approximations of pi. I wrote a program to test this, but



DON'T GET LOCKED IN



Pascal
Basic
Cobol
Forth
Pilot
Fortran
Spread Sheet
Inventory
Logo
"C"

Data Base Management
Word Processor
Communication Utility
Accounts Receivable
Accounts Payable
Engineering Utility
General Ledger
Mailing List
Macro Assembler
Education

TRSDOS*

CP/M**

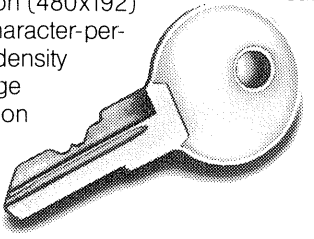
Open your doors to a world of SOFTWARE with LNW computers. You'll get **MORE PERFORMANCE**[1] than with the IBM PC[2] the Apple II[3] TRS80 MODEL II **or** TRS80 MODEL III[4] along with software support of TRSDOS or CP/M, the TWO MOST WIDELY USED OPERATING SYSTEMS. This means you, the user, can select from the largest base of business or personal software.

Standard Features: A serial RS232 communication port, parallel printer port, Hi-Resolution (480x192) B/W and COLOR graphics, an 80 character-per-line screen display along with Quad-density interface for 5" or 8" floppy disk storage offering immediate access to 3.5 million characters, or optional Hard disk interface to 5 or 10 million characters.

Standard Software: LNWBASIC and DOS PLUS operating system packages, commanding all the above features, are included.

The LNW computer will be the key to your success with the starting price at **$1695.00,** along with a full 6 month warranty.

Dealers: You too can open the door to a successful product. Call for our special dealer programs: (714) 544-5745.



LNW Computers
2620 Walnut Avenue
Tustin, California 92680
(714) 544-5744

*TRSDOS is a trademark of Tandy Corp
**CP/M is a trademark of Digital Research Corp
1 Performance is based on bench mark test in the JAN 1982 issue of BYTE magazine, pg 54, with LNW80 II as the comparison.
2. IBM PC is a trademark of IBM CORP
3. APPLE II is a trademark of APPLE COMPUTERS
4. TRS80 is a trademark of Tandy Corp

19

International orders please inquire for pricing/shipping cost.

because the allowable range for integers is +32767, an "overflow" error occurs after the fourteenth iteration and the displayed value of pi is still in the 3.14... level of accuracy.

In the sixteenth century, several series of converging fractions were found and used to compute values of pi. Francisco Vieta developed the following formula and, by hand, found nine decimal places for pi!

$$\frac{2}{PI} = \frac{SQR(2)}{2} + \frac{SQR(2 + SQR(2))}{2} + \frac{SQR(2 + SQR(2 + SQR(2))}{2} + \ldots$$

Around 1610, Ludolph van Ceulen of Germany computed pi to 35 places by the classical method of using polygons up to 2^{62} sides. He spent a large portion of his life on the task.

In 1650, the English mathematician, John Wallis, developed the series:

$$\frac{PI}{2} = \frac{2\cdot2\cdot4\cdot4\cdot6\cdot6\cdot8\cdot\ldots}{1\cdot3\cdot3\cdot5\cdot5\cdot7\cdot7\cdot\ldots}$$ (See Listing 1)

Lord Brouncker, the first president of the Royal Society, converted Wallis's series to the curious fraction:

$$\frac{4}{PI} = 1 + \cfrac{1^2}{2 + \cfrac{3^2}{2 + \cfrac{5^2}{2 + \cfrac{7^2}{\ldots}}}}$$

Gregory developed the series: arctan $x = x - x^3/3 + x^5/5 - x^7/7 + \ldots$ where x is equal to (or greater than) −1, and equal to (or less than) +1. Gottfried Liebniz noted that for x =1 the series becomes:

$$\frac{PI}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \ldots$$ (See Listing 2)

Two other interesting recipes I uncovered are:

$$\frac{PI}{6} = SQR(1/3)\cdot(1 - 1/3\cdot3 + 1/3\cdot3\cdot5 - 1/3\cdot3\cdot3\cdot7 + 1/3\cdot3\cdot3\cdot3\cdot9 - \ldots)$$ (See Listing 3)

and: $(PI - 3)/4 = 1/(2\cdot3\cdot4) - 1/(4\cdot5\cdot6) + 1/(6\cdot7\cdot8) - \ldots$ (See Listing 4)

In 1959, an IBM 704 computed pi to 16,167 decimal places; in 1961, an IBM 7090 computed pi to 100,265 decimal places; and in 1966, a STRETCH computer computed its value to 250,000 places.

There have been several famous mnemonics developed to help remember the values of the digits in extended values of pi. A. C. Orr published a 7-line poem in the 1906 *Literary Digest* which permitted recital of 30 places. A more mundane (and shorter) one is: "May I have a large container of coffee?" My contribution (for TRS owners) is: "TRS, I wish I could calculate in prompt BASIC and BYTES. Decimals decidedly deliver delayment."

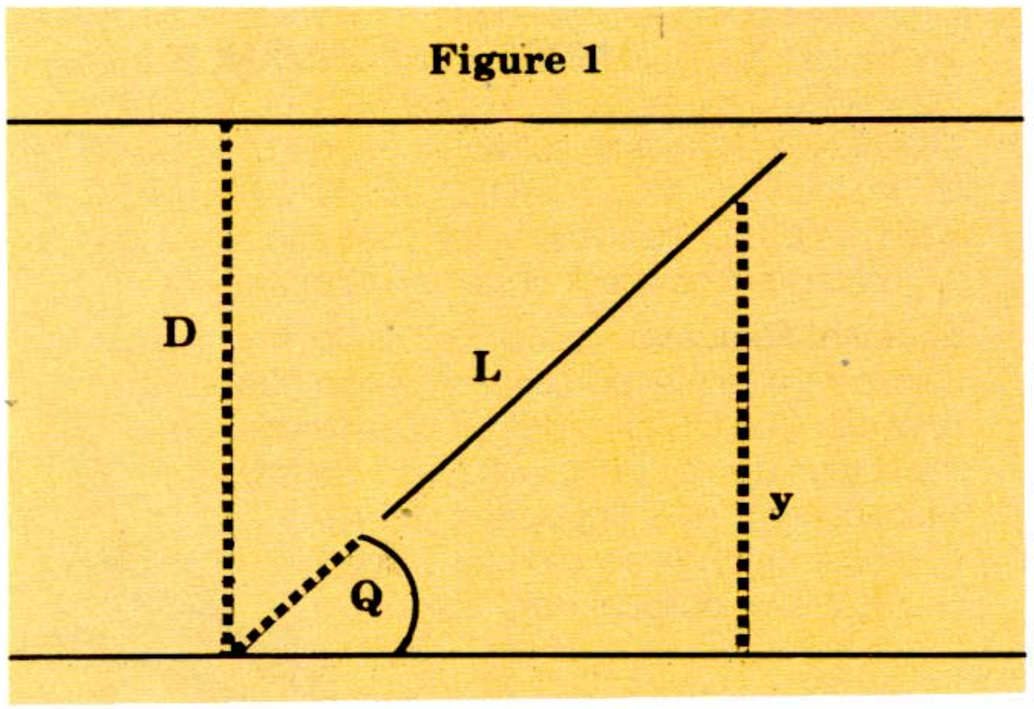
Buffon's Approximation

Buffon reported that he had dropped a needle which was slightly shorter than the distance between the ruled lines on a plank. The needle was dropped by hand. The total number of drops, and the number of times the needle touched a line, were recorded. He found that the total number of trials, divided by the number of touches, was equal to a fraction representing twice the length of the needle, divided by the circumference of a circle whose edges were marked by the ruled lines. Remember that the circumference of a circle equals pi times the diameter ($C = PI \cdot D$, or $PI = C/D$).

If we let T = the total number of trials; H = the total number of hits; L = the length of the needle; and D = the diameter of the circle defined by the lines on the plank, he found that the probability of a hit was $P = 2L/PI \cdot D$. If a given event can happen in H ways and not happen in F ways, in an assumption of each event being equally likely to occur, the mathematical probability (P) of the event happening is $P = H/(H+F)$. Therefore, he found that $H/T = 2L/PI \cdot D$. If the length of the needle is equal to the diameter of the circle (L=D) it can be seen that the probability of a hit is equal to 2/PI, or that $PI = 2T/H$.

The mathematical proof involves an understanding of probability theory and trigonometric functions. However, in simplified form, Figure 1 shows that whether the needle hits a line or not is a function of the distance y and the sine of the angle Q. The sine of an angle is the ratio of the side opposite the angle to the hypotenuse. The distance from the center of the needle to the ruled lines is "y", and the needle itself is the hypotenuse. The needle will touch a line if the distance from its center (y) is less than one-half L (Sin Q). "Y" can range from zero to D/2, and the angle Q from zero to PI. Therefore, the joint probability for both events producing a hit, on an assumption of random drops, is expressed as $P = 2L/PI \cdot D$.

Figure 1



These mathematical ideas are built into the program in lines 25 to 30. X is randomized to use the entire horizontal field; Y is randomized to cover the vertical field between the drawn lines, and A is randomized to

introduce the effect of the sinQ contribution. Because sin0 to 90 degrees equals 1 to 0 (zero), line 30 changes the length of Y according to the value of sinQ. The random value of Y establishes the starting point of the needle, and D establishes the length of the needle. All of the needles are dropped perpendicular to the drawn lines even though various angle values for Q have been introduced into the computation.

A hit, or miss, is identified in lines 40 and 45. If a piece of the needle (a value of Y(I)) coincides with a printed line, POINT (X,Y(I)) will be equal to −1 and a hit will be tallied. If no value of Y(I) coincides with a printed line, POINT (X,Y(I)) will return a 0 (zero) and a miss will be tallied. The needle is then SET and RESET in order to see it drop on the "plank." Lines 65 and 70 compute the value of pi and print both the current value and the cumulative value of pi over several sets of 100 trials. It takes about 40 seconds for 100 trials to be run and results computed.

The mathematical explanation in the 300-series lines may be omitted. However, they do present a synopsis of the mathematical reasoning involved in the program. This program is different from the other listings in this article. The others were trying to derive closer and closer approximations of the value of pi. Buffon's experiment provides a random opportunity for the correct value of pi to occur. I have seen close approximations after 30 drops and wide deviations from the true value after 300 drops. As you see the successive values of pi printed during the 100-trial sets, you will notice that they fluctuate around 3.14159... .

Comments

I wrote and tested these programs on a Model I, Level II TRS-80. I tested Listing 1 on a Model III Level II (at my friendly Radio Shack store), and the value given in Table 1 for 1000 trials was produced on it. I also tested Listing 1 (appropriately modified) on an Apple II Plus. The double-precision number capability made a difference of .000000078600599 when a sample size of 500 was used in both machines. The use of double-precision numbers uses up more memory space, but the accuracy of the results is increased.

Summary

Pi, like some of the other odd numbers of mathematics, is an interesting enigma. These programs let your computer do the number crunching for you, and let you examine the effect of increasing the size of the sample run on the accuracy of the results.

While it is challenging to try to develop new infinite series to provide longer lists of pi's decimals, no more than 10 decimal places are needed for even the most precise practical applications. It has been stated that 10 decimal places would describe the circumference of the planet earth with an accuracy within a fraction of an inch.

Since van Ceulen devoted most of his life to pi, here are the first 20 fruits of his endeavor: PI= 3.14159 26535 89793 23846 ... .

Bibliography

Eves, Howard, *An Introduction to the History of*

Mathematics, 4th Edition, N.Y., Holt, Rinehart and Winston, 1976.

Kasner, E., and J. R. Newman, *Mathematics and the Imagination*, N.Y., Simon and Schuster, 1956.

Newman, J. R. (Ed.), *The World of Mathematics*, Vol. 4, N.Y., Simon and Schuster, 1956.

Listing 1
Program Based on Wallis's Convergent Multiples

```
1 ' LISTING 1 PROGRAM BASED ON WALLIS'S CONVERGENT MULTIPLES
5 ' BASED ON A SERIES OF FRACTIONS OF THE FORM:
PI/2= 2/1 * 2/3 * 4/3 * 4/5 * 6/5 * 6/7 * ...
10 CLS: CLEAR 1000:DEFDBL A,B,J,K,P
15 INPUT "ENTER SIZE OF SAMPLE TO BE RUN";T
20 DIM A(T)
25 J=1: K=2: B=2
30 FOR I=1 TO T: A(I)=(K/J)*(K/(J+2)): K=K+2: J=J+2: NEXT I
35 FOR I=1 TO T: B=B*A(I)
40 PI=B
41 ' IF YOU WANT TO SEE ALL THE VALUES OF PI AS THEY ARE GENERATED USE LINE 44, IF YOU JUST WANT TO SEE THE FINAL VALUE OF PI USE LINE 45
44 PRINT PI: NEXT I
45 NEXT I: PRINT PI
50 END
```

As you can see from Table 1, this series is slow in generating an approximation. Using line 44, it took 27 seconds for 100 fractions and 54 seconds for 200 fractions. Using line 45, it took 105 seconds for 500 trials and I had 4287 bits of memory left.

Table 1
Sample Data from Program 1

T	PI	
50	3.126078900215411	
100	3.133787490528163	
200	3.137677900950937	
500	3.140023818600599	
500	3.14002374	(Apple II Plus)
1000	3.140807746030398	(TRS-80)

Listing 2
An Infinite Series of Converging Fractions

```
1 ' LISTING 2 AN INFINITE SERIES OF CONVERGING FRACTIONS
5 ' BASED ON A SERIES OF FRACTIONS IN THE FORM: PI/4 = 1 - 1/3 + 1/5 - 1/7 + 1/9 - ...
10 CLS: CLEAR 1000: DEFDBL A-C,J,K,P
15 INPUT "ENTER SIZE OF SAMPLE TO BE RUN";T
20 DIM A(T),B(T)
25 J=3: FOR I=1 TO T:A(I)=1/J: J=J+4: NEXT I
```

```
30 K=5: FOR I=1 TO  T: B(I)=1/K: K=K+4: NEXT I
35 C=1: FOR I=1 TO T: C=C-A(I)+B(I): NEXT I
40 PI=C*4
45 PRINT @ 325,T,PI
50 END
```

If you want to see the fractions as they are generated, insert print statements within the loops (25, 30, 35).

Table 2
Sample Data from Program 2

T	PI
20	3.165979272843216
100	3.146576774718296
200	3.142586415298768

Listing 3
An Infinite Series of Converging Fractions

```
1 ' LISTING 3 ANOTHER SERIES OF CONVERGING
FRACTIONS
5 ' BASED ON A SERIES OF FRACTIONS IN THE FORM: PI/6
= SQR(1/3)* (1-1/3*3 + 1/3*3*5 - 1/3*3*3*7 +
1/3*3*3*3*9 - ...
10 CLS: CLEAR 1000: DEFDBL A-D,J,K,P
15 INPUT "ENTER SIZE OF SAMPLE TO BE RUN";T
20 DIM A(T),B(T)
25 K=3: X=1
30 J=3: FOR I=1 TO T: A(I) = 1/(J↑X*K)
35 X=X+2: K=K+4: NEXT I
40 X=2: K=5: FOR I=1 TO T: B(I)=1/(J↑X*K): X=X+2:
K=K+4: NEXT I
45 CC=1: FOR I=1 TO T: CC=CC-A(I)+B(I): NEXT
50 C=SQR(1/3)
55 D=C*CC: PI=D*6
60 PRINT @ 325,T,PI
65 END
```

As Table 3 shows, this program gets to a close approximation after 10 trials and doesn't change after 14 iterations. Beyond 30 iterations, an "overflow" error is produced. The successive fractions must be so small that the computer rounds them off to zero after 13 iterations.

Table 3
Sample Data from Program 3

T	PI
10	3.14159296636464
15	3.14159296635879
20	3.14159296635879

Listing 4
An Infinite Series of Reducing Fractions

```
1 'LISTING 4 AN INFINITE SERIES OF REDUCING
FRACTIONS
```

```
5 ' BASED ON A SERIES OF FRACTIONS IN THE FORM:
(PI-3)/4 = 1/(2*3*4) - 1/(4*5*6) + 1/(6*7*8) -
1/(8*9*10) + ...
10 CLS: CLEAR 1000: DEFDBL A,B,C,J,P
15 INPUT "ENTER THE SIZE OF THE SAMPLE TO BE RUN";T
20 DIM A(T),P(T),B(T),C(T)
25 J=4: FOR I=1 TO T: A(I)=1/(J*(J+1)*(J+2)): J=J+4:
NEXT I
30 B(1)=1/24
35 J=6: FOR I=2 TO T: B(I)=1/(J*(J+1)*(J+2)): J=J+4:
NEXT I
40 FOR I=1 TO T: C(I)=B(I)-A(I): CC=CC+C(I): NEXT I
45 PI=(4*CC)+3: PRINT @325,T,PI
50 END
```

An array of fractions to be subtracted is produced in A(I), and an array of fractions to be added is produced in B(I) after B(1) has been defined. This series may be the most accurate approximation that you produce.

Table 4
Sample Data from Program 4

T	PI
100	3.141592627771901
150	3.141592649389693
184	3.1415992653581116 (accurate to 11 places)

Listing 5
Program for Buffon's Approximation of Pi

```
1 ' LISTING 5. BUFFON'S NEEDLE DROPPING
APPROXIMATION OF PI
5 CLS: CLEAR 1000: G$= STRING$(63,131)
10 DEFDBL C,M,P,R,S,T: DEFINT D,V-Y: GOTO 100: Z=1:
CLS
11 END
15 FOR I=1 TO 7: READ L(I): NEXT I
20 DATA 128, 256, 384, 512, 640, 768, 896
25 RANDOM: CLS: FOR I=1 TO 7: PRINT @L(I),G$: NEXT I:
N=0: T=1: M=0
30 X=RND(127): Y=RND(31)+6: A=RND(0): IF A=>.65
THEN D=5: ELSE D=4
35 FOR I=1 TO D: Y(I)=Y: Y=Y+1: NEXT I
40 FOR I=1 TO D: IF POINT (X,Y(I))=-1 THEN T=T+1:
M=M: GOTO 47 ELSE NEXT I
45 M=M+1
47 N=N+1: PRINT @ 2, "TRIAL # ";N;: PRINT @ 17, "SET #
";W+1
50 FOR I=1 TO D: SET (X,Y(I)): NEXT I
52 REM YOU CAN SET A TIME DELAY HERE IF YOU WISH
BUT TRY THE PROGRAM WITHOUT ONE FIRST
54 PI=2*(T+M)/T: PRINT @30, "PI = "; PI
55 FOR I=1 TO D: RESET (X,Y(I)): NEXT I
60 IFN=100 GOTO 65 ELSE 30
65 Z=Z+1: PI=2*(T+M)/T: R=R+PI: CR=R/Z
70 PRINT @ 960, "T = "; T,"M = "; M,"PI = "; PI, "CR = ";
CR
75 FOR D=1 TO 1000: NEXT D
80 INPUT "ENTER 1 TO RUN ANOTHER 100 TRIALS
 ENTER 2 TO SEE A MATHEMATICAL EXPLANATION
 ENTER 3 TO END THE PROGRAM";V
85 IF V=1 W=W+1: GOTO 25
90 IF V=2 GOTO 300
95 IF V=3 CLS: END
100 PRINT "THIS IS A REPLICATION OF BUFFON'S
EXPERIMENTAL DETERMINATION OF THE VALUE OF PI -
OBTAINED BY RANDOMLY DROPPING A NEEDLE MANY
TIMES ON A PLANK RULED WITH PARALLEL LINES"
105 PRINT: PRINT "IN THIS PROGRAM THE LINES ARE 6
SPACES APART AND THE NEEDLE IS LESS THAN 6 SPACES
LONG. THE PROGRAM DISPLAYS THE NEEDLE RANDOMLY
LANDING ON THE SCREEN AND COMPUTES THE TOUCHES
'T', THE";
110 PRINT" MISSES 'M', AND THE RATIO 'R' OF TOTAL
TOSSES TO TOUCHES"
115 PRINT: PRINT "THE PROGRAM TOSSES 100 RANDOM
NEEDLES AND THEN PRINTS OUT THE RESULTS. YOU CAN
RUN AS MANY SETS OF 100 TRIALS AS YOU WISH. THE
CUMULATIVE RATIO 'CR' WILL SHOW THE RATIO
DEVELOPED BY ALL THE SETS."
120 PRINT "WHEN YOU ARE READY TO GO ON PRESS ANY
KEY"
125 KY$=INKEY$: IF KY$="" GOTO 125 ELSE 130
130 CLS: GOTO 15
200 CLS: GOTO 300
205 PRINT @ 576, "FIGURE 1 SHOWS THE NEEDLE OF
LENGTH 'L', THE DIAMETER OF THE CIRCLE 'D', THE ANGLE
'Q', AND THE DISTANCE 'Y' FROM THE MIDDLE OF THE
NEEDLE."
210 PRINT: PRINT "THE PROBABILITY OF THE NEEDLE
LANDING ON A LINE IS A FUNCTION OF BOTH THE
DISTANCE 'Y' AND THE ANGLE 'Q'."
215 PRINT "PRESS ANY KEY TO CONTINUE WITH THE
EXPLANATION"
220 A$=INKEY$: IF A$="" GOTO 220 ELSE 225
225 E=512: EE=1
230 PRINT @ E, STRING$(63,128): E=E+64: EE=EE+1
235 IF EE>7 GOTO 240 ELSE 230
240 PRINT @ 576, "THE NEEDLE WILL TOUCH THE LINE IF
THE CENTER 'Y' IS LESS THAN 1/2L(SINQ).Y CAN RANGE
FROM 0 TO D/2 AND ANGLE Q CAN RANGE FROM 0 TO
PI."
245 PRINT "THEREFORE THE PROBABILITY 'P', ON AN
ASSUMPTION OF RANDOM DROPS, COULD BE EXPRESSED
AS P = 2L/PI D."
250 PRINT: PRINT "PRESS ANY KEY TO RUN MORE TRIALS.";
255 KY$= INKEY$: IF KY$="" GOTO 255 ELSE 260
260 GOTO 25
300 CLS: PRINT @ 220, "FIGURE 1": PRINT @ 250,"d":
PRINT @ 303,"L": PRINT @ 368,"Q": PRINT @ 371,"Y": PRINT
@ 491,"O"
305 FOR J=78 TO 121: SET(J,4): SET(J,19): NEXT J
310 FOR K=2 TO 23: SET(90,K): SET(91,K): NEXT K
320 K=6
325 FOR J=104 TO 91 STEP-1: SET(J,K): K=K+1: NEXT J
330 FOR K=10 TO 14 STEP2: SET(102,K): NEXT K:
SET(102,18)
335 SET(116,5): SET(116,7): FOR K=13 TO 17 STEP2:
SET(116,K): NEXT K
340 GOTO 205 ■
```

How to hunt for hidden treasure with your TRS-80.

Captain 80 Book of BASIC Adventures.

Thar's GOLD in them thar plots!

Eighteen tantalizing, challenging fantasy adventures in which YOU play the role of adventurer and hero! Finding treasures and achieving goals while trying to avoid deadfalls and other insidious obstacles are just a few of the things you'll experience in the *Captain 80 Book of BASIC Adventures.*

This 252-page book contains eighteen exciting adventures edited by Bob Liddil of The Programmer's Guild. Seven chapters tell you the what, when, where, why and how of adventures, including how to play, write and sell your own programs. Although written for the TRS-80 Models I and III, these adventure programs will easily adapt to most computers using Microsoft® BASIC.

The highlight of this book is the Adventure Generator. This program requires disk BASIC and will actually write another BASIC program around your imaginative instructions!

Please send ____ copies of the ADVENTURE BOOK for only $19.95 each, plus $2.05 shipping and handling. Check, M.O., VISA/MC# enclosed. C.O.D. orders accepted.

Name _____

Address _____

City _____ State _____ Zip _____

VISA MC _____ Exp. Date _____

Signature _____

SEND TO:
80-NW Books
3838 So. Warner St., Dept. C, Tacoma, WA 98409
DEALER INQUIRIES INVITED

You don't have to be a mathematician

A literary approach to structured programming

Tasha K. Taylor, Seattle, WA

I am neither a mathematician by education nor avocation. However, in the last year I have literally become a "computer addict." I am unable to function normally without access to a computer.

This all started 18 months ago, when I embarked on writing a novel, and began taking creative writing classes. As a result, I started pestering my husband for a new typewriter. Instead of investing in a high-class typewriter, we decided to upgrade our bare-bones TRS-80 Model I, and purchase a printer and SCRIPSIT.

In January 1981, my husband and I took the Radio Shack BASIC Programming courses. Slowly, I found myself emerging from a software "user" to a software "creator." Prior to those classes, I associated programming with staying up all night at a computer center, as I did for a class I took in college.

By the middle of last summer, I decided to pursue programming seriously. Consequently, I took more courses. In the process of doing so, I was presented with my own flow-chart template. As a dutiful student, I tried desperately to use it. I encountered two problems. First, I could never remember what all the boxes were for. I only used the three I could remember. Out of frustration, I resorted to drawing squares and labeling them with the functions they were to perform.

The second problem was the *size* of the squares. A "wordy" person, like me, could never fit everything into a box that small! As an alternative to flow-charting, I independently started applying the same techniques I used in my writing to my programming. I would begin by writing an outline, and pushing myself to be specific and succinct in describing the actions I wanted the program to perform. Then I would write the actual code for the program.

Subsequently, I have read a great deal about structured programming and top-down design. I am finding that my social science and humanities background has provided me with a level of expertise that I can draw on as a programmer.

Structured Programming versus Structured Writing

Organization is the key to performing any task in a successful manner. Structured programming is the term used to define the current state of the art in software engineering. To make the concept of structured programming more palpable to the layman, let's compare this strategy to a task that everyone is exposed to at one point in his/her education, that of writing a report.

The first step in writing a report is to identify the topic, not always an easy task. In writing a program, the first step is to specify the problem the program will solve. Don't let the word "problem" conjure up pictures of massive mathematical formulas. Let's use the following "problem" as an example: Prepare a program that would produce a simple mailing list.

The second step in writing a report is the research and development stage. Once the topic has been researched, the report needs to be designed. This can be done by an outline that maps out the structure of the report. Now, compare this process to the programming concept of top-down design.

Top-down design is the preferred approach to software design today. In top-down design, we attempt to define all of the requirements of the program before we attempt to implement it. Once the requirements are known, we can devise a

Figure 1
Top-Down Design

Design Level	Structured Writing	Structured Programming
Top Level	Topic Statement	General Program Description
2nd Level	Supporting Statements	Program Procedures
3rd Level	Specific Examples	Procedure's routines
4th Level	Conclusions	Subroutines or modules

program structure that is made up of the main procedure and skeletal subordinate procedures (sometimes called stubs). Then, each of the subordinate procedures is treated in turn.

From this point forward, use the chart in Figure 1 as a reference guide. A comparison is made between design strategies for a report and a program.

Before writing (or coding) a program, we should also design the structure that the program will follow.

Top Design Level — The top design level in a report is the topic statement — the subject (thesis) the report will attempt to prove. In a program, the top level is the general program description. In the example of a mailing list, the top design level will be referred to as the Main Menu, which will delineate the program strategy. The program will cover the following areas:

MAILING LIST
1. ADD/EDIT/DELETE NAMES
2. SAVE/LOAD DATA
3. PRINT LABELS

Second Design Level — The second design level in a report is providing support statements to validate the topic. In a program, one must establish the structure of the program procedures. The ADD/-EDIT/DELETE NAMES procedure will have three routines: one that will ADD names to the program, one that will EDIT names in the program, and one that will DELETE names from the program.

Third Design Level — This is the nitty-gritty level of a report. At this stage, it is necessary to provide specific examples to demonstrate whether or not the topic is correct. In programming, specific subroutines are designed at this level to perform the functions of each of the routines in the procedures. For example, one should plan the subroutines that will comprise the SAVE DATA routine in the SAVE/LOAD DATA procedure.

Fourth Design Level — The final step in writing a report is the conclusion, whereby all previous statements are drawn together and summarized. In a program, the fourth level is the solution to the problem the program was to solve. In most cases, this is done at the finest subroutine or module level. It is the finest subroutine that performs the function of alphabetizing the mailing list. It will be at the finest subroutine level that labels will be formatted and printed.

Language Skills — After the structure of a report is designed, the report must then be written. The actual process of writing a program is referred to as "coding" the program. In the process of learning to write any language, one learns the definitions of new words, and how those words are used. With constant use, these words become a part of your vocabulary.

The same type of procedure occurs in the acquisition of a programming language. Programming languages usually have limited lists of words, or terms, to learn. Having once developed successful subroutines (or modules), a programmer adds these routines to his "vocabulary" in that particular language. These modules can be used in other programs where similar functions are needed. Having once written a successful sort subroutine to alphabetize a mailing list, one could then utilize that same module in another program where a sort routine is needed.

Testing and Debugging — The final procedure a writer should perform is to edit the report to correct any errors in syntax, grammar or language. In order to edit (or debug) a program, it must be tested with data to determine whether or not it performs properly. Sometimes testing is a lengthy process; however, so can the process of rewriting a report.

Many educators today are starting to recognize that Computer Science does not belong exclusively in their science or math departments. Instead, it should be established as a separate entity, accessible by all students. Don't let stereotypes keep you from learning to program. Computers are a permanent part of our society. I find it much more enjoyable to be their master, rather than their slave. ■

All about input

How to structure, filter, blink and massage your INKEY$

Models I/II/III, PMC-80, LNW80

Richard Metzler, Albuquerque, NM

Level II BASIC has a lot of fine features but, unfortunately, the INPUT routine is not one of them. Even if the operator's input is of the correct form, there is no way to avoid the carriage return which follows the ENTER and wipes out the next line on the screen. And, if there is operator error, Level II gives that perplexing REDO? message. In this article, we'll see how to use the INKEY$ command to produce error-trapped, bomb-proofed, screen-preserving, and even encrypted input routines.

Our technique will be to build up our complicated routines from simple building blocks. The foundation of all the routines is this simple variant of the usual "press any key to continue" input routine: FOR J=0 TO 0: J=(INKEY$=" ") : NEXT. It looks like the FOR loop would only execute once, but the inner command returns a -1 as long as no key is pressed, and the NEXT increments J to zero and sends it back to the FOR. When a keypress occurs, the inner command sets J=0 and we exit the loop. The advantage this routine has over the usual: 10 IF INKEY$=" " THEN 10, is that it will go in the middle of a multi-statement line, allowing you to pack your program more effectively. Also, it helps to "structure" your program — you minimize the dreaded BASIC GOTOs. As a structured command, we might refer to it as REPEAT WHILE (INKEY$ = " ").

If you need to know what key caused the exit, we can make the following change: FOR J=0 TO 0: IN$ = INKEY$: J=(IN$ = " "): NEXT. Then the keypress is stored in IN$ on exit.

Now, suppose that you wish to have "timed input" for a game command. Consider the following: FOR J=1 TO TIME: IN$= INKEY$: J=J-TIME*(IN$> " "): NEXT. Now, if there is no input, the "J=" command sets J equal to J-0, so J is unchanged and after TIME cycles, the exit will occur with IN$= " " and J= TIME+1. However, if a key is pressed, J becomes J-TIME*(-1) and we exit with a J value of J+TIME+1. We need only subtract TIME+1 to find the number of cycles it took the operator to press a key.

It is often helpful to signal the operator that the machine is waiting on input by blinking the cursor. We can manage that as follows: FOR J=0 TO 0 :
IN$=INKEY$: J=(IN$= " ")
:JJ=(JJ+1)AND7: PRINT CHR$(14AND(JJ=1)OR15 AND(JJ =5));
:NEXT

The first "JJ" command increments JJ from zero to one, one to two, etc., until it increments from seven to eight when the "AND7" sends it back to zero. The next command PRINTs CHR$(0) (which does nothing) unless JJ=1 or JJ=5. If JJ=1, it turns on the cursor with PRINT CHR$(14) and if JJ=5, it turns it off with PRINT CHR$(15).

Suppose we wish to "bombproof" the routine: we will only accept certain keypresses. In the simplest case, we assume that the acceptable inputs form an interval in the ASCII list. We will only accept IN$ if MIN$<= IN$ <+ MAX$. The following does the job: FOR J=0 TO 0: IN$=INKEY$: J=((IN$<MIN$)OR (MAX$<IN$))
:NEXT

This would allow us to accept only octal digits. For example, if MIN$="0" and MAX$="7". If we wanted to filter out all but the hex digits, we could set L1$="0", U1$="9", L2$="A", U2$="F" and use: FOR J=0 TO 0 : IN$=INKEY$
:J=(IN$<L1$)OR(U2$<IN$)OR((U1$<IN$)AND(IN$< L2$))
:NEXT

If we were restricting input to "Y" for yes and "N" for no, this would do it:
FOR J=0 TO 0: IN$=INKEY$: J=((IN$<>"Y")AND (IN$< >"N"))
:NEXT

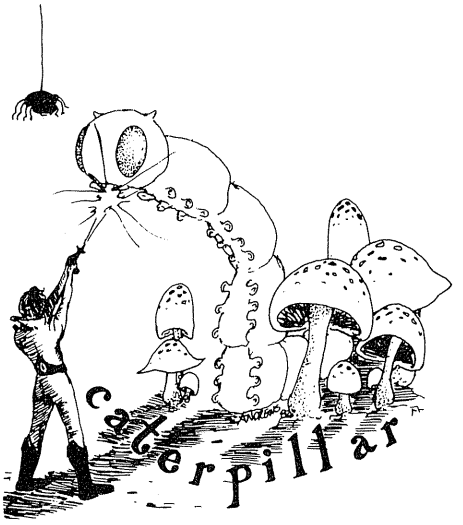
If we had a longer list of acceptable inputs, the above method would quickly become tedious. Suppose we wish to allow only the inputs "C" for copy, "N" for new, "K" for kill, and ENTER for "none of the above." The INSTR function of disk BASIC allows the following compact routine to be used when FL$="XCNK"+CHR$(13): FOR J=2 TO 2: J=INSTR(FL$, INKEY$) : NEXT: IN$=CHR$(J-1).

The reason for the change from zero to two in the FOR statement, is due to a peculiarity of the INSTR function. When INKEY$=" ", the null string, the INSTR function (which ordinarily returns the number of the first

AARDVARK

TRS-80 COLOR OSI VIC-64 VIC-20 SINCLAIR TIMEX

QUEST – A NEW IDEA IN ADVENTURE GAMES! Different from all the others. Quest is played on a computer generated map of Alesia. Your job is to gather men and supplies by combat, bargaining, exploration of ruins and temples and outright banditry. When your force is strong enough, you attack the Citadel of Moorlock in a life or death battle to the finish. Playable in 2 to 5 hours, this one is different every time. 16k TRS-80, TRS-80 Color, and Sinclair. 13K VIC-20. $14.95 each.

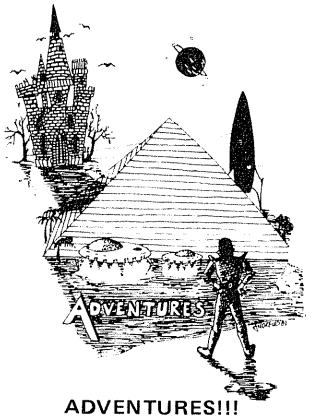


CATERPILLAR

O.K., the Caterpillar does look a lot like a Centipede. We have spiders, falling fleas, monsters traipsing across the screen, poison mushrooms, and a lot of other familiar stuff. COLOR 80 requires 16k and Joysticks. This is Edson's best game to date. $19.95 for TRS 80 COLOR.

TUBE FRENZY
(by Dave Edson)

This is an almost indescribably fast action arcade game. It has fast action, an all new concept in play, simple rules, and 63 levels of difficulty. All machine code, requires Joysticks. Another great game by Dave Edson. TRS 80 COLOR ONLY. 16k and Joysticks required. $19.95.



ADVENTURES!!!

These Adventures are written in BASIC, are full featured, fast action, full plotted adventures that take 30-50 hours to play. (Adventures are interactive fantasies. It's like reading a book except that you are the main character as you give the computer commands like "Look in the Coffin" and "Light the torch.")

Adventures require 16k on TRS80, TRS80 color, and Sinclair. They require 8k on OSI and 13k on Vic-20. Derelict takes 12k on OSI. $14.95 each.

PROGRAMMERS!
SEE YOUR PROGRAM IN THIS SPACE!!

Aardvark traditionally pays the highest commissions in the industry and gives programs the widest possible coverage. Quality is the keyword. If your program is good and you want it presented by the best, send it to Aardvark.

ESCAPE FROM MARS
(by Rodger Olsen)

This ADVENTURE takes place on the RED PLANET. You'll have to explore a Martian city and deal with possibly hostile aliens to survive this one. A good first adventure.

PYRAMID (by Rodger Olsen)

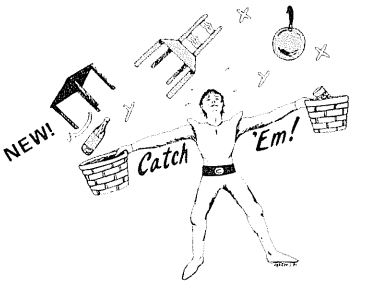
This is our most challenging ADVENTURE. It is a treasure hunt in a pyramid full of problems. Exciting and tough!

HAUNTED HOUSE (by Bob Anderson)

It's a real adventure—with ghosts and ghouls and goblins and treasures and problems – but it is for kids. Designed for the 8 to 12 year old population and those who haven't tried Adventure before and want to start out real easy.

DERELICT
(by Rodger Olsen & Bob Anderson)

New winner in the toughest adventure from Aardvark sweepstakes. This one takes place on an alien ship that has been deserted for a thousand years – and is still dangerous!



CATCH'EM
(by Dave Edson)

One of our simplest, fastest, funnest, all machine code arcade games. Raindrops and an incredibe variety of other things come falling down on your head. Use the Joysticks to Catch'em. It's a BALL! – and a flying saucer! – and a Flying Y!– and so on. TRS 80 COLOR. $19.95.

BASIC THAT ZOOOMMS!!
AT LAST AN AFFORDABLE COMPILER!

The compiler allows you to write your programs in easy BASIC and then automatically generates a machine code equivalent that runs 50 to 150 times faster.

It does have some limitations. It takes at least 8k of RAM to run the compiler and it does only support a subset of BASIC– about 20 commands including FOR, NEXT, END, GOSUB, GOTO, IF, THEN, RETURN, END, PRINT, STOP, USR (X), PEEK, POKE, *, /, +, -, >, <, =, VARIABLE NAMES A-Z, SUBSCRIPTED VARIABLES, and INTEGER NUMBERS FORM 0-64K.

TINY COMPILER is written in BASIC. It generates native, relocatable 6502 or 6809 code. It comes with a 20-page manual and can be modified or augmented by the user. $24.95 on tape or disk for OSI, TRS-80 Color, or VIC.

Please specify system on all orders

ALSO FROM AARDVARK – This is only a partial list of what we carry. We have a lot of other games (particularly for the TRS-80 Color and OSI), business programs, blank tapes and disks and hardware. Send $1.00 for our complete catalog.

AARDVARK - 80
2352 S. Commerce, Walled Lake, MI 48088
(313) 669-3110

Phone Orders Accepted 8:00 a.m. to 4:00 p.m. EST. Mon.-Fri.

23

character in FL$ matching the INKEY$ character, or a zero if there is no match) will return a one! This would cause an undesired exit if the FOR loop ran from zero to zero. That's why we need to run J from two to two. This means that we must fill the first spot in our "filter" FL$ with a dummy character (in our case, an "X") since a "find" in the first position would return a one, which would not cause an exit.

The next step is to fashion an input routine which will accept and store more than one keypress. We will set up a subroutine which allows LN keypresses, chosen from a string, FL$, of acceptable ones:

```
FOR I=1 TO LN:
:FOR J=2 TO 2: IN$=INKEY$: J=INSTR(FL$,IN$):
NEXT
:PRINT IN$; : MID$(OT$,I,1)=IN$
:FOR K=16438 TO 16444 : POKE K,0 :NEXT
:NEXT : OT$=LEFT$(OT$,LN)
```

The output is stored in the string OT$, which must be initialized to have length at least LN. The reason for the use of the MID$ technique is to avoid the generation of "garbage" strings which are associated with string concatenation. The loop on K is a little extra feature. It causes the INKEY$ buffer to be reset so that holding down a key will cause it to repeat. If you're troubled by keybounce, you might want to put in a delay loop.

As an interesting application of the above techniques, let's examine a short routine to change the usual TRS-80 keyboard into the super-efficient Dvorak keyboard. In Listing 1, we have a one-line program which "encrypts" the keyboard so that the display is the improved alternate to the usual "QWERTY" keyboard designed by Dvorak in the 1930s.

After clearing sufficient string space, we set up the translation string D$ of 122 characters. Then we insert the two control characters we will recognize: CHR$(8) for backspace, and CHR$(13) for ENTER. Then, we turn on the cursor with PRINT CHR$(14);. The loop is on J as it goes from zero to one, with a step of zero. The step of zero means that NEXT never increments J, and the only way we can exit from the loop is via the BREAK key. The command in the loop just prints CHR$(0) (the first character in D$) if there is no keypress. In this case, it is a do-nothing command. If there is a keypress, we pick out the character in D$ corresponding to the ASCII value of the keypress. This translates to the Dvorak keyboard.

The reason for the use of INKEY$+CHR$(1) is due to a peculiarity of the ASC function. The function gives an error when its argument is the null string. Since " " + CHR$(1) is just the same as CHR$(1), the ASC function on the concatenation just returns one, which prints CHR$(0), since the first character in D$ is CHR$(0). Because the ASC function returns the ASCII value of the first character of its string argument, no harm is done by tacking on the CHR$(0) when INKEY$ is not null.

Let's try to exploit the "filter" idea further. We will filter the keyboard so that only selected keypresses are recognized. These will be filtered (as in the Dvorak example above) before being printed on the screen. (This might be useful for a situation where the operator types in a password and the filter changes the screen display for security.) We will also provide for a filter on the characters delivered to the output string in case they need to be encrypted or changed for RS-232 or printer output. Assume that we have predefined filter strings FKBD$, FSCR$, CODE$, and FOTPT$.

Consider the following routine:

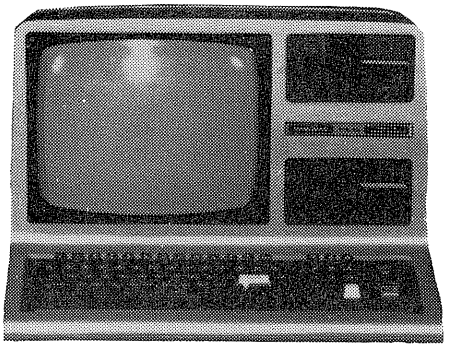
```
FOR I=1 TO LN
:FOR J=1 TO 1
:J=ASC(MID$(FKBD$,ASC(INKEY$+CHR$(1)),1))
:NEXT
:PRINT MID$(FSCR$,J-1,1);
:MID$(FOTPT$,I,1)=MID$(CODE$,J-1,1);
:NEXT
:FOTPT$=LEFT$(FOTPT$,LN)
```

As in the Dvorak example, we adjust FKBD$ so that the element in the Jth position is nonzero only if J=ASCII (INKEY$) where INKEY$ results from an acceptable keypress. A nonzero value of J causes an exit from the loop and we print the corresponding (possibly coded) value of FSCR$, then store the value (possibly coded with a different code) in FOTPT$.

Let's put the above ideas to work in a multipurpose input program. In Listing 2, we have a routine which will define an "input area" on the screen, with graphics blocks, and will accept only certain characters depending on the value of the flag string SS$. Clear 500 bytes for string input, then initialize the length of the

Apparat, Inc.

4401 So. Tamarac Parkway, Denver, CO 80237 (303) 741-1778



MODEL III PRICE LIST

(All 48K with Tandon drives & the Apparat Disk Controller)

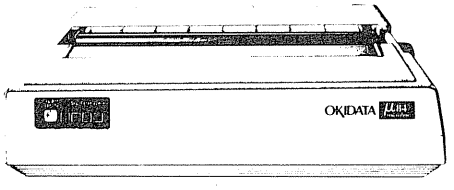
2 single sided 40 track drives	**$1895.00**
2 dual sided 40 track drives	**2145.00**
2 dual sided 80 track drives	**2395.00**
with RS-232 C add	**115.00**

Price includes NEWDOS/80 & TRSDOS

NOTE: These Model III computers contain Apparat installed disk drives and memory. They are warranted by Apparat, Inc. for 120 days.

OKIDATA ML 84

The best. For less.



The printer with correspondence quality **and** data processing throughput! The 200 cps bidirectional ML 84 is the multi-mode printer you can afford. Okigraph™ dot-addressable graphics are standard. Six program-selectable character sizes with true decenders, enhanced and emphasized print. The ML 84 proves top of the line doesn't have to mean top dollar. **$1125.00**

SPOOL/64

Saving time, increasing productivity and lowering costs.

SPOOL/64, a 64K byte hardware print spooler, keeps your computer processing while your printer is printing. No more valuable time lost by waiting for a print run to finish.

This unit plugs directly into your personal computer and printer with one easy connection. It can then transfer up to 3,000 characters per second. The data is stored in the spooler's memory and sent to the printer at a rate the printer can accept. **$319.00**

TANDON DISK DRIVES

Complete with power supply, chassis & configured for TRS-80* Model I or III

- Single sided 40 track **$285.00**
- Dual 40 or Single 80 track... **$415.00**
- Dual sided 80 track..... **$515.00**
- **Special** - 2 Dual sided 80's.. **$999.00**

(Adds almost 1.5 megabytes of storage to a double density model I or model III)

NEWDOS/80

Version 2.0

The most sophisticated DOS ever produced for the TRS-80* Models I and III. It provides the user with "MAINFRAME" power on a "MICRO".

Some Features available are:

- Jobstream Control Language
- Mod I/Mod III Diskette interchangability
- Double Density Support on Model I
- Pagenation of BASIC listings on the screen
- Basic program single stepping
- Dynamic variable manipulation
- Multiple array sorts with BASIC CMD "O"
- Complete technical support provided

All this plus much more for only

$149.00

NEWTERM /80, MODEL III $49.95

- Runs under popular operating systems
- Interrupt driven to 19,200 baud
- Translation table
- Automatic sign-on
- Status line

CHEXTEXT®, $59.95

"Surefire Spelling Checker"

See Desktop Computing Sept. '82 issue, p. 52-53 for review. . . ."Chextext, a program that avoids the limitation brought about by hashing/since it retains the integrity of the words . . . The dictionary, or any part of it, may be displayed on the screen."

FLEXTEXT/PLUS, $34.95
For the Mod I & II for all EPSON printers with GRAFTRAX/PLUS

CREATIVE CUISINE, $29.95
Electronic Cookbook

InfoWorld
Software Report Card

NEWDOS/80
Version 2.0

	Poor	Fair	Good	Excellent
Performance	☐	☐	☐	☑
Documentation	☐	☐	☐	☑
Ease of Use	☐	☐	☑	☐
Error Handling	☐	☐	☐	☑

25

MEAL MASTER, $24.95
Meal planning & grocery shopping aid

UNITERM/80, $89.00
Universal terminal program that takes advantage of the extended NEWDOS/80 commands

MICRO CLINIC, Mod I $24.95, Mod III $29.95
The ultimate in memory & disk diagnostics

APPARAT'S PROM BLASTER
An eprom programmer for most 25 X X and 27 X X chips. TRS-80

MOD I & III	**$119.00**
Cable	**17.95**
Bus Extender	**69.95**

MODEL I DOUBLE DENSITY PACKAGE
Converts a standard TRS-80 Model I to Double Density.
Apparat Doubler & NEWDOS/80 V 2.0. $278.00 value for only **$219.00**
Doubler alone **$129.00**

VISA MasterCard

Scripsit* & TRS-80* are registered trademarks of Tandy Corporation Freight F.O.B. Denver, call for shipping charges. Foreign Orders shipped Air Freight

input string to be eight. Initialize FF$ to contain three control characters, three spaces for later insertion of special characters, all the possible keyboard entries, and five characters on the end which we can use (together with ARW$ in the next line) to print arrows and underlines on the screen. This last is accomplished by holding down the shift key and downarrow key with one hand while pressing one of the keys from B to F with the other hand.

Line 50 is our executive program to demonstrate the input subroutine. We call the subroutine with all possible values of SS$ and some different values of LN, to illustrate its operation. Lines 70 through 110 are the actual routine. In line 70, we chop LN down to sixty-three or less (if necessary), call a subroutine which prints LN graphics blocks, and set up our output storage string. In line 80, we set up our filters for integers, real numbers, or hexadecimal numbers. We choose only that part of FF$ which is needed and put the "special treatment" characters in positions four and five in FL$. Line 90 assigns the filter FL$ when we are expecting a disk filespec and puts "/" in special character position number six.

All the action is in line 100. We stay in that line until LN characters have been entered or until ENTER has been pressed. We first blink the cursor as usual, then obtain the keypress in IN$ and check to see if it appears in FL$. The "AND (IN$>" ")" which is tacked on is there to correct for the fact that INSTR(FL$, " ")=1.

The next four lines are a series of ON GOSUBs. I prefer these to IF statements in the present situation, because there is no need to split into separate lines. If there is no line number corresponding to the expression following the ON, we go to the next statement, not to the next numbered line, as with an IF statement.

The first ON GOSUB will branch only if I=1, and we are expecting a disk filespec. Subroutine 220 rejects anything but letters for the first character of a disk filespec. The next branch provides for backspace, ENTER, line-clearing with shift left-arrow, and special treatment for "-", ".", and "/" (depending on the value of SS$).

In the next line, we allow for the possibility of printing the arrows and the underline character. The final ON GOSUB provides for printing the character on the screen, storing it in an output string, and incrementing I by 1. The last line decrements I to compensate for the automatic increment done by the NEXT. We exit the loop when LN characters have been typed, or when ENTER is pressed and we set bit number six with I=(I OR 64).

In line 100, we decrement I after the increment by NEXT and zero bit number six with an "AND 63" command. The effect of this is to give us the correct character count even if ENTER caused an early exit. We chop the output string to the correct length, erase any

remaining graphics blocks, reset SS$ and LN to default values, and return. The subroutines called by the ON GOSUB lines are found in lines 120 to 270.

Well, that's the end of our guided tour through the land of structured, filtered, timed, blinked, and otherwise massaged INKEY$ input. I hope you find it useful in your next program.

Program Listing 1 — Input

```
5 '*** ONE-LINE DVORAK KEYBOARD 3/8/82 ***
 *** BY RICHARD C. METZLER ***

10
:CLEAR 300: D$ = STRING$ (31,0)+
" *"+CHR$ (34)+"#$%&'()!Sw-vz6: 753190248sW =
VZAXJE.UIDCHTNMBRL?POYGK,QF; @axje>
uidchtnmbrl/povgk< qf +"
:MID$ (D$,8,6)= CHR$(8) +" "+ CHR$(13): PRINT
CHR$(14);
:FOR J=0 TO 1 STEP 0
 :PRINT MID$(D$,ASC(INKEY$+ CHR$(1)),1);
:NEXT
```

Program Listing 2 — Input

```
5 '*** INFILTER 3/7/82 ****
 '*** BY RICHARD C. METZLER ***

10 '*** INITIALIZATION ROUTINE ***
20 :
:CLEAR 500: LN=8
30 :
:FF$= CHR$(8) + CHR$(13) + CHR$(24) +
STRING$(3,128) +
"0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ !"+
CHR$(34) + "#$%&'(): *-<>? + : @;,./("+ CHR$(2) +
CHR$(3) + CHR$(4) + CHR$(5) + CHR$(6)
:ARW$ = CHR$(91) + CHR$(92) + CHR$(93) + CHR$(94)
+ CHR$(95)

40 '*** EXECUTIVE PROGRAM TO EXERCISE INPUT
SUBROUTINE ***
50 :
:LN=99: GOSUB 70: PRINT OT$
:SS$="INT": LN=4: GOSUB 70: PRINT OT$
:SS$="DEC": LN=23: GOSUB 70: PRINT OT$
:SS$="HEX": LN=4: GOSUB 70: PRINT OT$
:SS$="FILESPEC": LN=12: GOSUB 70: PRINT OT$
:STOP

60 '*** INPUT SUBROUTINE ***
70 :
:LN=((LN AND (LN<63)) OR (63 AND (LN>=63)))
:I=1: GOSUB 170
:OT$= STRING$(64,0)
80 :
:IF SS$="INT" THEN FL$= LEFT$(FF$,16): MID$(FL$,4,1)="-"
ELSE IF SS$="DEC" THEN FL$= LEFT$(FF$,16):
MID$(FL$,4,1)="-"
 :MID$(FL$,5,1)="." ELSE IF SS$="HEX" THEN
FL$=LEFT$(FF$,22) ELSE FL$=FF$
90 :
:IF SS$= "FILESPEC" THEN FL$= LEFT$(FF$,42):
MID$(FL$,6,1)="/"
100 :
:FOR I=1 TO LN
 :J=(J+1) AND 7: PRINT CHR$(14AND(J=1) OR 15 AND
(J=5));
 :IN$= INKEY$: K=(INSTR(FL$,IN$) AND (IN$>""))
 :ON I AND (SS$= "FILESPEC") GOSUB 230
 :ON K GOSUB 130,150,170,190,210,210
 :ON -(K>66) GOSUB 260
 :ON -(K>3) GOSUB 250
 :I=I-1
:NEXT
110 :
:I = ((I-1)AND63)
:OT$=LEFT$(OT$,I)
:PRINT STRING$(LN-I,32)
:SS$="":LN=8
:RETURN

120 '***** BACKSPACE SUBROUTINE ***
130 :
:PRINT CHR$(136) CHR$(24);
:IF I>1 THEN I=I-1: PRINT CHR$(24) CHR$(136)
CHR$(24);: RETURN
 ELSE RETURN

140 '*** "ENTER" SUBROUTINE ***
150 :
:I =(I OR 64) :RETURN

160 '*** CLEAR-LINE SUBROUTINE ***
170 :
:PRINT STRING$(I-1,24) STRING$(LN,136) STRING$(LN,24);:
I=1: RETURN

180 '*** "MINUS ONLY IN FIRST POSITION" SUBROUTINE
***
190 :
:IF I>1 THEN K=0 :RETURN ELSE RETURN

200 '*** "ONLY ONE OCCURRENCE ALLOWED"
SUBROUTINE ***
210 :
:IF (I>1) AND (INSTR LEFT$ (OT$,I-1), IN$)>0) THEN K=0:
RETURN ELSE RETURN

220 '*** "ONLY LETTERS IN FIRST POSITION" SUBROUTINE
***
230 :
:IF INSTR(17,FL$,IN$)=0 THEN K=0: RETURN ELSE RETURN

240 '*** "PRINT AND STORE CHARACTER" SUBROUTINE ***
250 :
:PRINT IN$;: MID$(OT$,I,1)=IN$: I=I+1: RETURN

260 '*** CHANGE SHIFT-DOWN-ARROW B-F TO
ARROWS & UNDERLINE ***
270 :
:IN$= MID$(ARW$,K-66,1): RETURN ■
```

80-U.S. Journal interviews Bob Snapp

80-U.S. Staff

Bob Snapp has earned the respect of many in the industry for his "enhancements" to TRSDOS. He has provided many of us with utilities and application packages that are truly first class and well documented. During a recent conference in Wisconsin, 80-U.S. had the opportunity to sit down with Bob and discuss a little about where we have been and where we are going. —Ed

Q. How about a little personal history?

A. It was about four years ago that the company I was working for was in the process of being sold to a larger company. I enjoyed being a big fish in a small pond, running the data processing department and the man I reported to was "The Boss," I enjoyed that. I didn't really want to be reporting to some guy working in another city who is reporting to someone else six layers down in management.

Bob Snapp

So, a friend of mine invited me to set-up a consulting business and we agreed on a contract that would guarantee enough revenue to get started.

I set up my own software consulting business at that time. This was also the time that the TRS-80 was being introduced and, of course, I had one. In one sense it was a toy, but in another, I could see the possibilities. I had been accustomed to dealing and selling packages for mainframes. I saw this as missing, there wasn't anything like it for the micros. I was just filling in something that should have been there in the first place.

That's really how we got started. I wrote for my own use and sold to a few businesses. Of course, they were delirious with it and then I thought, "Hmmm, maybe there is something here we can sell." It was a long road. Credibility makes a successful software business. We were building credibility to get some real revenue. We finally did that and built up an expanding product line.

I think that if I made any real mistakes in getting started in the TRS-80 software business, it was that I cut off my old business too soon. My advice is to hold on to your other source of revenue until you're really, really sure. Revenue that can really support you and your family.

Q. It appears that you feel someone can still jump into this business, start from scratch and do quite well.

A. I think so. There are still a lot of needs to be met and that's the secret to being successful. There will be opportunities for many years to come.

Q. Can you tell us a little about Snappware's current products or projects?

A. The primary thrust of our current product line is tools to assist in developing custom applications, to do it more quickly and more cost effectively. This is an expansion of a philosophy I developed while I was working with mainframes. We were not really using our people effectively and we were spending all our time trying to develop applications to save the end-user's time. It was getting to the point that the machine became more expensive. Something had to be done or eventually we wouldn't be able to get any application code out on the market. People would be buying machines for $50 and paying $5000 to develop software for it.

Q. What market are you aiming for? You have produced a number of good items for the Model II. Is there work on any others?

A. Well, we're into the Model I and III and LDOS. We support TRSDOS on the Model III, although we're not delirious about it. We are adapting significant portions of the product line to the new IBM personal computer, not because we particularly like that machine but because we know we will sell a bundle of them.

Q. Is Snappware your main occupation?

A. It's my only occupation, other than playing the stock market.

Q. How do you see the future of the TRS-80 market and its peripheral support groups?

A. I see it as a continuously expanding market. I sincerely hope that Tandy maintains the larger share or increases it. I have my worries, but the microcomputer market, generally, is going up and up and up. I like Tandy's computer products. It's good machinery, reasonably priced and there is a lot of good, off-the-shelf software for it. It sits very well in the marketplace. I

BEFORE YOU GET TOO EXCITED ABOUT LOBO'S NEW COMPUTER, THERE'S SOMETHING YOU SHOULD KNOW.

There's plenty to be excited about in Lobo's new MAX-80,™ as you'll see in just a minute.

But first we want to warn you: you can't get one right away. Already, orders are coming in faster than we can build systems. However, if you can appreciate an incredible price/performance bargain, you'll agree the MAX-80 is well worth waiting for.

WHAT'S ALL THE EXCITEMENT ABOUT?

We're glad you asked. And the answer is pretty simple. Just look at this list of *standard* features:

- *5 MHz Z-80B processor.* That's 2½ times the speed of a TRS-80 Model III or SoftCard/Apple!
- *64k RAM.* 128k is a low-cost option.
- *CP/M included.* A few more dollars get you LDOS, an incredibly powerful operating system that lets you run standard Radio Shack software.
- *Software-selectable 25 x 80, 16 x 64, and 16 x 32 screen formats.* For full compatibility with CP/M and TRS-80 applications.
- *All disk interfaces built in.* Plug in any combination of 5¼" floppies, 8" floppies, and Winchester disk.
- *Two RS-232 serial ports.* Ready to plug in modems, printers, or what-have-you.
- *Centronics-type parallel port.* For any printer using this standard interface.
- *Plus:* numeric keypad with 4 function keys, software definable text and graphics characters, built-in clock/calendar with battery backup, and buffered I/O expander port.

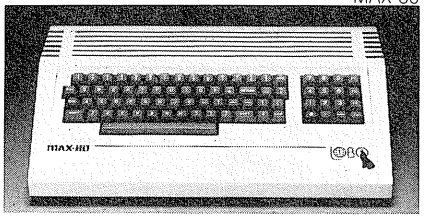
Now for the best part: *the factory-direct price for all this power is just $820—including shipping and Lobo's standard 1-year hardware warranty!*

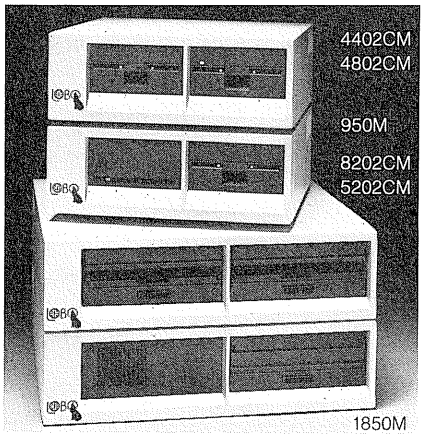
WHAT TO DO NOW.

Call Lobo toll-free. Tell us what hardware and software configuration you're interested in (see below), and we'll give you an approximate shipping date. A $100 deposit will hold your place on the waiting list.

Then get a good book to help you pass the time.

MAX-80





CPU and Accessories

MAX-80 computer with 64k RAM and CP/M	**$ 820.00**
64k expansion RAM (installed)	**$ 95.00**
12" (diag.) high-resolution anti-glare green phosphor monitor	**$ 175.00**
LDOS operating system *instead of* CP/M	**$ 39.00**
LDOS operating system *in addition to* CP/M	**$ 69.00**

Dual 5¼" Floppy Disk Systems

4402CM single-sided, 40 track; 180 kB per diskette	**$ 690.00**
4802CM double-sided, 80 track; 720 kB per diskette	**$1,175.00**

Dual 8" Floppy Disk Systems

NOTE: Lobo CP/M permits reading and writing standard single-sided, single density CP/M disks with either of these systems.

8202CM single-sided, double density; 577 kB per diskette	**$1,185.00**
5202CM double-sided, double density; 1155 kB per diskette	**$1,485.00**

Winchester Disk Systems

950M 5¼" system: 4.8 MB hard disk plus 720 kB floppy	**$2,405.00**
950MX same as 950M above but no floppy drive	**$2,100.00**
1850M 8" system: 8.0 MB hard disk plus 1155 kB floppy	**$3,085.00**

The Lobo Warranty

All Lobo hardware products carry a limited 1-year parts and labor warranty. Call or write for complete warranty statement.

ALL PRICES INCLUDE SHIPPING WITHIN THE U.S.A.
California residents add 6% sales tax.
Payable by credit card, check, or money order.

TOLL-FREE ORDER NUMBERS:
U.S. (except California),
800-235-1245
In California, **800-322-6103 or 800-322-6104** Hours: 7AM - 5PM Pacific Time

MasterCard VISA

LOBO drives INTERNATIONAL

Lobo Drives International
Dept. US183
358 S. Fairview Ave.
Goleta, CA 93117

Prices subject to change without notice.

don't think that Tandy is doing as well as they could in marketing, but I do like the package.

Q. What things do you see coming in the microcomputer field that personally excite you?

A. The Model 16. I'm really excited about the Motorola MC68000. It's the most fascinating thing to come about in years. I cut my teeth on the IBM 370s and when you read the technical specifications on the 68000 it looks just like a 370. It's incredible.

Q. Do you think that the Radio Shack implementation of the 68000 is well done?

A. I have not studied the hardware. I've heard some tales that it may not be as good as it could.

Q. Is Snappware going to be working on the Model 16?

A. Absolutely.

Q. Operating systems?

A. No.

Q. Languages?

A. We might offer a BASIC interpreter for the Model 16 with the facilities and features that people have become accustomed to with our current product lines. We have not yet made a decision of whether we are going to aftermarket a Tandy offering or one of our own. But there will definitely be some offerings.

Q. The micro industry is pretty young and it's done some very good things and some questionable things as it grows. Do you have any pet peeves? Are there specific problems that you feel need to be addressed?

A. There are a lot of rip-offs in this industry and it grieves me because most of the aftermarket merchandise is sold by mail order. It puts a big black eye on the entire industry and once burnt, twice shy. It hurts all of us.

Q. Do you have any suggestions to the user about how he can protect himself?

A. Use credit cards. Back out of credit transactions if you've been cheated. I don't really know what else to suggest, even COD doesn't guarantee that you are going to get something of value. It just guarantees you're going to get a package.

Q. It seems like every month you can read a story either about machine compatibility or software protection and piracy. Do you have any views on either topic?

A. I'll start with the second one. As a software vendor, I'm deeply concerned about protection and piracy. I don't know what the answer is and I know that there is a lot of piracy that goes on. I'd like to see it stopped.

We've put some barriers in our product that makes it more difficult to steal than most. But it's still going to be true for a long, long time that anything a clever programmer can do to lock-up software, another equally clever programmer can undo.

I personally object to getting software that can't be backed up. That really bothers me as a software purchaser. You can't make any other disk out of it, you can't do anything with it. Then it crashes and your business is tied around it. That's unacceptable.

On ours, you make multiple working copies and if the master gets crashed, you can use the working copies while the master is being replaced.

I think that part of the answer is going to be to get some highly publicized court cases before the eyes of the public. Many pirates don't even consider what they are doing to be wrong.

Q. They call themselves collectors.

Yea, the collectors. I hate to say it, but what I think would help is to find some poor turkey and put him in jail for 30 days or 3 days or something. Get it splattered all over the media. That would go a long way and maybe people would think a little more about it.

Q. What about compatibility?

A. That's been a problem since the first Univac. I don't think we're ever going to be able to escape it. People who are writing systems programs are always going to have to deal with the fact that every machine is different. Higher level languages are the key to reducing the problem. One of the big mistakes that I can see is trying to standardize things way too early.

I understand that there is an ANSI standard BASIC being composed. In my opinion, standard usually becomes the least common denominator with minimal requirements. Nobody wants to take a Model II and use it like a Model I. To me, that just doesn't cut it. Assume you were trying to design a language or operating system that would allow total transportability from the Model I, II, III, MX80, even the Apple. What you do is find the weakest subset that will run all those machines. In every case, you'll be losing a significant portion of the power of that machine.

Q. What about all the newcomers to the industry?

A. Frankly, I feel sorry for people who are buying all these new machines. Most companies are not going to make it. People are going to wind up with something that nobody is going to support, and that is sad.

Q. So your word to the buyer is to check the software, and check the support first?

A. My word to the buyer, the ultimate end-user, is to buy a machine that is very popular. You're going to have a lot more, good, off-the-shelf software and you'll have continuing support. The company won't go out of business tomorrow. Even if another machine is half the price, looks like it has a 500-megahertz CPU, stay away from it. You're just going to be sorry. ■

LSNAPP

An evaluation of Snapp-Ware's enhancements to LDOS

Models I/III, PMC-80, LNW80

Pete Carr, Port Orange, FL

Okay, you LBASIC users! Tired of seeing the DOSPLUS, NEWDOS/80 and MULTIDOS users doing all those neat things from BASIC that LBASIC doesn't provide? One might view LDOS LBASIC as a little lean in BASIC features and BASIC utilities compared to these other DOSs. LBASIC does have quite a few really nice features that are missing from standard TRS-80 Microsoft BASIC and some features that no other DOS BASIC has, but I do miss having some of the more advanced programmer utilities from within LBASIC. Well, Snapp-Ware has just released something LDOS BASIC users will definitely appreciate.

I had been reading about Snapp Extensions for the Model II for quite a while. Snapp has now released their extension packages for Model III TRSDOS 1.3 and Model I/III LDOS. There has also been added a new extension called "reverse compression" since the first TRS-DOS 1.3 release. This review will deal with the new LDOS Snapp package, but it does relate to the TRSDOS version, except for price differences.

There are six packages advertised, consisting of:

Extended BASIC	$39.50
Built-in Functions	$34.50
Screen Mapping Support	$34.50
File Mapping Support	29.50
College Educated Garbage Collector	$34.50
Reverse Compression	$19.50

There is also an LDOS trial demo disk for $10. It allows you to make one working Snapp LDOS disk to use and see which, if any, of the extensions you want to buy. This is a great option. The buyer can see exactly what these packages offer. These packages are not cheap, and if you bought all six, they would cost close to $200.

Upon receiving my LSNAPP (LDOS Snapp Extensions) package, I found it included a seventy-six-page manual and one floppy disk. The manual is well written, with a table of contents pointing you to the features of Snapp extensions. Each extension has its own section in the manual with clear instructions and examples. In the back of the manual are a few real-world programs. Like all other computer program manuals, it could use more examples.

Before you can use the new package, you have to install it on an LDOS system disk. After Snapp extensions are installed, you can put any programs or files you want on the disk. The installation procedure is quick and easy. I was a little surprised, because I thought it would be more involved. There is a PATCH file that you use to patch LBASIC, which takes all of about ten seconds.

The next part of the installation creates a file named SNAPPINC on the LDOS disk. This file consists of the Snapp library commands and takes four cylinders of disk space. This is the only part of the installation procedure that takes preplanning. You must preplan the starting cylinder that the file SNAPPINC will start on.

This is easily done with the LDOS FREE map command. You simply find four empty cylinders, starting as close to the directory cylinder as possible. These four cylinders should also be in a contiguous, unbroken format. Placing the SNAPPINC file as close to the directory as possible is done for efficiency. Some Snapp routines are pulled into LBASIC, as they are needed, similar to the way the DOS system uses overlays. This saves valuable computer memory by loading just the code that is needed for a certain function. If the routines (SNAPPINC file) are located too far from the directory and broken up into extents, this will slow the speed of LBASIC considerably because of inefficient disk read–head movement.

After I purged my unwanted files from my LDOS disk, the FREE:0 command showed plenty of room starting at cylinder 27. All we do now is enter the starting cylinder and press ENTER. You can make backups from Snapp LDOS, but you will have to initialize the backup with the Snapp distribution master before it will run correctly. The master distribution disk cannot be copied, so it would be a good idea to make a few run-time LSNAPP disks. This way, if the master distribution disk is damaged, you will still have some LSNAPP disks to use while you send back the distribution master for replacement.

Extended Built-in Functions

I loaded LBASIC and started trying the extended built-in functions package. This thing was more powerful than I had first thought. It had some great POKE and PEEK extensions. With normal BASIC, you can POKE or PEEK into RAM only one byte at a time. Now, I had commands like PEKW, that would read two bytes (word) from memory, and PEK$, that would extract any number of bytes from RAM, up to a length of 255.

I looked in the manual for the POKE command. You could POKE a string of characters up to 255 bytes

New Book

IT'S JUST GREAT

№ 6 IN THE SERIES

AUTHORIZED EDITION

TRSDOS® 2.3 DECODED & Other Mysteries.

FARVOUR EXPEDITION EXPLORES TRSDOS, AND YOU ARE THERE!

Dateline: California, December 1982.

James Farvour . . . Programmer, Author and Adventurer Extraordinare', who two years ago last spring came to popular public attention with his much heralded and nationally acclaimed publication Microsoft Basic Decoded & Other Mysteries, has done it again.

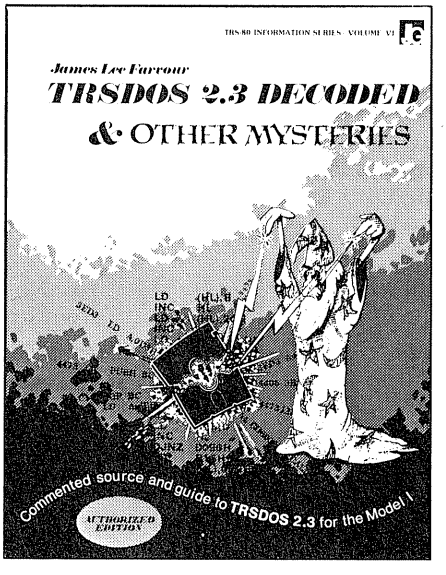
Leaving no stone unturned in his never ending quest to bring to the common man and programmer the information hidden for so long in the dark unchartered expanse of computer software, which was thought before to be unobtainable to mortal men, James Farvour brings to you . . .

TRSDOS 2.3 Decoded & Other Mysteries.

The diary of our intrepid explorers' journey through the internal operations of the TRSDOS operating system (used on the TRS-80 Model I computer) and what he found while he was there.

With the assistance of his trusted friend and long time companion Tandy, our hero brings to you, in the comfort of your own home, such wonders as . . .

a fully commented source code listing at the end of each chapter along with (where appropriate) diagrams and descriptions of the data structures for the modules described, a discussion of the Model I system architecture as it pertains to operating system implementation, a general description of the Z-80 CPU register sets and interrupt modes, a description of the command line interpreter and the subroutines used by the TRSDOS commands module, the modules used by the file management system, the error message processor, the debug module available with TRSDOS, and much, much more.



Can you believe it? . . . Believe it!

Discover yours today!

TRSDOS 2.3 DECODED & Other Mysteries is available for $29.95 at computer stores, B. Dalton Booksellers and independent book dealers around the world. If your dealer is out of stock, order direct from IJG.

Include $4.00 for shipping and handling. Foreign residents add $11.00 plus purchase price. U.S. funds only please.

IJG, Inc. 1953 West 11th Street
Upland, California 91786
Phone: 714/946-5805

Helping you help yourself.

29

in length! It actually gives you the power of doing BLOCK moves of memory in a very simple, straightforward fashion. All of this from within BASIC, not machine language. I was starting to get really excited about Snapp-Ware.

The next thing that I tried was a new command called SYSTEM CLEAR. With the CLEAR command, you can change memory size more than once in your program. It lets the program allocate its own files and protects a block of high memory for machine code without the user having to bother with remembering how many files and what memory size the program uses! I have always thought it would be great to set number of files and memory size from within a BASIC program and I guess Snapp did, too!

All right, what other surprises do we have here? LBASIC comes with a CMD"O" command, like TRSDOS 1.3, which will sort a single variable array. As it stands, the CMD"O" sort works just fine, but a big problem with this type of sort is that you almost always need to sort more than one variable.

The Snapp sort takes care of this problem. It allows you to sort up to thirty-two arrays at the same time, with tagalong options, etc. It's superfast, powerful, and it doesn't have to be loaded into high memory to use it.

Another function I like is ETIM$. It calculates the difference between two times. A new command called SYSTEM"ERASE" removes any or all arrays from the array table, freeing space or allowing them to be redimensioned. The only other Model I/III TRS-80 BASIC I know of that has this feature is NEWDOS/80 version 2.0.

Functions MAX and MIN return the largest or smallest value from a user-supplied list. These functions can save a lot of IF...THEN...ELSE, <=<>=>, program statements.

The function FMT$ arranges data into a string variable. It's kind of like a super PRINT USING command. Snapp says this is really a powerful addition that, if experimented with, will add versatility to string handling in your programs. The manual supplies a good example of its use that prints zip codes in the new 12345-1234 format.

The FN HEX$ is nice, too. It converts a decimal into a hex string. I, personally, really like this one, because I can't convert number bases in my head that fast, and I get tired of looking at a paper chart. It works great inside your BASIC program, also. (Note: Logical Systems, Inc. now offers a filter package for LDOS which contains a super utility called CALC. It allows you to convert and do math with hex, dec, and bin number bases. It's fantastic.)

I am sure you have read advertisements about the new operating system NEWDOS/80 version 2.0 from Apparat, which has commands that allow you to SWAP variables, DELETE lines from a program and keep running, MERGE a new program block and keep running, etc. The Snapp built-in function package gives LDOS most of these features. It adds SYSTEM DELETE, SYSTEM"SWAP", which works like the NEWDOS/80 commands. The only thing I really miss from Snapp extended built-in functions is a dynamic MERGE command. With SYSTEM DELETE already present, the system really could use a SYSTEM MERGE command.

A new feature, that falls a little short to me, is the FN PK$/UPK$. It packs a string into a compressed form which saves computer memory and disk space, then allows you to decompress the string for printing, but it only works with upper case chr$. Not being able to work with lowercase greatly reduces FN PK$/UPK$ usefulness.

There are other functions like FN PW (password encoding), FN ID$ (gives the disk name), FN FILES (tells how many and what kind of files you have open). There is a SYSTEM RESTORE command that allows you to pick which line of data you want the restore to start at. This is super for TRSDOS BASIC, but LBASIC already has a restore line number feature. I don't really see the reason to duplicate it again. (I talked to Snapp about this duplication and it was left in for TRSDOS compatibility.) Overall, built-in functions is my favorite package, and I

recommend it as one of the most useful Snapp extension packages.

Extended BASIC

Another package, called Extended BASIC, adds commands that give the user a more versatile RENUM function. It allows you to move multiple or single lines from one place to another in your program. Of course, it allows other options (block moves, etc.) in renumbering your program, also.

Both NEWDOS/80 and DOSPLUS have a function called DI-DU, that will move single lines. Snapp adds this useful feature to LDOS with its RENUM RI command. Moving a single line is not as easy with Snapp's renum, but it gives you the option.

Extended BASIC also adds abbreviations for a whole list of commands like A-AUTO, C-CLS, K-KILL, M-MERGE, N-NEW, P-LLIST, L"-LOAD"prog", etc. These shorthand features will spoil you.

Now, I love this next one! If you accidentally NEW a program, you just enter U, and you've got it back. The U command, or one like it, should be in every BASIC. The only other TRS-80 DOS I know of that has a program renew feature is NEWDOS/80 version 2.0. It might save your sanity some morning at 3:00, when in a daze, you enter NEW mistakenly!

The Snapp cross reference function is also very versatile. By entering X with a period (to video), or comma (to printer), you will get a listing of where all of your program variables and/or line numbers are used. It can also give you single variable/line references of your choice.

The Z command is somewhat like X, but it gives you your variables *along with* their current values. This is fantastic when writing and debugging a program. It is a great help when you are trying to figure out what's happening in someone else's program too!

The FIND function is a string/keyword facility. It allows you to find strings like "ENTER NAME" and keywords like GOTO. It also allows you to find and replace all occurrences of a keyword, and replace it with another of your choice.

This is a very welcome feature, but I find the find and replace function not quite as useful as the DOSPLUS search and replace option. DOSPLUS allows you to search and replace just about anything in a BASIC program, including strings, keywords, even control codes. I miss this capability in the Snapp FIND function, but overall, the FIND feature gives you most things you need for everyday use.

XCOMPRESS allows you to compress spaces, remarks, etc., from your program. It also makes multiple statement lines from smaller lines, if you want. It's one of the best program compressors I've used, and like all of Snapp's features, it runs *fast*.

Reverse Compression

This is a good time to mention the Snapp Reverse Compression package, even though it is a separate package from the Extended BASIC package. The Snapp Reverse Compression package really complements the COMPRESS feature. It will decompress a compressed program into multiple lines, where possible, making easy-to-read statements with spaces added for clarity. It also adds indents to FOR... NEXT loops, etc.

There is another side effect to the reverse compression feature that might not be obvious. When program lines are packed to the limit, and you save the program to disk using the ,A option, you will get a "direct statement in file" error when you try to load the ASCII-saved file back into memory. The reverse compression option will save you from this problem (and it is a common problem), by unpacking your program lines into smaller lines that pose no problem to ASCII loading. Compiler users, who save the BASIC source program in ASCII, can now be saved a lot of headaches!

There are DOSs that already contain some of these features, or

variations, and $39.50 (not including the reverse compression package), to get them enhanced for LDOS, might be a little expensive for some people. It is a fine package, though, and might be exactly what you want and need to enhance your LDOS.

Extended File Mapping

The file mapping support automates a lot of tedious programming that we now have to go through to write random file programs. It's unbelievable. It gets rid of the FIELD, LSET statements, and the CVI,MKI-type functions. It also allows you to directly PUT and GET your program variables, doing away with having to use two variables for each data field. By defining your variables and their lengths in a string, you can PUT more than one data element to a file at one time.

The system does all of this automatically. The more field elements your program has, the more the Snapp File Mapping feature will show its usefulness. With a large file, you can do more in three statements using Snapp file mapping than it would take you in twenty or thirty statements, the old way.

The SYSTEM GET function works the same way, but as an extra bonus, it removes any unnecessary trailing blanks from a string. This really can give you a big savings in memory when filling arrays from a disk file.

As Snapp says in the manual, "Leave the driving to us." If you write a lot of LBASIC data file programs, I guarantee you will love it.

Extended BASIC Screen Mapping

The screen mapping function is really nice. It allows you to predefine a screen format for data entry, like the way Profile lets you draw and define your program's data entry form right on the video screen. With Snapp's Screen Mapping, you are furnished a program that allows you to define where on the video your entry fields will go, what they say, what type of variables, and their field lengths. Using commands like SYSTEM INIT, SYSTEM RECEIVE, SYSTEM SEND, you have an automated screen data-fetching program. It can take much less memory than if you wrote a routine in BASIC to handle your data entries the way that Snapp Screen Mapping does. You do have to protect a small portion of high memory with this function. You can also have as many different video forms saved on disk as you wish and call them in and out of your program when you wish.

I think that more user-defined options would be useful in this package. I didn't particularly like the * character for unused field slots (example: NAME **********), and wished that I had the option of changing it to an underline character (5FH), which, on my Model III, is a thin solid underline.

DOSPLUS 3.4 has a command called INPUT@, which gives a programmer some of the capabilities of Snapp's Screen Mapping, but it does use the underline instead of the * character. The DOSPLUS INPUT@ feature is easier to use, but it doesn't give you as much power and automation as Snapp's Screen Mapping. At the same time, INPUT @ doesn't take any high memory away, either.

It depends on how much automation you need in programming video forms to decide how useful this package would be to you. I find that DOSPLUS' INPUT@, in smaller programs, does a real good job. For big programming jobs, Snapp Screen Mapping is a joy to use. It makes it easy to have very professional-looking video input forms from within your BASIC program and, at the same time, automates your input statements. You don't usually see things like this on a TRS-80 Model III!

College Educated Garbage Collector

The string garbage collector is a very unique approach to the age-old

TRS-80 problem of string reorganizing. When string space is filled, the TRS-80 has to take time out to reorganize string space, get rid of unused strings, and move the strings that are in use into a contiguous format. This can be a long, frustrating process.

The string garbage collector gets rid of this problem. I tried it in a lot of different experiments, and it worked great. It takes 576 bytes of main memory and about five cylinders of diskette space, if needed. The process requires four bytes per active string as a work area, and will borrow this amount from RAM to do its reorganizing. If there is no room left in RAM, the garbage collector will temporarily "page out" to a disk file, part of the BASIC interpreter, and, if needed, part or all of your resident BASIC program, to make room for its work area. After it is finished working with the string garbage collection, the "paged out" information will be returned to its correct location for program continuation.

This, for most cases, gets rid of the string collection problem. It is definitely a unique approach to a long-time problem. If you have a program that you use all of the time and experience the computer string collection syndrome (lockup during sorting, etc.), this package would probably be the answer to your problem.

What Did I Think?

My strongest impression is of the ingenuity of Bob Snapp, the author of this fine package. I don't believe he knows what the words, "it can't be done" mean. When there's a better way, he takes it. His software has impressed me, in some ways, more than any I have used. I think anyone with LDOS should make a point to try the Snapp demo disk. You should take a couple of weeks for experimenting and find out which of these packages will be useful to you.

There are some greatly needed features in these packages that are added to LDOS, but there are also a lot of these same features already available when you buy DOSPLUS, NEWDOS/80 version 2.0, MULTIDOS, etc. If one of these DOSs contain most of the features that you miss from LDOS, it might be a good idea to buy that DOS. For LDOS BASIC programmers, this package turns LDOS into a Mercedes Benz. If you only need one or two of the packages, it doesn't cost that much. The product is fantastic. You may find that you will want one or more of the packages. For program development, these packages might be the best investment you ever made. ■

LSNAPP, by Bob Snapp, is sold by Snapp-Ware. The trial demo disk is $10, and prices vary from $19.50 to $39.50, as noted in the article. Also available for Model III TRSDOS.

Model II supervisor calls

Part I: A function grapher

Model II

T. R. Dettmann, Associate editor

A question I have frequently been asked by people who are trying to use their Model II is, "How do I get at the SVC's?" The second most frequent is, "What the #$/@ are the SVC's?"

The owner's manual describes SVC's in this way (page 4/9 of the TRSDOS section): "Supervisor Calls (SVCs) are Operating System routines available to any user program. The routines alter certain System functions and conditions; provide file access; perform I/O to the keyboard, Video Display, and Printer; and perform various computations." Starting from this point in the manual and working through from there, most people very quickly get discouraged.

First, the SVCs are described in terms familiar to machine language programmers and there is no indication that there may be a way to get at them from BASIC. Second, all the examples are for machine language implementations.

Because of this, most people have ignored them. But that isn't necessary. There are a lot of useful routines among the SVCs. If we just have a way to get at them from BASIC, we could do things like turn off the BREAK key, read disk ID's, display system error codes and messages, get special printer information, examine the directory of a disk, do specialized input and output, set scroll protect for the screen, or copy RAM to the screen (or screen to RAM). There's more that can be done, but most of the other system calls can be done as easily using BASIC commands.

This article shows you a few things that can be done with SVCs directly from BASIC without you having to be a machine language programmer. We'll demonstrate accessing the SVCs and give some direction on using them.

The demonstration illustrates the scroll protect function by displaying a graph of Sin(x) vs x on the screen, while continuously maintaining a heading on the top four lines. Unfortunately, the editor has told me that the typesetter doesn't have a scroll-protected font, so I can't demonstrate it for you in the magazine. In a second article, I will explain more about SVC access and lead you through building SVC calls to routines that will flip screens and access the Model II disk directory. In the process, we'll cover a little about machine language, but only enough to help the more advanced user get more from the SVC calls. Even without a knowledge of machine language techniques, anyone will be able to use the most important SVC calls without problems.

To start with, let's look into the SVCs themselves and see how we go about accessing them.

How Do We Get At SVCs?

In some respect, the fears most people have about getting to the SVCs are right. The only way to invoke SVC is to go to machine language. What isn't clear is that it is really very simple and you don't even have to know about machine language to do it.

The SVC calls that most people would like to use are all very simple because they don't require much from the user. The best example is the scroll SVC which allows you to protect a number of lines at the top of the screen from scrolling as you print on the screen.

Scroll protecting is such a useful feature that it's surprising that you find very few people using it. Oh, I've seen some programs that load machine language routines and include scroll protect in them, but very few that make it part of the BASIC program without the need for assembling machine language routines.

To see how it works, let's look at the instructions in the owner's manual (TRSDOS, page 4/44) and the first demonstration program.

The manual tells us that you can protect from zero to twenty-two lines at the top of the display from scrolling if you will just put the number of lines in the B register, put a 27 (the SVC code) in the A register, and execute a RST8. Simple, huh?

To a machine language programmer, this is a piece of cake, but if everyone is going to do this, we've got to make it easier than that. As it turns out, the steps are really very simple. In fact, they're exactly as stated above.

If you look at line 130 of the demonstration program, you will see a list of six numbers. This is the machine language code to execute the scroll SVC. In simple terms, the code can be interpreted this way:

Load the B register with 0	6	0
Load the A register with 27	62	27
Do a RST 8	207	

Return to the BASIC program 201

Congratulations, you're now a machine language programmer! Well, maybe not quite, but you have enough here to make use of something that can only be done from machine language.

To do a scroll protect, we need to take these numbers, each of which is supposed to go in one byte of memory (one memory address), place them so they fit together correctly, and then tell BASIC where to get them. That's the function of subroutine 1000.

In subroutine 1000, we take the numbers which exist in memory as integers and compact them so that they are stored as one byte per number. To accomplish the trick, a function we might not ordinarily think of comes to the rescue — CVI. CVI is explained in the section of the BASIC manual on random access files as a function that allows us to bring an integer back from a random access file.

When coming off random access files, an integer looks like a two-byte string. When we put it there, we encoded it with MKI$, which made it look that way (See Files and foibles if you want to know more about random access files). The important thing is that if we give CVI a two-byte string, it will make that string look like an integer.

In line 1010, we take the machine language numbers (stored in array CD) and combine them into two-byte strings which we store in the integer array SV. (It's an integer because of the DEFINT at the beginning of the program, but you could have used a % to make it clearer.) This puts all six bytes of the machine language program together for us in the correct way so that it forms a legal program. Now we have to let BASIC know where it is.

Line 1020 points BASIC to our program and then executes it. First, though, it has to set the variable Y equal to zero to make sure BASIC has created it. We'll get back to why this happens in a moment.

After Y is created, we use the DEFUSR command to tell BASIC that the machine language program we want is located starting at the zero position of the array CV. This works because integer arrays are always stored going from low to high memory, two bytes for each integer in array sequence from zero through the dimensioned index.

Once BASIC knows where to find it, we execute the USR command to scroll protect the screen. If we hadn't created the variable Y at the beginning of the line, it would have been created just before the USR call executed. This would have caused BASIC to move the array with the service routine after we had pointed to it! Our USR call would have been sent somewhere else. Oops!

As we've created the routine, though, we have zero loaded into the B register, so we are asking for zero lines to be scroll protected. That's not very useful. To get more lines protected, we simply set CD(2) equal to the number of lines we want protected from zero to twenty-two (see line 260 in the program) and then GOSUB1600 to execute the scroll protect.

Now that it has been explained, it really is simple, isn't it? Even if it isn't, if you copy the routines carefully and study the example, you should be able to easily add

scroll protect to your own programs. Try the scroll protect example in your own computer. It really works.

Program Listing for SVCs

```
10 REM******************
20 REM
30 REM SCROLL PROTECT DEMO
40 REM (C) 1982 BY TERRY R. DETTMANN
50 REM
60 REM VERSION 0.009/82
70 REM FILENAME: SCROLL/BAS
80 REM
90 REM******************
95 REM DEFINE C TO BE AN INTEGER FOR THE CD ARRAY
AND S FOR SV
100 CLEAR 1000: DEFINTC,S
105 REM SV WILL HOLD THE ACTUAL SVC CALL CODE
106 REM CD WILL HOLD THE BYTE BY BYTE VALUES OF THE
CODE
110 DIM SV(2),CD(6)
115 REM WE READ THE BYTE BY BYTE MACHINE LANGUAGE
CODE IN HERE
120 FOR I=1 TO 6: READ CD(I): NEXT I
130 DATA 6, 0, 62, 27, 207, 201
135 REM A USEFUL HEADING FUNCTION, THE CHR$(2)
TURNS OFF THE CURSOR
140 DEFFNHDR$(X$)= STRING$((78-LEN(X$))/2,150)+"
"+X$+" "+ STRING$((77-LEN(X$))/2,150)+ CHR$(2)
200 REM - - - - - - - - - - - MAIN LOOP - - - - -
- - - - - - - - - - - - - - - - - -
205 REM THE LINES THROUGH 250 WILL BE THE HEADING
FOR THE
206 REM SAMPLE PLOT (SIN(X) VS. X)
210 CLS: PRINT FNHDR$("SCROLL PROTECT DEMO")
220 PRINT TAB(38)"SIN(X)"
230 PRINTTAB(13)"-1.0 0.0 +1.0"
240 PRINT"X (DEG)";
250 PRINT TAB(15)"!----+----!----+----!----
+----!----+----!----+----!"
255 REM SET THE NUMBER OF LINES TO PROTECT TO 4
AND THEN CALL
256 REM THE PROTECTION SUBROUTINE
260 CD(2)=4: GOSUB 1000
265 REM NOW LET'S GENERATE A GRAPH OF SIN(X) VS X
270 FOR I=0 TO 500
275 REM FIGURE X IN RADIANS (180 DEGREES = PI
RADIANS)
280 X=3.14159*I/20
285 REM COMPUTE Y AND SCALE TO A RANGE OF 0 TO
50
290 Y = 25 + 25*SIN(X)
295 REM ROUND OFF Y AND ADD 1 TO GET RANGE 1-51
300 Y = INT(Y+.5)+1
305 REM COMPUTE THE DEGREES CORRESPONDING TO
X
310 DG = X*180/3.14159: IF DG>=360 THEN DG=DG-
INT(DG/360)*360
315 REM EVERY 5 LINES (45 DEGREES), PRINT THE X VALUE
320 IF IMOD5=0 THEN PRINTUSING"###.##";DG;
325 REM EVERY 5 LINES, PUT A LINE ACROSS THE GRAPH,
OTHERWISE BLANKS
330 IF IMOD5=0 THEN GR$= STRING$(51,"-") ELSE GR$=
STRING$(51," ")
335 REM PLACE VERTICAL LINES IN THE GRAPH
340 MID$(GR$,1,1)="!": MID$(GR$,51,1)="!":
MID$(GR$,26,1)="!"
345 REM PLACE THE SIN(X) VALUE IN THE GRAPH
350 MID$(GR$,Y,1)="*"
355 REM PRINT THE GRAPH LINE
360 PRINT TAB(15)GR$
370 NEXT I
375 REM THE CLS IS TO END SCROLL PROTECT
380 CLS: END
1000 REM - - - - - - - - - - - SCROLL PROTECT -
- - - - - - - - - - - - - - - - -
1005 REM FIRST WE MUST SET UP THE SVC FOR CALL BY
COMBINING
1006 REM THE BYTES INTO A MACHINE LANGUAGE
PROGRAM
1010 J=0: FOR I=1 TO 6 STEP2: CV(J) =
CVI(CHR$(CD(I)) + CHR$(CD(I+1))): J=J+1: NEXT I
1015 REM NOW WE SET UP THE USR CALL, THIS LINE MUST
REMAIN
1016 REM AS IS, CHANGING ANY PART OF IT WILL SEND
THE USR CALL
1017 REM TO AN UNDETERMINED PLACE
1020 Y=0: DEFUSR3 = VARPTR(CV(0)): Y = USR3(0)
1030 RETURN ■
```

ACCEL3/4

NEW

$99.95

+ $1.50 Shipping, CA add 6%
specify Mod I or III

Space Frontier

ACCEL3 compiles *big* programs, 30K +. This is by design; subset compilation and use of ROM routines keeps code growth low (and ensures the highest compatibility with the BASIC source program). Also, the NOEXPR option can limit compilation to the program flow operations (GOTO, GOSUB, RETURN, FOR-NEXT, IF, ON) over part of your program, or in toto. 5.5K, all DOS.

EDIT

Full-Screen Editor for TRS-80 Basic, Mod I/III

You need better-than-wordprocessor control when you write and edit BASIC programs. That's why EDIT has lots of better-than-wordprocessor features.

- Full-floating cursor with autorepeat (including function keys).
- Navigation by cursor-controlled scrolling, by line number or by program content.
- Overtype, Delete or Insert characters of text (including line numbers).
- Join or Split lines. Copy, Delete, Insert, Move, Position or Replicate lines.
- Copy, Delete, Move, Position or Replicate blocks of lines.
- Find and/or Replace occurrences of strings.
- Over 30 commands and functions, supports lower-case.
- 3.3K, relocatable (all DOS), also includes TRSDOS, LDOS overlay version.

You owe it to your programming.

$40

+ $2 Shipping, CA add 6%
specify Mod I or III

NEW

ACCEL4, overlay version of ACCEL3, works with TRSDOS, LDOS *only*, no user memory requirements. Anyone for 40K + compilations?

You owe it to your programs.

ALGORIX
(415) 387-3131
Box 11721 San Francisco CA 94101

NEW

ENB Southern Software's NEW relational database manager. Integrated data dictionary, recursive command structure, more. **$140**

Developed in England by Southern Software. TRS-80, TRSDOS, tm Radio Shack LDOS, tm Logical Systems

70

File copy utility

A COPY command on a single drive system

Model I/III, 32K+ with disk drive

Charles Edwards, Jacksonville, FL

TRSDOS 2.3 users with only one disk drive quickly discover one of the limitations of their system: the lack of a single-drive copy utility. TRSDOS does have a COPY command, but at least two drives are required in order to use it. Radio Shack's two utilities, GETTAPE/BAS and GETDISK/BAS, are not much help. For one thing, it is a nuisance to have to dump a file to tape before you can load it to a different disk. Additionally, these programs do not preserve the end of file pointer, which can be disastrous for data files.

Well, do not fear, DIP is here. No, DIP is not that kid in ninth grade who sat in the back of the room studying the zipper on his pencil case. DIP stands for Disk Interchange Program. It is a BASIC program which will copy *any* file from one diskette to another and what's more, it preserves the end of file pointer. In order to learn how DIP accomplishes this, let's examine how BASIC allocates memory for its files.

After you respond to the "HOW MANY FILES?" prompt, BASIC sets aside 288 bytes of RAM for each file. The first 32 bytes form the Device Control Block (DCB) and the next 256 comprise the file buffer. The layout of the DCB is shown below.

Device Control Block Layout

Bytes	Purpose
0-2	Reserved
3-4	Address of file buffer
5	Location of record delimiter in current record
6	Disk drive number
7	Reserved
8	EOF — last valid byte in the last record
9	Logical record length — usually 256
10-11	Next record number to read or write
12-13	ERN — last valid record in the file
14-31	Reserved

For the purpose of this discussion, the only fields we are concerned about are ERN and EOF. ERN (Ending Record Number — bytes 12 and 13) is a two-byte field which contains the last record number in the file. EOF (End Of File — byte 8) is a one-byte field which contains the number of the last valid byte in the record pointed to by the ERN. In order to preserve the end of file pointer, all that is necessary is to insure that these three bytes are the same in the new file as they are in the old one.

But, "How do we find the DCB?" you ask. It couldn't be simpler. Listing 1 is the complete source for DIP. At line 120, we FIELD the variable A$ to point to the first 128 bytes of the buffer. Like any string variable, A$ is composed of three bytes: a one-byte length field and a two-byte address. In this case the two-byte address points to the beginning of the buffer. So, we just take this address, subtract 32 (the size of the DCB) from it, and voila, we have the DCB address.

That's enough of the technical details. In order to run DIP: enter BASIC, respond to the "HOW MANY FILES?" question with 1, and enter RUN "DIP/BAS." DIP will ask for the names of the source and destination files and will prompt you when to exchange diskettes. DIP was written for a 48K system, so make the changes shown below for 32K.

```
40 CLEAR 16384: DIMA$(50), B$(50): DEFINTA-Z

60 LW=0: J=0: FORI=1TOL: J=J+1: IF J=51 THEN
GOSUB 100
```

There is one interesting thing to note about this program. At line 40 I CLEAR 32767 bytes of string space. I had first tried CLEARing 37000 bytes but got a run-time error. So I thought that perhaps numbers greater than 32767 have to be represented in negative form, as in USR calls. But I tried CLEARing -31464 bytes (37000-65536) to no avail. Apparently 32767 is the upper limit. I have never seen this fact documented anywhere, so perhaps I am the first person dumb enough to ever try CLEARing that much string space.

DIP has been extremely useful to me as a single drive owner and I hope you find it as valuable as I do.

Program Listing for Disk Interchange Program

```
40 CLEAR 32767: DIM A$(100), B$(100): DEFINT A-Z
45 CLS
50 LINEINPUT "Enter source file name: ";F1$ :LINEINPUT
"Enter destination file name: ";F2$: L=1: I=0: GOSUB 110:
EF=PEEK (DC+8): E1=PEEK (DC+12): E2=PEEK (DC+13):
L=LOF(1): IF L=0 THEN CLOSE 1: PRINT "File does not
exist or is empty": KILL F1$: GOTO 50
55 GOSUB 60: GOTO 45
60 LW=0: J=0: FOR I=1 TO L: J=J+1: IF J=101 THEN
GOSUB 100
70 GET 1, I: A$(J)=A$: B$(J)=B$: NEXT
100 CLOSE 1: LINEINPUT "Insert destination disk and
<ENTER>";A$: OPEN "R",1,F2$: GOSUB 120: FOR J=LW+1
TO I-1: LSETA$=A$(J-LW): LSETB$=B$(J-LW): PUT 1,J: GET
1,J: IFB$<>B$(J-LW) OR A$<>A$(J-LW) THEN PRINT
"Error in record #";J: STOP ELSE NEXT: LW=J-1: CLOSE 1:
J=1
110 IF I>L THEN OPEN "R" ,1,F2$: GOSUB 120: POKE
DC+8,EF: POKE DC+12,E1: POKE DC+13,E2: CLOSE 1:
LINEINPUT "Copy complete, hit <ENTER> to continue"
;A$: RETURN ELSE LINEINPUT "Insert source disk and
<ENTER>";A$: OPEN "R" ,1,F1$: GOSUB 120: RETURN
120 FIELD 1,128ASA$, 128ASB$: DC=PEEK (VARPTR(A$)+1)
+256* PEEK (VARPTR(A$)+2) -32: RETURN ■
```

Short cut

PERT analyze your way to work

For all models

Arnold Maddox, St. Louis, MO

If you are like me, you want to stay in bed in the morning until the last second and still try to make it to work on time. There are several things I can do. I can skip breakfast and save fifteen minutes, shave and put on my tie in the car, park in someone's reserved space, or I can plan my trip to work better. The latter seems to be the best alternative.

Like most people, I can take several routes to work. For several weeks, I timed the various segments of these routes. The object is to put the right combination of segments together to minimize the total travel time. The program in Listing 1 does the job nicely. It is an adaption of the PERT *(Program Evaluation and Review Technique —Ed.)* management technique.

The data is read as follows. First read the number of intersections on the various routes (line 120). Next, number these intersections and read the segments in three parts as follows: Intersection FROM, Intersection TO, Travel time. See lines 170 to 230.

The program counts these segments. You will have to put three "dummy" paths at the end of the data to stop the path reading. The program does the rest very well thank you.

The map in Figure 1 is represented by the data on lines 490 to 530. The results are shown in Figure 2. However, there is a caution. The various paths must be sorted by FROM intersection first and then, within each of these, sorted by TO intersections.

In collecting your data, you may note that some intersections have stop lights or signs in one or both directions. A left turn may be present which might slow down your trip. We can account for these by introducing extra dummy intersections. In Figure 3 there is a stop sign on the path from D - J - M and not one on the path through E - J - P.

We can introduce a new intersection called Z and make its time to intersection J equal to the estimated time the stop sign delays us. The resulting map will be as seen in Figure 4.

You can easily enhance this program by reading in intersection names like "Main and 12th" along with your times and have the program make an actual intersection list for you.

One final note. You may have to take your time measurements over several different days of the week. For some unknown reason, I've found that Tuesday is a "light" day for traffic.

Figure 1

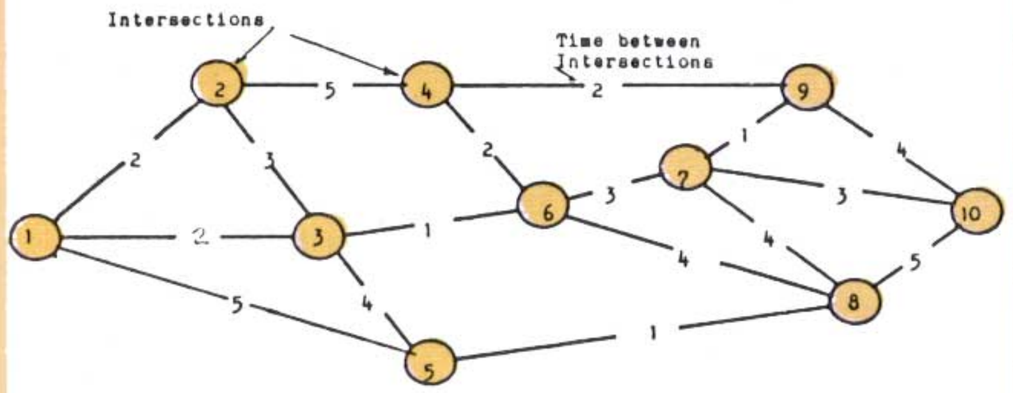


Figure 2

```
No. INTERSECTIONS= 10
No. PATHS= 17
TOTAL TIME= 9
1---> 3 ---> 6 ---> 7 ---> 10
```

Figure 3

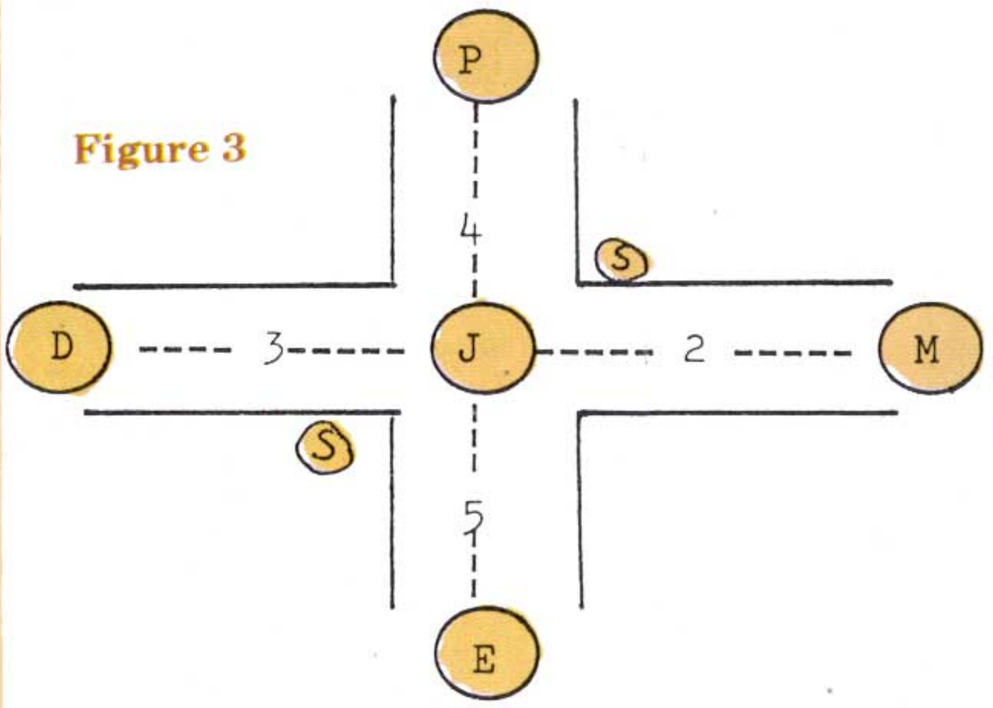
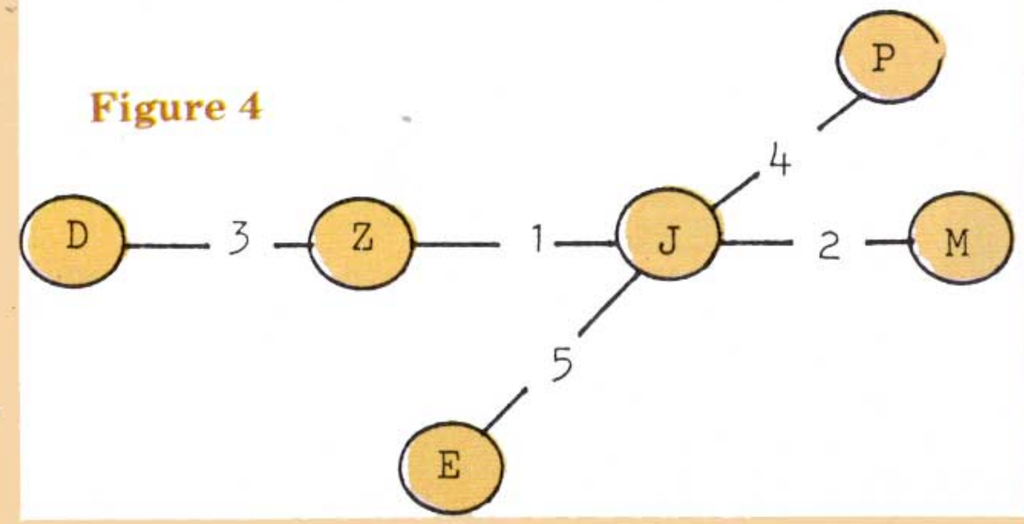


Figure 4



Program Listing for Short Cut

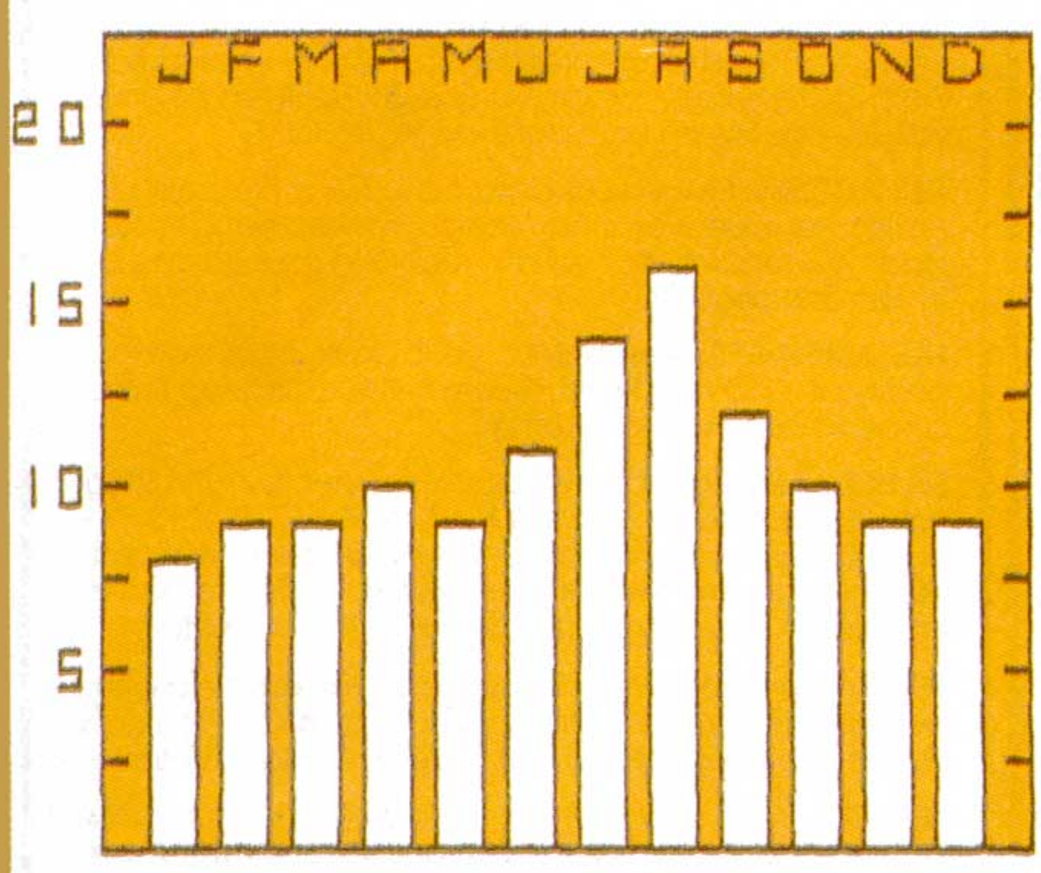
```
5 REM ******************************
10 REM TO RUN THIS PROGRAM ON A COLOR
20 REM COMPUTER CHANGE THE LPRINT TO
30 REM PRINT#-2, IN LINES 130, 240,
40 REM 330, 430, 460
50 REM IF YOU DON'T HAVE A PRINTER
60 REM THEN CHANGE THE LPRINTS TO
70 REM PRINT STATEMENTS.
80 REM ******************************
100 '***** SHORT *****
110 CLS
120 READ NI 'NUMBER OF INTERSECTIONS
130 LPRINT"NO. INTERSECTIONS = ";NI
140 DIM TE(NI),TL(NI),SK(NI) 'EARLY, LATE, AND SLACK
TIME
150 SP=5*NI 'SPACE FOR PATHS
160 DIM I1(SP),I2(SP),CT(SP) 'INTERSECTIONS
FROM-TO-TIME
170 READ F,T,C 'FROM, TO, TIME
180 IF F>NI THEN 240 'ARE WE THROUGH READING?
190 NP=NP+1 'NUMBER OF PATHS
200 I1(NP)=F 'FROM F
210 I2(NP)=T ' TO T
220 CT(NP)=C 'IN TIME C
230 GOTO 170 'READ SOME MORE
240 LPRINT"NO. PATHS = ";NP
250 FOR I=2 TO NI 'SET INITIAL
260 TE(I)=9999 'EARLY
270 NEXT I ' TIMES
280 FOR J=1 TO NP 'FIND
290 TS=TE(I1(J))+CT(J) 'SHORTEST
300 IF TS>TE(I2(J)) THEN 320
310 TE(I2(J))=TS 'THROUGH
320 NEXT J 'NETWORK
330 LPRINT "TOTAL TIME = ";TE(NI)
340 TL(NI)=TE(NI)
350 FOR J=NP TO 1 STEP -1 'DO
360 TS=TL(I2(J))-CT(J) 'THE
370 IF TS<TL(I1(J)) THEN 390 'REVERSE
380 TL(I1(J))=TS 'PATHS
390 NEXT J
400 FOR I=1 TO NI 'COMPUTE
410 SK(I)=TE(I)-TL(I) 'THE
420 NEXT I 'DIFFERENCES
430 LPRINT"1";
440 FOR I=2 TO NI 'PRINT
450 IF SK(I)<>0 THEN 470 'FINAL
460 LPRINT "--->";I; 'ROUTE
470 NEXT I
480 END
490 DATA 10: 'NUMBER OF INTERSECTIONS
500 DATA 1, 2, 2, 1, 3, 2, 1, 4, 5, 2, 3, 3, 2, 4, 5, 3, 5, 4, 3, 6,
1
510 DATA 4, 6, 2, 4, 9, 2, 5, 8, 1, 6, 7, 3, 6, 8, 4, 7, 8, 4, 7, 9,
1
520 DATA 7, 10, 3, 8, 10, 5, 9, 10, 4
530 DATA 99, 99, 99: 'END OF DATA ■
```

Bar graphs

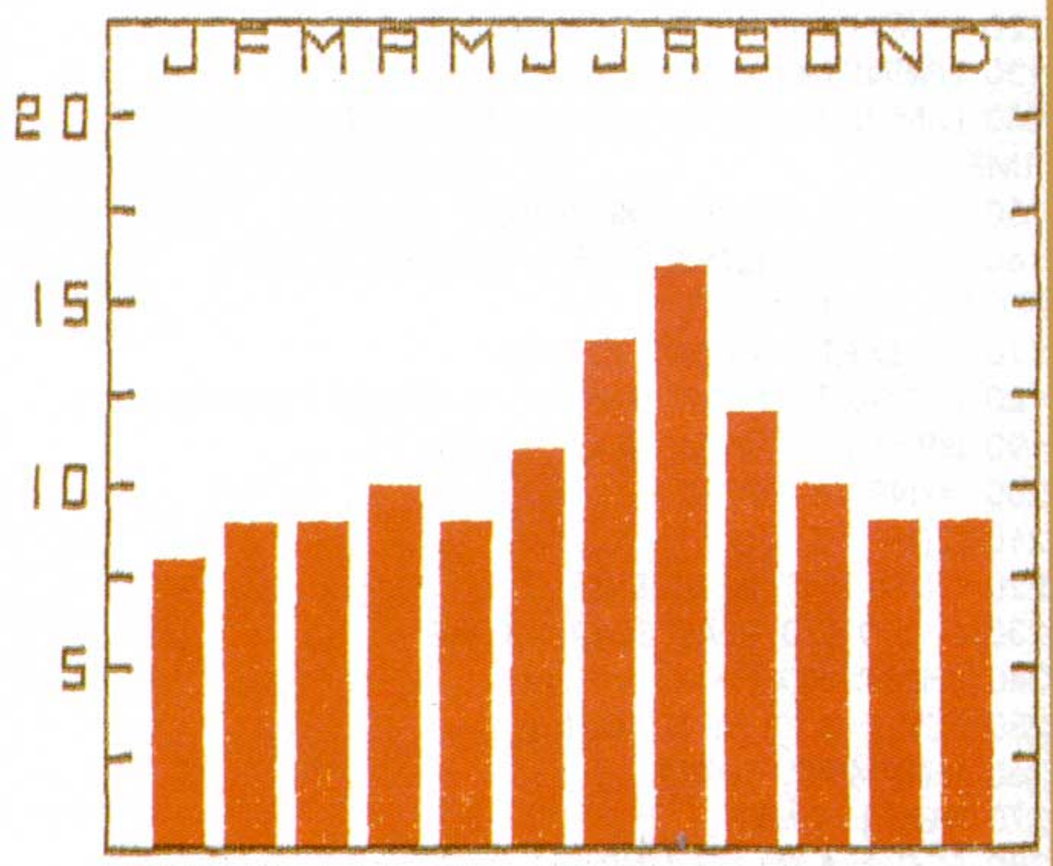
Creating them on your Color Computer

Color Computer

Richard Scales, Wabash, IN



1981 Water Bills ($)
Figure 1



1981 Water Bills ($)
Figure 2

The Color Computer's graphic capabilities are well documented in this and other publications. Unfortunately, most of the articles deal with graphics for games. Can it be that I am the only one using the CC in a business environment? You should have seen the eyebrows go up when I purchased the CC and Line Printer VII for my office!

Computers provide our firm with copious amounts of statistical data. So much, in fact, that a lot is wasted due to the volume of information available. I was looking for a way to condense and display summaries in a manner that shows trends and variations from the norm. The bar graph is a popular way to accomplish this.

The illustration on the cover of *80-U.S. Journal* (September 1982) showed exactly what I wanted. You can imagine how my enthusiasm waned when I found that the "Multi-Pen Plotter" cost about $2000. Unable to find a program that produces Bar Graphs and permits a printed output, I decided to write my own. The program listing that follows will produce up to 12 "bars," each one representing one month of data. Extended BASIC is required and you must use a "Screen Print" program/routine to get a hard copy.

Figure 1 shows the program's output, in this case Water Bills. Note that the bars are not "filled-in." This permits coloring of the printed copy using a straight-edge and marking pens.

The bars are simply long skinny boxes of variable height. Consider program line 560, which draws the "February Box":

```
560 LINE(50,D)-(60,180),PSET,B
```

The figures inside the brackets specify screen coordinates (across,down) of opposite corners of the box. The B at the end of the line causes a box to be drawn instead of a line. The variable D is assigned a value in line 230. If you want solid bars, change the B to BF and the computer will fill-in the boxes with the foreground color. The printout will have solid black bars. See Figure 2.

PMODE 4 (Line 190) causes High Resolution Graphics to be produced. Changing PMODE 3 will give less attractive letters and numbers, but will permit multi-colored bars on the screen display. Use B, not BF, when drawing the boxes. Now you can add a series of

Innovation...

MULTIDOS...C.E.C.'s deluxe disk operating system for the TRS 80 Model I and Model III. Has forty library commands, twelve utilities, a MIGHTY-MULTI mini-dos, supports three densities (MODEL I version), read/write to/from practically any DOS format and is compatible with most double density boards. Supports clock speeds up to 5.32 MHz. MULTIDOS has, standard, two super BASICs, including BBASIC, the popular "BOSS" single step king and a host of Basic program editing utilities. Versatile, user friendly, fast, efficient. ***Get into the world of DOS FANTASTIC!...*$99.95**

Z'DOS...economical disk operating system for TRS 80 Model I and Model III. Inexpensive and feature packed. Forty library commands, supports three densities (Model I version), read/write compatible with most DOS formats and double density boards. Has SUPERBASIC II with 41K of free memory, cross reference, renumber and global editing. ***The fastest DOS in town!...*$39.95**

EBASIC...extended disk Basic. Adds high-speed graphics, versatile special functions, twenty-four types of formatted keyboard INPUT, program line labeling, array operations, and multi-key SORT of unlimited arrays. All in low RAM along with Disk Basic and still leaves over 38K of free memory. ***EBASIC just...*$29.95**

(EBASIC requires Z'DOS or MULTIDOS Model I version 1.4 or greater, or MULTIDOS Model III version 1.3 or greater.)

BOSS / RENUM90... Best operated single stepper. This machine language BASIC program debugging utility allows you to single step BASIC programs, trace in a controlled area on the display, trace to the line printer, save the screen and display variables, control program execution speed, save Basic programs to high RAM and insert breakpoints into your Basic program. RENUM90 allows you to renumber all or part of your Basic program. **On tape or disk, BOSS/RENUM90 for $24.95 or BOSS only, $15.95**

— Available from the System Innovators —

COSMOPOLITAN ELECTRONICS CORPORATION
5700 PLYMOUTH ROAD
ANN ARBOR, MICHIGAN 48105

Telephone: (313) 668-6660

Add $3.00 shipping/handling. Foreign orders add $10.00. Michigan residents add 4% sales tax. C.O.D. add $1.50. Model I orders specify single, double or P density. Personal checks take 2 weeks to clear. VISA & Master Card accepted.

AUTHORIZED DEALERS

ABC SALES
13349 Michigan Avenue
Dearborn, MI 48126
(313) 581-2896

COMPUTER SHACK
1691 Eason
Pontiac, MI 48054
(313) 673-2224
1-800-392-8881

POWER SOFT
11500 Stemmons Expressway
Dallas, TX 75229
(214) 484-2976
1-800-527-7432

BYTES & NAILS
5110 6th Avenue
Sioux City, IA 51106
(712) 274-2348

SIMUTEK
Computer Prod., Inc.
4877 East Speedway
Tucson, AZ 85712
(602) 323-9391
1-800-528-1149

MICRO-PROG-80
P.O. Box 606
Sunnymead, CA 92388
(714) 653-9429

37

TRS-80 is a registered trademark of Tandy Corp.

Bar graphs

PAINT statements for an attractive display. You will have to add several lines as follows:

```
535 PAINT(40,179),1,1

565 PAINT(55,179),1,1

595 PAINT(70,179),1,1

625 PAINT(85,179),1,1

655 PAINT(100,179),1,1

685 PAINT(115,179),1,1

715 PAINT(130,179),1,1

745 PAINT(145,179),1,1

775 PAINT(160,179),1,1

805 PAINT(175,179),1,1

835 PAINT(190,179),1,1

865 PAINT(205,179),1,1
```

The PAINT statements instruct the computer to begin "painting" at screen locations (across,down) and to paint with color #1 (green) and to stop painting at border color #1. Experiment with the "1,1" combinations until you get just what you want. Note that the painting begins at (down 179). This is just above the baseline (180) and could cause a problem if you enter a *very* small number during data entry (program lines 220-430). Try entering .001 or smaller to see what happens.

No provisions were included to label the printer output. Normally, I print the graph and then boot-up Color Scripsit so a short description can be added. Scripsit will center your label automatically and avoid having to decide how far to tab when using PRINT#-2 statements.

Several remarks have been included in the program to help you find your way around the listing. Hopefully, you will find lots of things to tinker around with and improvements to make. Perhaps this program will inspire you to create (and share) more "business graphics" programs. One improvement that comes to mind would be to permit changing the X-Y labels through input statements. The alphabet (and numbers) could be put into strings and then add a labeling routine to the program. Most of my Bar Graph needs are met by the program as it is.

Program Listing for BAR GRAPH

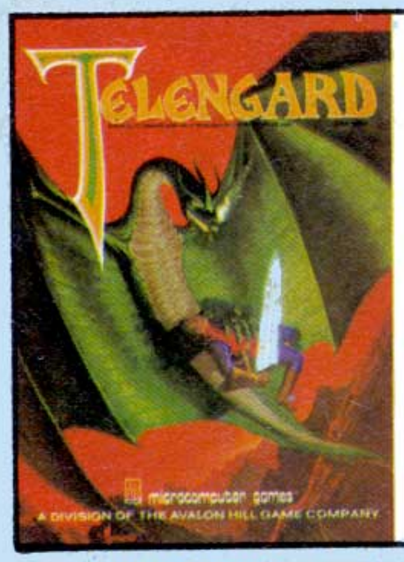
```
10 CLS
20 PRINT @203,"BAR GRAPH"
30 FOR Z=1 TO 1000: NEXT Z: CLS
40 PRINT:PRINT
50 PRINT" YOU WILL BE PROMPTED TO ENTER"
60 PRINT" VALUES FOR EACH MONTH OF THE"
```

```
70 PRINT" YEAR. YOU MUST ENTER A NUMBER"
80 PRINT" BETWEEN 0 AND 20.IF YOU DON'T"
90 PRINT" HAVE A FULL YEAR OF DATA,THEN"
100 PRINT" ENTER A ZERO FOR THE FIRST"
110 PRINT" MONTH WITH NO DATA. IF YOU"
120 PRINT" WISH TO SKIP A MONTH,ENTER-1"
130 PRINT" FOR THE MONTH TO SKIP.REMEMBER"
140 PRINT" DATA MUST BE BETWEEN 0 AND 20"
150 PRINT: PRINT
160 PRINT" <PRESS ANY KEY TO CONTINUE>"
170 Z$=INKEY$: IF Z$="" THEN 170
180 CLS
190 PMODE 4,1
200 PCLS
210 PRINT: PRINT:' DATA INPUT
220 INPUT "ENTER JAN VALUE";A: B=180-8*A
230 INPUT "ENTER FEB VALUE";C: D=180-8*C
240 IF C=0 THEN GOTO 440
250 INPUT "ENTER MAR VALUE";E: F=180-8*E
260 IF E=0 THEN GOTO 440
270 INPUT "ENTER APR VALUE";G: H=180-8*G
280 IF G=0 THEN GOTO 440
290 INPUT "ENTER MAY VALUE";I: J=180-8*I
300 IF I=0 THEN GOTO 440
310 INPUT "ENTER JUN VALUE";K: L=180-8*K
320 IF K=0 THEN GOTO 440
330 INPUT "ENTER JUL VALUE";M: N=180-8*M
340 IF M=0 THEN GOTO 440
350 INPUT "ENTER AUG VALUE";O: P=180-8*O
360 IF O=0 THEN GOTO 440
370 INPUT "ENTER SEP VALUE";Q: R=180-8*Q
380 IF Q=0 THEN GOTO 440
390 INPUT "ENTER OCT VALUE";S: T=180-8*S
400 IF S=0 THEN GOTO 440
410 INPUT "ENTER NOV VALUE";U: V=180-8*U
420 IF U=0 THEN GOTO 440
430 INPUT "ENTER DEC VALUE";W: X=180-8*W
440 SCREEN 1,1
450 ' DRAW FRAME
460 LINE(25,0)-(220,180),PSET,B
470 GOSUB 880' LEFT CALIBRATION
480 GOSUB 1190'Y-AXIS VALUES
490 GOSUB 1060'LTRS ACROSS TOP
500 GOSUB 970'RT CALIBRATION
510 'DRAW THE BARS
520 IF A<0 OR A>20 THEN 540
530 LINE(35,B)-(45,180),PSET,B
540 IF C<0 OR C>20 THEN 570
550 IF C=0 THEN GOTO 870
560 LINE(50,D)-(60,180),PSET,B
570 IF E<0 OR E>20 THEN 600
580 IF E=0 THEN GOTO 870
590 LINE(65,F)-(75,180),PSET,B
600 IF G<0 OR G>20 THEN 630
610 IF G=0 THEN GOTO 870
620 LINE(80,H)-(90,180),PSET,B
630 IF I<0 OR I>20 THEN 660
640 IF I=0 THEN GOTO 870
650 LINE(95,J)-(105,180),PSET,B
660 IF K<0 OR K>20 THEN 690
670 IF K=0 THEN GOTO 870
680 LINE(110,L)-(120,180),PSET,B
```

```
690 IF M<0 OR M>20 THEN 720
700 IF M=0 THEN GOTO 870
710 LINE(125,N)-(135,180),PSET,B
720 IF O<0 OR O>20 THEN 750
730 IF O=0 THEN GOTO 870
740 LINE(140,P)-(150,180),PSET,B
750 IF Q<0 OR Q>20 THEN 780
760 IF Q=0 THEN GOTO 870
770 LINE(155,R)-(165,180),PSET,B
780 IF S<0 OR S>20 THEN 810
790 IF S=0 THEN GOTO 870
800 LINE(170,T)-(180,180),PSET,B
810 IF U<0 OR U>20 THEN 840
820 IF U=0 THEN GOTO 870
830 LINE(185,V)-(195,180),PSET,B
840 IF W<0 OR W>20 THEN 870
850 IF W=0 THEN GOTO 870
860 LINE(200,X)-(210,180),PSET,B
870 GOTO 870
880 LINE(25,20)-(30,20),PSET
890 LINE(25,40)-(30,40),PSET
900 LINE(25,60)-(30,60),PSET
910 LINE(25,80)-(30,80),PSET
920 LINE(25,100)-(30,100),PSET
930 LINE(25,120)-(30,120),PSET
940 LINE(25,140)-(30,140),PSET
950 LINE(25,160)-(30,160),PSET
960 RETURN
970 LINE(215,20)-(220,20),PSET
980 LINE(215,40)-(220,40),PSET
990 LINE(215,60)-(220,60),PSET
1000 LINE(215,80)-(220,80),PSET
1010 LINE(215,100)-(220,100),PSET
1020 LINE(215,120)-(220,120),PSET
1030 LINE(215,140)-(220,140),PSET
1040 LINE(215,160)-(220,160),PSET
1050 RETURN
1060 DRAW"BM43,2;D8L6U2":'J
1070 DRAW"BM58,2;L6D4R6L6D4":'F
1080 DRAW"BM66,10;U8F4E4D8":'M
1090 DRAW"BM82,10;U8R7D8U4L7"'A
1100 DRAW"BM97,10;U8F4E4D8":'M
1110 DRAW"BM118,2;D8L6U2":'J
1120 DRAW"BM133,2;D8L6U2":'J
1130 DRAW"BM142,10;U8R6D8U4L6"'A
1140 DRAW"BM163,4;U2L6D4R6D4L6U2":'S
1150 DRAW"BM178,2;D8L6U8R6":'O
1160 DRAW"BM187,10;U8F8U8":'N
1170 DRAW"BM202,2;R5F2D4G2L5U7":'D
1180 RETURN
1190 DRAW"BM20,136;L4D4R4D4L4":' 5
1200 DRAW"BM10,96;D8":'1
1210 DRAW"BM20,96;L4D8R4U8":'0
1220 DRAW"BM10,56;D8":'1
1230 DRAW"BM20,56;L4D4R4D4L4":'5
1240 DRAW"BM6,18;U2R4D4L4D4R4":' 2
1250 DRAW"BM20,16;L4D8R4U8":'0
1260 RETURN ■
```

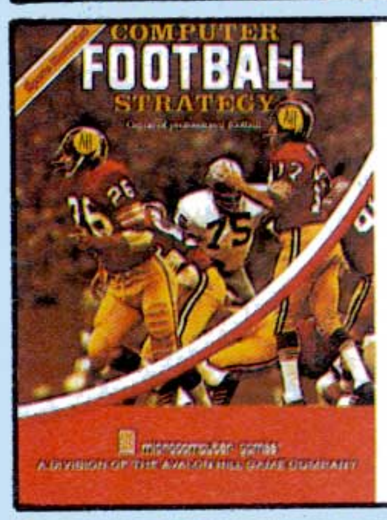
The Thinking Man's Gamemaker

WARNING: Do not buy Avalon Hill Microcomputer Games unless you are above average!



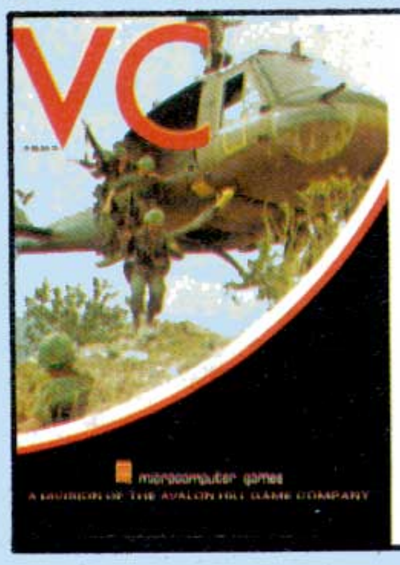
TELENGARD – Microcomputer Dungeon Adventure game. In the mysterious underworld of TELENGARD there are fifty levels of ever-more complex mazes for mighty adventurers to explore. Various means of survival tactics are at the adventurer's disposal and all of the necessary ingredients have been incorporated into this solitaire **real time fantasy and role-playing game.**

G.F.S. SORCERESS – The first **science-fiction adventure game** of the continuing saga of Joe Justin and Selena Sakarov aboard the Galactic Federation Starship Sorceress. In the game, the player takes the part of Joe Justin as he attempts to clear himself of a false charge of mutiny. Beautiful full-color manuals are included giving the player useful clues in his or her attempt to prove Joe Justin's innocence.

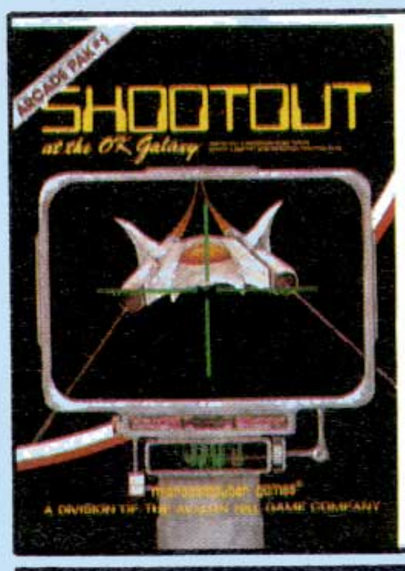


COMPUTER FOOTBALL STRATEGY – Thrilling computer version of Avalon Hill's famous board game. Based on the award-winning Sports Illustrated game of professional football, **Computer Football Strategy** forces the player to constantly make the right decisions about his team's offensive and defensive formations. Match wits against the computer or against a live opponent.

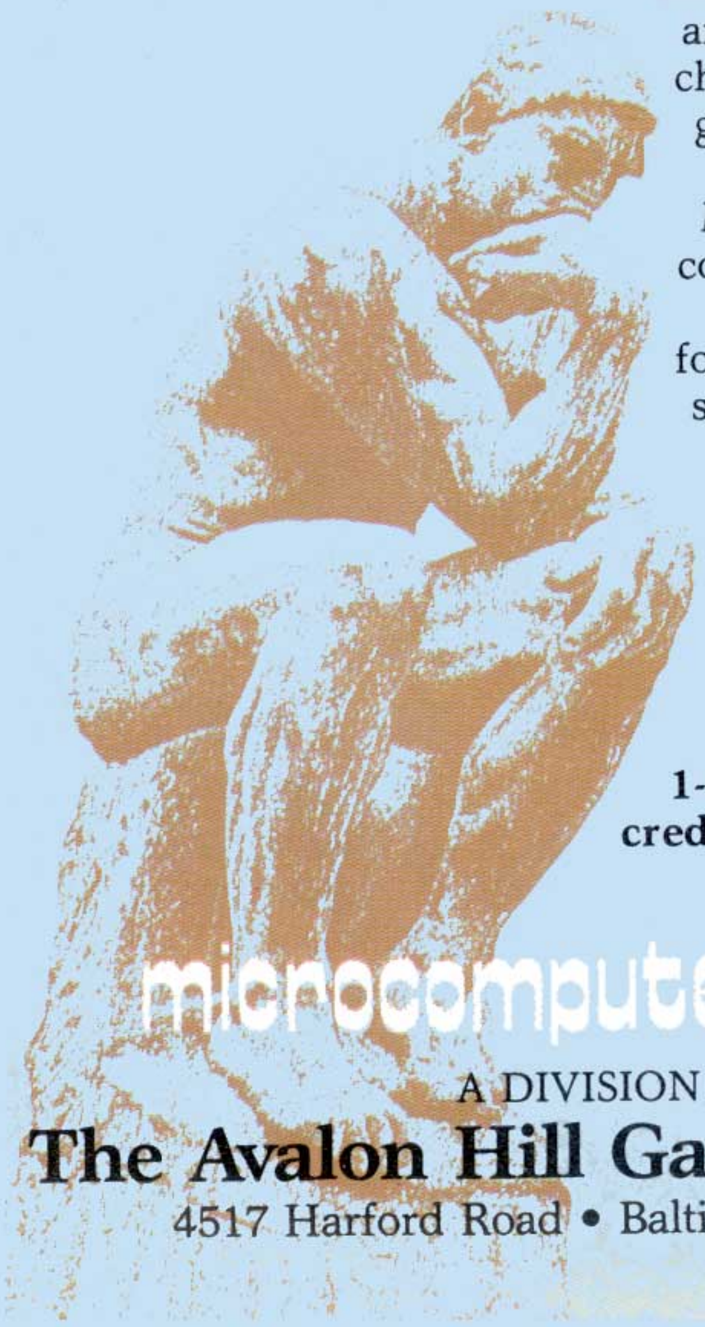
VOYAGER – A solitaire computer game that challenges the human player to explore the four levels of an alien spacecraft's maze-like corridors and rooms in 3-D simulated graphics, all the while avoiding robots programmed to blast any intruders. In order to win, the human must destroy all power generators and escape or hunt out and annihilate the killer robots. **VOYAGER** comes with color-animated graphics and sound capabilities.



V.C. (short for Viet Cong), is Avalon Hill's first game ever on the controversial Viet Nam War. Under your command is the chopper based air-mobile and heavily armed 1/509th Air Cavalry and 9/15th Field Artillery for fire support. It is an **unconventional conflict.** You have the task of bringing the civilian population under your protection where the enemy (played by the computer) can hide amongst the people, and where the politics of terrorism and friendship can turn the people you want to save against you.



SHOOTOUT – 30 ALIEN WARSHIPS HAVE ENTERED YOUR PATROL ZONE. OK, shields up?, energy level . . . check, azimuth set?, yup. This may sound like the latest summer space movie thriller but in fact it's the preparations YOU will make when playing Avalon Hill's new solitaire arcade strategy game **SHOOTOUT AT THE OK GALAXY.** Over 2 years in the making, **SHOOTOUT** is purely graphical combining arcade excitement with just the right touch of strategy.



These games are for the sly and clever. Check on the chart at right to see if the game is compatible with your home computer. Most Avalon Hill Microcomputer Games are also compatible with the following home computer systems: Apple II®; Atari 400 & 800®; IBM P.C.®; and Commodore VIC-20, PET CBM & 2001®. **Available at finer computer stores everywhere! Or call TOLL FREE: 1-800-638-9292 for fast credit card purchase. Ask for Operator 80.**

microcomputer games®

A DIVISION OF

The Avalon Hill Game Company

4517 Harford Road • Baltimore, MD 21214

SOFTWARE GAMES	CASS. TRS-80 Color	CASS. TRS-80 Models I & III	DISK TRS-80 Models I & III	PRICE
B-1 Nuclear Bomber		16K		16.00
			32K	21.00
Midway Campaign		16K		16.00
			32K	21.00
North Atlantic Convoy Raider		16K		16.00
			32K	21.00
Nukewar		16K		16.00
			32K	21.00
Planet Miners		16K		16.00
			32K	21.00
Lords of Karma		48K		20.00
			48K	25.00
Computer Acquire		16K		20.00
Computer Statis Pro Baseball		16K		25.00
			32K	30.00
Tanktics		16K		24.00

SOFTWARE GAMES	CASS. TRS-80 Color	CASS. TRS-80 Models I & III	DISK TRS-80 Models I & III	PRICE
Computer Baseball Strategy		32K		17.00
Galaxy		16K		20.00
			32K	25.00
Voyager	16K	16K		20.00
Foreign Exchange		16K		20.00
Draw Poker		16K		16.00
Andromeda Conquest		16K		18.00
			32K	23.00
Telengard				
			48K	28.00
VC				
			32K	25.00
G F S Sorceress		48K		30.00
			48K	35.00

SOFTWARE GAMES	CASS. TRS-80 Color	CASS. TRS-80 Models I & III	DISK TRS-80 Models I & III	PRICE
Fredericksburg			32K	35.00
Conflict 2500		16K		16.00
			32K	21.00
Empire of the Overmind		48K		30.00
			48K	35.00
Stocks and Bonds		16K		20.00
			32K	25.00
Computer Football Strategy			32K	21.00
Guns of Fort Defiance		16K		20.00
			32K	25.00
Dnieper River Line		32K		25.00
			32K	30.00
Shootout at the OK Galaxy	16K	16K		20.00
Tank Arkade		16K		15.00

I hereby certify that I am one sharp dude. Therefore, please send me _____ (number) copies of _____ (game title) on _____ (cass. or disk.) for _____ (type of computer).

I enclose a total of $_____ (include 10% for postage and handling).

☐ Let me take an IQ test first. In the meantime please send me your FREE full-color game catalog with details on all your software.

Name _____

Address _____

City _____ State _____ Zip _____

☐ CHECK ☐ MONEY ORDER ☐ CHARGE (see below)

☐ AMERICAN EXPRESS ☐ CHOICE ☐ VISA ☐ MASTERCARD

Account No. _____ Expiration Date _____

For fast credit card ordering call Operator 80 TOLL FREE at 1-800-638-9292.

Send Check or Money Order to:

AVALON HILL MICROCOMPUTER GAMES

Dept. 80 • 4517 Harford Road • Baltimore, MD 21214

TRS-80® Printers—No On

DMP-500
1795.00
Cat. No. 26-1252

Daisy Wheel II
1995.00
Cat. No. 26-1158

DMP-200
799.00
Cat. No. 26-1254

DMP-400
1195.00
Cat. No. 26-1251

Triple-Mode Dot Matrix Printers

Our three new dot-matrix printers—the DMP-200, 400 and 500—offer incredible versatility! Each one allows you to select from a *word processing* mode which emulates Daisy Wheel features like proportional spacing, boldface, underline, super and subscripts; a *bit image* mode for impressive, dot-addressable graphics; and a *data processing* mode for fast, efficient throughput. These printers feature bi-directional, logic-seeking 9-wire print heads and support the use of single sheets or fanfold paper.

You get the 96-character ASCII set with 32 special and 30 block graphics characters. Lower case characters have descenders. Print density can be set at 10, 12 or 16.7 cpi in a 9×9 or 15×9 dot matrix. Print proportionally in an n-23 dot matrix. (See samples at right.) Repeat and column-addressing codes make graphics programming easier. Horizontal resolution is 60-100 dots per inch.

Best of all, a handy switch allows you to "custom set" modes, pitches and interface options to make it easier to use a TRS-80 printer with your software. The DMP-500 features a standard parallel interface. The DMP-200 and 400 include selectable parallel and TRS-80 Color Computer-compatible serial interfaces (600/1200 baud). All offer front panel on-line switching and paper advance, including form feed.

DMP-200. An amazingly low price for a full-performance 9 1/2" printer. Prints 120 characters per second at 10 cpi—55 lines per minute at 80 columns. Removable tractor and friction platen. Line feed in 1/6, 1/8 and 1/72 increments.

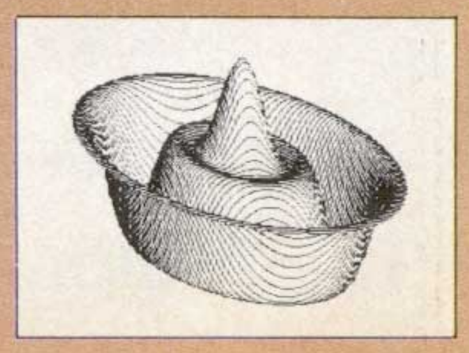
```
10 CPI STANDARD !"#$%&'()*+,-./0
12 CPI ELITE !"#$%&'()*+,-./0123456789
10 CPI CORRESPONDENCE !"#$%&'()*
PS MODE  ABCDEFGHI abcdefghi vwxyz{|}~
```

DMP-400. Full 132-column output for under $1200. Prints 140 cps—51 lpm at 132 columns. Compact, low-profile styling with noise-reducing cover. Line feed in 1/6, 1/8, 1/72 and 1/216 increments. Adjustable tractor is also removable. Friction platen.

DMP-500. A business printer that's all business. High performance design gives 220 cps—76 lpm at 132 columns. Adjustable tractor accepts paper from 4" to 15" wide. Line feed in 1/6, 1/8, 1/72 and 1/216 increments.

Low-Cost Dot-Addressable Printer

DMP-100. Graphic density is 60×63 dots per square inch. Prints 80 upper and lower case 5×7 dot-matrix characters at 50 cps, 27 lpm (10 cpi). Features underline, 480-byte full-line dot buffer and selectable parallel or Color Computer-compatible serial interfaces. Adjustable tractor handles 4 1/2" to 9 1/2" fanfold paper.



ne Offers a Wider Variety

DMP-100
399.00
Cat. No. 26-1253

DWP-410
1495.00
Cat. No. 26-1250

```
ABCDEFG abcdef 01234 :;<=>?
ABCDEFGH abcdefg 01234 :;<=>?
ABCDEFGH abcdefg 01234 :;<=>?
ABCDEFGH abcdefg 01234 :;<=>?
```

Daisy Wheel Word Processing Printers

Our letter-quality printers give your correspondence and reports that crisp, clean "electric typewriter" look. Interchangeable 124-character print wheels let you select the typeface you want (shown above). "External Program Mode" lets you use print wheels with different pitch or special characters. Switch selectable print density for fixed 10 or 12 cpi or proportional spacing (to give justified right margins). All standard word processing features are included: forward and reverse paper feed, 1/2-line feed, underline and backspace. Use friction-feed platen or an optional tractor-feed. Parallel interface. Each printer includes Courier 10 print wheel and a carbon ribbon cartridge.

Daisy Wheel II. Our finest! Ideal for the business, legal or medical office. Prints at over 500 words per minute. For added convenience, just snap in a tractor-feed (26-1447, $289.95) to use fanfold paper, pin-fed labels, pre-printed and multi-part forms. Fast, optimized carriage movement for surprisingly efficient throughput. The DW II can also be used with our microprocessor-controlled automatic sheet feeder or envelope feeder.

DWP-410. Full-featured letter-quality printouts at a remarkably low price. Prints at over 300 words per minute. 1/120" space and 1/48" line feed. Automatic Paper Set makes paper insertion easy. Or use an easily removable tractor-feed (26-1459, $239.95)—it's ideal for labels, multi-part forms and long DP runs with minimal supervision.

Available Nationwide

You can choose the TRS-80 dot-matrix or daisy wheel printer that meets your needs at any Radio Shack Computer Center, store or participating dealer. Be sure to see our complete selection of color plotters, too. Come in today!

Radio Shack®
The biggest name in little computers™
A DIVISION OF TANDY CORPORATION

Send me your free TRS-80 Computer Catalog today!

Mail To: Radio Shack, Dept. 83-A-185
300 One Tandy Center, Fort Worth, Texas 76102

NAME _____

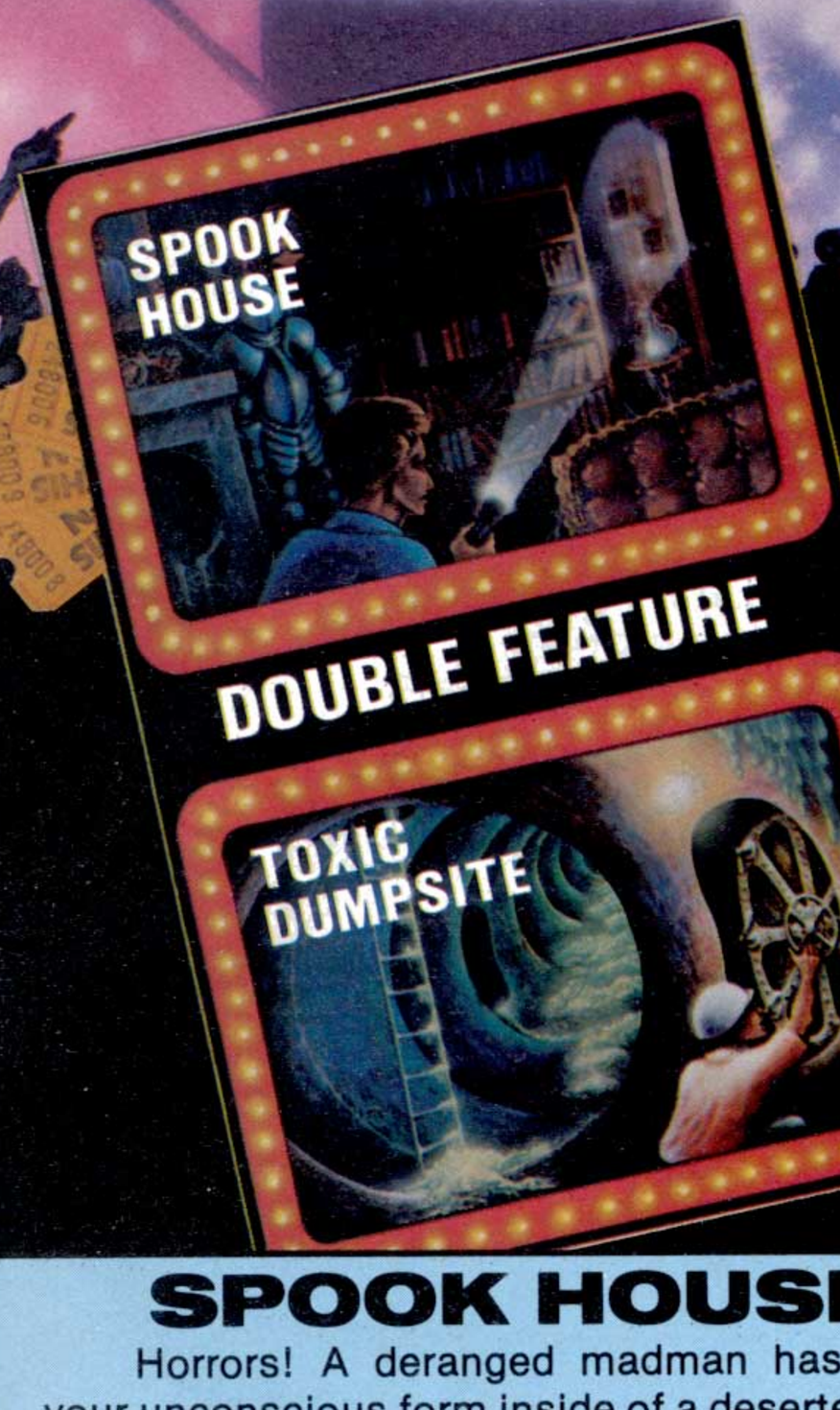
ADDRESS _____

CITY _____ STATE _____ ZIP _____

PHONE _____

Retail prices may vary at individual stores and dealers. Special order may be required at some stores.

DOUBLE FEATURE
FOR YOUR TRS-80



HERE'S YOUR TICKET TO MANY THRILLING HOURS OF ENJOYMENT

TWO BRAND NEW GRAPHIC ADVENTURES IN ONE PACKAGE — FOR YOUR TRS-80

By Roger Schrag

SPOOK HOUSE

Horrors! A deranged madman has locked your unconscious form inside of a deserted creep house at a crazy carnival. You awake to some sobering facts: a bomb is set to explode, and only 30 real-time minutes stand between you and eternity. Somewhere, somehow you must locate and defuse the explosive — but we'd be less than honest if we thought you had more than a ghost of a chance! Brrrr — a real cliffhanger!

TOXIC DUMPSITE

Something's gone very, very wrong at the Toxic Dumpsite where the treatment and burial of deadly contaminants take place. The entire plant will explode like the Fourth of July in less than 30 minutes — unless you can avoid the many traps and protection systems and shut the plant down in time. Of course, time is the one luxury you haven't got . . . !

This is it — the blockbusting debut of the long-awaited Graphic Adventures! Now, your TRS-80 can hobnob with the best of 'em. Over 50 full screens of sharp, quality graphics breathe a realism and vitality into Adventuring, the likes of which you've yet to see on the TRS-80. Both Graphic Adventures feature game save, extensive vocabulary, real-time action (yes — we said real-time!) and accept FULL sentences! Exclusive area showings brought to you by Adventure International. Now playing at a TRS-80 near you!

TRS-80 48K Models 1 & 3 disk 012-0164 — $39.95

To order, see your local dealer. If he does not have the program, then call 1-800-327-7172 (orders only please) or write for our free catalog.

Adventure INTERNATIONAL

PRICES SUBJECT TO CHANGE

Published by ADVENTURE INTERNATIONAL
a subsidiary of Scott Adams, Inc.
BOX 3435 • LONGWOOD, FL 32750 • (305) 830-8194

Printer hangups

Avoiding the printer-not-ready problem

Model I only

Jerry Latham, Midwest City, OK

We all know of the consequences of attempting to send data from the TRS-80 to a nonexistent or temporarily out of service printer. You end up hanging the system and losing your program and data. The culprit in this situation is the routine in ROM that handles printer output. This ROM routine only checks to see if the printer is ready for data and if it is not, it sits patiently waiting. If the printer doesn't exist, or mysteriously went out to lunch, then the system hangs up and it is "bye bye" to in-memory data.

Obviously, what we need is some type of routine that can check the printer before trying to send data to it. The routine would also make sure that all is right, and if not, let us know so that we can take better action than just pressing RESET and retyping all of our data.

The Model I/III is a memory mapped computer, as are many on the market today. Memory mapping only means that rather than sending data to a port (a la serial interface), data for a particular device is sent through a memory location in RAM. For the video, these are locations 15360 to 16383, and for the parallel line printer, the location is 14312 decimal. That's 37E8H for you hex nuts out there.

Like PEEKing and POKEing into video memory, we can PEEK at location 14312 and see what the status of our line printer is. We can do this because when you aren't sending data to the printer to be printed, the printer is sending data to the computer to tell it whether or not it is ready to handle data.

Now that we know where to look to check the condition of our printer, the question remains, how do we use this information to prevent printer hangups? First, we need to know just exactly what a particular printer's READY status value is. With so many printers on the market, it is a fair guess that there is more than one READY signal being sent.

Listing 1 is a complete and formal test procedure for determining several possible status values for various printer conditions. Listing 1 is self prompting and will allow you to determine values for the following conditions: printer ready, printer in off-line mode, printer power switch off, printer power line unplugged, printer initiating, printer busy/buffer full, printer initiating without paper, printer out of paper and no printer attached.

Depending on the sophistication of the printer, there are several other conditions that could be tested. These might be conditions like: no ribbon installed, paper hanging, print head overheat, etc.

So that we can carry the same thoughts and values from program to program, see Table 1 for a list of variable names and their uses that I derived for the Epson MX-80 printer. If you want to hurry and run the tests to find out all of your printer's status values by using Listing 1, please do so. A word of caution—test for "no printer attached". It is necessary to disconnect the cable from the printer to the computer expansion interface. This should be done with the power OFF to prevent possible damage to the printer, the computer or both. The program in Listing 1 waits until the end to do that test.

If you do not have a TRS-80 Model I, it will be necessary to do a little research and find where the printer is memory mapped. In all listings, the variable PP% is used to hold this value and it is set up in line 10 of each program. Change this value as required for your particular computer. Once this change is made, you may proceed with the tests.

Another possible area of conflict with other systems is in line 690 of Listing 1. Here, we get a string 255 characters long. The purpose of the section in Listing 1, lines 640-760, is to fill the printer's buffer and force it to print and send busy signals to the computer. I was using an MX-80 for these tests, and its buffer is 80 characters long. Sending 510 characters in a row was more than enough to do the job.

Once you have a list of the status codes your particular printer sends to your computer, you can start planning how to use them in your programs. Listing 2 shows how to do your printer checks and how to set up a subroutine that will diagnose your printer's status and control further processing based on that diagnosis. It is a fully commented version.

The important thing to notice in this example program is that checks on the printer are done *before* trying to send any data to it. If this is not done, then you

risk the possibility of system lockup.

Listing 2 was designed using the Epson MX-80 values for printer status checks. Substitute the values you got with Listing 1 for the constants in line 10. Of course, don't forget to substitute your system's printer memory mapped location for the constant PP% in line 10. No other changes should be necessary.

Since Listing 2 is so well REMarked, I won't duplicate the effort with a line-by-line account of the program's operation. I will, however, make note of a couple of things that are not readily apparent in the program. First, no explicit check is done to see if the printer is just initiating (ST%). Instead, a check is done in line 120 to see if it is initiating without paper. If the printer is found to not be ready (and yet no specific error is found), then line 140 sends the program back for another loop. This will cause a wait-until-ready condition while the printer is powering up normally. Line 100 informs us that the printer is off or unplugged. It doesn't mention the fact that you may have a blown fuse. This could possibly cause this status value to be sent to the computer also.

One suggestion I offer if you elect to use this routine is to locate the subroutine (lines 80-200) close to the front of your program to speed execution time. Use a GOTO statement to skip the routine when first starting the program run. Another suggestion is to put a check for PRINTER READY as one of the first executed statements in your program so you can catch any errors before you start typing in data.

Table 1

Constant	Use/Status Indication
OK%	Printer ready
OFF%	Printer power switch off
PO%	Printer unplugged from AC supply
IP%	Initiating without paper
NP%	Out of paper
OL%	Printer in off-line mode
ST%	Printer initiating normally
BZ%	Printer busy/buffer full
UP%	Printer not attached to computer

Listing 1

```
5 REM PRINTER STATUS TEST PROGRAM
WRITTEN BY JERRY L. LATHAM
 1409 EVERGREEN CIRCLE
 MIDWEST CITY, OK 73110

10 CLEAR 2000: PP%=14312: REM MEMORY LOCATION
THAT PASSES INFORMATION FROM THE PRINTER TO THE
COMPUTER AND FROM THE COMPUTER TO THE PRINTER.
20 CLS:PRINT"THIS PROGRAM WAS DESIGNED AT THE
REQUEST OF THE EDITORS OF"
30 PRINT"80—U.S. JOURNAL TO DETERMINE THE STATUS
SIGNALS PASSED FROM A"
40 PRINT"PARALLEL PRINTER TO THE TRS—80 MODEL I
COMPUTER TO ASSIST IN THE"
50 PRINT"PREVENTION OF SYSTEM HANGUP DUE TO
PRINTER ERRORS."
60 PRINT"TO START, CHECK THAT YOUR PRINTER IS
PROPERLY ATTACHED TO THE"
70 PRINT"COMPUTER:"
80 PRINT"CABLE HOOKED UP FROM THE PRINTER TO
EXPANSION INTERFACE."
90 FOR X=0 TO 200: NEXT
100 PRINT"PRINTER PLUGGED IN."
110 FOR X=0 TO 200: NEXT
120 PRINT "PRINTER HAS PAPER IN IT."
130 FOR X=0 TO 200: NEXT
140 PRINT "PRINTER TURNED ON."
150 FOR X=0 TO 200: NEXT
160 PRINT "PRINTER IN THE ON—LINE MODE."
170 FOR X=0 TO 200: NEXT
180 PRINT "OK, WE ARE READY FOR THE FIRST TEST."
190 FOR X=0 TO 500: NEXT
200 CLS: PRINT "FIRST WE WILL DETERMINE WHAT VALUE
THE PRINTER SENDS TO THE"
210 PRINT "COMPUTER TO TELL IT THAT IT IS OK TO SEND
DATA TO THE PRINTER."
220 PRINT "WE WILL CALL THIS VALUE OK% FOR FUTURE
REFERENCE."
230 PRINT
240 INPUT "WHEN THE PRINTER IS READY, PRESS <ENTER>
TO BEGIN";A$
250 FOR X%=0 TO 63: PRINT PEEK(PP%);: NEXT X%
260 PRINT: PRINT "WRITE THIS VALUE DOWN AND ASSIGN
IT THE NAME OK%."
270 PRINT "THIS IS YOUR <PRINTER READY> SIGNAL TO
THE COMPUTER."
280 PRINT: INPUT "PRESS <ENTER> WHEN READY TO
```

```
CONTINUE";A$
290 CLS: PRINT "NOW FOR THE NEXT TEST. IF YOUR
PRINTER HAS AN ON-LINE/OFF-LINE"
300 PRINT "SELECT SWITCH, SET IT TO THE OFF-LINE
MODE. IF IT DOES NOT HAVE"
310 PRINT "THIS SWITCH, THEN IGNORE THIS TEST."
320 INPUT "PRESS THE <ENTER> KEY TO DETERMINE THE
OFF-LINE STATUS SIGNAL";A$
330 FOR X%=0 TO 63: PRINT PEEK (PP%);: NEXT X%: PRINT
340 PRINT "WRITE THE DISPLAYED VALUE DOWN AND
ASSIGN IT THE NAME OL%"
350 PRINT "THIS IS YOUR OFF-LINE SIGNAL TO THE
COMPUTER"
360 PRINT: INPUT "PRESS <ENTER> WHEN READY TO
CONTINUE TESTING";A$
370 CLS: PRINT "NEXT WE WILL TEST THE ON/OFF SWITCH
SIGNAL FROM THE PRINTER."
380 PRINT "TURN OFF YOUR PRINTER WITH THE ON/OFF
SWITCH."
390 INPUT "PRESS <ENTER> WHEN THE PRINTER IS OFF";A$
400 FOR X%= 0 TO 63: PRINT PEEK (PP%);: NEXTX%: PRINT
410 PRINT "WRITE THIS VALUE DOWN AND CALL IT OFF%
FOR FUTURE USE."
420 INPUT "PRESS <ENTER> WHEN READY TO CONTINUE
TESTING";A$
430 CLS: PRINT "NOW TO SEE IF YOU GET A DIFFERENT
SIGNAL WHEN THE AC LINE IS"
440 PRINT "UNPLUGGED FROM THE WALL SOCKET THAN
THE ONE YOU GOT WHEN THE"
450 PRINT "SWITCH WAS TURNED OFF. WITH THE SWITCH
STILL IN THE OFF"
460 PRINT "POSITION, UNPLUG THE PRINTER FROM THE
SOCKET."
470 INPUT "PRESS <ENTER> WHEN READY TO RUN THE
TEST";A$
480 FOR X%= 0 TO 63: PRINT PEEK (PP%);: NEXT X%: PRINT
490 PRINT "WRITE THIS VALUE DOWN AND CALL IT PO%
(POWER OFF) FOR LATER USE"
500 INPUT "PRESS <ENTER> WHEN READY TO GO TO THE
NEXT STEP";A$
510 CLS: PRINT "PLUG THE PRINTER BACK INTO THE
SOCKET BUT *DO NOT* TURN IT ON"
520 PRINT "JUST YET. WE ARE NOW GOING TO DETERMINE
THE INITIATION STATUS"
530 PRINT "VALUE RETURNED TO THE COMPUTER."
540 PRINT: PRINT "THE TRICK TO THIS TEST IS TO FIRST
PRESS THE ENTER KEY AND THEN"
550 PRINT "IMMEDIATELY! REACH OVER AND TURN THE
PRINTER ON."
560 PRINT "IF YOU MISS, HIT THE BREAK KEY AND TYPE
GOTO 510 TO RESTART THIS";
570 PRINT "PART OF THE TEST."
580 INPUT "PRESS <ENTER> WHEN YOU AND THE PRINTER
ARE READY";A$
590 FOR X%=0 TO 100:PRINT PEEK(PP%);: NEXT X%: PRINT
600 PRINT "YOU SHOULD SEE THE OFF% CODE FOLLOWED
BY SOME SERIES OF NUMBERS"
610 PRINT "FOLLOWED BY THE <PRINTER READY> (OK%)
```

```
CODE. THAT GROUP OF"
620 PRINT "OF NUMBERS IS THE INITIATING STATUS CODE
(ST% FOR LATER USE)"
630 PRINT "WRITE IT DOWN AS USUAL."
640 INPUT "PRESS <ENTER> WHEN READY FOR THE NEXT
PART OF THE TEST";A$
650 CLS: PRINT "NOW WE ARE GOING TO TRY TO
DETERMINE THE PRINTER BUSY SIGNAL."
660 PRINT "YOU MUST FILL THE PRINTER BUFFER SOMEHOW
TO DETERMINE THIS VALUE";
670 PRINT "ADJUST LINE 690 SO THAT YOU ARE ASSURED
OF SENDING AT LEAST"
680 PRINT "ENOUGH CHARACTERS TO DO THIS"
690 B$=STRING$(255,"T")
700 INPUT "PRESS <ENTER> WHEN YOUR PRINTER IS
READY TO PRINT";A$
710 LPRINT B$;B$
720 FOR X%=0 TO 64: PRINT PEEK(PP%);: NEXT X%: PRINT
730 PRINT "HOPEFULLY THIS STRING OF 510 CHARACTERS
KEPT YOUR PRINTER BUSY."
740 PRINT "YOU SHOULD HAVE A SERIES OF NUMBERS
YOU HAVEN'T SEEN BEFORE."
750 PRINT "THIS IS THE PRINTER BUSY/BUFFER FULL SIGNAL.
CALL IT BZ% FOR USE";
760 PRINT "LATER ON. WRITE IT DOWN ALSO."
770 INPUT "PRESS <ENTER> WHEN READY FOR THE NEXT
TEST";A$
780 CLS: PRINT "THIS IS THE NEXT TO LAST TEST. TURN THE
PRINTER OFF AGAIN."
790 PRINT "NOW REMOVE THE PAPER FROM THE PRINTER.
YES, WE'RE GOING TO MAKE"
800 PRINT "A LITTLE NOISE (IF YOU HAVE AN OUT OF
PAPER BELL/BUZZER)."
810 PRINT "YOU MAY NOW SEE TWO VALUES WE HAVEN'T
SEEN BEFORE,THE FIRST WILL";
820 PRINT "BE THE INITIATING WITHOUT PAPER SIGNAL,
AND THE LAST GROUP WILL"
830 PRINT "BE THE OUT OF PAPER SIGNAL THE PRINTER
SENDS TO THE COMPUTER."
840 PRINT "TO KEEP THE NOISE LEVEL TO A MINIMUM WE
WILL RUN THIS LIKE THE"
850 PRINT "TEST FOR ST% (INITIATION), THAT IS PRESS
<ENTER> AND RAPIDLY"
860 PRINT "TURN THE PRINTER ON."
870 INPUT"PRESS <ENTER> WHEN READY TO START THIS
TEST";A$
880 FOR X%=1 TO 100: PRINT PEEK (PP%);:NEXT X%: PRINT
890 PRINT "IF YOU HAVE TWO SETS OF NUMBERS AND THE
FIRST IS NOT THE SAME AS";
900 PRINT "THE INITIATION (ST%) VALUE, THEN IT IS AN
INITIATION WITHOUT"
910 PRINT "PAPER SIGNAL IP%. THE LAST GROUP IS THE
ACTUAL OUT OF PAPER"
920 PRINT "SIGNAL, NP%. WRITE IT DOWN WITH ALL THOSE
OTHERS."
930 INPUT "PRESS <ENTER> WHEN READY FOR THE LAST
TEST ";A$
940 CLS: PRINT "FOR THIS LAST TEST WE SHOULD SHUT
DOWN BOTH THE PRINTER AND THE"
950 PRINT "COMPUTER. BUT FIRST SAVE THIS PROGRAM
SO YOU CAN RE-ENTER HERE"
960 PRINT "LATER. TO DO SO LOAD THE PROGRAM, TYPE
RUN, HIT THE <BREAK> KEY";
970 PRINT "AT THE FIRST PROMPT AND THEN TYPE GOTO
940. THIS WILL GET YOU"
980 PRINT "BACK TO THIS POINT IN THE PROGRAM."
990 PRINT "WHILE THE SYSTEM IS SHUT DOWN,
DISCONNECT THE CABLE FROM THE"
1000 PRINT"PRINTER TO THE TRS-80 AT EITHER THE
INTERFACE CARD EDGE OR THE"
1010 PRINT "PRINTER ITSELF, WHICHEVER IS EASIER."
1020 INPUT "PRESS <BREAK> TO SET UP FOR THIS TEST, OR
<ENTER> TO RUN THE TEST";A$
1030 CLS: PRINT "THIS IS A TEST OF THE PRINTER NOT
THERE SIGNAL. IT SHOULD BE 255";
1040 FOR X%= 0 TO 48: PRINT PEEK (PP%);: NEXT X%:
PRINT
1050 PRINT "THIS IS THE <NO PRINTER ATTACHED>
SIGNAL AND IS ORDINARILY 255."
1060 PRINT "YOUR RAM MAY SET UP DIFFERENTLY, HENCE
THE NECESSITY FOR THIS"
1070 PRINT "TEST. WRITE THE VALUE DOWN WITH THE REST
AND CALL IT UP%"
1080 PRINT "THAT IS, UNPLUGGED PRINTER."
1090 PRINT: PRINT "AND THAT IS ALL THERE IS TO IT. NOW
YOU HAVE A SET OF NUMBERS"
1100 PRINT "THAT YOU MAY USE TO SET UP ROUTINES IN
YOUR PROGRAMS TO PREVENT"
1110 PRINT "PRINTER MALFUNCTIONS FROM CAUSING
YOU TO LOSE DATA OR TIME."
```

Listing 2

```
5 REM PRINTER HANG-UP PREVENTION PROGRAM FOR
THE TRS-80 MODEL I
WRITTEN BY JERRY L. LATHAM
  1409 EVERGREEN CIRCLE
  MIDWEST CITY, OK 73110

10 CLEAR 2000: PP%=14312: OFF%=15: PO%=15:
OK%=63: OL%=143: ST%=159: BZ%=191: NP%=207:
IP%=233: UP%=255
15 REM LINE 10 SETS UP VARIABLES TO CHECK PRINTER
STATUS. EXAMPLE IS FOR THE EPSON MX-80 WITH
TRACTOR FEED, ATTACHED TO A 48K TRS-80 MODEL I VIA
THE EXPANSION INTERFACE CARD EDGE.

20 IF PEEK(PP%) <> OK% THEN GOSUB 80
25 REM LINE 20 CHECKS PRINTER STATUS BEFORE
BEGINNING A LINE OF PRINT. THIS WILL PREVENT SYSTEM
LOCKUP IF THE PRINTER IS NOT ATTACHED OR READY.

30 LPRINT "THIS IS A TEST OF THE MX-80 ERROR CODES"
35 REM LINE 30 SHOULD CAUSE NO REAL PROBLEMS AS IT
WILL NOT FILL THE 80 CHARACTER BUFFER ON THE MX-80.

40 IF PEEK(PP%) <> OK% THEN GOSUB 80
45 REM IN LINE 40 WE AGAIN CHECK THE STATUS OF THE
PRINTER BEFORE SENDING DATA TO IT.

50 LPRINT STRING$(255,"T")
55 REM LINE 50 SENDS 255 T'S TO THE PRINTER. THIS WILL
OVER-LOAD THE MX-80'S BUFFER AND FORCE US TO GO
LOOKING TO SEE IF ANYTHING SERIOUS IS WRONG WITH
THE PRINTER.
```

```
60 GOTO 20
65 REM LINE 60 SETS US IN AN INFINITE LOOP SO WE CAN
WATCH THE PRETTY LINES ON THE SCREEN AND EAT LOTS
OF PAPER WITH OUR PRINTER TESTS.

70 END
75 REM LINE 70 IS JUST A SAFETY VALVE TO KEEP US OUT
OF SUB-ROUTINE COUNTRY.

80 IF PEEK(PP%)=OK% THEN RETURN ELSE
PRINT"*******CHECKING @ 80******"
85 REM LINE 80 CHECKS IF THE STATUS OF THE PRINTER
HAS CHANGED BACK TO READY WHILE WE WERE GETTING
HERE. THE LINE TO BE PUT OUT TO THE SCREEN WAS MADE
LONG SO THAT WE WILL NOTICE IT AS IT FLASHES BY ON
THE SCREEN.

90 IF PEEK(PP%)=BZ% THEN GOTO 150
95 REM LINE 90 CHECKS FOR THE BUFFER FULL/PRINTER
BUSY SIGNAL AND SENDS US TO A WAIT LOOP DOWN AT
LINE 150 IF THIS IS THE CASE IN POINT.

100 IF PEEK(PP%) = OFF% THEN PRINT"PRINTER SWITCH
OFF OR PRINTER IS UNPLUGGED.":GOTO 180
105 REM LINE 100 CHECKS TO SEE THAT POWER IS
APPLIED TO OUR PRINTER. IF IT ISN'T THEN WE GO TO A
WAIT LOOP AT 190.

110 IF PEEK(PP%)=OL% THEN PRINT "PRINTER IS IN OFF
LINE MODE.":GOTO 180
115 REM LINE 110 CHECKS TO SEE IF THE ON-LINE
SWITCH IS IN THE OFF-LINE POSITION AND NOTIFIES US IF
IT IS. THIS IS PROBABLY THE SINGLE MOST OFTEN
ENCOUNTERED ERROR (FOR ME ANYHOW).

120 IF PEEK(PP%)=NP% OR PEEK(PP%)=IP% THEN PRINT
"PRINTER IS OUT OF PAPER.": GOTO 180
125 REM LINE 120 CHECKS FOR AN OUT OF PAPER
STATUS, AND GOES TO A WAIT. LOOP IF THIS CONDITION
IS FOUND.

130 IF PEEK(PP%)=UP% THEN PRINT "PRINTER CABLE IS
UNPLUGGED AT EITHER THE PRINTER OR TRS-80 E.I. TURN
OFF POWER!! CORRECT PROBLEM AND RESTART
PROGRAM.": END
135 REM LINE 130 SEARCHES FOR A REAL PROBLEM: THE
CABLE FROM THE INTERFACE TO THE PRINTER HAS
BECOME LOOSE OR DISCONNECTED. SINCE THIS MAY
CAUSE PROBLEMS IF CORRECTED WITH POWER ON THE
PROGRAM ENDS INSTEAD OF WAITING.

140 GOTO 80
145 REM LINE 140 SETS THE WHOLE THING UP AGAIN. THE
ONE TEST THAT WASN'T MADE WAS TO SEE IF THE PRINTER
WAS INITIALIZING (159 IN LOCATION 14312). THIS WILL
CATCH THAT AND WAIT UNTIL READY.

150 IF PEEK(PP%)=BZ% THEN PRINT "WAITING @ 150":
GOTO 150
155 REM LINE 150 DOES A SIMPLE WAIT UNTIL PRINTER IS
NOT BUSY LOOP IF THE PRINTER IS BUSY, THAT IS. THE
PROGRAM GOT HERE BY FINDING THE PRINTER BUSY, SO
IT SHOULD STILL BE, BUT MAY RUN OUT OF PAPER WHILE
WE AREN'T LOOKING.

160 IF PEEK(PP%) <> OK% THEN 80
165 REM IF THE PRINTER ISN'T BUSY, AND IT ISN'T READY
THEN GO BACK AND FIND OUT WHAT THE MATTER IS.

170 RETURN
175 REM LINE 170 RETURNS US TO THE MAIN PROGRAM
FLOW ONCE THE PRINTER IS AGAIN IN THE READY (NOT
BUSY) CONDITION.

180 PRINT"CORRECT PROBLEM THEN PRESS ENTER TO
CONTINUE";
185 REM LINE 180 TELLS THE OPERATOR THAT THE SYSTEM
IS WAITING FOR THE PROBLEM TO BE CORRECTED BEFORE
GOING ON. THIS PREVENTS SYSTEM LOCKUP WITHOUT
LOSING DATA.

190 A$=INKEY$: IF A$<>CHR$(13) THEN 190
195 REM LINE 190 JUST SITS THERE UNTIL THE ENTER KEY IS
PRESSED.

200 RETURN
205 REM AND AT LAST, LINE 200 RETURNS TO THE MAIN
PROGRAM FLOW WHEN THE OPERATOR SIGNALS THAT THE
PROBLEM HAS BEEN FIXED. ■
```

Captain 80

The adventures of a software secret agent

Bob Liddil

I was working quietly in the attic lab of Professor Megabyte's old Victorian house when I got a call from the chief on my two-way wrist modem and pocket computer. An anonymous tipster had revealed a clue to the whereabouts of the missing academician and hardware inventor, for whom I had been searching lo, these many months.

On my desk, I had found three new program disks. Two of the three are published by Computer Shack, the third by Trend Software. They were entitled Jovian, Cyborg, and Demon Seed. According to the chief, somewhere on these disks I would be able to find the long-lost professor.

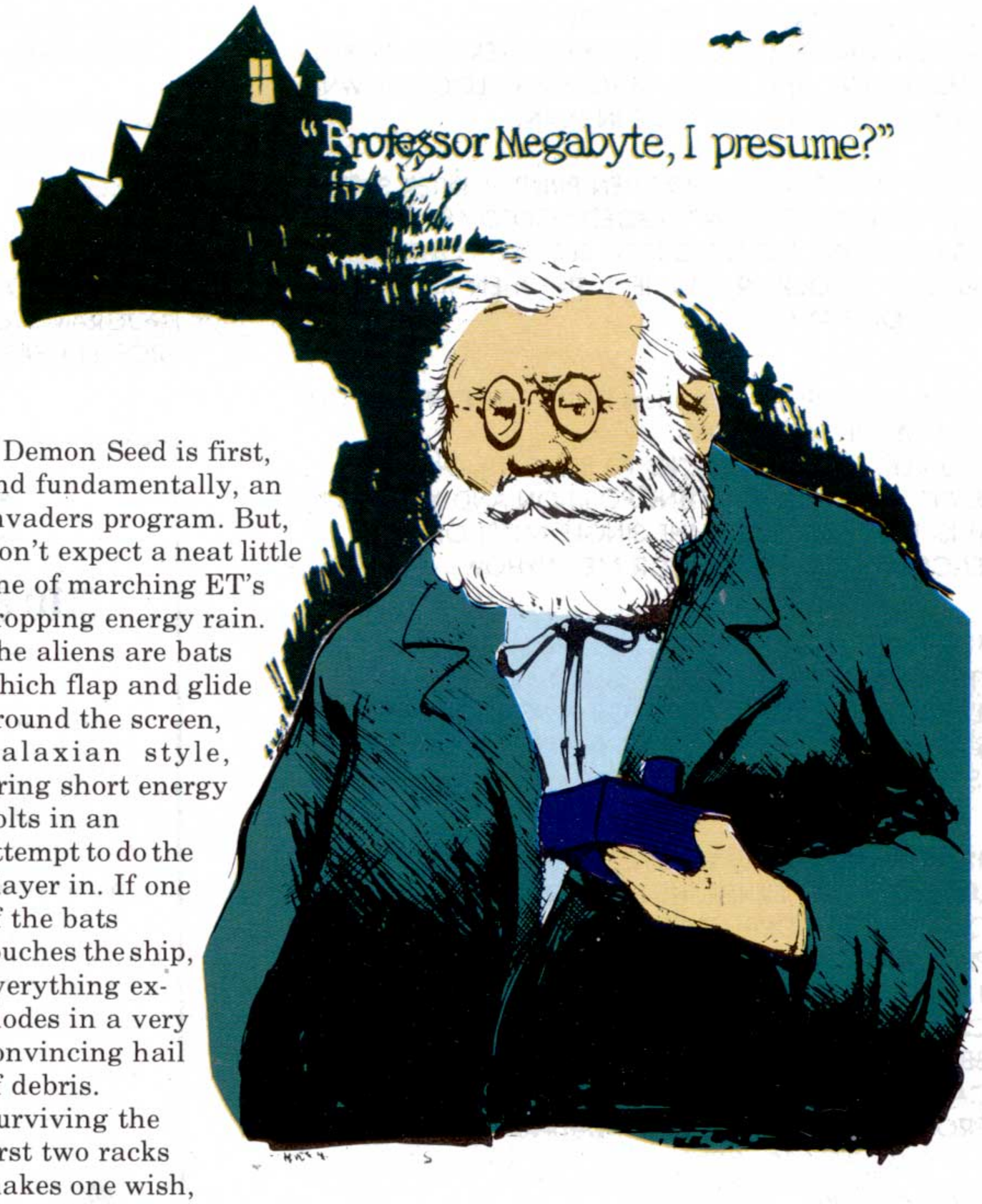
I didn't really know what to expect when I booted up Demon Seed. Max wasn't around to brief me on the details. He's been wandering the phone lines more and more. Then it hit me — that sinking feeling I get whenever I am downloaded into the world of a computer program.

"Max! What are you doing?" I shouted in vain as I de-rezzed in real time. There was no answer. I just knew I was in for a hard time.

I re-rezzed in the padded seat of a Class VII Laserfighter, inside Demon Seed. I was all set for the fight of the year when I heard a voice over the radio. It was Max.

"Captain, don't transmit, just listen. I've been kidnapped! I've downloaded you into the first program booted and provided sideslip codes to access my prison. Help me, Captain, you are my only hope."

I didn't have time to absorb the reality of the situation. Somebody pressed <ENTER> and the game was on.



Demon Seed is first, and fundamentally, an Invaders program. But, Don't expect a neat little line of marching ET's dropping energy rain. The aliens are bats which flap and glide around the screen, Galaxian style, firing short energy bolts in an attempt to do the player in. If one of the bats touches the ship, everything explodes in a very convincing hail of debris. Surviving the first two racks makes one wish, wistfully, that he hadn't.

Round three features a wave of large, bewinged demons which are also equipped with energy bolts and destructo bodies. Unlike their smaller precedessors, however, if you shoot a wing or two off these guys, they keep flying, gliding and shooting. Hitting them dead center is the only way to get them.

The final variant consists of an awesome mother ship which looms down from screen top. The laserfighter must shoot through the vulnerable bottom of the battlestar and crack the reactor before the guardian birds can swoop from the

sides to get him, or before the onboard defenses of the station itself blow him away. The laserfighter has a defense screen which comes in very handy in this sequence.

Demon Seed is a well done, animation quality, professional arcade game, which will provide endless hours of fun to those who own it. It is action-packed, user-friendly and full of challenge. Watch out Big Five, these from Trend (distributed by Computer Shack) are on your trail!

Dodging electric bats and laser bolts is not my idea of a pleasant Sunday jaunt. I do have some experience at this kind of shoot-em-up and with some effort, I was able to work my way through several waves of alien nasties. I was about to dispatch Momma ship, herself, when the first sideslip code engaged.

The scene around me de-rezzed, whereas I was quite intact. It was my first sideslip and I didn't know what to expect. When the world fuzzed back into focus, I found myself still in space, now at the controls of a Cybership Model L50. My sideslip had taken me into the second Computer Shack program, Jovian.

Jovian is a strange mixture of Star Castle and the myriad hybrid Galaxian-Invader-Phoenix genres. It features a continuous movement ship controlled by keyboard numbers. The object seems to be to blow up the cluster stations which multiply like tribbles as the game accelerates, and fend off smaller attack fighters, all laser-equipped.

Though not as exciting, or as diverse as other twitches, Jovian is still a well-crafted offering. With direction controlled by arrow keys and firing executed by space bar, this program escapes the "cramped finger, search all over the board for the controls" syndrome that plagued early arcade look-alikes.

Jovian is not destined to be a classic by any stretch of the imagination. But, it is largely original in content and will be a solid dollar value for anyone who decides to add it to their collection of arcade variants.

Making right-angle turns and fighting aliens will surely be the death of me, I think. I had blasted a fair number of cluster stations and dispatched a large number of aliens to bug-eyed monster heaven when I felt a jolt. The second sideslip code had just booted.

Clever fiends, these kidnappers! Wherever they had Max hidden, he was hidden well. It was going to take several programs to get there . . . if I survived.

I found myself among robots. This was obviously the other Computer Shack disk, Cyborg. I decided to see how fast I could get out of here, *if* I could. I never got along with robots very well, except for Andy from Android Nim — the kindly old geezer I met recently at the old programs' home. These cybernauts looked positively grim. Seeing my chance, I stole an anti-gravity sled. I hoped it was not a mistake.

Lovers of Pac-man will be positively consumed by Cyborg. Picture, if you will, ten or twelve screens of a maze grid. No dots or distracting ghosts, just signal modules to connect and mines to explode, and the screens are interesting and varied. There are all manner of barriers to dodge, shapes to remember, and robots to ram. One gets the feeling he could play for a month and not become bored.

Cyborg shares movement commands with Jovian, but with an added buffer in the form of alpha key-controlled up and down shifting. The spacebar and activates shields (which are the only weaponry here). To knock something off, you have to ram it, shields on. ENTER puts the sled in neutral — a welcome stasis after charging about all over the place.

All in all, Cyborg has the overwhelming feeling of bigness, the same bonus that you got when you saw how far you had to go to get to the end of Sea Dragon. It is dynamic action, well animated and as sophisticated as any advanced user would demand. It is different from any currently-running twitch, and is sure to satisfy its user.

It didn't take long for me to get the hang of operating the anti-grav sled. Soon, I was happily zipping and zooming about, wiping out all kinds of robots and mines and things. I kicked it into high gear and charged all up and down the complex combat grid, doing elated battle with everything in sight. But, then the sled sputtered and stopped. There were robots everywhere and they were converging on me. One touch from any of them and I'd be just so many scattered electrons. I was backed against the seat of the sled trying desperately to think of what to do when everything froze.

From amidst the tangle of motionless robots, I saw a human approaching. He was short, white bearded, pot bellied and bespectacled. Holding a small metal box with a large red button, he was muttering to himself.

"Better hurry boy, not much time before the freeze wears off."

I didn't move for a second until I found my voice. Then, I offered a handshake to my rescuer, which he accepted with unusual strength for what appeared to be an elderly man. It dawned on me who he was.

"Ahem," I said in the best of my newfound voice. "Professor Megabyte, I presume?" The old man smiled. ■

Tandy topics

Ed Juge, Director,

Computer Merchandising

1500 One Tandy Center,

Fort Worth, TX 76102

This month's column is being prepared a bit early. During the time it should be written, I'll be in New York for our yearly New Products Show. No, that doesn't mean we're going to introduce a new generation of hardware. Once a year, we host a meeting for the trade press and financial community, where we display the products which are new in the latest Radio Shack catalogs. The show lasts two days, and is a great opportunity to visit with our press and financial friends in a more leisurely fashion than is possible at trade shows. Of course, they get to ask extensive questions of some of the primary players in our new products game.

I thought you might find it interesting to know some of the behind-the-scenes things we do from time to time. Speaking of such things, the real aim of this column is not to hype products and bring you three-month overdue information on bugs and fixes. I'd really like to give you a flavor of things that happen here in Fort Worth — how we think, and why we do some of the things we do. If there are subjects you'd like to see me discuss, please let me know. I admit that free time here is approximately minus three hours each day, and the pace is very quick, but I still do lots of head scratching each month when I sit down to write "Topics". It's not easy to guess what you'd find interesting, so please don't be shy!

Getting Started In Computing

Cam Brown told me several months ago that this issue would be heavily targeted toward this subject. So, for once, I'll discuss the monthly theme. I think I'm somewhat of an expert in getting started . . . the hard way!

About ten years ago, I was in business for myself, and decided to computerize my inventory. I chose an early microcomputer from Hewlett-Packard, which was about 50% slower than Model I, had 3803 bytes of user memory, two computer-controlled cassette tape drives, and with a thermal printer it cost almost $15,000. Now, my familiarity with computers had been limited to one trip through Tarleton State College's computer room with a co-worker, seven or so years earlier, where I observed a room full of vacuum tubes and a programmer programming by plugging wires into a big black board. Had the state of the art not allowed keyboard programming by the time I was ready to get "computerized", I wouldn't have.

At any rate, the folks at H-P had written a great manual which allowed me and a co-worker to get the idea of BASIC programming fairly easily. It wasn't a "course" in any sense, but when we decided what needed doing, we had to search the book for some function which would allow us to do it. Tedious, but effective. After nine months of seven-day weeks, programming (and learning) before and after hours, and from 6:00 a.m. to midnight on Sundays, we became fairly competent, self-taught, hack

programmers. And we had a very effective inventory system. Armed with a slightly fuzzy recollection of college accounting courses, and a mother-in-law who was an accountant, we put together a general ledger system, too.

Am I recommending this as a way to start? Absolutely not! My marriage survived, luckily. Yours might not. I'm relating my experience to show you how far we've come. There's no comparison in what's available today to help you get started.

There are some specific points I would recommend to anyone starting out, as I did, from absolute "scratch". First, let's define two classifications of "starters".

First, there are those (usually business users) who only want a specific job done, and a computer is the means to an end. Easy job! Buy a turnkey system from someone you have confidence can help you if you have trouble, and is competent to repair a broken computer when necessary ... and who will be around to do it! Buy, if possible, from your friendly local outlet, where you plan to turn for service. Unless you're competent to handle your own problems and repairs, do give this last bit of advice some *serious* consideration.

With the same "unless", you should be aware of one other potential "gotcha". It is a definite maybe — not common, but it can happen. If you own a TRS-80 computer, printer, cables and software, you can return it all to us and say, "I don't care what's wrong, make it work." The same is true for Apple, IBM, or any single-brand system. While there are lots of outstanding third-party add-ons available for every major brand of CPU, you've got to choose your dealer carefully. He's the only guy who will guarantee (maybe) that it all works together. And there can be cases where the computer works to specs, ditto the printer and ditto again the software, yet your system doesn't work. You'll be totally dependent on your dealer, if you're a beginner, so choose him carefully.

The second category of beginner is the complete, or semi-hobbyist, who wants to learn something about computers, or wants to get into the absolutely fascinating world of programming. I won't tell you it's addictive, but there's no known cure for a computer hobbyist. Some spouses have achieved a partial cure with a baseball bat, but not without certain side effects. My warning is that you will need to learn. Learning to program is like learning to type or receive Morse code. A serious effort will pay off in about the same length of time — a month, maybe two.

A very disappointed writer in the *Wall Street Journal* a couple of years ago considered the whole "personal computer thing" hype when he couldn't write a complex program the first night. As I recall, he typed, "My name is John" and, "Hello," and all he got was "syntax error". Hrumph! Who needs it? (I wonder how he'd review a piano?)

So, just be aware that there is a learning period — not long, but it's there. Professional programmers, obviously, spend much more time and effort and get formal intensive training in design, coding, and many areas the hobbyist won't usually explore in depth.

Excellent books are available today (some even come with the computers) which will teach you all you need to know to enjoy learning, and to do some programming you'll find fun and useful. Our Color Computer Learning Lab is one of the best courses we've seen. It goes beyond just teaching you how to use each command, and gets into the "now that I know the commands, how do I use them" area. We'll be doing more courses like it. Check your local computer club, too.

Hardware is, of course, a subject of great concern to users. Our philosophy with Model I and Model III was to provide a low-cost system which could be expanded in easy-to-take steps, all the way up to our biggest and most powerful business system in that class. A $699 4K, Level I Model III (yes, they're still available on special order) will expand to a 48K RAM, 20-plus megabyte hard-disk system capable of running a business.

Each manufacturer has his own philosophy, and there is something to fit everyone. We think service is of extreme importance, and it's available through any Radio Shack Store, and our chain or more than 200 Tandy-owned service centers nationwide, that fix TRS-80s. When you need software, add-ons, or just some advice, we have over 6000 stores in the U.S.A. to help.

We happen to think there's never been a better time to get started in computing, no matter what your reason. It's a whole new business tool, bringing advantages to the small businessman that he wouldn't have dreamed of ten years ago. It's a tool of the mind for efficiency. It's productivity for the busy manager or professional. And for the hobby types, it's stimulating, fascinating, and educational. The art of programming teaches youngsters to solve problems with a structured, organized approach. Above all, it's fun.

End of commercial and column. Good luck, and good computing. ■

Files and foibles

Multi-disk handling techniques

Models I/II/III, PMC-80, LNW80

T. R. Dettmann, Associate editor

Over the past two years, I have been trying to lay out simple techniques for advanced programming on the TRS-80, concentrating on, but not limiting myself to, random access files. In that time, I've received many questions about various techniques. A common one has been, "What do I do when I run out of disk space?" My answer is, "Why run out of disk space?"

When dealing with files that are expected to be very large, it makes sense to provide for the possibility of exceeding the disk space available to a single file. You might take the approach that Profile uses in making segments of each record occupy different disks. But what if you simply have short records which don't easily break up into segments?

There's another answer. Why not use the multi-disk capability you have on your system, and place files on each disk which look to the user as if they were a single file? Does that sound too hard? Well, it's not. Galactic Software did a beautiful job with that in their Model II mass mail program.

What we're going to do here is present some simple techniques for achieving the goal of making several files look like one. The demonstration program creates three files and allows you to add, edit, delete, and print records as if they were a single file. It's not as sophisticated as Galactic's program, but it amply demonstrates that even *you* can do it.

How Is It Done?

To make three files look like one, we have to organize our access technique so that it knows about three files in a way that the user never sees. To do this, we create an index.

We've used indexing techniques over and over again since the beginning of Files and foibles. Each time, though, we've indexed a single file. In a single file index, we need two pieces of information: the key we're going to look for, and the disk record number in the file where we're going to find the data the key is for. To make a multiple file index, we have to add information about which file the entry refers to.

An index also needs a structure. We could store the index as a binary tree, a linked list, a hash table, or any other form that will allow us to find things when we want them. The example uses a linked list structure to maintain the index in sorted order at all times. This isn't the most efficient, but it is sufficient for the demonstration.

Once we have a structure for our index file, we can store information in any file by locating free space in the files, putting the information there, then adding it to the index.

It sounds simple enough when we say it, but a practical example will make it more concrete. The sample program creates three files which will have names of seven characters plus a number 2, 3, or 4. These are the data files which will hold our information. In the sample program, they have a ten-character identifier (the key) and a 100-character data field in each record. No attempt has been made to limit the record size or use sub-records.

Two other files are also created with the same name as the other files, but different extensions. One file (/IDX) is the linked list index file. The other (/STK) is a push-down stack that will be used to remember where deleted records are.

The five files can appear anywhere on your disks. For example, on a four-disk system, the index and stack files could be kept on drive zero and the two, three and four data files could be on drives one, two, and three, respectively.

The Program

Let's look at the program one section at a time to see how it operates. The lines through 170 initialize the program by establishing arrays for the data fields, useful functions, as well as some strings. The string LT$ will be used to check for allowed characters in filenames and PC$ is the list of possible commands for the program.

The Main Control Loop

The main program is contained in lines 200-250. It follows this pattern:

1. Get a command line.
2. Interpret it and check for errors.
3. Execute the command.
4. Repeat the process from step 1.

This is a very common pattern for programs and can be implemented in many ways. In this case, command lines will consist of a keyword (one of those in string PC$) and a modifier which may, or may not, be needed.

The subroutine in lines 300-380 decodes the command line into two strings, C1$ and C2$. C1$ is the keyword command and C2$ is the modifier. Subroutine 400 is used by 300 to remove leading and trailing blanks before processing.

The major subroutines in the program are at lines 1000, 2000, 3000, 4000, 5000, and 6000. Subroutine 5000 is a simple HELP routine that displays the allowed commands for someone using the program.

As Easy As...

TRS-80 Disk & Other Mysteries

by H.C. Pennington

This book is the definitive authority on data recovery for the TRS-80 Model I disk system. In almost every case, lost data can be recovered and this book tells you how to do it. From clobbered directories to parity errors, this profusely illustrated data recovery cookbook includes examples and step-by-step instructions for both beginners and professionals.

132 pages. $22.50

Microsoft BASIC Decoded & Other Mysteries

by James Farvour

This book is the only one of its kind. You will learn how the TRS-80 Model I BASIC interpreter works from power-up to power-down. It has the complete disassembled Microsoft BASIC Level II interpreter with over 13,000 lines of detailed comments in one volume. The math, arithmetic and utility routines are fully explained with examples on how to use them. An incredible learning aid for the beginner and a valuable reference for the professional.

312 pages. $29.95

The Custom TRS-80 & Other Mysteries

by Dennis Bathory Kitsz.

A comlete hardware/software how-to-do-it like reverse video, high res graphics, audible keystrokes, 8-track mass storage music synthesizer, and real time clock. Dozens of project hints, tips, and fixes complete with the interface software. If you are a beginner, there are chapters on wire-wrap, soldering and construction methods. If you are a professional, there is a wealth of technical detail.

332 pages. $29.95

BASIC Faster and Better & Other Mysteries

by Lewis Rosenfelder

BASIC is not nearly as slow as you think! This book shows you how to make it faster and better with techniques and tricks that you never dreamed of for Model I, II and III disk systems. You won't find trivial, poorly designed "checkbook" programs here – only useful BASIC techniques and code ready for use in YOUR programs. Tutorial for the beginner, instructive for the advanced, and invaluable for the professional. All routines are available on disk.

290 pages. $29.95

Learn about your TRS-80 computer – it's as easy as 1–2–3–4. These books are easy to read and understand because they are written for you by people just like you. Here are answers and solutions without jargon. There are no end of chapter "tests" – no incomprehensible math formulas – no textbook explanations – just straightforward, plain English.

You will learn how to recover "lost data" on disk, how the BASIC interpreter works, how to make BASIC run faster and better, and how to modify and interface your TRS-80. In other words, you will learn how to make your computer do all the things you want it to do. It's as easy as 1–2–3–4.

Available at computer stores, B. Dalton Booksellers and independent book dealers. **BASIC Faster and Better** is also available at all RADIO SHACK Computer Centers and selected RADIO SHACK stores. (Cat. No. 62-1002) If your dealer is out of stock, order direct. Include $4.00 for shipping and handling. Foreign residents add $11.00 plus purchase price, in U.S. funds.

TRS-80 trademark TANDY Corp.

JG

1953 West 11th Street Upland, CA 91786 # 50

The FILE Command, Open Files

Normally, the first subroutine called is 6000. This subroutine makes a file ready for use by the program. C2$ is first checked to see if it will be an allowable filename (<=7 characters, all allowed). If the tests are passed, F$ is set to C2$, any files that are open are closed, and a GOSUB500 is done to open the new set of files.

Subroutine 500-590 is called from the major subroutine 6000 to open a set of files, field them for use, and initialize them if necessary. F$ is a filename passed to it which serves as a root name for all the files (much like Profile). The parameter FO is set to one, to indicate that this routine has been executed. It serves as part of the error check in line 220 to prevent executing file commands without having a file open for use.

The ADD Command: Adding Data

Once the files are open, we can add items to the files. Subroutine 1000 controls the process of adding items to the file, but it is very simple. C2$ is the key when adding to a file. Then, subroutine 1000 gets the data (DA$) and passes control to subroutine 600 to actually add the data.

If we follow the process of adding a new item, it will lead us through a tangle of requirements, each met by a single subroutine. Starting at subroutine 600, we first look in the index to see if the item is already there (subroutine 700). If it is, EF will be returned as one. In that case, we simply replace the data record with the new data.

If we don't find the item (EF=0), LG will be the location of the last linked index location before where the new item is to go. We next need to find storage for the new item (subroutine 800).

Subroutine 800 first looks at the stack file (file number five) to see if there are any deleted records waiting to be used. If there are, it takes the first available one and returns it as the location of the new record. If there is no deleted record, the program looks at each of the three data files to see if it is less than its maximum allowed length. (LF, the maximum length, is arbitrarily set to ten for demonstration purposes. It could be set differently for each file, based on disk tests or whatever.) If a file is less than its allowed length, that file, and the next record number in the file, are returned as the place to put the new record.

If a location is found to put the new record, the EF is again set to zero. However, if no space is available, EF is set to one, and an error message is printed.

Once subroutine 800 is done, if it found a record to put the new data in, subroutine 900 will be executed to add the item into the linked list in the index.

Subroutine 900 calls subroutine 960 to find an empty index record to put the new index record in. It simply searches through the file to find an empty record and if it doesn't find one, it adds it to the end of the file. Once a

record is found, the record before it, in linked order, is called into memory. The link is updated and the new index record is written to disk.

The EDIT Command: Editing Existing Data

Subroutine 2000 is provided to edit an existing record. First, we use subroutine 700 to look for the item we want (C2$). If it isn't found, an error message is printed and we return to the command loop. Otherwise, we get it, display it, and give the user a chance to change the data (but not the key). Just pressing <ENTER> will cause no changes, but anything typed will replace the existing data.

The DELETE Command: Deleting A Record

To delete something, we use subroutine 700 to locate it. If it's found, we display it and make sure it is really to be deleted. If it is, the data record is blanked (line 3050), the record number is saved in the stack file (subroutine 650), and then it's removed from the index and the index record is set to zero to mark it for future use.

The PRINT Command: Printing the File

To print the file, we print a heading and then follow the linked list through the file to print everything in alphabetical order. To follow a linked list, we follow the steps:

1. Start with the header record (1).
2. Get the current record's link field.
3. If the link is zero, we're done.
4. If the link isn't zero, get the record with that link number.
5. Get the data record pointed to by the current index record.
6. Print the data.
7. Repeat, starting at step 2.

We've covered a very simple method for making multiple files look like a single file. This is particularly useful if you are dealing with files that have the potential to grow larger than your disk capacity. This program could be extended to create disk size files for more drives than you have available. It could know what files are on-line, and ask for disks to be mounted as needed.

To make practical use of the program, you would certainly want to improve the add routine to store the information you want, and possibly change the indexing technique to the one most suited to your problem. No matter what use you finally make of the technique, have fun with it.

Program Listing for Files and Foibles

```
10 REM*******************
20 REM
30 REM LARGE FILE DEMO
```

Files and foibles

```
40 REM (C) 1982 BY TERRY R. DETTMANN
50 REM
60 REM VERSION 0.010/82
70 REM FILENAME: BIG/BAS
80 REM
90 REM**********
100 CLEAR 10000: DEFINTA-Z
105 REM SET UP NUMBER OF LINES (NL) AND COLUMNS
(NC) ON VIDEO
106 REM LF WILL BE AN ARBITRARY LIMIT ON THE NUMBER
OF RECORDS
107 REM PER FILE, SET TO 10 FOR DEMONSTRATION
108 REM FO IS ZERO IF NO FILE IS OPEN
110 NC=80: NL=24: LF=10: FO=0
115 REM DATA ARRAYS FOR FIELDING
120 DIM I$(4),D$(4)
125 REM SOME USEFUL FUNCTIONS FOR HEADINGS
130 DEFFNHDR$(X$)= STRING$ ((NC- LEN(X$)-2)/2,
"=") +" "+X$+" "+STRING$ ((NC- LEN(X$)-3)/2, "=")
140 DEFFNCTR$(X$)= STRING$ ((NC- LEN(X$))/2," ")+X$
145 REM ALLOWED LETTERS FOR FILENAMES
150 LT$="ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789"
155 REMPC$ IS THE LIST OF POSSIBLE COMMANDS
160 PC$="ADD EDIT DELETE PRINT HELP FILE END CLS "
165 REM WELCOME TO THE PROGRAM!
170 CLS: PRINT FNHDR$("LARGE FILE DEMO"): PRINT
FNCTR$("NO FILE OPEN"): PRINT
200 REM - - - - - - - - - - MAIN PROGRAM - - -
- - - - - - - - - - - - - - - - -
205 REM GET A COMMAND
210 LINEINPUT "COMMAND ==> ";CD$
215 REM DECODE IT AND ERROR TRAP
216 REM IF NO FILE IS OPEN, THEN ONLY FILE & HELP
COMMANDS ARE
217 REM ALLOWED
220 GOSUB 300: IF CD=0 OR (FO=0 AND (CD<>6 AND
CD<>5))THEN PRINT "COMMAND ERROR, TRY HELP":
GOTO 210
225 REM 7 IS THE CODE FOR END OF PROGRAM, 8 IS A
CLEAR SCREEN
230 IF CD=7 THEN CLOSE: CLS: END ELSE IF CD=8 THEN
CLS: GOTO 200
235 REM EXECUTE THE COMMMAND
240 ON CD GOSUB 1000, 2000, 3000, 4000, 5000, 6000
245 REM GO BACK FOR MORE
250 GOTO200
300 REM - - - - - - - FIGURE OUT WHICH
COMMAND IT IS - - - - - - - - - - - - - - -
305 REM STANDARD - GET RID OFF ALL LEADING &
TRAILING BLANKS
310 GOSUB 400
315 REM IS THERE A BLANK LEFT? IF SO THEN THERE ARE 2
WORDS
316 REM IF THERE IS NO BLANK, THEN SET THE SECOND
WORD TO NOTHING
320 L=INSTR(CD$," "): IF L=0 THEN C1$=CD$: C2$="":
GOTO 360
325 REM SEPARATE OUT THE 2 WORDS, C1$ IS THE
COMMAND,
326 REM C2$ IS A MODIFIER (A KEY FOR STORAGE)
330 C1$= MID$(CD$,1,L-1): CD$= MID$(CD$,L+1):
GOSUB 400
```

```
340 C2$=CD$
345 REM SINCE WE'VE ONLY ALLOWED A 10 CHARACTER
KEY IN THE INDEX
346 REM FILE, CHECK FOR IT
350 IF LEN(C2$)>10 THEN PRINT "NAME TOO LONG":
CD=0: RETURN
360 CD= INSTR(PC$,C1$): IF CD=0 THEN RETURN
370 CD=INT(CD/6)+1
380 RETURN
400 REM - - - - - - - - - - - - REMOVE LEADING &
TRAILING BLANKS - - - - - - - - -
410 IF LEN(CD$)=0 THEN RETURN
420 IF MID$(CD$,1,1)=" " THEN CD$= MID$(CD$,2):
GOTO 410
430 IF LEN(CD$)=0 THEN RETURN
440 IF MID$(CD$,LEN(CD$),1)=" " THEN CD$=
MID$(CD$,1, LEN(CD$)-1): GOTO 430
450 RETURN
500 REM - - - - - - - - - - - - OPEN FILES - - - - -
- - - - - - - - - - - - - - - - -
510 OPEN "R",1,F$+"/IDX",16: OPEN "R",5,F$+"/STK",4:
FO=1
520 OPEN "R",2,F$+"2/DAT": OPEN "R",3,F$+"3/DAT":
OPEN "R",4,F$+"4/DAT"
525 REM INDEX FILE
530 FIELD 1,10 AS ID$,2 AS DK$,2 AS RN$,2 AS LK$
535 REM DATA FILES
540 FIELD 2,10 AS I$(2),100 AS D$(2):FIELD3,10 AS
I$(3),100 AS D$(3)
550 FIELD 4,10 AS I$(4),100 AS D$(4)
555 REM STACK FILE FOR DELETED RECORDS
560 FIELD5,2 AS DS$,2 AS RS$
565 REM INITIALIZE FILES IF NEEDED
570 IF LOF(1)=0 THEN LSETID$="": LSETDK$=MKI$(0):
LSETRN$=MKI$(0): LSETLK$=MKI$(0): PUT1,1
580 IF LOF(5)<1 THEN LSETDS$=MKI$(1): PUT5,1
590 RETURN
600 REM - - - - - - - - - - - - ADD OR REPLACE DATA
- - - - - - - - - - - - - - - - - -
605 REM LOOK FOR THE ITEM
606 REM IF NOT THERE, THEN GET SPACE FOR IT & LINK IN
607 REM IF NO SPACE LEFT THEN CAN'T GO ANY FURTHER
610 GOSUB 700:I F EF=0 THEN GOSUB 800: IF EF=0 THEN
GOSUB 900 ELSE RETURN
615 REM STORE THE DATA
620 LSET I$(DK)=C2$: LSETD$(DK)=DA$: PUTDK,RN
630 RETURN
650 REM - - - - - - - - - - - - PUSH STORAGE ON
STACK - - - - - - - - - - - - - - -
655 REM FOR DELETED RECORDS
660 GET5,1: SP=CVI(DS$)+1: LSETDS$=MKI$(SP): PUT5,1
670 LSETDS$=MKI$(DK): LSETRS$=MKI$(RN): PUT5,SP
680 RETURN
700 REM - - - - - - - - - - - - FIND AN ITEM - - - -
- - - - - - - - - - - - - - - - -
705 REM START AT THE HEAD (RECORD 1)
710 GET1,1:LK=1
715 REM FOLLOW THE LINKS TO EACH NEW RECORD TO
THE END (LK=0)
720 LG=LK: LK=CVI(LK$): IF LK=0 THEN EF=0:RETURN
725 REM GET THE RECORD
730 GET1,LK
```

Files and foibles

```
735 REM IF IT'S WHAT WE WANT, THEN WE'RE DONE
740 IF INSTR(ID$,C2$)<>0 THEN EF=1: DK=CVI(DK$):
RN=CVI(RN$): RETURN
745 REM IF WE'VE PASSED IT, THEN IT ISN'T THERE
750 IF C2$<ID$ THEN EF=0:RETURN
760 GOTO 720
800 REM - - - - - - - - - - - FIND STORAGE - - -
- - - - - - - - - - - - - - - -
810 EF=0
815 REM FIRST, ARE THERE DELETED RECORDS ON THE
STACK?
820 GET5,1: IF CVI(DS$)<=1 THEN 850
825 REM IF YES, THEN RETURN THAT RECORD NUMBER
830 DS=CVI(DS$)-1:LSETDS$=MKI$(DS):PUT5,1
840 GET5,DS+1: DK=CVI(DS$): RN=CVI(RS$): RETURN
845 REM NOTHING ON STACK, SO FIND FIRST FILE WITH
SPACE FOR IT
850 FOR DK=2 TO 4: IF LOF(DK)<LF THEN
RN=LOF(DK)+1: RETURN
860 NEXTDK
865 REM NO SPACE IN ANY FILE
870 EF=1: PRINT "ERROR - SYSTEM FULL":RETURN
900 REM - - - - - - - - - - - LINK RECORD INTO LIST
- - - - - - - - - - - - - - - -
905 REM GET SPACE FOR THE INDEX RECORD
910 GOSUB 960
915 REM LINK THE NEW INDEX RECORD TO THE LAST ONE
920 GET1,LG: LK=CVI(LK$): LSETLK$=MKI$(IR): PUT1,LG
925 REM NOW PUT IN THE NEW RECORD
930 LSETID$=C2$: LSETDK$=MKI$(DK):
LSETRN$=MKI$(RN)
940 LSETLK$=MKI$(LK): PUT1,IR
950 RETURN
960 REM - - - - - - - - - - - FIND AN EMPTY
INDEX RECORD - - - - - - - - - - -
965 REM FOR THE DEMO, WE'LL JUST DO A LINEAR
SEARCH FOR AN OPEN RECORD
970 IF LOF(1)<2 THEN IR=2: RETURN
980 FOR IR=2TO LOF(1): GET1,IR: IF CVI(DK$)=0 THEN
RETURN
990 NEXT IR: RETURN
1000 REM - - - - - - - - - - - ADD DATA - - - - -
- - - - - - - - - - - - - - - - -
1005 REM GET SOME DATA THEN SIMPLY STORE IT
1010 LINEINPUT "DATA ===> ";DA$
1020 GOSUB 600: RETURN
2000 REM - - - - - - - - - - - EDIT DATA - - - -
- - - - - - - - - - - - - - - - -
2005 REM EDITING IS MORE COMPLEX, FIRST LOOK FOR
THE
2006 REM KEY, IF IT ISN'T THERE THEN ERROR
2010 GOSUB 700: IF EF=0 THEN PRINT"NOT IN THE FILE":
RETURN
2015 REM GET THE DATA AND DISPLAY IT
2020 GET DK,RN: PRINT D$(DK)
2025 REM ASK FOR CHANGES
2030 LINEINPUT "CHANGES ==> "; IN$
2035 REM NO CHANGES MADE IF JUST HIT ENTER
2040 IF IN$="" THEN PRINT "NO CHANGES MADE":
RETURN
2045 REM MAKE THE CHANGES
2050 LSETD$(DK)=IN$: PUTDK,RN: RETURN
```

```
3000 REM - - - - - - - - - - DELETE DATA - - - -
- - - - - - - - - - - - - - - - - -
3005 REM LOOK FOR THE ITEM
3010 GOSUB 700: IF EF=0 THEN PRINT"NOT IN FILE":
RETURN
3015 REM GET AND DISPLAY IT
3020 GET DK,RN: PRINT I$(DK): PRINT D$(DK)
3025 REM DOUBLE CHECK TO BE SURE IT'S THE RIGHT
ONE
3030 INPUT "ARE YOU SURE (Y/N)"; YN$
3040 IF YN$="N" THEN PRINT "NO CHANGES": RETURN
ELSE IF YN$<>"Y" THEN 3030
3045 REM BLANK OUT THE DATA
3050 LSETI$(DK)="": LSETD$(DK)="": PUT DK,RN
3055 REM REMEMBER WHERE THE DATA RECORD WAS
3060 GOSUB 650: LP=LK
3065 REM BLANK OUT THE INDEX RECORD (REMEMBER
THE LINK THOUGH)
3070 LSETID$="": LSETDK$=MKI$(0): LSETRN$=MKI$(0):
LK=CVI(LK$): LSETLK$=MKI$(0)
3080 PUT 1,LP: GET 1,LG
3085 REM LINK AROUND THE DELETED INDEX RECORD
3090 LSETLK$=MKI$(LK): PUT 1,LG: RETURN
4000 REM - - - - - - - - - - PRINT DATA - - - -
- - - - - - - - - - - - - - - - - -
4010 PRINT "DSK REC INDEX KEY KEY DATA"
4020 GET 1,1
4030 LK=CVI(LK$): IF LK=0 THEN RETURN
4040 GET 1,LK
4050 DK=CVI(DK$): RN=CVI(RN$)
4060 GET DK,RN
4070 PRINTUSING"### ### ";DK,RN,ID$,I$(DK),D$(DK)
4080 GOTO 4030
5000 REM - - - - - - - - - - HELP COMMAND - -
- - - - - - - - - - - - - - - - - -
5010 PRINT: PRINT
5020 PRINT TAB(10) "ALLOWABLE COMMANDS ARE:"
5030 PRINT TAB(15) "ADD <NAME> WHERE <NAME> IS
10 CHARACTERS OR LESS"
5040 PRINT TAB(15) "EDIT <NAME>"
5050 PRINT TAB(15) "DELETE <NAME>": PRINT
TAB(15)"FILE <FILENAME>"
5060 PRINT TAB(15)"PRINT "
5070 PRINT: PRINT
5080 RETURN
6000 REM - - - - - - - - - - OPEN A FILE - - - -
- - - - - - - - - - - - - - - - - -
6005 REM FILENAME CAN ONLY BE 8 CHARACTERS
6006 REM AND SOME FILES WILL HAVE A CHARACTER
ADDED
6010 IF LEN(C2$)>7 THEN PRINT "FILE NAME ERROR -
TOO LONG": RETURN
6015 REM CHECK FOR ILLEGAL CHARACTERS
6020 FOR I=1 TO LEN(C2$): C$=MID$(C2$,I,1)
6030 L=INSTR(LT$,C$): IF L=0 THEN PRINT "FILENAME
CHARACTER ERROR": RETURN
6040 NEXT I
6045 REM F$ IS WHAT WE'LL OPEN
6050 F$=C2$: PRINT "OPENING FILE SERIES "+F$
6055 REM CLOSE ANY OPEN FILES, THEN OPEN THE FILES
NEEDED
6060 CLOSE: GOSUB 500: RETURN ■
```

Scripsit your programs

An easy way to edit on your Model II

Model II, two drives

Waldo T. Boyd, Geyserville, CA

There is a remarkably easy way to check the variables in a BASIC program written on the TRS-80 Model II, and to replace them with others as desired. This method has not been noted either in the TRSDOS Owner's Manual or in the Scripsit reference manual, and therefore, may well escape the attention of many users of this business-level computer.

Scripsit 2.0 provides an efficient "global search and replace" function, to be found beginning on Page 22 of the reference manual. With this function, strings may be found sequentially, observed, passed, deleted or replaced. The choice of search string runs from a single letter through the numerals and "shift" characters in any combination, to words and phrases. Thus, variables such as A, A$, AA$, A1$, I% and hundreds of others in a BASIC program may be examined, changed throughout the program, or deleted without risk of missing one or more. Keeping variables in order is a particularly irksome problem when writing a long program, even when care is used in selecting them.

Another helpful use for this feature of Scripsit is in exchanging the PRINT statements that are generally employed while writing and debugging a program for LPRINT statements that will print out program copy. While there are short utility programs available for this purpose, the "global" function of Scripsit is right at hand.

A further use for Scripsit while writing programs involves the format line. TRSDOS format provides a full-screen line with wrap-around. Publication in in some magazines, for instance, requires the line to be thirty-nine characters in length, with second and subsequent lines indented two spaces. Writing to format order can be a chore; with change of format made easy through Scripsit's "Block Format" capability (Reference Manual, Page 15) the program can be written without regard for format, and then formatted exactly to publishing requirements with ease. Formatting can be done while picking up the program for the Global Search and Exchange function to save the Block Format step, if the requirement is known in advance.

How to do it

The BASIC program is usually filed as written, on the Drive 0 disk, but this is not a requirement. SAVE the program to disk in ASCII, as described in the BASIC section of the TRSDOS manual, using a "filespec" (file name) that is easily remembered, such as: SAVE "MOONGLOW/TXT:1", A. The ", A" files your program in ASCII.

Place the disk containing the file in one of the Expansion Unit drives. Insert the Scripsit 2.0 disk in Drive 0 (the CPU drive slot), and RESET. Answer the opening prompts until the directory appears.

Open a new document, preferably using the "filespec" name selected for the BASIC program, as a memory aid. Open this document in the same manner as with any new Scripsit document.

At this point the format line may be changed to fit publication requirements, using the Format Line Editing function of Scripsit, Page 40. To fit publishing manuscript requirements the "o" will be placed two spaces outside of and to the left of the left bracket. The "cursor:" counter below the format line will count the spaces. Place a right bracket 48 spaces to the right of the "o" marker, then <ENTER>. Be sure the mode indicator at right below the format line shows "Mode:O" (<CTRL> o will toggle this function).

Keyboard the following:

<CTRL> U C

The "Convert Document" prompts will appear on screen. The first prompt is answered with a Capital A. The second is answered with the exact name of the SAVE'd file (the "filespec"). Do *not* use quotation marks around this entry of the name. The above filespec name would be entered in this manner:

MOONGLOW/TXT:1 <ENTER>.

If the TRSDOS disk is in another drive than 1, use that drive number following the colon.

The BASIC program will now be

copied to the Scripsit disk, with the original line format changing to fit the page format selected. The program appears on screen, page by page.

When the program is all in the Scripsit document, open the Scripsit reference manual to Page 22, Global Replace, and follow the instructions.

It is best to run through each search string once, just looking. There may be uses for the string selected which were not anticipated when the program was in writing. Also, some strings show up as portions of words having nothing to do with variables. A better definition of the string may be in order, such as providing a space before, after or both before and after the string. Experimentation is the best teacher at this point.

After a satisfactory "find only" run-thru is made, the "F" global search prompt may be exchanged for an "R" if substitution is desired, and the string will be automatically exchanged wherever it appears with the next run-thru.

Any wording additions, alterations or deletions (corresponding to TRSDOS BASIC line editing) may be done while working on the document in Scripsit, and will be functional as a command line when returned to the BASIC file, provided all programming rules are followed.

Note that a count is made of each string as it ends its search through the program. This can be quite useful. Also, there are times when extra variables are unnecessarily used; they will begin to show up when seen in context with strings during search. For instance, "FOR S=1 to 10" may have been written when other "FOR...NEXT" loops have used "Xs," anticipating some kind of conflict that does not pan out. The global search provides means for replacing the "S" with an "X" and checking to see that any following use of "X" is initialized to its required value.

When all work is completed, the document is returned to TRSDOS. If the identical filespec (file name) is used, the "file already exists" prompt will appear, so use a variation on the name, as for instance follows:

<CTRL>U C

The "Convert Document" prompts reappear. Answer:

GLOWMOON/TXT:1<ENTER>

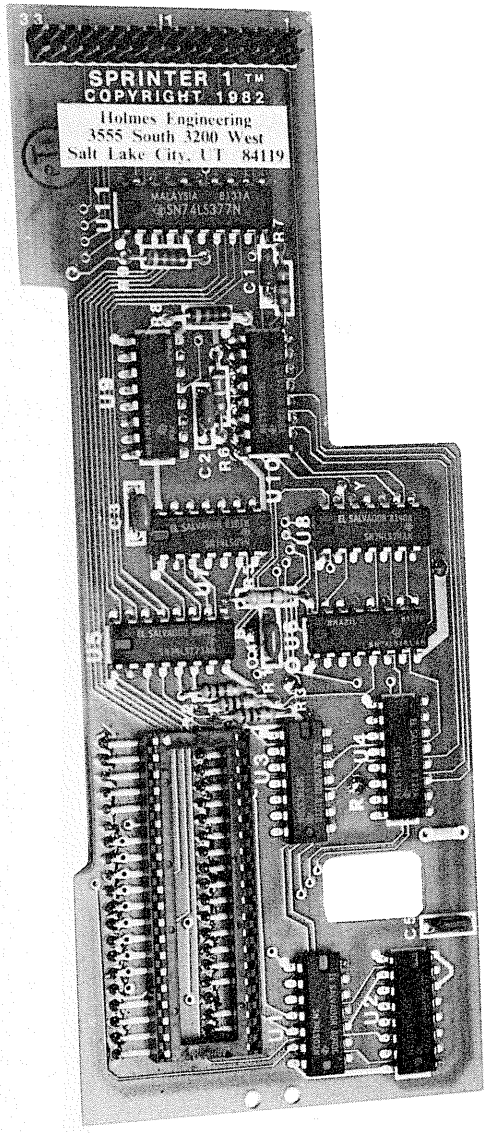
The revised document will now be filed to the TRSDOS disk, but in the original TRSDOS format. It may be RUN after booting the TRSDOS disk and calling up BASIC appropriately. The Scripsit document remains intact, and the PRINT commands of Scripsit may be used as with any plain-language document to print the program in hard copy conforming to the new format. ■

Sprinter II

Triple your CPU speed with this board from Holmes Engineering

Models I/III, PMC-80, LNW80

Jim Klaproth,
Associate editor



It seems that nowadays the owners of TRS-80 microcomputers have an unlimited selection of add-on hardware to enhance their tiny machines. Any day now, I expect to be able to upgrade my Model I to a Model III just by plugging in a board. My Model I is already so crammed full of hardware modifications that it is becoming difficult to even get all the guts back in the case. However, I had never taken the plunge and installed a speed-up modification. When I saw the ad for a new speed modification that simply plugged in, already had the faster Z-80B chip on it, and allowed operation at 5.3 Mhz, with an unconditional thirty-day money back guarantee, I simply had to try one out.

A slow clock

One of the limiting factors of the Models I and III is the slow clock speed of the the CPU. The Z-80 microprocessor is rated for 4.0 megahertz (Mhz) operation, but Radio Shack chose to run it at 1.77 Mhz on the Model I and 2.03 Mhz on the Model III. By simply allowing the Z-80 to run at its full rated speed, we can see a two-fold improvement in actual computing time. Taking it one step further, if we triple the clock speed we should get a three-fold improvement. The only way to accomplish this is to replace the Z-80 chip with a Z-80B, which is rated for 6.0Mhz. Enter the Holmes Engineering Sprinter II. What this board does is to take the original clock signal, which is at 10.6445 Mhz (Model I) and send it through a programmable divide-by counter. This counter then divides the 10.6445 Mhz pulses by 2, 3, 4, 5, 6, 7, 8 or 9 to arrive at 5.32, 3.55, 2.66, 2.13, 1.77, 1.52, 1.33 and 1.19 Mhz respectively. These clock pulses are then routed to the Z-80B mounted on the Sprinter II, which replaces the original Z-80 in the TRS-80.

A software gear shift

The actual clock frequency is determined by a simple software command that can be given in BASIC or machine language. The BASIC command is simply: OUT 254,(X), where X is an integer from 0 to 15. That means there are actually sixteen separate combinations available to the user. There are eight different clock speeds, each with the option of running with or without wait states. Running with wait states means that a small delay will be added to every memory access, slowing down the actual speed by about 10%. Holmes guarantees that any computer will run at 3.55 Mhz with wait states. Anything above that speed, it is hit or miss.

Hardware limitations

Every machine is different and if even one component is marginal, it may interfere with high-speed operation. The maximum speed that an unmodified TRS-80 will run is 5.32 with wait states. When I called Holmes to get the information on the device, they asked what my configuration was. I told them I had a suffix G keyboard with 3 chip ROM set, expansion interface with buffered cable and DIN plug, with a mix of Motorola and NEC RAM chips. The technician told me that my machine would probably have to be modified in order to run at 5.32 Mhz, but to try it out unmodified first. I did, and was pleasantly surprised to discover that it worked perfectly!

Installation

Installation was a snap. The Sprinter II simply plugs into the socket after removing the Z-80 from the computer. Then there are four wires to attach to the computer

circuit board. This is accomplished with solderless clips that simply push on. If you are skillful with an iron, you can cut off the clips and solder if you wish. Several photos are supplied with the documentation to aid you in locating the proper points to fasten the wires. It took me about twenty minutes to do the job. There is one modification to the board that is recommended for all machines that involves simply cutting a trace on the circuit board. There are other modifications in the documentation, but none of them applied to my machine. If you have trouble getting it to work properly, Holmes will recommend solutions or they will modify your interface for only $25.

Testing the machine at various speeds involves simply issuing the command OUT 254,(X) and then running a program to see if it crashes. My machine ran at every speed except at 5.32 Mhz without wait states. I have tested out every one of my programs using 5.32 Mhz with wait states and none of them have crashed. I am also happy to report that Holmes has provided an option of hardware switch to disable or enable the slowdown of the clock during disk accesses. Since NEWDOS/80 will run successfully with no slowdown, I decided to order this option with the unit. By setting SYSTEM option BJ=3 and disabling slowdown, I can now run at three times normal speed all the time. It seems to speed up disk I/O considerably and is required to run eight-inch disk drives in double density.

Holmes also has an option of an on-board parallel printer port. This is for those who do not have an expansion interface. It is the same as the Centronics port on the expansion interface; however, the edge connector is different. It uses the same connector as the Model II does, so a Model II cable will work. Holmes will supply a cable for only $19.50 versus $39.95 for the Model II cable. It uses the same commands, LIST and LLIST, as the standard printer port and no initialization commands are necessary as in the speed-up modification. Holmes also makes a 48K memory expansion board that plugs right into your keyboard.

After running all of my programs without a single glitch at triple speed, I can say without reservation that I highly recommend this product. The improvement in computing speed is fantastic! Graphics games in BASIC are greatly enhanced but sometimes delay loops will be needed to slow them down. All in all, I was very impressed with the quality of the hardware, the helpfulness of the staff, and (best of all) the performance of the product. Now, Holmes, how about that Model III converter board? ■

Ed. note — The Sprinter II is available from Holmes Engineering, 3555 South 3200 West, Salt Lake City, UT 84119. The price is $99.95. The optional Parallel Printer Port is $24.50.

BASIC bits

String packing rules and printer/video routing

Models I/III, PMC-80, LNW80

Thomas L. Quindry, Burke, VA

For the last two months, I have been explaining string packing techniques. Before we leave string packing you should be aware of the following key points:

1. String packing is not confined to graphics. You can pack machine language code to perform functions from within your BASIC program. Some uses include sound generation through the cassette port and small utilities. Use the VARPTR function to find the location of your string and use it to define the starting address for the USR function. For Level II BASIC, POKE 16526, PEEK (VARPTR (?)+1): POKE 16527, PEEK (VARPTR(?)+2). For disk BASIC, calculate the address = PEEK (VARPTR (?)+1) + PEEK (VARPTR (?)+2) * 256 as we have done in previous columns. Then use the form, DEFUSR=nnnn where ? is the string variable name and nnnn is the computed address, respectively.

2. When using graphic characters which travel across the screen, you will need a null graphic character. That is, a graphic character the same size and shape as the one to be displayed but containing all spaces or ASCII character 128 (80H). When you want your character to move, you must make it disappear from its previous location by printing the null character in its current position. In the jumping men example given last month, this was not needed. A null character for the man would be thirty-one characters with all the graphic codes (the three-digit codes) replaced with 128. The other numbers, 26s and 24s would remain the same.

3. You cannot use an ASCII 0 (0H) or an ASCII 34 (22H) within the string function. The BASIC interpreter sees the 0 as the end of a BASIC statement and the 34 computes to the quote mark. If you need a register defined to zero use the assembly language command XOR A to set the A register to zero and then define the proper register, say C, to zero by a command LD C,A. The 34 is a little harder. Its main use is in the command LD (nnnn), HL. Instead do a PUSH HL, POP BC, LD (nnnn), BC. The hardest thing to keep away from is defining the address nnnn which may contain 0H or 22H in part of its code.

4. Never edit a line that has been string packed. Due to BASIC's way of displaying tokenized words with their ASCII equivalent in the input buffer, some of your packed code will be changed to ASCII characters when you get out of the EDIT function. They will no longer be what you expect.

Another technique somewhat akin to string packing is to pack a remark statement. This makes it a little more difficult to locate the code for USR functions since the VARPTR function can't be used. And, of course, it can't be used for graphics in the sense that string functions can. One advantage is that the ASCII code 34 can be used within the packed code although the code 0 cannot.

The best way to use the remark statement to pack machine code is to define the first line of your program as a remark using the apostrophe (') form of remark statement. Again you must reserve space for your machine code to be packed using a form similar to that used for strings:

```
10 '123456789 1234567890
```

To locate your code use the start of BASIC pointer and add seven to it. The number seven derives from the way BASIC statement lines are represented in memory. The first two bytes at the start of BASIC gives the location of the next BASIC statement. This is followed by two bytes giving the current BASIC statement line number. Three more bytes are taken up by the coding for the (') type of remark statement signifier. Thus, in line 10 above, the digit 1 is at the start of BASIC + 7 bytes.

The start of BASIC pointer is located at address 16548 and 16549. As an example look at Listing 1. This program, which I call ZAP, sets up a machine language routine which whites out the screen. When alternately calling the defined USR function and CLS, the screen will flash on and off. This can be used in games to represent crashes or zaps when a wrong answer or move is entered by a game player.

Again, as in our example last month, the BASIC program statements needed to set up the packed code (lines 30 thru 80) are no longer needed and can be deleted from the program after the packed remark statement is set up by the first running of the program.

Listing 2 gives the assembly code used to white out the screen. One reason for listing this is to show how I avoided a potential problem with packing the code in my remark statement. If you recall rule three above, an

ASCII code 0 cannot be included within the packed code. The video screen starts at address 3C00H and ends at address 3FFFH. If you were putting this program in high memory rather than in a BASIC program statement, you would use the code:

LD HL,3C00H

LD DE,3C01H

instead of that given in lines 120 to 150. In defining HL, the ASCII equivalent is 21H, 00H, 3CH. As you can see, this would put an ASCII code 0 in our packed line of code. I got around this by using LD HL,3C01H, defining DE the same as HL by the PUSH and POP commands, then decrementing HL by one to make it equal to 3C00H. It accomplishes the same thing but avoids the deadly zero.

Question: I don't have a printer but want to use a program that has a large number of LPRINT statements in it. How can I keep the program from hanging up every time an LPRINT is reached without changing the program?

Answer: It's easy! If you want to disable the LPRINT statement do a POKE 16422,67: POKE 16423,0. On both the Model I and Model III computers, the address pointed to (0043H or 67 in decimal) contains the machine language code for RETURN. You just skip the LPRINT routine. As an alternative, do a POKE 16422, PEEK (16414): POKE 16423, PEEK (16415). This will route your LPRINTs to the video instead. A word of caution. The video location pointer is upset so it may not appear on the screen where you want it. Another way to route the lineprinter to the video would be to replace all of the values in the Device Control Block (DCB) for the LPRINT with the DCB for the video. That would be to POKE values from the consecutive addresses starting at 16421 through 16428 with the values in 16413 through 16420 respectively. This is not easy to restore to original values unless you memorize them.

Another trick is to route your PRINT statements to a lineprinter. This would be done by a POKE 16414, PEEK(16422): POKE 16415, PEEK(16423). Don't try this after routing the printer to the video as the proper value won't be at the memory address.

To restore to the proper vector values (for the LPRINT to go to the lineprinter again), POKE 16422,141: POKE 16423,5 for the Model I or POKE 16422,194: POKE 16423,3 for the Model III. The PRINT function is restored to the video by POKE 16414,88: POKE 16415,4 for the Model I and POKE 16414,115: POKE 16415,4 for the Model III.

Remember to send your questions and tips to me, in care of *80-U.S. Journal*, 3838 South Warner Street, Tacoma, WA 98409. Send a self-addressed, stamped envelope and I'll try to give you a personal handwritten reply as long as the answer is not too long and involved.

Listing 1

```
10 '123456789 1234567
20 M= PEEK (16548) + PEEK (16549)*256+7
30 FOR N=0 TO 16
40 READ A
50 POKE M+N,A
60 NEXT
70 DATA 205, 127, 10, 125, 33, 1, 60, 229, 209, 43
80 DATA 119, 1, 255, 3, 237, 176, 201
90 POKE 16526, M- INT(M/256)*256: POKE 16527,
INT(M/256) 'DEFUSR=M FOR DISK BASIC
100 FOR N=1 TO 10
110 X=USR(191)
120 CLS
130 NEXT
```

Listing 2
Assembly Code for White Out

```
100  CALL  0A7FH
110  LD    A,L
120  LD    HL,3C01H
130  PUSH  HL
140  POP   DE
150  DEC   HL
160  LD    (HL),A
170  LD    BC,03FFH
180  LDIR
190  RET ■
```

@ News

First and last — an informative farewell

ESF Owners

Spencer Hall, Associate editor

Here is the first issue of *80-U.S. Journal* for 1983 and . . . yes . . . the last issue of @ News. Dry your tears, friends, and I'll tell you why. The wise old men in the executive suite at *80-U.S.* (who may edit this all out anyway) have decided that, based on a statistical analysis of the incoming mail, two pages every month is just too much space to devote to a minority. This is partly a gut feeling, because nobody really knows how many active ESF users there are . . . or how many disk, tape or Models I, II and III owners there are either.

Rest assured that by no means will *80-U.S.* turn its back on ESF owners. The editors are committed to affirmative action! They promise to treat all minorities fairly. From time to time there will be articles concerning ESF. There is one in the works right now which contains a badly needed utility. Watch for it soon.

New Year's Day is a good time to confess sins and resolve to right wrongs. My sin has been the shabby way I have treated all the people who have written me letters of appreciation for my modest and often faulted efforts. Your patience has been long and this week it will be rewarded, I promise.

I'll devote most of this column to some candid and possibly valuable observations about Exatron Corporation. First I'll respond to the most common complaint in letters. "Why don't they get their act together?" Well folks, it just isn't an act. They're like that down there. I've seen several packets of info for new owners. Every one was different. All were crammed with valuable information, not the least of which is their toll-free hot line (1-800-538-8559). These packets are collated, apparently with the use of a random number generator, so as to make reference to them very difficult if you haven't sorted the stuff into categories. Many queries have come to me which could have been answered from this packet. More could have been cleared up by a free call to Sunnyvale.

Be advised: Hardware question? Ask for Jim Howell or "a technical person" on whatever equipment is bugging you. Software questions? Ask for Bill Burnham who knows nearly everything. If you manage to stump him, which isn't easy, he'll know whom to contact for the answer.

Actually, Exatron wears two faces these days. There is the old Exatron, an extension of Bob Howell, Sr.'s personality. It loves computers and programming and those who share this love. It supports the Owners Association, occasionally forgets to answer mail, always answers the Hot Line with generous assistance, and likes nothing better than to offer new and sophisticated utilities to run on its products.

The other face is a tough-minded business management which is trying, with remarkable success, to get ESF into other computers as OEM (that's jargon for "part of the package you buy") equipment. Financially, they've done wonders for what was frankly, until they arrived, one of the largest "cottage" operations in the personal computer field. This group is cool to the whole ESFOA concept and has forced some tough policy changes on the old guard. Give them credit, however, for upgrading the ESF and lowering its price . . . also for the recent improvement in wafer quality.

Incidentally, the most negative letter I ever received was from Robert H. Rheinlander of Clinton, PA. It was only a copy. The original went directly to Exatron. He blames me for "suckering" him into buying a drive 1 at $199.50 just weeks before the price was dropped to $99.50!

That hurt me, too. However, rest assured that neither I nor, I suspect, anyone at Exatron knew that price break was coming until hours before the decision was reached. The economies of scale resulting from OEM contracts made it possible. And that's what we have to expect in the squashy electronics business! I'm printing this article on my "bargain" Okidata Microline-80, without lowercase descenders yet, which cost me only $659. Have you checked the ads lately?

Like Mr. Rheinlander, I find that files @SAVEd on the old style drive often won't @LOAD on the newer one. This happens even though speeds of the two are within 2% of each other. Then again, they occasionally will load. I've marked most of my wafers with the drive they were made on. It's not too bad a deal.

As for the speed-up kit that was taken off the market, I mourn it too. It was a peach! Several of my friends have tried to get one but no soap! I suspect that the gung ho side of Exatron management wants to push the 64K in-keyboard upgrade

with its built in speed-up. Unfortunately, not only is this piece of equipment expensive ($199.95), but to use it effectively in the absence of available software requires a degree of knowledge that most of us don't have. How about a write-in campaign folks?

I'll conclude this farewell column by calling your attention to a number of Exatron programs that are worth buying an ESF just to have available. They are all available from Exatron Corporation.

ESF MONITOR: One of the best monitors around for the money. With it you can copy machine language programs from tape to wafer, *or* from wafer to tape. You can print screen panels or hard copy panels of machine language programs in hexidecimal, write programs in hexidecimal, debug them with break points, etc.

WHAT'S ON IT: Displays all files on a wafer, stating whether they are BASIC, machine language or data files, showing the byte count of each and the ASCII values of the first few bytes.

COPY CAT: Copies an entire wafer (programs only) automatically with one command.

LOWERCASE DRIVER: Occupies 345 bytes of high memory vs. Tandy's 1024 but does the same thing.

@FREEZE: Allows you to "can" a program when you need sleep and go back to the exact spot where you knocked off yesterday. Program, pointers, stack, CPU registers and variables are all just where they were.

EASY DOES IT: The best string packer around. Make picture strings for use in your programs, using either pixels, graphics blocks or keyboard characters in any combination. Save screen panels (for instructions, etc.) as though they were programs, work in normal or large character modes, create boxes and circles automatically, etc.

PATCHWORD for tape Scripsit: Store and load files to wafer (tape was never like this!).

EDTASM patch: Store source and object code to wafer or tape and get a lowercase driver built in.

ESOS (ESF Operating System): Builds directories, loads and stores programs by filename, and contains several utilities commonly available in the various DOSs.

Next month I'll begin a new series, tentatively titled "In the Chips." *80-U.S.* wants someone who can talk to beginners at their own level about operating systems, machine code, high and low bits, DE and BC registers without scaring the HL out of them or putting them to sleep. They figured I was just simple enough to pull it off. Some compliment! ■

We have recently been told that Exatron has been most generous in helping Mr. Rheinlander solve the problems noted. —Ed.

Exploring VisiCalc

Three dimensional modeling with VIZ.A.CON

Models I/II/III, PMC-80, LNW80

Timothy K. Bowman, Spokane, WA

Have you ever thought of how you might use VisiCalc[1] in three-dimensional modeling? Let's consider several examples. You want to prepare sales projections for three departments and then combine the results into a company total. Or payroll information is reported weekly and then summarized into monthly reports and then into yearly reports. Or you are analyzing the potential effects of a merger on four companies each of which has different tax impacts of the proposed merger. Each of these situations, and more, are easily completed using your VisiCalc program and a recently introduced product called VIZ.A.CON to produce these three-dimensional results.

How it works

In order to explain how the VIZ.A.CON program works, let's consider combining three departments' sales projections into one total company projection. Using VisiCalc, perform the following three steps. First, create and save an original model with all column and row headings but without any data entries or formulas. Second, using a copy of the original model, create the company level model again without any formulas or data entries. Save this model in both the Data Interchange Format DIF[2] and the standard VisiCalc (VC) format. Third, using the original model, create each of the departmental models with the appropriate data and formulas. Again, each of these should be saved using the DIF and VC formats.

Load VIZ.A.CON and answer the prompts. VIZ.A.CON will automatically use the previously created DIF files in the consolidation process. You can then print out the consolidated results with either VIZ.A.CON or VisiCalc (much faster). If you don't like the consolidated results change one or more of the department models and rerun the consolidation process. It's literally that easy.

VIZ.A.CON features

Several of the key VIZ.A.CON features are worth special note. It allows the user to save the consolidation process so that it is not necessary to reenter those steps. This feature is especially important if the consolidation process includes many sub-consolidations. Also, these consolidation processes can be saved and combined later to build a higher level consolidation. You can have up to fifty files consolidated in one process! The order of consolidation is not dependent on the order in which files were created in VisiCalc. VisiCalc precision is maintained throughout the processing and output and you can recall a saved consolidation process to provide updated results. In the example we used above, you might save the consolidation process for combining the three departments and use it to run monthly, quarterly or annual updates. Also, you can save the consolidation results in ASCII format so that they can be loaded into a word-processing program and be further edited or combined with other text material.

VIZ.A.CON comes on a diskette (that can be backed up) along with a forty-three-page notebook. My evaluation version was printed on a LPVI, but current versions are being printed using a daisywheel printer. The documentation is straight forward, easy to understand, and includes three sample sessions. I would recommend that the first-time user simply browse through the manual and then run the program using the sample sessions. The consolidation process is actually easier to understand once you have run it, as compared to just reading about it in the manual.

The program handles filename syntax and input/output errors and the error messages are adequately explained in the manual. There is also an appendix in the manual which gives several helpful hints that allow the program to run faster or obtain better disk utilization.

For the serious user of VisiCalc, this is an important product. It makes good use of DIF, runs fast, and opens many new possibilities for the use of VisiCalc. The program requires 48K and one disk drive. Two drives make it easier to use and allow for a greater number of files to be stored. I recommend it and believe that it will make decision-making easier. After all, isn't that our objective?

For further information, contact Abacus Associates, 6565 West Loop South, Suite 240, Bellaire, Texas 77401, phone number (713) 666-8146. The Model I and III versions (specify which DOS) sell for $89.95 and the Model II/16 version sells for $119.95. Add $3.95 shipping for any version.

Has anyone been able to adapt the enhanced version of VisiCalc for the Model III to run on the Model I? If you have been able to adapt it or have any other question, write me in care of *80-U.S. Journal*, 3838 South Warner Street, Tacoma, WA 98409. Please include a SASE.

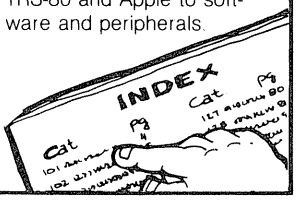
[1] *VisiCalc is a registered trademark of VisiCorp.*

[2] *DIF is a registered trademark of Software Arts, Inc.* ■

How to Buy or Sell Computer Equipment and Software

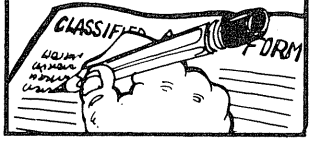
START HERE WITH...

Begin your search in the index of Computer Shopper's bargain filled pages. Locate the category and page number of items that interest you from TRS-80 and Apple to software and peripherals.



Start or add to your computer system by finding money saving bargains in each month's issue from individuals who no longer need their personal equipment.

As you outgrow your system or want to trade up (most dealers won't take trade-ins), use Computer Shopper ads to sell your items to 20,000 readers nationwide for the low cost of 12 cents per word.



If you need help with any computer related problem whether it's an interface problem or advice on the right peripheral for a home-brew system, use the free HELP column especially designed for that purpose

You've got your computer hardware, but what about the software? Use a Computer Shopper ad to find what you need. Someone advertising in Computer Shopper probably has what you want.

Computer Shopper is THE nationwide magazine for buying, selling and trading Micro and Mini-computer equipment and software. Each issue has over 60 pages full of bargains of new and used equipment.

You can save hundreds of dollars by getting the equipment you need from the hundreds of classified ads individuals place in Computer Shopper every month.

Now is the time for you to join over 20,000 other computer users who save time and money with a subscription to Computer Shopper.

Subscribe today and get your first issue and a classified ad absolutely FREE. Type or print your ad on a plain piece of paper and send it along with your subscription.

Just fill in the coupon or MasterCard or VISA holders can phone **TOLL FREE 1-800-327-9920** and start making your computer dollar go further today.

Cut out and mail to: **COMPUTER SHOPPER**
P.O. Box F115 • Titusville, FL 32780

Yes, I'll try Computer Shopper, I understand that if I'm not satisfied with my first issue I can receive a full refund and keep the first issue free.

☐ 1 year $10.00 ($30.00 in Canada)

☐ I have enclosed my free classified ad.

☐ I want to use my free ad later, send me a coupon.

NAME: _____

ADDRESS: _____

CITY: _____

STATE: _____ ZIP: _____

66

COMPUTER SHOPPER
P.O. Box F115 • Titusville, FL 32780
Telephone: 305-269-3211

Basically BASIC

Using string functions

James A. Conrad, Seattle, WA

Last month we discussed the TRS-80 string functions. Now we will play with some of them to analyze and manipulate strings.

Concatenation

Concatenation (pronounced kon-kat-e-NAY-shon) is a fancy word which means a combination, joining, linking or chaining. We concatenate (link) strings. To do this we use the concatenation operator, the "+" sign. When it works on strings it joins them; it doesn't perform numeric addition.

We can concatenate strings from string variables or characters enclosed in quotes, for example:

```
10 FI$ = "DIANA"

20 LA$ = "JONES"

30 NA$ = FI$ + ", " + LA$

40 PRINT NA$
```

Line 30 of this little program concatenates "JONES" in variable FI$, a string in quotes consisting of a comma and a space, and "DIANA" in variable FI$. It assigns the manufactured string to variable NA$. Line 40 prints it as JONES, DIANA.

What's in a name?

Let's assume we have a name stored in variable NA$; in a mailing list, for example. (In an actual program we would probably have a series of names stored in an array, e.g. NA$(1), NA$(2), etc.) We want to separate the first and last names into the variables FI$ (first name) and LA$ (last name).

Here's a routine to get the first name:

```
10 NA$ = "JOHN FITZGERALD COSGROVE"

100 REM Routine to get FI$

110 FOR X=1 TO LEN(NA$)

120    IF MID$(NA$,X,1) =" "
          THEN FI$ = LEFT$(NA$,X-1)
          : X = LEN(NA$)

130 NEXT X

140 PRINT FI$
```

How does it work? Lines 110 and 130 set up a FOR...NEXT loop which will sequentially examine each character of NA$. Line 120 is the "do" portion of the loop. It uses the MID$ function to test each character one at a time (the last "1" in the argument) for a space (" ") in NA$. When the space is found we have the first name.

Let's step through the loop and see what happens. On the first pass the index variable (counter), X, equals 1. The IF portion of line 120 looks at the first character, MID$(NA$,1,1), which is a "J." It's not a space so X is set to 2 for the second pass, and the second character, "O," is tested. On the fifth pass X=5, and the MID$ function finds the space it's looking for. Because the 5th (Xth) character is the space, we know that the first name is X-1 characters long. The THEN portion executes and assigns the left X-1 (4) characters of NA$ ("JOHN") to variable FI$.

The next part of the THEN portion, X = LEN(NA$), advances the counter to its final value (the number of characters in NA$) to end the loop. (Simply jumping out of the loop e.g., GOTO 140, is a bad programming practice.)

Now let's find the last name. We'll use a different technique:

```
200 REM Routine to get LA$

210 FOR X=1 TO LEN(NA$)

220    X$ = RIGHT$(NA$,X)

230       IF LEFT$(X$,1) = " "
             THEN LA$ = RIGHT$(NA$,X-1)
             : X = LEN(NA$)

240 NEXT X

250 PRINT LA$
```

Here line 220 sets up variable X$ as a test string,

sequentially assigning it the right-side characters of NA$. Line 230 tests the left character of X$ for a space. When it finds it, the THEN portion executes and assigns LA$ the right X-1 characters of NA$. (We could just as well have used RIGHT$(X$,X-1), the right X-1 characters of X$.) Again we end the loop by setting our counter, X, to the length of NA$ — its final value in the loop.

Chopping and padding

Suppose we have a field sixteen characters wide for a name. We want to change the string so it has exactly sixteen characters. How do we do it? Let's try it first with a long name, then with a short one:

```
10 NA$ = "KORZENIOWSKI, STANISLOV" (twenty-three chrs.)

10 NA$ = "DOE, JANE" (nine chrs.)
```

We'll assign the result to variable X$. The long name is easy — just use the LEFT$ function to chop it down to sixteen characters:

```
100 REM Routine to make sixteen character string

110 X$ = LEFT$(NA$, 16)
```

And we'll print the string from line 130 and print its length from line 140:

```
130 PRINT X$

140 PRINT "LENGTH OF X$ IS "; LEN(X$)
```

Fine for Stan. We've chopped him down to size. Now what about Jane? When we run her name through the routine we find that it is still nine characters long. So we have to pad X$ with spaces to get the sixteen characters we want in the string. Insert line 120 to do this:

```
120 X$ = X$ + STRING$(16 - LEN(X$), " ")
```

Now the program works for any size name. Line 120 does several things:

1. Assigns a new value into variable X$ (the X$ on the left of the equal sign) consisting of:
2. The old value of X$ (which has a maximum length of sixteen characters — longer names having been chopped by line 110).
3. Plus (concatenated by the "+" concatenation operator):
4. A string of spaces whose length is sixteen characters minus the length of old X$ — built with the STRING$ function:

a. The first part of its argument is the number of characters it creates. We make this 16 - LEN(X$). Jane's name has nine characters — it adds seven. Stan's has already been chopped to sixteen, so it doesn't add any.

b. The second part of the STRING$ argument is the character we want to use — in this case a space (" ").

We've created a string of spaces and concatenated it to the old X$ to make our new X$ exactly sixteen characters long.

Experiment

Here's one to try on your own. Give Jane and Stan middle names. Put their first names first in NA$ (e.g. "JANE CAROL DOE"). Now write a routine that puts into a twenty-character field: last name, comma, first initial, period, middle initial, period. The result for Jane's name should be: "DOE, J.C." followed by ten spaces.

Programming tips

1. The STRING$ function provides a handy way to make output attractive with lines of the same character, such as "*," "-," or "=." You can even make lines of graphics blocks by including the graphics code of the character you want as the second part of the STRING$ argument. For example, STRING$ (20, 191) produces a twenty-character solid line.
2. If you're going to be printing a STRING$ line more than once, assign it to a variable early in your program. You can print it whenever you want just by printing the variable.
3. Use the LEFT$ function in input checking routines to allow yes/no inputs of either the first letter or the entire word, e.g.:

```
120 INPUT "ANSWER 'YES' OR 'NO' "; R$

130 IF LEFT$(R$,1) = "Y" THEN ...
```

4. The CHR$ function is handy for printing graphics blocks (codes 128 - 191). Here's a little program that shows all of them:

```
10 REM Print graphics blocks

20 PRINT "CODE", "CHARACTER"

30 FOR X=128 TO 191

40 PRINT X, CHR$(X)

50 NEXT X
```

This program could also print ASCII characters by using X=32 TO 127 in line 30.

5. You can't normally print a quotation mark by using a PRINT statement (the statement reads it as a string enclosure mark). But if you have a burning desire to print them, try: PRINT CHR$(34).
5. Your manual contains a lot of information on many of the fine points of string functions. Study it when you wonder if a function can do something or why it does it differently than you thought it would.

String functions are especially useful in writing input checking and output formatting routines — areas where many programs are weak. A few hours invested in studying string functions will pay dividends in creating user-friendly programs. That's BASIC. ■

Proctor products

A case history

T. R. Dettmann, Associate editor

We are going to look at one typical business, a metal fabricating operation, which has been using the TRS-80 almost since its introduction. We'll take a look at how the system has been used and how it evolved from a few small programs to a major system involving job costing, accounting and even engineering functions.

The company we're going to look at is Proctor Products of Kirkland, Washington. The history of this company's TRS-80 is typical of many.

Our story begins with the owner, Maury Proctor. While looking into computers for his business, Maury came across the TRS-80 Model I when it was still new on the market. He had seen some larger systems and was duly impressed, but the TRS-80 seemed to provide power at a price he could afford.

Recognizing the system as more than a toy, he got a full system with two disk drives, 48K of memory, and a Line Printer I. After putting it in his shop, he started looking for software. Among the first packages he bought was an inventory program.

The next major package he wanted was a job costing system. At that time, there really weren't any to run on the TRS-80, so he began looking for a programmer. There weren't many programmers around either so he eventually wound up with a programmer who didn't own a TRS-80. He was assured that it didn't make any difference and so he was off.

There's no need to guess what happened. As the months went by, Maury became more and more upset because his system wasn't doing job costing. Oh, there was a program alright, but the programmer could never quite get it to work. After nearly nine months of effort, Maury paid off the programmer and started looking again.

This time, Maury asked for recommendations from the man who sold him the computer. A short time later he got in touch with another programmer, the program was finished and Proctor Products was happy once again.

It was also another learning period for him. In searching for packages to meet his needs, he often looked at software and found it seriously lacking. One program couldn't print reports he needed. None of the programs would talk to the others. Some were poorly documented and none were really satisfactory.

During the time we're talking about, Radio Shack discovered that the Model I had disk problems unless they modified the expansion interface and put in a special cable from keyboard to interface. Sure, you guessed it, Maury had the problems. Programs wouldn't run, disks would get ruined; it was enough to upset a saint.

One modification wasn't enough, there was a second modification to augment the first. Maury had that done and still had problems. He installed a dedicated power line for the computer, a SOLA transformer for regulated power, and a power strip to further regulate it. Even with all this, he had problems.

Proctor Products emerged as a testing ground without peer for the TRS-80. They started replacing components in an effort to fix the system. It would work for a while, then break down. Frustrated is a mild word for how Maury felt. Eventually, he wound up with a completely new system before it worked again without problems.

Just at the time Maury felt pressed by the capability of the system, Radio Shack introduced the Model II computer. With a lot of soul searching, Maury thought about whether he should upgrade. With the software he had acquired he knew that he could put together a good system to sell, so he decided to become a dealer for the Tandy II computer (that's a Model II of a different name and color).

After over a year of development and testing, Maury now put together a viable job costing system for his plant. During a year of daily use, he has pushed for refinements, better displays, more friendly prompting, and greater integration.

Problems solved? Not really. As with the Model I, the Model II had software problems. After trying two payables systems, he finally found one he could accept. But only after extensive modification. After looking at general ledgers, he settled on one only to find it wasn't compatible with TRSDOS 2.0 when he upgraded the operating system.

Maury has had his hardware problems too. A disk unit went bad and required a new part. When it went, it took a disk with it. This didn't show up until that bad disk was backed up over and over again and all the data disks were bad.

Still, through it all, Maury hasn't complained. He's convinced that his system has saved him time and money in spite of the effort and frustration. He continues to look for new software to expand his capability.

New ideas for additions to the job costing system continue to come out of his fertile mind and extensive experience in metal fabricating and construction.

Maury has completed an estimating program on VisiCalc that will price construction jobs with only a few entries. His estimators are overjoyed.

I'm sure that Proctor Products will go on. Despite all the problems and frustrations, Maury has realized that he has come a long way down the road and that it has been well worth it. ■

In the past few years, things have been going smoothly and some of the software for job costing, accounting and other functions are now for sale. Contact Mr. Maury Proctor, Proctor Products, Box F, Kirkland, WA 98033 for more details. —Ed.

CompuKids™

Magazine and Club

(Special school rates available)

MEMBERSHIPS INCLUDE:

1 year magazine subscription
Contests with GREAT prizes
Free Computer Advice
Program Exchange
Educational Programs
....and lots more....

- ☐ One year subscription *plus* one year membership in CompuKids Club — $24.00 (Canada — $31.00)
- ☐ One year subscription only — $16.00 (Canada — $20.00)
- ☐ Six month trial subscription — $9.00 (Canada — $11.00)
- ☐ Payment enclosed (check or money order)
- ☐ Please bill me later (a $1.00 billing fee will be added)

Name _____

Age _____

Address _____

City _____ **State** _____

Zip Code _____ **Phone** _____

Signature (if billing) _____

Mail check or money order today to:

CompuKids Magazine
P.O. Box 874
Sedalia, Mo. 65301
67

Or Call TOLL FREE:
1-800-822-KIDS

Speller

Have your Color Computer talk to non-readers

Color Computer

Charles Hohn, Lowry AFB, CO

I originally bought my Color Computer with one purpose in mind, to educate my six-year-old son. Unfortunately, his reading and spelling abilities are limited. So, the first goal became a program to improve my son's reading and spelling skills.

SPELLER was developed to satisfy this goal. This is a simple program that will "tell" you what to do, and waits for you to do it. It even gives you a second chance! And since it is written in Color BASIC, it will run on any TRS-80 Color Computer.

This program is designed to "talk" to non-readers. But with a little imagination, it could be used to quiz anyone on definitions, terms, or equations. With the CLOAD command at the end of the program you can link a group of programs on one tape. This will allow you to load the program into the computer, leave the play button down on the cassette player, and let a child run the program.

Type the program in and insure there are no bugs before saving it on tape and making the audio recordings. Once debugged, you're ready to CSAVE. I suggest you use a sixty-minute tape to allow the linking of a number of programs.

After CSAVEing, leave the play and record buttons on. At this point, disconnect the EAR and AUX jacks from the tape recorder. This will leave only the MIC (remote) jack plugged in (it's the small gray one). The audio recordings will be made using the built-in microphone of the recorder. However, I found a much higher quality recording can be made by using another microphone. Radio Shack's Tie Clip Mike (Cat. 33-1058) works great.

RUN the program. When the cassette motor comes on, give the instruction (spell 'house'). The motor will stop and the computer waits for input. Press ENTER, and the motor will come on again. This is the second chance. Repeat the same instruction, or add to it (try again, spell 'house'). The motor will stop and the computer waits for input. Press ENTER and the correct spelling and score are displayed. After a few seconds, the routine starts over with the next word. Repeat this procedure until all words in the DATA lines (lines 50 and 60) have been recorded on tape.

At the end of the program, the computer asks if you would like to play again. If you enter "Y" the computer will CLOAD, but there is nothing to CLOAD. So instead, enter "N" and change the words in the DATA lines. If more than ten words are used, change the value of X in line 70. If you like, you can increase the count in lines 120, 230 and 290. This will give you more time for your instructions. *Note:* The count in line 230 *must* match the count in line 290. This allows the computer to skip over the second-chance audio recording when the first answer is correct.

Once all changes have been made, CSAVE, RUN, and make the audio recordings as before.

Whether you're trying to study a foreign language, learn equations, or teach your children, I believe you'll find this program very useful.

For Extended BASIC users, here are possible enhancements. Between lines 110 and 120, DRAW and box a PAINT it. In the instruction, ask the shape or color. You could also use the CIRCLE command. This could make the program usable by very young children.

Program Listing for SPELLER

```
10 'SPELLER
20 'WRITTEN BY C. HOHN
30 'LOT D-71 LMHP
40 'LOWRY AFB, CO. 80230
50 DATA HOUSE, JUMP, PHONE, SCHOOL, BOOKS
60 DATA COMPUTER, LIST, ENTER, EXIT, CRAWL
70 X=10
80 CLS: PRINT @ 44,"SPELLER"
90 IF P=X THEN 430
100 READ A$
110 MOTOR ON: AUDIO ON
120 FOR Z=1 TO 4000: NEXT
130 MOTOR OFF: AUDIO OFF
140 P=P+1
150 INPUT "SPELL ";B$
160 IF B$=A$ THEN GOTO 170 ELSE GOTO 270
170 IF B$=A$ THEN C=C+1
180 PRINT "THAT IS "C" OUT OF "P" CORRECT"
190 IF C=X THEN GOTO 370
200 'CORRECT ANSWER
210 SOUND 200,5
220 MOTOR ON
230 FOR Z=1 TO 5000:NEXT
240 MOTOR OFF
250 IF P=X THEN GOTO 420
260 GOTO 80
270 '2ND CHANCE
280 MOTOR ON: AUDIO ON
290 FOR Z=1 TO 5000:NEXT
300 MOTOR OFF: AUDIO OFF
310 INPUT "SPELL ";B$
320 IF B$=A$ THEN SOUND 200,4 ELSE SOUND 2,5
330 IF B$=A$ THEN GOTO 350
340 PRINT "THE CORRECT SPELLING IS.."A$
350 IF B$=A$ THEN C=C+1
360 PRINT: PRINT "THAT IS "C" OUT OF "P" CORRECT"
370 IF C=X THEN 380 ELSE 400
380 FOR D=1 TO 200 STEP 5
390 SOUND D,2: NEXT D
400 FOR Z=1 TO 3000: NEXT
410 IF P=X THEN GOTO 430
420 GOTO 80
430 PRINT: PRINT: INPUT "DO YOU WANT TO PLAY AGAIN
(Y/N)";Y$
440 IF Y$="N" THEN END
450 CLS: PRINT @ 430,"TYPE 'RUN'"
460 CLOAD ■
```

Baudot with your computer

An evaluation of Richcraft Engineering Ltd.'s teletype program

Models I/III, PMC-80, LNW80

Dr. Richard D. Jameison, Washington, D.C.

Advanced Baudot Radio Teletype for the TRS-80 Model I and III, Volume 5 — Disassembled Handbook for TRS-80 is a fascinating book. It is aimed at both the newcomer and experienced assembly-language programmer who wish to utilize their TRS-80 in the fascinating field of amateur radio Baudot teletype.

What is this newcomer and experienced programmer nonsense? Why quite simply, chapters one to five are for the TRS-80 user who need only know where the on-off switch is located and who is able to answer a BASIC loader-program's questions. Questions such as, call sign, name and address, what type of transmitter and receiver they are using, type of antenna, ARRL section, and two user's-choice optional messages.

The BASIC loader-program POKEs this data in memory and then prompts the user to return to DOS where the program is permanently stashed away, one time only, as a /CMD disk file. The loading routine does not have to be repeated unless you wish to change the contents of one, or more, of the permanent messages. There are 26 prepared messages the user may select.

Chapters six to ten are for the the experienced assembly-language programmer and cover the entire 11000-byte object-code program, the source code (with comments), expanded program commentary, and the Morse code identification subroutine's object and source code. It also includes a unique way of concatenating those programs whose length is too great for EDTASM to assemble together in the 48K available. Actually, the longest source-code program that EDTASM is capable of assembling at one time is equal to an object-code program of approximately 9000 bytes.

Advanced features of the program include:

— Type-ahead during *both* transmit and receive at speeds of 60, 66, 75 and 100 words per minute.

— Full editing of type-ahead if desired.

— Automatic storage in high memory of the incoming message.

— Editing of this message.

— Retransmission of the message if desired.

— No RS-232 interface is required. Parallel to serial and vice-versa data conversion is possible.

— Optional input of three messages from the keyboard.

This book is Volume 5 of Richcraft's series, *The Disassembled Handbook for TRS-80.* It is also available in German, French and Japanese. This is a big book in spite of its 258 pages. It is printed on heavy standard size paper and bound so that is lies flat when opened. This is a decided convenience.

Personalizing the program takes only about ten minutes *if* you have ordered the disk along with the book. The disk includes both the BASIC loader program and the 11000-byte BAUDOT/CMD object code program.

Operation

It is as straightforward and simple as first-class, top-down programming can afford. From DOS, the user types in BAUDOT and he is then off and running. The video display is cleared and reads: TRANSMIT MODE — 60 SPEED, HI-MEM BYTES REMAINING = 25535. The clear key is the transmit-receive switch. Downarrow puts the transmit program in the type-ahead mode and displays an *. At 60 speed, type-ahead is a most useful feature as even relatively slow typists can go faster than that. It never misses a character.

The program is *always* in the type-ahead mode during receive mode. You can type-ahead in transmit mode while the type-ahead input from the receive mode is being transmitted.

Pressing shift and the uparrow displays the following menu:

CQ = A
CLEAR/EDIT TYPE-AHEAD =CLR/C
QRZ =E
OPERATOR CHOICE =G

QRX =I
CQ I.D. FOLLOWS > K
QSY UP = M
SECTION = O
PROGRAM HERE = Q
RYRYRY = S
HANDLE = U
INPUT MESSAGE = X & Y
CLEAR/EDIT HI-MEM = 3 & 4/5
QTH =B
XMIT TYPE-AHEAD = D
73 = F
OPERATOR CHOICE = H
QRM = J
QSY =L
QSY DOWN = N
TRS-80 = P
RIG HERE = R
QUICK BROWN = T
CALL LETTERS? = V & W
INPUT MESSAGE = 1 & 2
XMIT HI-MEM MESG = 6
ENTER PREPARED MESSAGE DESIRED?

Most of the prepared messages are obvious to amateur radio operators. The operator's choice, (G or H), are whatever was input by the user with the BASIC loader program. The three input message options (V/W, X/Y and 1/2) allow the user to input a message with the first letter (V, X or 1) and to transmit by pressing W, Y or 2. Normally V is used to input the call letters and handle of the station being worked, and W to transmit this message. Pressing K automatically sends the I.D. message, plus DE your call letters in fifteen-word per minute Morse code. This satisfies the FCC requirement that each teletype station identify itself once every ten minutes in Morse.

The CLEAR/EDIT HI-MEM function allows the operator to edit the received message, or to input and edit a HI-MEM message. Edit commands are just like those in Electric Pencil. Pressing 3 clears HI-MEM, 4 gives edit instructions, and 5 goes directly to the edit mode. The HI-MEM message is transmitted by pressing the numeral 6. Upon returning to the transmit mode (and each time from RECEIVE MODE), the program automatically advises the number of HI-MEM bytes remaining on the video display.

Model I and Model III Compatibility

This program includes its own object code keyboard decoding routines. As such, program operation is identical for *both* the Model I and Model III TRS-80. The only major difference between the Model I and Model III are the ROM CALLs. CALLs to 28A7H in the Model I are now CALL 021BH in the Model III with 03 delimiter.

Richcraft has intentionally *not* used memory locations 20936 through 28999 to allow for future growth. Options currently planned for this area in memory include automatic, unattended MSO (message storage operation) and remote tuning and/or control of the ICOM series of transceivers, including the models 730 (HF bands), 551 (6 meter band), 251A (2 meter band), and 451 (3/4 meter band).

You must have a Model I or Model III with 48K memory, a port zero encoder/decoder such as the Telesis VAR/80, and a teletype terminal such as the Flesher TU-170. Obviously, an amateur receiver and transmitter is also required. If 425 and/or 850 cycle frequency shift keying is desired, HAL Communications terminal units are recommended.

This is an excellent book and program. For the time being it is not available in 16K, though Richcraft is considering a version for release next year. This cassette version, if it comes to pass, will be produced by Richcraft's West German licensee, 'The Ingenieurburo fur Datentechnik' in Leverkusen, F.R.G., who now translate and publish the German language editions of Volumes. one through five of *The Disassembled Handbook for TRS-80.* ■

Ed. note: The handbook is available from Richcraft Engineering, Ltd., Drawer 1065, Chataugua, NY 14722, for $20 or $49 with disk postpaid.

Microcomputers in church

How the TRS-80 has proven useful

Dr. Edward Simpson, Columbia, MD

A free computer? Most folks would jump at the chance to have free use of a microcomputer, but when one of my church members needed a place to set up his TRS-80 Model I, Level II complete with printer and two drives — I wasn't so sure. More than one business has drowned in a sea of gadgets, and churches are not immune to such asphyxiation. Computers were fine for big churches but small churches (i.e., 150 members) like ours would waste more time with the machine than it would save. A year and six boxes of minidisks later I wonder how I ever managed to get along without it.

Initially we began converting our stencil mailing system to the TRS-80. Updates with the old system required at least five minutes for a secretary to find the old stencil plate, type a corrected stencil, ink the new stencil and refile it. Radio Shack's Business Mailing List allows us to easily manage the 1200 names in our files with changes taking a minute or less once the data is on line.

We use both the compressed (900 names) and expanded (660 names) formats, compressed for people only on the mailing list and expanded for folks we keep additional information on. The expanded format allows us to set up eight different categorical groups along with giving us space for remarks. Our categories breakdown our data by member, college student, bus rider, nursery worker, mailing list, Missionary, Phoster Club and Sunday School worker. These categories allow us to send out mailings or get listings of any one or group of categories in our files. Recently a third disk drive was added to our system enabling us to have 1800 names in the compressed files and 1320 in the expanded.

Our mailing ministry has also made use of Tandy's Profile program. Typing 12000 to 15000 addresses into the computer for a resident mailing list for our city seemed insurmountable. The member who made the system available to the church, and later donated it to us, wrote a program to interact with Profile allowing us to put the entire city's resident mailing data on two disks along with simplifying the input of the information. The street name is input once followed by the house numbers, and apartment numbers. A mailing label addressed "Resident" will be produced for each house or apartment building number and for each individual apartment. Mailing labels can be produced for a single street, for all apartments, for all streets, or for the entire list.

The mailing program has also seen use in producing attendance record sheets for our Sunday School classes. A minor modification has the program printing the listing of class members with "blocks" to be checked indicating attendance. An in-house program has developed allowing each class's attendance to be kept in the computer. This program would also give hard copy of the absentees by the number of weeks missed.

Profile has also been modified to help us keep track of the home visits we make and of the return calls on prospective members. Our church's Bible Institute keeps its records on Profile, allowing a student's transcript to be produced in a minute.

Two other of Radio Shack's programs have proven useful in the work of our church. General Ledger I

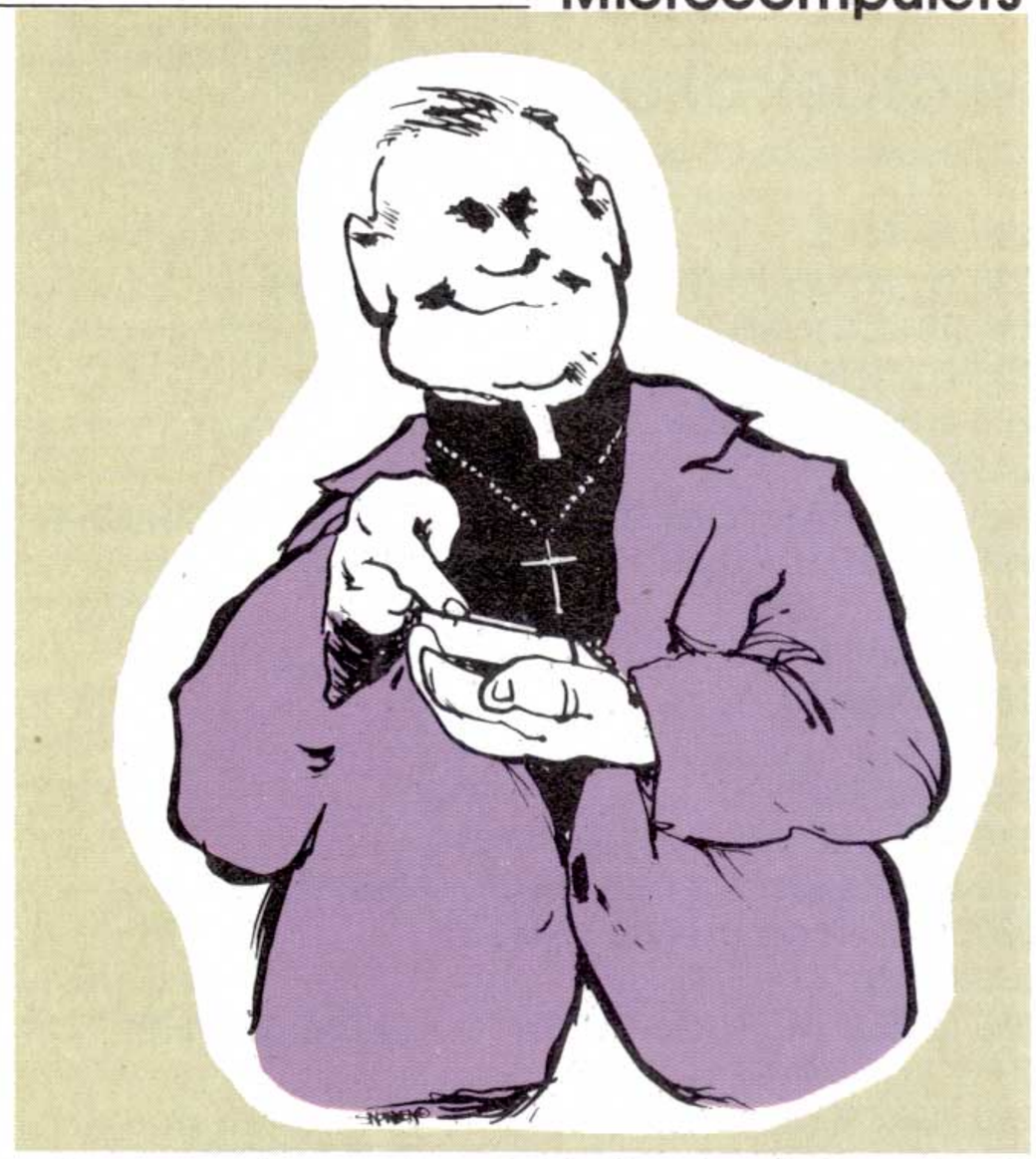
simplifies the production of Balance Sheets and Income Statements. Inventory Control I allows us to maintain better records in our church's small bookstore.

When we bought Lazy Writer our TRS-80 became our word processor handling our newsletters, form letters, bulletins and brochures.

Several in-house programs have been produced to handle the donation records of the members and a weekly prayer list. The program for the donations enables our Financial Secretary to input each person's giving for the week. The program does all of the addition along with giving the contributor a breakdown of his gifts by week and a quarterly summary for income tax purposes.

We will eventually be getting a faster letter-quality printer along with adding lowercase to our keyboard. Those additions will see the system's utility increasing greatly in the word processing areas. Remote terminals may be added later allowing the church's Treasurer and Financial Secretary to input data from their homes over the telephone lines rather than having to come to the church office.

Thousands of dollars can be invested by churches in separate typewriters, VariTypers, addressing machines and other office equipment. It makes good sense to buy one computer system that will fulfill all of these requirements with greater flexibility as well as providing additional applications and expandibility.



Our computer was donated to the church, but what if the entire system suddenly "gave up the ghost?" I have no question but that next week we would be out buying a new one. We could certainly operate without a computer, but we accomplish so much more with it. ■

Microstrokes

Continued from page 30

```
210 TS=TS+2
220 TP=65536: K=(TP+CC)/256: MS=INT(K):
LS=(TP+CC)-MS*256
230 POKE TS,LS: POKE TS+1,MS
240 IF IN<>CO GOTO 160
241 CLS
250 POKE IM,1: POKE SC,0: POKE SW,0 :' SET UP IMAGE 1
WITH SCALE SIZE 0 AND SW SET TO DISPLAY
260 POKE Y,20
270 X1=65:Y1=20
280 POKE X,65
290 '
300 A=USR1(0)
310 A$=INKEY$:IF A$="" GOTO 310
320 POKE 16444,0: POKE 16440,0: POKE 16438,0
330 POKE 16442,0: POKE 16443,0
340 TG=0
350 GOSUB 410 : 'TOGGLE SW TO ACHEIVE SET RESET
EFFECT OF IMAGE
360 POKE X,X1: A=USR1(0)
370 TG=0 :POKE SW,TG
380 GOSUB 740
390 GOTO 290
400 END
410 TG=(NOT TG): TG=TG*-1: POKE SW,TG: RETURN:
'TOGGLE SW ON AND OFF
420 READ BB
430 IF BB=-1 RETURN
440 POKE X,BB
450 X=X+1
460 GOTO 420
470 DATA 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 41, 33, 0, 240
480 DATA 126, 50, 3, 240, 35, 126, 50, 4, 240, 42, 5, 240,
41, 229, 193
490 DATA 33, 246, 240, 183, 237, 74, 94, 35, 86, 213, 225,
26, 183, 202, 220
500 DATA 240, 229, 205, 103, 240, 225, 229, 205, 61, 240,
195, 74, 240, 195, 41
510 DATA 240, 237, 91, 3, 240, 126, 203, 7, 210, 162, 240,
195, 192, 240, 229
520 DATA 237, 91, 7, 240, 42, 9, 240, 223, 40, 9, 33, 9, 240,
52, 225
530 DATA 43, 195, 41, 240, 33, 9, 240, 54, 0, 225, 195, 41,
240, 230, 15
540 DATA 203, 71, 194, 126, 240, 203, 79, 194, 135, 240,
203, 87, 194, 144, 240
550 DATA 203, 95, 194, 153, 240, 201, 33, 4, 240, 53, 203,
135, 195, 103, 240
560 DATA 33, 4, 240, 52, 203, 143, 195, 103, 240, 33, 3,
240, 53, 203, 151
570 DATA 195, 103,  240, 33, 3, 240, 52, 203, 159, 195,
103, 240, 58, 2, 240
580 DATA 183, 194, 192, 240, 205, 223, 240, 33, 215, 240,
229, 62, 128, 245, 58
590 DATA 3, 240, 245, 58, 4, 240, 33, 11, 240, 195, 80, 1,
205, 223, 240
600 DATA 33, 215, 240, 229, 62, 1, 245, 58, 3, 240, 245, 58,
4, 240, 33
610 DATA 11, 240, 195, 80, 1, 209, 225, 35, 213, 201, 195,
154, 10, 33, 3
620 DATA 240, 126, 254, 128, 210, 240, 240, 35, 126, 254,
64, 210, 240, 240, 201
625 REM THESE ARE THE IMAGE STRINGS TO BUILT YOUR
OWN CREATE A DATA STATEMENT ENDING WITH -1
630 DATA 225, 33, 215, 240, 229, 201,-1
640 DATA 4, 4, 4, 4,-1: 'LINE GOING FOUR PIXELS LEFT
650 DATA 2, 2, 2, 2, 2, 2, 2, 2, 2,-1: 'LINE GOING DOWN
660 DATA 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 136, 136, 136, 136, 136,
136, 136, 8, 6, 6, 6, 6, 6, 6, 6, 6, 6,-1: ' AN 'X'
670 DATA 4, 4, 4, 5, 4, 1, 1, 5, 4, 2, 2, 2, 10, 4, 4, 5, 8, 1, 5,
4, 6, 6, 2, 4, 4, 4, 4, 4, 4, 4, 4, 1, 1, 10, 8, 8, 8, 8, 8, 1, 1, 9,
9, 1, 1, 9, 9, 1, 9, 1, 8, 8, 1, 8, 8, 10, 8, 8, 1, 1, 1, 1, 1, 1, 8,
8, 8, 130, 2, 129, 1, 8, 8, 8, 2, 8, 8, 8, 8, 8, 2, 4, 4, 4, 4, 2, 2,
8, 8, 8, 8, 4, 4, 4, 4, 4, 2, 2, 2, 2, 8, 8, 10, 132, 132, 132, 4,
1, 2, 2, 8, 8
671 DATA 2, 132, 4, 6, 6, 6, 6, 4, 4, 6, 10, 10, 4,-1:
'DINOSAUR
680 DATA 6, 6, 6, 4, 6, 1, 4, 4, 2, 1, 4, 4, 2, 1, 4, 4, 2, 1, 4, 4,
2, 1, 4, 4, 2, 1, 4, 4, 2, 1, 4, 4, 2, 1, 4, 4, 2, 1, 4, 4, 2, 1, 4, 4,
2, 1, 4, 4, 2, 1, 4, 4, 2, 1, 4, 4, 2, 1, 4, 4, 2, 1, 4, 4, 2, 1, 4, 5,
5, 5, 5, 1, 8, 1, 8, 1, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8
681 DATA 1, 1, 4, 4, 4, 4, 1, 4, 1, 8, 1, 8, 8, 8, 8, 8, 8, 1, 8, 8,
8, 8, 8, 8, 2, 8, 8, 2, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8,
8, 1, 8, 8, 8, 2, 2, 2, 4, 4, 4, 1, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 2, 2, 4, 2, 2, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 9, 6,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4
682 DATA 4, 4, 4, 4, 4, 4, 4, 4, 4, 2, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8,
8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8,
8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 6, 6, -1:' TANK
690 DATA 2, 1, 1, 1, 1, 1, 9, 9, 9, 9, 10, 10, 10, 5, 5, 5, 8, 8,
8, 8, 8, 8, 8, 10, 10, 10, 10, 2, 2, 2, 2, 2, 4, 4, 4, 4, 4, 4, 1, 1,
1, 1, 1, -1: 'aA SMALL HOUSE
700 DATA 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8,
8, 130, 2, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
130, 2, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8,
130, 2, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
-1: ' FOUR HORIZONTAL LINES
710 DATA 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 136, 8, 6, 6,
6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 136, 8, 9, 9, 9, 9, 9, 9, 9, 9,
9, 9, 9, 9, 9, 136, 8, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, -1:
'DIAGONAL PATTERN
730 DATA 04, 04, 1, 1, 8, 8, 2, 2, -1: ' A SMALL BOX
740 A$=INKEY$
750 IF A$="F" THEN F=F+1
760 IF A$="S" THEN F=F-1
770 B=VAL(A$): IF B>0 AND B<10 POKE IM,B
780 IF A$=CHR$(8) THEN X1=X1-F: IF X1>0 POKE X,X1
ELSE X1=X1+F
790 IF A$=CHR$(9) THEN X1=X1+F: IF X1<127 POKE X,X1
ELSE X1=X1-F
800 IF A$=CHR$(10) THEN Y1=Y1+F: IF Y1<47 POKE Y,Y1
ELSE Y1=Y1-F
810 IF A$=CHR$(91) THEN Y1=Y1-F: IF Y1>0 POKE Y,Y1
ELSE Y1=Y1+F
820 IF A$="U" S1=S1+1
830 IF A$="D" S1=S1-1:IF S1<0 THEN S1=0
840 POKE SC,S1
850 POKE 16444,0: POKE 16440,0: POKE 16438,0
860 POKE 16442,0: POKE 16443,0
870 RETURN ■
```

Reviews

Radio Shack Computer Classes

Part I $29.95
Parts II, III $49.95

Getting started and learning new ideas go hand in hand. Recently we had the opportunity to talk to some people who took a class from a local Radio Shack Computer Center. Their observations help confirm what all computer owners already know, it's fun to work with a TRS-80.

We decided to jump into the middle of the course and take a look at Part II. In this ten-hour class, you are assumed to have a very rudimentary knowledge of the TRS-80. If you know what to do when the computer asks "How many files?" (press ENTER, of course), you are ready. The four sessions are devoted to an introduction of the commands and demonstrating what the machine can do. You will learn how to perform calculations, go over the BASIC commands such as GOTO, FOR...NEXT, IF...THEN...ELSE, AND, OR, and some others. There is also an excellent tutorial on how to take advantage of the editing and AUTO features in Level II BASIC. The class also looks at variables, numeric and string, but does not go into detail on the string functions. POKE is introduced but not worked in detail. The Part III class did more work in string manipulation and poking, but no work with graphics was included.

Before going any further, note this: it is NOT a sales gimmick. There is no pushing of Radio Shack products or software. It is assumed that you have or are using a TRS-80, but even that is not required to take the class.

For many students there is a Tandy Grant available that will make the class tuition free. If, you are a teacher and as a job requirement you have to use a TRS-80, Parts I and II can be underwritten by the Tandy Corporation. In some cases the store manager may extend discounts for other reasons. At the center we investigated, there was one case where a whole family (father, mother and three children) took the class and was given a lowered tuition.

The classes were small, only eleven students in one and six in the other. All students had their own computer to use. In the Part III class the computers were networked together. The workbook that you use is oriented to hands-on experience and you get a lot of it. Our students were adults, but the class had quite a mix. There were three kids, some teachers, a secretary using a TRS-80 in her job, a fireman taking up the computer hobby and a few business people. Instruction was on a Model III (for I/III owners), but there are also classes for Model II users as well as classes on various programs.

The material is well designed and carefully selected for the student. It is for the anxious and uncomfortable. In these classes, the teacher was outstanding, he knew his machine and understood his students. He was a Radio Shack employee, but his concern for his students was obvious and genuine.

One student said: "It gave me the confidence to go on and teach myself. It couldn't have gone better."

If you have heard horror stories about computer classes, especially those that were taught in college, don't despair. These classes are successfully fighting computer

Reviews

anxiety. You can even obtain graduate level college credit for taking Parts I and II of the introductory three-part program. Completing Parts I, II and an extra project will earn three-quarter hours of credit, but be ready to pay the additional tuition that is charged by the university involved. It may even be possible to get credits for taking Part III.

Part I costs $29.95, Parts II and III are $49.95 each and they all involve ten hours of instruction. Courses for Model II owners, or users of Scripsit, Profile, or VisiCalc range from $40 to $150 depending on the number of hours. The centers are not making a profit. The classes are viewed as a benefit and aid to sales, but they are definitely a "loss leader."

Even if you already know it all, tell your friends to take it. They won't learn as much as you, but they will at least begin to understand you. All participants get a certificate (what a way to impress friends!), a workbook, and most of all, CONFIDENCE.

Cameron C. Brown

Problem Solving Principles for BASIC Programmers
Applied Logic, Psychology and Grit
Author: William E. Lewis
Hayden Book Co., Inc.
50 Essex Street
Rochelle Park, NJ 07662
(800) 631-0856
$10.95 (cat. #5200)

There is no lack of books that tell you how to program. Structured Coding, HIPO charts, modular programming, etc., etc., etc., are appearing in books and articles all over the place. Unfortunately, many of these are poorly written as aids for people trying to learn to use any of these techniques.

Problem Solving isn't a book on any of these topics. Rather, it deals with programming from a standpoint I have been using for several years in classes I teach. That is, programming is really nothing more than a form of problem solving!

I guess I just can't say many bad things about someone whose book agrees so well with what I've been trying to teach about programming. In fact, I've made this a required book for my courses.

So what's so good about it? To put it in a nutshell, the book deals with applying the techniques we all use for solving everyday problems in programming. Would you be surprised if I told you that includes Modular, Structured Programming? You just don't give yourself enough credit.

Modular programming is really nothing more than breaking down a problem into smaller chunks and solving those first.

The book uses a very nice method for presenting the techniques. Each method is discussed in detail under a heading that suggests what it's about. These headings are called the problem solving "Prescriptions."

For example, Prescription 1 in chapter two (BASIC Problem Solving Prescriptions) is "Make sure there's a method to your madness." Be organized. Lay out your plan. Think before you program. Most prescriptions are illustrated with examples taken from programming (with example programs) and examples from non-computer situations. There are nine prescriptions, thirteen advanced prescriptions, and sixteen debugging prescriptions.

Careful reading of this book will make you a more careful programmer, as well as a more successful one. Versions of the book are also available specifically for PASCAL, FORTRAN and for general languages.

T.R. Dettmann

MAXPRINT
Model I or III
32K w/disk
MX-80 Printer
Peggytronics
381 First Street, Ste. 5147
Los Altos, CA 94022
(408) 737-2253
$27.95

The MX-80 is a very popular printer because of its many capabilities, exceptional print quality, and low price. Sadly, most word processors don't support all of its features. MAXPRINT is a printer driver (a machine language

program that tells the printer what to do) for the MX-80 printer and TRS-80 Models I and III with 32 to 48K and at least one disk drive, which supports all the MX-80 features — including some it isn't supposed to have!! The GRAFTRAX option is not necessary, although MAXPRINT works just as well when it is installed.

MAXPRINT supports all MX-80 print styles (normal, condensed, expanded, enhanced and double strike in any combination) on a character-by-character basis, underlining, superscripts and subscripts, line spacing adjustment in 1/72 inch increments, setting left and right margins, and line widths, centering text on the line, and proportional right margin justification.

MAXPRINT can perform its magic from its own menu, or from two keystroke commands embedded in the text. It can work as a stand-alone program accessed from DOS or BASIC, or may be used with a word processor like Scripsit or Electric Pencil. Although not documented in the menu, pressing "B" will exit from MAXPRINT to BASIC, leaving a BASIC program intact. SYSTEM /61951 will return to the MAXPRINT menu, so MAXPRINT can format text printed by LPRINT.

MAXPRINT has many useful features, some of them not available any other way, but its most unique feature is its ability to provide proportional right justification with the MX-80. Most word processors justify the right margin by adding spaces between the words of each line until all the lines are the same length and the right margin is even. This process often leaves long, unsightly gaps between words. With proportional justification, extra spacing is added between letters giving a professional, typeset look.

Since the MX-80 does not have a proportional print mode, a trick is used to get the appearance of proportional spacing. Normal enhanced printing prints ten characters per inch (cpi), while condensed, expanded printing provides 8.25 cpi. The difference between the two widths is barely apparent to the eye. For right justification, MAXPRINT begins printing each line at ten cpi, then switches to the wider 8.25 cpi at just the right letter to make all the lines the same length, and all the right margins line up. The result is right justification without any long gaps between words. However, there are two disadvantages. The first is that no other feature (underlining, super or subscript, or change of print style) can be mixed with justification. The second is that lines of widely varying length cannot be justified. The most that MAXPRINT can do is print the entire line at 8.25 cpi. If the line is still too short, MAXPRINT will print it, but it won't be justified. This means that some rough right justification using hyphenation must be done before MAXPRINT is used for the final printing.

MAXPRINT patches itself into the printer Device Control Block (DCB) at 4026 Hex and replaces the address of the Radio Shack ROM printer driver (058D Hex) there with the starting address of MAXPRINT. This means that any CALL to the ROM driver — BASIC's LPRINT and many word processors, for example — will automatically use MAXPRINT. Word processors that use their own custom printer drivers (except for Scripsit) are patched by the GENMOD utility that comes with MAXPRINT. The printer is "memory mapped" to address 37E8 Hex, which means that any character loaded into that address is sent to the printer. GENMOD searches the word processor for the D (37E8H), A instruction, and replaces it with a CALL to MAXPRINT. Scripsit is patched by a special SCRMOD program included with MAXPRINT.

MAXPRINT comes with a comprehensive thirty-seven-page manual prepared entirely using MAXPRINT in its right justification mode and an MX-80. It is a beautiful example of what MAXPRINT can do. The manual is well written and very clear. It first explains how to make disk copies of customized versions of MAXPRINT, then how to modify the MX-80 for use with the program. It goes over each of the MAXPRINT features and commands in detail. There are examples of lines of text with embedded commands, and what those lines look like when printed. Finally,

there is a listing of the source code of the SCRMOD program and a partial listing of the source code of MAXPRINT itself (the I/O and printer driver patching sections).

MAXPRINT comes on a disk with a minimal operating system. Included are 32K and 48K versions of MAXPRINT, SCRMOD32 and SCRMOD48 to patch Scripsit, and GENMOD32 and GENMOD48 to patch other word processors and programs. SCRPMOD/SRC, the source code to SCRMOD48 in a form that can be read by the NEWDOS enhanced version of the Editor Assembler package, is also provided. Having the source code makes it much easier for machine language programmers to use MAXPRINT with other programs. I modified SCRMOD48 so I am able to use MAXPRINT with SCRIPT/CMD (Scripsit modified by SuperScript).

One small hardware modification to the MX-80 will be necessary to have it work with MAXPRINT. In order to change print styles within a line, MAXPRINT has to clear the MX-80 print buffer by sending a carriage return (CR) character, and this might be necessary several times during the course of printing one line. Unfortunately, the Radio Shack MX-80 cable provides a signal on pin 14 which causes the printer to generate a linefeed (LF) after each CR. This makes the printer advance the page one line every time MAXPRINT clears the buffer. The only way to defeat it is to cut that line. I opened the printer, cut the connection to pin 14 of the cable connector, and wired in a microminiature SPST toggle switch (R.S. cat. no. 275-624), so I can run the MX-80 in "normal mode" (LF after CR) or "MAXPRINT mode" (no LF after CR) at the flip of the switch. The added switch is not necessary, however. If the pin 14 connection is cut, MAXPRINT can be used, and all that is necessary to return to "normal" is to set the two MX-80 internal switches to their "LF after CR" positions, although that involves opening the printer's case each time the printer is to be used without MAXPRINT.

It may seem like a lot of trouble to make a hardware modification to the printer to use MAXPRINT, but the modification is a small one and it is well worth the effort. I use my TRS-80 and MX-80 mostly for word processing and correspondence. MAXPRINT allows me to easily access all of the MX-80 print styles, and do underlining, superscripts and subscripts from within Scripsit. Most important, it provides a very professional looking proportional right-justification mode which is not available any other way. The author of MAXPRINT, C.E. Krehbiel, strongly supports his product. I had a problem while trying to patch my word processor so I sent him a letter asking for help. A few days later I received a disk containing a custom version of MAXPRINT, modified to work with my program. A few days after that, I got a telephone call from Mr. Krehbiel to ask how it was working! MAXPRINT is a moderately priced, very useful addition to my system, and I don't think any MX-80 owner should be without it.

John T. Phillip, M.D.

LNW 5/8 Doubler
Model I
LNW Research Corporation
2620 Walnut
Tustin, CA 92680
(714) 641-8850, (714) 544-5744
$219.95 (includes DOS+ operating system)

It had long been my dream to add 8″ disk drives to my Model I. Along came LNW with a 5/8 Doubler that promised double density, boot from either 5″ or 8″ drives, and a low cost. It was too much temptation for me to resist.

I quickly procured and installed the 5/8 Doubler, which converted my two ancient Vista 5¼″ drives to double density. Up to this point, I had been having a lot of trouble formatting track thirty-nine on drive one. After the doubler, no problems.

The external data separator is excellent. What an improvement in reliability! The doubler is worth the price even if you are not going to add the big drives to your system. It also doubles your present storage capacity. Installation is so simple that I think my Aunt Martha could do it, and she is certainly no mechanic. Simply remove the old floppy disk controller chip, plug it into the empty socket on the doubler, and plug the doubler board into the empty socket on the expansion interface board. That's it!

The next step was to order the remaining hardware for the 8″ drives. The LNW advertisement says that you need a special cable for 8″ operation. What you really need is a fifty-pin cable (LNW part 1099, $39.95) and a thirty-four- to fifty-pin cable adapter (LNW part 1096, $24.95). The latter part adapts the standard thirty-four-pin cable to the larger fifty-pin cable. That is all that you need from LNW.

Now comes the expensive part. The 8″ disk drives that I am using are Shugart model 800/801. These drives can be purchased for as little as $350 each from the discounters. Add about $70 for a power supply and you are in business. The cost to add two 8″ drives is about $1050. One last item — a speed-up mod — is required to run the 8″ drives in double density mode. That brings the total to about $1100.

What does our $1100 buy? How about 1182720 additional bytes, plus the added bytes gained by your 5¼" drives going to double density. With two, forty-track, 5¼" drives, and two, seventy-seven-track, 8" drives, we now have a total storage capacity of 1.55 megabytes. That is more capacity than a three-drive Model II! By using one 5¼" and three 8" drives, we can have over two megabytes.

What does that mean in practical terms? Well, I got sick and tired of having to swap diskettes all the time, not to mention the problem of having to catalog all the different files on about seventy disks. My first project was to generate four disks that would hold all of my most-used programs and files. After doing that, I still have over 154 free granuals left, which is more storage than I originally had on my drives. Now, I can leave the four disks in all of the time, boot up, call up any of my programs without worrying about which disk it is on, and have a complete index of all four drives. The only way to go!

How does it all work out? Very well. I am running the Holmes Engineering Sprinter II speed-up board at 5.32 Megahertz (the double density 8" drives require the CPU to run at least at 3.55 MHz). My disk operating system is NEWDOS/80 version 2, with system parameter BJ =3 (for triple CPU speed) and with pdrive parameters TI=EH, TD=F, TC=77, SPT=30, TSR=3, GPL=4, DDSL=17 and DDGA=6 for the 8" drives. This setup requires the speed-up board to run all the time, even during disk accesses. I have had absolutely no problems with either the DOS or the hardware.

There are a few small prices to pay for such a great increase in performance and capacity. One is occasional keybounce, due to the higher clock speed. The second is increased noise from the large drives that are constantly turning. The third is having to make sure that the speed-up is turned on and that the DOS is configured correctly. These minor details are greatly offset by the benefits of tremendous capacity, faster processing, and less disk I/O access times. I highly recommend it.

Jim Klaproth

The Custom TRS-80 and Other Mysteries
Author: Dennis Kitsz
IJG, Inc.
1953 West 11th Street
Upland, CA 91786
(714) 946-5805
$29.95 + $4.00 s/h

Available at Computer Stores, B. Dalton Bookseller

The newest TRS-80 "mysteries" publication from IJG is hardware oriented and is aimed at only the Model I owner. Having enjoyed reading Harv Pennington's volume on disk organization and being a minor league "hardware hacker" of sorts (limited to installation of memory chips, LNW expansion box construction and doing an uppercase modification), I ordered the book. My first surprise came when my local computer store told me that the book wasn't available and had been back-ordered some sixty days. After I placed the order, I received what proved to be a very readable book, written in the best of Dennis Kitsz' style.

While the bulk of the writing in "Custom TRS-80" comes from articles and columns previously published, this book is far more than a collection of short stories. The organization isn't immediately apparent but might be nicknamed, "learning with Dennis." This is a gentle approach to a tutorial on the TRS-80 architecture, physical construction strengths and weaknesses. It is prefaced with a short review (in just eighteen pages) on how to read circuit diagrams, what the various electronics parts and logic chips do and how to select hand tools. Along the way through the book, and after sufficient background content has been established, hardware and/or software projects are introduced.

If the reader follows this path, the Level II, 16K TRS-80 gets extra memory, high resolution graphics, lowercase video and a real-time clock. Analog I/O interfacing and a musical composing/editing capability round out the big additions to the machine. Several of the changes improve operating convenience and long term

Reviews

reliability of the Model I. Some "stand-alone" projects include a printer port, a paper tape reader and a novel mass storage device made out of an 8-track tape deck. The reader receives an outstanding microcomputer education.

There are plenty of good crisp photographs to build self-confidence in project execution and Kitsz provides good advice on how to handle accidents or failures. Troubleshooting approaches are clear and there is an excellent annotated bibliography to guide further reading. This book supplements and enhances the information provided in the TRS-80 Technical Reference Manual and many of the discussions in that manual which may have seemed dry and boring (like video chain and address multiplexing for dynamic memories) are made almost enjoyable by Dennis Kitsz. If you aren't interested in the technical end of the TRS-80 and if the current performance you receive from Scripsit and Sargon meet all your needs, this book isn't for you — save your thirty dollars. On the other hand, owners of early and balky expansion interfaces will be repaid in full by the content of Chapter five. The section titled "111 cures for the common crash" has something for everyone. I personally checked off a dozen things to be done for my own Model I.

On the minus side, the editing and proofreading jobs aren't quite complete in the second printing. However, the reader can easily work around these minor typographical oversights. Many of the customizing projects are interdependent rather than stand-alone and the prospective customizer should spend some time mapping out memory before cutting traces! I personally objected to once more purchasing data sheet copies from Zilog, Western Digital and National Semiconductor. The excerpts weren't complete enough to stand on their own merit.

If you would like to know which 'speed-up' schemes will work on certain production versions of the keyboard (and how to identify these versions) this is your book. If you want to develop a finer understanding of the Model I before you jump into major video modifications or before buying a CP/M conversion, this book will help you along the way.

Capt. Paul Hine

The Temple of the Sun from Programmer's Guild Instant Software, distributor Peterborough, NH 03458 (800) 258-5473 Model I cassette $19.95 Model I/III disk $29.95

It was 9:45 p.m. I'd just come in my office after a long stakeout in front of the computer game store. I took off my hat and hung it on the nail behind the door. Throwing my trench over the back of my old wooden office chair, I sat down. I was about to prop my tired feet up on my messy desk when, suddenly, my office door flew open with a crash. There in the doorway stood two of the biggest, meanest-looking bouncers I've ever had the misfortune of meeting. They paused in the doorway for a moment, huge and gruesome against the faint 60-watt light of the hallway.

I reached for my rod. It was gone! I jumped to my feet and made a mad dash for the fire escape. Too late. I didn't even get to the window when the two Neanderthals were on top of me, pinning both my arms behind my back, causing considerable pain. Then the lights went out.

When I came to, I was sitting in an airplane seat with the same two goons on either side of me. Might as well try a little conversation.

"Nice weather we're having," I said in a thin, weak voice.

The one on my right reached slowly inside his blood-stained overcoat. I thought I was a goner. He pulled out an envelope and handed it to me. Whew! I opened it, unfolding the parchment-like paper slowly. It read: "C.W. Flatts — Others are trying to discover the secrets of the Shaman. They have tried to steal the gold and treasures within. Many have disappeared, or were killed. You must come. Signed, Professor Phungame."

"That's it? That's all the clues you're gonna give me? Well, it's not enough. Not for C.W. Flatts," I said,

folding my arms across my chest.

The mammoth on my left reached into his coat.

"All right, all right," I said quickly. "I'll do a little checking, okay?" He pulled his hand out of his coat and rested it on the arm of the seat. "That was close," I thought.

"What's a Shaman, anyway?" I asked.

Looking at both of them, I could tell from their twin stone-faced expressions they had nothing more to say to me. I'd have to play their game. Come to think of it, they hadn't said a word to me. Probably only grunt anyway.

After several hours, we landed on a dirt airfield in the middle of a tropical jungle. The two anthropoids dumped me head first into a World War II jeep with a driver who only rambled in some foreign jibberish. We took off like a rocket sled, moving pell-mell through the jungle on a bumpy dirt road. Every pothole seemed to have our name on it. The insects attacked us, kamikaze style. I didn't have my rod, or my hat (I always think better with my hat), or my trench . . . not even my Cutters. This was some adventure.

"I must warn you," I said, "I know karate, and judo — I'm a black belt in each. My hands are lethal weapons." My confidence was waning. Sure, I'm lethal. I know exactly six other Japanese words I learned from reading food labels.

Finally, the driver jerked the jeep to a stop, shoved me out on my head, then left me alone in the tropical jungle at the steps of the Temple of the Sun. I had nothing. Not even an American Express card.

The hot, South American sun shown high at mid-day. The smell of the lush, green foliage filled the air. Suddenly, a bolt of pure terror ripped through my body as I heard the crack of a branch breaking behind me. I turned quickly, expecting to find some deadly exotic animal circling me. To my relief, there stood a dark-haired man, in his early thirties, wearing khaki shirt and pants, with deep brown eyes and a thin black mustache. His eyes bothered me. They conveyed death.

After exchanging some friendly greetings neither of us understood, we entered the temple. The awesomeness of the different rooms filled me with visions of finding golden masks, precious gems, sacred items, charred wood that had been used for blood sacrifices, altars of stone, ceremonial robes, golden crowns — the possibilities seemed endless.

I had to get a grip on myself. This was an investigation, not some night on the town with a few of my drinking buddies. What started out as a routine missing persons gig turned suddenly into a full-fledged historical find of considerable greatness. I was beginning to feel a little out of my league. I was now Stanley, looking for Livingston. I was expecting Indiana Jones to come running through at any moment.

While visiting the rooms of the temple, I was picking up lots of clues, but none of them pointed toward my client, Professor Phungame. My brain was starting to respond to my questions, but what did the answers mean? Only Professor Phungame could supply the answers.

The rooms were all different. Some had golden walls, others had fountains, waterfalls, and bottomless pools. Several had altars of stone and charred wood. There were rooms filled with fine apparel. I found a feathered robe that I put on to take home.

"Boy, will this get rave reviews from the guys at the bowling alley," I thought.

In all of the excitement, I had lost my guide. I kept looking for Professor Phungame. It was hard concentrating on my assignment. Finding gold and ivory statues was a distraction not easily overcome. Needless to say, I grabbed whatever I found that seemed to be of some value. After all, I had expenses to meet.

When I finally came to the end of my search, I had so much stuff, and was wearing so many crazy things, I wished my guide were still with me. If he were only there with a One-Step, I would have him take a picture for my Aunt Emma.

I still hadn't found Professor Phungame. But, I had to admit, through all the tense moments, especially with the snakes and the jaguars, I was having fun — a real adventure. After all, what's life without a little excitement and

MAILING LIST & COIN INVENTORY — MODEL I PROGRAMS —

Mailing List — high capacity sort by any field, 30-char. last & company name, 20-char. remarks, prints labels specific printing orders.

Coin Inventory — high capacity, quick M.L. sort print any or all coins.

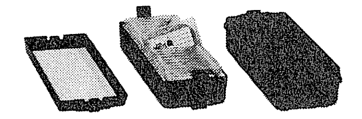
Arctic Computer Systems
5031 Nemiskam Rd. NW
Calgary, Alta, Canada T2K-2P8

$29.95 each or **$45** for both.
Add $2.50 for shipping.
NEWDOS/80 2.0 & Disk required.

\# 81

SECUR-IT

Stringy File



Secur-It *Stringy File storage containers offer safe, no-spill portability and secure, orderly storage for your wafers. High-impact plastic construction. 20 wafer capacity. Black only. $14.95 ea.*

Model I/III Software

• Label Maker, $12.95 • Page Maker (disk only), $19.95 • Home Inventory, $39.95 • Document Maker, $39.95 •

• Send For Free Catalog
• For Fastest Service Send Money Order Or Certified Check • Add $2.00 Shipping Charge Per Order • Calif. Residents Add 6½% Sales Tax • All Merchandise Shipped From Stock •

REM Industries, Inc.
9420 "B" Lurline Ave., Chatsworth, Ca., 91311
(213) 341-3719

\# 82

MULTI DOS*

is the New DOS LEADER with more features than any other DOS and at the best price–99.95

ORDER TODAY FOR SURPRISE FREE BONUS.

TRS80 MOD I/III VISA/MC

C D C
13715 Vanowen Street
Van Nuys, CA 91405
(213) 873-6621

* TM

\# 83

Quatrk Computer Systems, Inc.
P.O. Box 638
Bixby, Oklahoma 74008-0638
(918) 369-1359

TAX PROGRAMS FOR TRS-80 MOD III
Complete programs for tax preparers only
Programs (screen print only) for anyone who wants to do their taxes on the TRS-80 MOD III

BASIC PACKAGE
SMAL-LDOS OPERATING SYSTEMS
FORM 1040 — U.S. INDIVIDUAL TAX RETURN
SCHEDULE A & SCHEDULE B
TAX PREPARER'S WAIVER & INVOICING

FEATURES
MENU DRIVEN PROGRAMS
DIRECT PRINTING OF THE FORMS & SCHEDULES WHEN USED WITH MOST DOT MATRIX PRINTERS
THE 1040 FORM REQUIRES PROPER SETUP TO BE ACCEPTED BY THE IRS
ALL TAX RECORDS ARE STORED ON DISK

COST

TAX PREPARERS BASIC PACKAGE	$320.00
SCREEN PRINT ONLY	170.00
MANUAL & PRINT OUTS ON THE IDS PRINTER	50.00

ADDITIONAL SCHEDULES (*requires extended system)

SCHEDULE C	$ 80.00
SCHEDULE D	90.00
SCHEDULE E	110.00
SCHEDULE F	n/a
SCHEDULE G	80.00

BASIC SYSTEM REQUIREMENTS
TRS-80 MOD III WITH 48K & TWO DISK DRIVES

***EXTENDED SYSTEM REQUIREMENTS**
TRS-80 MOD III WITH 48K & THREE DISK DRIVES

TRS-80 — TRADEMARK OF TANDY CORP
SMAL-LDOS — TRADEMARK OF LOGICAL SYSTEMS, INC
PRICE OF MANUAL & PRINTOUTS APPLICABLE TOWARD COST OF PACKAGE PURCHASED

\# 84

LARGECAPACITYSYSTEMSLARGECAPACITYSYSTEMSLARGECAPACITYSYSTEMSLARGECAPACITYSYSTEMSLARGECAPACITYSYSTEMS

MI M3

GENERAL LEDGER

400+ACCOUNTS
5000+TRANSACTIONS/MONTH
NO OTHER SYSTEM OFFERS....
+REPORT FLEXIBILITY/CAPACITY
+DEPARTMENT P & L (UP TO 5)
+UNLIMITED ACCOUNT CATEGORIES
+STATEMENT OF CHANGES (ASSETS)
+PERCENT P & L comparison
100% sales MTD vs YTD
or net sales QTD VS YTD
or total Exp. MTD VS QTD
+WITH TDOS (A MINI DOSPLUS)

149.95
reg. 32K 2 drives
manual $30.00 test set $50.00

HOLMAN D-P SERVICE
2059 W. LINCOLN
OROVILLE, CA 95965 VISA/MC 3.00 S&H
(916) 533-5992 COD

\# 85

Get

FREE

Programs for the

Color Computer

Join the

East Texas Color Computer Club

2101 East Main St.
Henderson, TX 75652

\# 86

WILDCAT

REMARKABLE DISK DIRECTORY CATALOG SYSTEM
For TRS 80 Model I & III

- Unique abstract feature permits positive file identification. Catalog includes human readable remarks from machine code files. Contains ID, name, date, free space, filespec, and remark.
- Read any size directory from any size disk of any density with any number of tracks and any number of sides.
- Machine language sort by any field. Multi-format print in any sorted order. Single sheet or continuous forms. Save to disk option. Very fast machine code. Self-prompting. User friendly. Complete easy to understand documentation.

Please Specify MODEL and DOS When Ordering.
Requires 2 Drives and 48K RAM.

Satisfaction Guaranteed - Ten day return privilege

Introductory Price Until January '83
39.95 *(plus 2.00 shipping and handling)*

Check - Visa - Mastercard

Don Fielding • 2207 N.W. 61st Place • Margate, FL 33063
(305) 972-6744

\# 87

Arrow Ambush

an exciting new game by SOFTLEX Games

- TRS-80 mod. 1 level II
- gets harder as you get better
- written in unprotected BASIC for easy user modifications
- $9.95 (cassette, postage paid)

SOFTLEX Games
40 Highland Avenue
Lexington, MA 02173

Name__________
Address__________
City_____State_____Zip_____

Send $9.95 NOW!

\# 88

ribbons

For Radio Shack Printers

New Ribbons For:
- Daisywheel II
- Line Printer III & V
- Line Printer VI & VIII
- Line Printer VII

Recycle Your Used Ribbons:
- Service available for most printer ribbon types.

Buy directly from a major manufacturer and save!
FREE CATALOGUE!

*Aspen Ribbons, Inc. is not affiliated with the makers of Radio Shack products.

Aspen Ribbons, Inc. 1700 N. 55th St.
Boulder, CO 80301 (303)444-4054
End User Orders: 800-525-0646
Wholesale Orders: 800-525-9966

\# 89

adventure, huh?

On my way out of the temple I found my guide, dead as a doornail, crushed between two walls only three centimeters apart. This kind of adventure made life interesting (and deadly for some). I was lucky.

The temple was full of rooms I hadn't visited yet. Some of them were deadly, others full of mystery and treasures just waiting to be discovered. There were plenty of moments that made this adventure interesting and fun.

I was about to leave the steps of the temple when something strange happened. A voice spoke to me, saying, "Oh, great and wise Shaman, the earth and its elements are at your command." I didn't want the earth, or its elements — I wanted a Snickers.

Looking down from the temple steps, I saw my jeep and driver waiting.

"Where have you been," I demanded. I forgot that he didn't speak English. Good thing. Some of the things I said to him couldn't be repeated to a Marine.

I jumped into the jeep and we drove back to the airfield.

"I'm still looking for Professor Phungame, and I'll find him, sooner or later. He sure missed out on one exciting adventure. Man, Shaman's tombs, gold, shrinking rooms — you have to be there to experience it."

The driver just nodded and smiled in agreement. I could tell he knew what I'd experienced. Some languages are universal . . . like the thrill of adventure.

I hopped on the plane in my wild outfit and flew home. Finally, in my tiny office, I sat down at my computer and entered the case. This was one adventure I wanted to remember. Though I didn't meet Professor Phungame, I had a feeling we'd meet, somewhere down the road of life's adventures.

I looked at the messages my secretary left on my desk. Good thing she only comes on Tuesdays, or the whole place would be ruined with organization. I looked over the past-due bills for lights and telephone. Then I noticed another parchment-type envelope and opened it.

There was a check for my services, $200 per day plus expenses, and a short note: "I'll be in touch about another adventure." It was signed 'Professor Phungame.'

I felt a grin come across my unshaven face. I grabbed my trench off the back of my chair, sat down, leaned back and pulled my trench over me. Right now, I needed a little shuteye. I propped my feet on the desk top and closed my eyes. Tomorrow, another case, another adventure.

Grey Smith and C. W. Flatts,
Computer Eye

Bounceoids
Model I/III tape or disk
The Cornsoft Group
6008 North Keystone Ave.
Indianapolis, IN 46220
(317) 257-3227
$15.95 tape
$19.95 disk
Plus $1.50 s/h

"What's that?" "It looks like a snake from outer space!" "Uh oh . . . duck, here comes something else!" "They got me." "Yeah look, they're laughing at you too!"

Bounceoids is a fascinating new arcade game from The Cornsoft Group. The only thing more challenging than playing Bounceoids, with its super-fast action graphics, is trying to describe it! What appears to be a combination of several popular arcade games turns out to be like none of them. Just when you think you have the game figured out, another mysterious monster comes out of the woodwork to the attack.

Bounceoids starts out innocently enough. At skill level 0 a giant asteroid appears and bounces randomly across the screen as you attempt to blow it to pieces. In the meantime, a combination of tiny bouncing monsters and stationary guns commence attacking and before you know it, all heck breaks loose!

"Challenge stage, what's that?" "I don't know but you better get ready, something is bound to hap . . . LOOK AT THOSE BIRDS!!!"

If you manage to survive the initial onslaught and reach 20000 points, you will quickly find out what the challenge stage is. A stream of birds come soaring out of the corner of the screen for you to use

as target practice. If you shoot them all down, bonus points will accumulate and you will then be faced with a similar stream of flying saucers. The normal (could this be called normal!?) game continues after you complete the challenge stage.

The Cornsoft Group has come up with a very unique game in Bounceoids. The action is non-stop and sure to leave you with tired fingers. The graphics are very high quality and super-fast!!! This is a very professionally designed game.

"Huh oh, here he comes, hide it!!!" Another thing more challenging than Bounceoids is hiding it when the boss comes into the office!

Bob Perez

Space Castle
Model I or III
The Cornsoft Group
6008 North Keystone Ave.
Indianapolis, IN 46220
(317) 257-3227
16K tape $15.95
32K + Disk $19.95
Plus $1.50 s/h
Joystick Compatible, Sound

As your Jeteye fighter breaks out of hyperspace you find yourself facing a floating space fortress surrounded by three levels of pulsating force fields. It is the Space Castle of Yugdab, the evil warlord who all thought had been destroyed in the Freedom Wars. You try to contact the rest of the fleet, but either they are too far or Yugdab has found a way to suppress sub-space communications. It is up to you to try to destroy the castle. So you do what any peace loving Jeteye knight would do, you attack without warning!

Space Castle is one of the newest programs to come from The Cornsoft Group. The game plan is simple. Keep firing at the force fields until you finally get a shot at the castle in the center. But, to keep things interesting you will soon find out that the castle sends out intelligent mines which work their way through the force fields to eventually get to you. You can't destroy these mines while they are within the bounds of the shields, but they can be destroyed once they get outside of them. Then you find that there is another catch to the game. If you spend too much time trying to destroy the mines the force fields are rebuilt. Well, nobody said it was going to be easy.

You are awarded ten points for hitting the outer shield, twenty points for a hit on the middle shield, and thirty points for the inner shield. Destroying an intelligent mine will get you another fifty points. If you get through the shields and actually score a hit on the castle then you get five hundred points and another ship. You can never actually destroy the castle. As the instructions say, Yugdab "can immediately resurrect himself to continue the battle." So, if you are good enough, one game could last forever, just like Yugdab. Oh, I almost forgot . . . Yugdab has a special surprise if you manage to knock a hole through all three shields and if they line up to give you a shot at him!

Your controls are simply rotate left or right with the left and rightarrow keys or joystick, thrust with the @ key, uparrow or by pushing the joystick forward, and fire with the spacebar or joystick fire button. You will probably find, as I did, that there is some disorientation when you rotate your ship for any reason. Because of the Model I's low resolution graphics, your ship really gets distorted when it is rotated to one of its forty-five degree axes. Also because of the low resolution, it is possible for one of the intelligent mines to come in at an angle that you cannot fire at. The sound effects are not what you would call spectacular, but they do fit the action and do not cause any slowdown of the game.

I ordered the Model I disk version and promptly received my copy. The disk turns out to be one of those self-booting, auto-starting disks with its own peculiar operating system built in. This may cause some users problems if they want a backup copy of the disk. It cannot be backed up by normal means.

Would I recommend this game to my friends? Yes. If you enjoy arcade games then this is one you wouldn't want to leave out of your library. Just be prepared to own only one copy.

Jerry Latham

PMC SOFTWARE

Proven Software Solutions for the TRS-80, PMC-80 & 81 User At VERY Special Prices

GAMES

Title	List	Our Price
Advanced Operating Systems		
+ VOYAGE OF THE VALKYRIE		
D 32K I & III	39.95	32.00
T 16K I & III	29.95	24.00
Adventure International		
ARMOR PATROL		
D 32K I & III	24.95	19.00
T 16K I & III	19.95	15.00
COMBAT		
D 32K I & III	20.95	16.00
T 16K I & III	19.95	15.00
CONQUEST OF CHESTERWOOD		
D 32K I & III	20.95	16.00
T 16K I & III	19.95	15.00
+ DEMON VENTURE #1— REIGN OF THE RED DRAGON		
D 32K I & III	24.95	19.00
ELIMINATOR		
D 32K I & III	24.95	19.00
T 16K I & III	19.95	15.00
GALACTIC EMPIRE		
T 16K I & III	14.95	12.00
GALACTIC REVOLUTION		
T 16K I & III	14.95	12.00
GALACTIC TRADER		
T 16K I & III	14.95	12.00
+ GALACTIC TRILOGY		
D 32K I & III	39.95	30.00
LUNAR LANDER		
D 32K I & III	20.95	16.00
T 16K I & III	14.95	11.00
MACES & MAGIC #1—BARLOG		
D 32K I	29.95	23.00
MACES & MAGIC #2— STONE OF SISYPHUS		
D 32K I	29.95	23.00
MACES & MAGIC #3— MORTON's FORK		
D 32K I	29.95	23.00
MISSILE ATTACK		
D 32K I & III	20.95	16.00
T 16K I & III	14.95	11.00
OTHER VENTURE #2— CROWLEY MANOR		
D 32K I & III	20.95	16.00
T 16K I & III	19.95	15.00
OTHER VENTURE #3— ESCAPE FROM TRAAM		
D 32K I & III	20.95	16.00
T 16K I & III	19.95	15.00
OTHER VENTURE #4— EARTHQUAKE SF 1906		
D 32K I & III	20.95	16.00
T 16K I & III	19.95	15.00
OTHER VENTURE #5—		
+ DEATH PLANET		
T 16K I & III	19.95	15.00
PLANETOIDS		
D 32K I & III	20.95	16.00
T 16K I & III	19.95	15.00
SCOTT ADAMS' ADV. #1, #2 & #3		
D 32K I & III	39.95	31.00
SCOTT ADAMS' ADV #4, #5 & #6		
D 32K I & III	39.95	31.00
SCOTT ADAMS' ADV #7, #8 & #9		
D 32K I & III	39.95	31.00
SCOTT ADAMS' ADV #10, #11 & #12		
D 32K I & III	39.95	31.00
SEA DRAGON		
D 32K I & III	24.95	19.00
T 16K I & III	19.95	15.00
SHOWDOWN		
D 32K I & III	20.95	16.00
T 16K I & III	14.95	12.00
+ SILVER FLASH-PINBALL 1		
D 32K I & III	19.95	15.00
T 16K I & III	14.95	12.00
SKY WARRIOR		
D 32K I & III	20.95	16.00
T 16K I & III	14.95	11.00
SPACE INTRUDERS		
D 32K I & III	20.95	16.00
T 16K I & III	19.95	15.00
STAR FIGHTER		
D 32K I & III	29.95	23.00
T 16K I & III	24.95	19.00
STAR TREK 3.5		
D 32K I & III	19.95	15.00
T 16K I & III	14.95	11.00
TREASURE QUEST		
D 32K I & III	19.95	15.00
T 16K I & III	14.95	11.00
Automated Simulations (EPYX)		
CRUSH, CRUMBLE & CHOMP		
D 32K I & III	29.95	23.00
T 16K I & III	29.95	23.00
DUNJONQUEST— HELLFIRE WARRIOR		
D 32K I & III	39.95	30.00
T 16K I & III	39.95	30.00
DUNJONQUEST— KEYS OF ACHERON		
D 32K I & III	19.95	15.00
T 16K I & III	19.95	15.00
DUNJONQUEST— SORCERER OF SIVA		
D 32K I & III	29.95	23.00
T 16K I & III	29.95	23.00
DUNJONQUEST— TEMPLE OF APSHAI		
D 32K I & III	39.95	30.00
T 16K I & III	39.95	30.00
DUNJONQUEST— UPPER REACHES OF APSHAI		
D 32K I & III	19.95	15.00
T 16K I & III	19.95	15.00
+ RICHOCHET		
D 32K I & III	19.95	15.00
T 16K I & III	19.95	15.00
Big 5		
ATTACK FORCE		
D 32K I & III	19.95	15.00
T 16K I & III	15.95	12.00
COSMIC FIGHTER		
D 32K I & III	19.95	15.00
T 16K I & III	15.95	12.00
DEFENSE COMMAND		
D 32K I & III	19.95	15.00
T 16K I & III	15.95	12.00
GALAXY INVASION		
D 32K I & III	19.95	15.00
T 16K I & III	15.95	12.00
METEOR MISSION		
D 32K I & III	19.95	15.00
T 16K I & III	15.95	12.00
ROBOT ATTACK		
D 32K I & III	19.95	15.00
T 16K I & III	15.95	12.00
+ STELLAR ESCORT		
D 32K I & III	19.95	15.00
T 16K I & III	15.95	12.00
SUPER NOVA		
D 32K I & III	19.95	15.00
T 16K I & III	15.95	12.00
Cornsoft		
+ BOUNCEOIDS		
D 32K I & III	19.95	15.00
T 16K I & III	15.95	12.00
SCARFMAN		
D 32K I & III	19.95	15.00
T 16K I & III	15.95	12.00
SPACE CASTLE		
D 32K I & III	19.95	15.00
T 16K I & III	15.95	12.00
Med Systems		
ASYLUM		
D 32K I & III	22.95	18.00
T 16K I & III	19.95	15.00
+ ASYLUM II		
D 32K I & III	22.95	18.00
T 16K I & III	19.95	15.00
DEATHMAZE 5000		
D 32K I & III	17.95	14.00
T 16K I & III	14.95	11.00
LABYRINTH		
D 32K I & III	17.95	14.00
T 16K I & III	14.95	11.00
WARRIOR OF RAS— DUNZHIN		
D 48K I & III	29.95	23.00
T 48K I & III	29.95	23.00
WARRIORS OF RAS— KAIV		
D 48K I & III	29.95	23.00
T 48K I & III	29.95	23.00
WARRIORS OF RAS— THE WYLDE		
D 48K I & III	29.95	23.00
T 48K I & III	29.95	23.00
Melbourne House Software		
PENETRATOR		
D 32K I & III	24.95	20.00
T 16K I & III	24.95	20.00

Now Twice As Many Programs!

EDUCATION

Advanced Operating Systems		
MOSTLY BASIC EDUCATIONAL PKG. 7 PRG.		
T 16K I & III	24.95	20.00
MOSTLY BASIC SCIENTIFIC PKG. 7 PRG.		
T 16K I & III	24.95	20.00
TIME DUNGEON— AMERICAN HISTORY		
T 16K I & III	24.95	20.00
TIME DUNGEON—		
+ WORLD HISTORY		
T 16K I & III	24.95	20.00
Automated Simulations (EPYX)		
JABBERTALKY		
D 32K I & III	29.95	23.00
T 16K I & III	29.95	23.00

BUSINESS

Small Business Systems Group		
ACCOUNTS PAYABLE		
D 48K I & III	195.00	155.00
ACCOUNTS RECEIVABLE		
D 48K I & III	195.00	155.00
GENERAL LEDGER		
D 48K I & III	195.00	155.00
INVENTORY CONTROL		
D 48K I & III	195.00	155.00
PAYROLL		
D 48K I & III	195.00	155.00

WORD PROCESSING

Aspen Software		
GRAMMATIK		
D 32K I	59.00	46.00
PROOF EDIT		
D 32K I	30.00	23.00
PROOFREADER		
D 32K I	54.00	42.00
SOFT-SCREEN		
D 48K I	69.00	54.00
SOFT-TEXT		
D 48K I	69.00	54.00
Michael Shrayer Software		
ELECTRIC PENCIL		
T 16K I	100.00	24.00

INFO. PROCESSING

Adventure International		
+ MAXI CRAS		
D 48K I & III	99.95	79.00
MAXI MANAGER		
D 48K I & III	99.95	79.00
+ MAXI MANAGER UTILITY PACK #1		
D 48K I & III	49.95	39.00
+ MAXI STAT		
D 48K I & III	199.95	147.00
Auk's		
+ COMPUTER FILING SYSTEM (CFS)		
D 32K I & III	69.00	55.00
Dan Haney Associates		
ELECTRIC SPREADSHEET		
D 32K I & III	69.95	65.00
T 16K I & III	34.95	30.00

OPERATING SYS.

Micro Systems Software		
DOS PLUS V3.3S		
D 32K I	100.00	80.00

UTILITIES

Adventure International		
+ DIRECTORY INFO MGT II (DIM II)		
D 32K I & III	24.95	19.00
Howe Software		
MON3		
T 16K I	39.95	31.00
MON4		
D 32K I	49.95	39.00
STERM		
T 16K I	69.95	55.00
SYSTEM DIAGNOSTIC		
D 32K I & III	99.95	79.00

MISCELLANEOUS

Adventure International		
HINT SHEET for single Adventure #1 thru #12	1.00	1.00
Software Affair		
ORCHESTRA-85		
T 16K I	99.95	95.00
D 16K I	99.95	95.00

+ NEW # 93

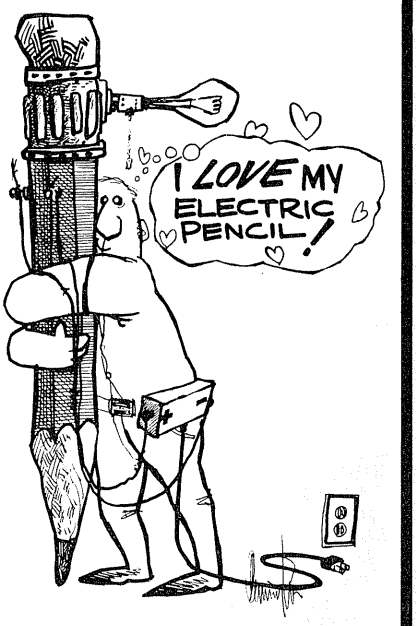
PMC SOFTWARE

475 Ellis St., Mt. View, CA 94043

Orders: (415) 968-1604

Terms FOR FAST DELIVERY, send certified checks money orders VISA or MasterCard number and expiration date Personal checks require 3 weeks U S A sales only PRICES INCLUDE UPS continental delivery (do not use P O Box) CALIFORNIA customers add 6½% tax Prices subject to change

Word Processing Software for TRS-80* Model I



World's most popular word processing package. Fully proven Electric Pencil* now enhanced and offered in 16K cassette version only.

- Easiest to learn
- Simple to operate
- Full screen editing
- Global search & replace
- Powerful insert/delete
- Dynamic print formatting
- No control key mod.

Send check or MO for $24.00 plus sales tax for Calif. cust. UPS shipping in USA prepaid.

$24.00

$10.00 for manual only

*Electric Pencil licensed to PMC Software by Michael Shrayer. # 94
TRS-80 trademark of Tandy Corp.

PMC SOFTWARE

475 Ellis St. Mtn. View, CA 94043 | Orders (415) 962-0318 | Info (415) 962-0220

NO LIMIT **Epson Giveaway** NO LIMIT

Epson MX70/80 Cartridges

$5.00 EACH

Min. 3 of same color

Reloads $2.50 each Min. 12
$30.00 a Doz. of Same Color
Cartridges and Reloads Available
In Black, Red, Green, Blue, Brown

AR systems

Dept. 8A, 35 Cherry Court
East Northport, N.Y. 11731

N Y S Residents Add Tax. Add $2.00 Shipping & Handling
Prices Subject to Change
Allow Clearing Time for Personal Checks
Money Orders & Certified Checks shipped same day

103

SMITH-CORONA TP-1
DAISY WHEEL PRINTER

- Serial RS232 / Parallel Centronics
- Ask about our discount catalog

$575

MICRO TECHNOLOGY
7817 Ivanhoe Ave., San Diego, CA 92037
(714) 457-2149

104

Save Grief & Aggravation

—Know Before You Buy—

READ

"Consumer Protection for the Microcomputer Owner"

COVERS:

1. Preliminary Considerations before Buying
2. Legal Differences between Mail Order and Local Store
3. Pitfalls and Protection of Paying by Cash, Check, Credit Card, etc.
4. What a Warranty Really Is
5. How to Complain Effectively
6. How to Legally Reject and Revoke Acceptance of Goods
7. Miscellaneous Hints and Suggestions

ONLY $15 Post Paid (COD $3 Extra)

Send Check or Money Order to

L. J. KUTTEN, ATTORNEY
201 South Central
P.O. Box 16185
St. Louis (Clayton), MO 63105
314-721-6644
Mo. Residents add 4.625% sales tax
Compuserve ID #70655,376 Source ID# TCK092

105

COLOR CROSS REFERENCE

Color Cross Reference is a machine language program which will build a cross reference listing for the following in a Basic program:

- All Basic Variables
- All Reserved Words
- All Line Number References

Listings can be sent to either the video display or to an optional printer.
REQUIRES: 16K Color Computer
CASSETTE VERSION . . . $19.95
DISK VERSION $21.95

MDFS INC
S.R. 79, Box 103
Orland, ME 04472

106

LARGECAPACITYSYSTEMSLARGECAPACITYSYSTEMSLARGECAPACITYSYSTEMSLARGECAPACITYSYSTEMSLARGECAPACITYSYST

SMALL & LARGE CAPACITY
ACCOUNTS RECEIVABLE
FOR MODEL I/III
5000+CUSTOMERS
15000+ TRANSACTIONS

(REQ. 32K 2 DISKS)

BALANCE FORWARD. 99 TRANS. CODES. 30-60-90-120 AGED STATEMENTS. SHOW DATE/INV#/DESCRIP/AMT (WITH AGEING). SELECTIVE FINANCE CHARGES & RATES. RATES. FAST ENTRY. POSTING W/AUDIT REPORT. SUB-ACCTS. % OF CREDIT LIMIT. DATE OF LAST PAYMENT. SALES ANALYSIS. SPECIAL 90 DAY ACCOUNTS. LABELS.

$149.95

TEST SET $50.00 MAUAL ONLY $30.00

HOLMAN D-P SERVICE
2059 WEST LINCOLN
OROVILLE, CA 95965
916-533-5992

3.00 S&H
VISA OR MC
COD*

*COD (CASH, CERTIFIED, OK, MONEY ORDER)

107

MICRO-MINI ADS SELL YOUR PRODUCTS

All ads in our Micro-Mini section are 2 inches wide by 3 inches deep. Price is $75 on a 1x rate. This low, low price includes reader service. Submit ads with check or money order to:

80-U.S. Journal
3838 South Warner
Tacoma, WA 98409
(206) 475-2219

Toucan Presents:

A new generation of adventures — with graphics!!!

Subterranean Encounter — A fantastic medievai adventure which will take you from above ground with it's castles, wizards, moats, and alligators to the subterranean depths whose evil and danger knows no bounds. A new plateau in adventuring — from Toucan Software, who else?? (TRS-80* Model I or III (Please specify), 32K, 1 Disk Drive). Just $29.95 at fine software stores everywhere or directly from us.

TOUCAN SOFTWARE
4024 Canonero Court
Fair Oaks, CA 95628
(916) 966-4241

*TRS-80 is a registered trademark of Radio Shack, a division of Tandy Corporation.

109

TAXES

Software to assist you in organizing and preparing an accurate Tax Return.

TAX I

- 1040, Schedule A and B
- For 16K Timex-Sinclair, Commodore, Texas Instruments, Apple, Radio Shack Color, Mod I/III
- Program on cassette, tax deductible **$29.75**

TAX II

- Tax Estimator
- 1040 and frequently used schedules
- For 48K Apple II, Radio Shack Mod I/III.
- Program and Tax Tables on Disk, tax deductible **$34.75**

Specify machine with your order.
(Pa. Residents add 6% Sales Tax)

Eugene H. Bock & Associates
Consultants **215-362-5082**
135 Heartwood Dr., Lansdale, PA 19446

BONUS 1983 H & R Block 266 page Tax Workbook ($5.95 value) included if you order before Mar. 15th.

110

BARCLAY WHYTE ASSOCIATES

WORDPROC I-III VERSION 3.0

Basic plus machine language Word Processor
TRS-80* Model I/III 48K Disc
Special Version Now you can embed control codes in your text for
Superscript Subscript Underline Italics More
Supports all MX-80** typefonts
Still only **$14.95 each, with disc**
$10.95 each, documentation only

BARCLAY WHYTE ASSOCIATES
P.O. Box 948,
New Westminster
B.C. Canada. V3L 5C3.
Add $1.00 for postage
B.C. Residents add tax at 6%

*Trade mark of Tandy Corporation
**Trade mark of Epson America

111

Architectural Engineering Library

50 programs in a total of 10 categories including:

Heat Loss/SLR
Timber Beams,
Steel Beams,
Wood Trusses (4 types),
Concrete: Beams, Walls,
Slabs and Footings.

TOTAL COST: **$225** (U.S.)
Individual Categories: **$35**
Disk, ESF or Cassette

ERIC CLOUGH
Box 52, Winlaw, B.C.
Canada VOG 2J0
or
DHU-GLAS — Box 1664
Sausalito, CA 94966

112

Dental Computer Newsletter

E. J. Neiburger, D.D.S., Editor
1000 North Avenue
Waukegan, IL 60085

The D.C.N. is an international group of dentists, physicians and office management people who have interests in office computers. Though the emphasis is on microcomputers, many members use minis. We cater to all makes and brand names.

Annual membership dues $15.00. Membership runs from January to January. If you join mid-year, we will supply you with the year's back issues.

113

TRS-80 MODEL I T.M.* GOLDPLUG - 80

Eliminate disk re-boots and data loss due to poor contact problems at card edge connectors. The GOLD PLUG - 80 solders to the board card edge. Use your existing cables.

CPU/keyboard to expansion interface $18.95
Expansion interface to disk, printer, RS232, screen printer (specify) . . . . . .$9.95 ea
Full set, six connectors. . . $54.95

VISA master charge

EAP COMPANY
P.O. Box 14, Keller, TX 76248
(817) 498-4242
*TRS-80 is a trademark of Tandy Corp.

114

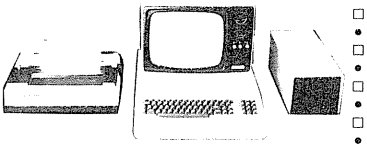
COLOR COMPUTER COLORFORTH

FORTH is a high level computer language like BASIC or PASCAL. **COLORFORTH**, a version of figFORTH, is available NOW for the TRS-80C computer. **COLORFORTH** execution time is as much as 10 times faster than BASIC. **COLORFORTH** requires a minimum of 16K ram, but does not require either Extended Basic or disk system. When you purchase **COLORFORTH**, you receive both cassette and disk versions, the standard figEDITOR, and an extensive instruction manual.
ALL FOR ONLY..... $49.95

Armadillo Int'l Software
P.O. BOX 7661 PH. (512)459-7325
AUSTIN, TEXAS 78712

115

LNW80 COMPUTERS



• IN STOCK NOW!! CALL FOR SPECIAL PRICES ON LNW80s, NEC PRINTERS, RGB COLOR MONITORS, AND TANDON DISK SYSTEMS.

***** NEW LNW SOFTWARE! *****

EXCELLONIX

7180 Woodrow Wilson Dr
Los Angeles, CA 90068

(213) 650-5754 - (714) 973-1939

116

For Educators, Program Developers, and Hobbyists
Presenting mainframe-like JCL control on a Micro!

"BABY BLUE 80*" and "SUPER BLUE 80*"

The BASCOM or RSCOBOL COMPILERS BATCH COMPILER CONTROL DISKS

Two versions for LDOS owners with Radio Shack's COBOL or Microsoft's BASIC COMPILER

"BABY BLUE 80*"

Automatic Job Control Language COMPILER — 9-item compile menu. Batch compile 10 source code programs automatically! FORMAT and BACKUP automatically. Lucid documentation

"SUPER BLUE 80*"

All BABY BLUE hands off compilation features plus User in compile mode selects custom compilers In batch mode compile up to 10 files from the auto-directory Support features include: JOB LOG, ROUTE, or LINK, PRINTER, SPOOLER & SYSTEM TRS-80 MODEL I/III 35 track single density data disk. Model III owners use CONVER Special introductory prices: "BABY BLUE" only $45 for each version, both $80 "SUPER BLUE 80" for each version $100, or together $160 (NY res add 8% sales tax)

Comnet Software Associates
© 1982 Write to: Steve Abramowitz
175 Adams St., Brooklyn Heights, NY 11201

117

MODEL I/III SOFTWARE BY: *Rip*

*** RIPCHECK ***.....$59.95
SUPER FAST MENU DRIVEN PAYABLES SYSTEM (4) USER DEFINED ACCOUNTS - REVIEWS - (156) VENDOR NAMES/ADDRESSES - LPRINT CHECKS/REG. - (4) SEARCHES WITH TOTALS

*** RIPBONDS ***.....$89.95
A COMPUTERIZED PORTFOLIO FOR THE BOND INVESTOR - YTM - CURRENT ACCRUED INT. OUTSTANDING CURRENT/SOLD BOND REPORTS

*** PAYROLL5 ***.....$79.95
CONVERT RADIO SHACK'S® DISK PAYROLL TO OPERATE 5 SEPARATE DIVISIONS/DEPTS. WITH INDIVIDUAL & CONSOLIDATED RECAPS - SEND ORIGINAL PROGRAM DISKETTES -

*** SHUTTLE COMMAND ***.....$19.95
TRANSPORT YOUR GRP FUEL CELLS THROUGH BI-DIRECTIONAL METEOR SHOWER - MACHINE LANGUAGE ARCADE TYPE GAME - AUTOMATIC HIGH SCORE SAVE - ONLY ORDER IF YOU'RE READY FOR "THE ULTIMATE CHALLENGE"

- - SEND CHECK OR MONEY ORDER - -

GLENN/CLIFF ASSOCIATES
8301 EAST MONTEBELLO
SCOTTSDALE, ARIZONA 85253

118

Easy File: Database, Mail List, ect... **$15.00***
Easy File: Disk, 32K Min., Model I/III . **$15.00**
Stock Quest: 39-week Data File..... **$11.00**
Household Inventory: Database Program **$7.00***
Zaars Citadel: RND Situation Game.... **$5.00**
Miners: Hammurabi Type Game..... **$4.00**
Stock Game: Easy Type Stock Market Game **$5.00**

DATA MANIA, INC.

51625 Chestnut Road
Granger, IN 46530

Add $2.00 P/H
* 16K or 32K, Specify
For 16K TRS-80™ Color and Model I/III. Tape Only Specify

119

IEEE-488 TO TRS-80* INTERFACE

Everything needed to add *powerful* BASIC GPIB-488 controller capability to TRS-80 Model 1 or 3, Level 2 or DOS with a minimum of 16K.

488-80B
For Model 1 Operation

488-80C
For Model 3 Operation

Model 488-80B or 488-80C Price: $375.
+ shipping, insurance & tax

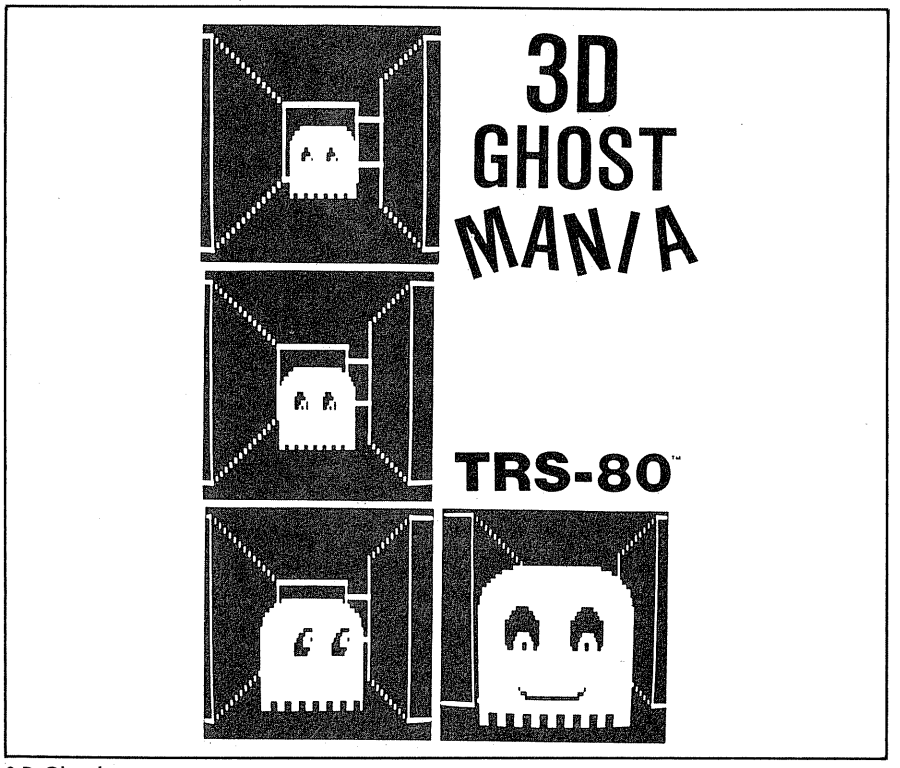
WHEN ORDERING SPECIFY DISK OR TAPE

SCIENTIFIC ENGINEERING LABORATORIES
11 Neil Drive • Old Bethpage, NY 11804
Telephone: (516) 694-3370

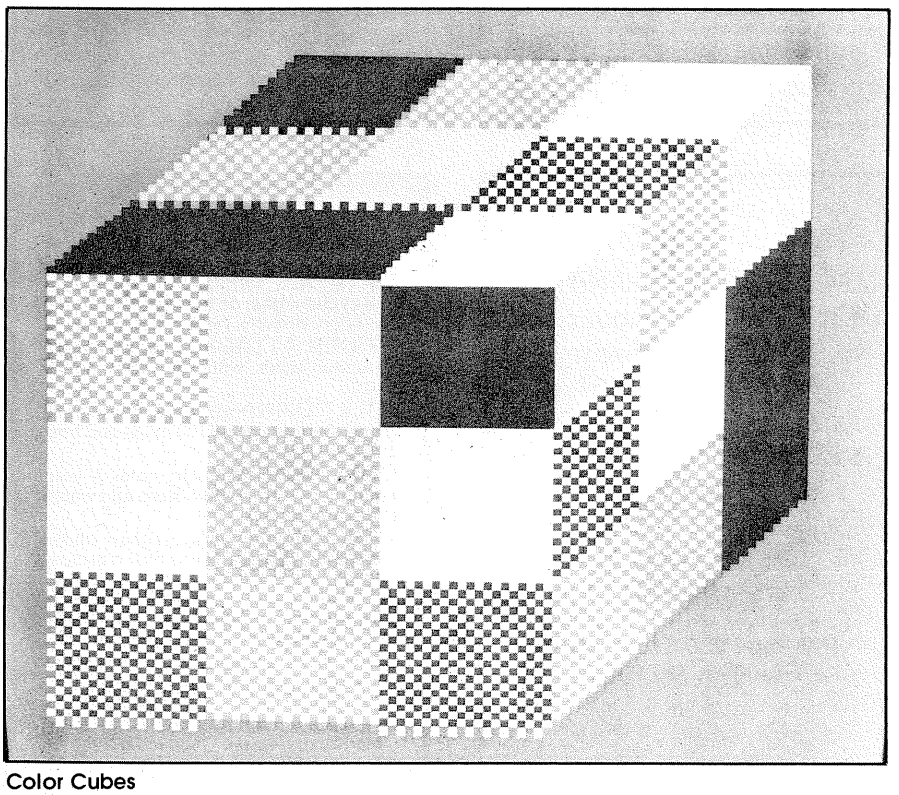
*Trademark of Tandy Corp.
There is no affiliation between Scientific Engineering Laboratories and Tandy Corp. or Radio Shack.

120

New products



3-D Ghosts



Color Cubes

Business Management VI

This is a complete system for analyzing the financial statements of a company. It aids in planning for future activities, investments and alternatives. Balance sheet and income statements from up to seven financial periods can be used for input. The system then provides percent analysis, ratios pertaining to liquidity, efficiency, profitability, computes compounded growth rates, linear regression, and gives graphical representation of key financial data.

Dividends, earnings per share, average market price, net sales and income, and per share analysis of key data are just some of the calculations performed.

It is available on diskette for the Model II, 64K in either TRSDOS 2.0a or a CP/M version. This user-oriented system has a price of $185. For further information and free brochures about this and other business software, contact Century Software Systems, 1875 Century Park East, Suite 1730, Los Angeles, CA 90067 or phone (213) 879-5911, or Telex 18-1380.

 # 200

Three-dimensional ghosts

3D-Ghost mania is a three-dimensional game for the Models I and III. It gives you the experience of running through an endless maze of hallways while trying to match wits with pursuing ghosts. The company claims that the game has been known to produce nightmares when played in excess. Retail price is $29.95 for cassette (16K) or disk (32K, Model III only). Contact Computer Price Index, 5350 South 3600 West, Salt Lake City, UT 84118, (801) 968-2252.

 # 201

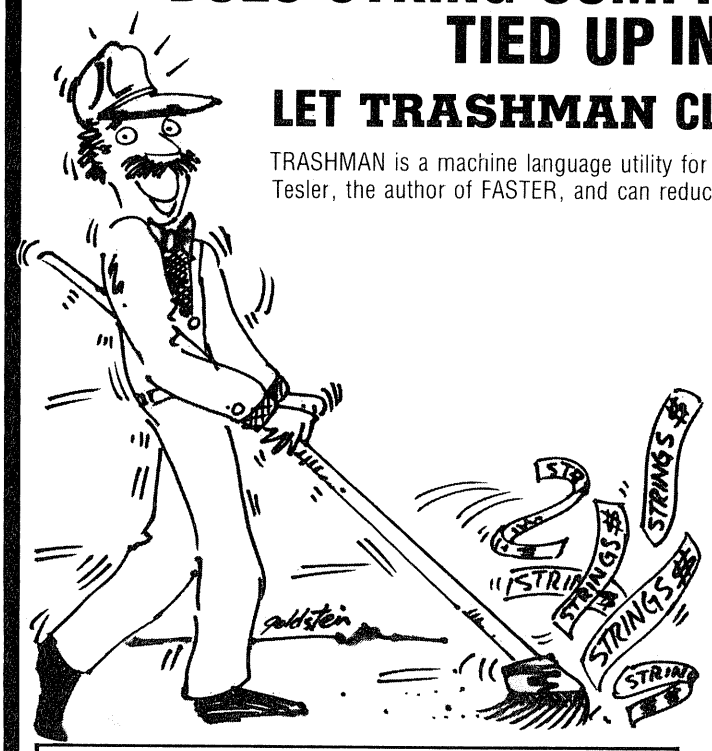
Color Cubes game

Radio Shack announces a computer game version of the cube puzzle. Color Cubes (#26-3075) is available for all versions of the Color Computer for $29.95. It offers a colorful video representation of a scrambled 3-dimensional cube, itself made up of twenty-seven smaller "cubies" in six different colors. The

DOES STRING COMPRESSION HAVE YOU TIED UP IN KNOTS?

LET TRASHMAN CLEAN UP THE MESS!

TRASHMAN is a machine language utility for the TRS-80 Models I and III. It was written by Glenn Tesler, the author of FASTER, and can reduce BASIC's string compression time by 95% (see table below).



# STRINGS	SECONDS DELAY NORMAL	TRASHMAN	PERCENT IMPROVEMENT
250	11.8	0.7	94
500	45.8	1.6	96.5
1000	179.6	3.5	98
2000	713.2	7.8	98.9

WHAT'S STRING COMPRESSION?

When a BASIC program changes a string (words, names, descriptions), it moves it to a new place in memory, and leaves a hole in the old place. Eventually, all available memory gets used up and BASIC has to push the strings together to free up some space. This takes time. Lots of time. The computer stops running for seconds or minutes, and you may even think it's "crashed". The keyboard won't work, and until all the strings have been collected, you just have to sit and wait. Then things run for a while, until string compression is needed again. And again.

If you're using your computer for business, that wastes your money. If you're using it personally, it wastes your time.

WHAT'S THE SOLUTION?

As soon as you start using TRASHMAN, those delays almost disappear. It uses less than 600 bytes of memory, plus 2 bytes for each active string. It works with other machine language programs and with all major operating systems. It's easy to use, comes with complete instructions, and can be copied to your own disks.

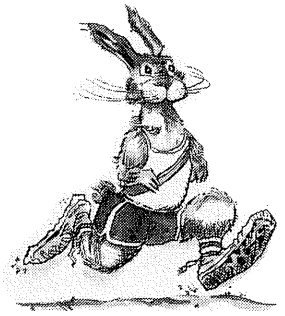
WHAT'S THE CATCH?

If a BASIC program uses only a few strings, very little time is wasted in string compression, and TRASHMAN won't be helpful. But, if hundreds of strings, including large string arrays, are used, TRASHMAN is just what you need.

TRASHMAN is available on disk for just **$39.95.**

(All timings done on TRS-80 Model I. Model III 15% faster, but pct. improvements identical. Listing of timing program available on request.)

SAVE TIME WITH FASTER



"FASTER" speeds up most TRS-80 BASIC programs by 20-50%. It's helped hundreds of satisfied people and it can help you. Detailed instructions make it easy to use. FASTER analyses your BASIC programs *while they run*, then displays a simple change, usually one line, that sequences program variables so the ROM will find them faster.

You can use FASTER to speed up programs you've bought, as well as programs of your own. Since it isn't a compiler, your BASIC programs can be read and changed afterwards. FASTER works on business programs, models, and games. The more complex your program, the better the results.

Does FASTER really work? Yes! Just check the reviews in *Personal Computing*, May, 1981, p. 116: **"FASTER is effective and easy to use"**; *80 U.S. Journal*, April, 1982, p. 106: **"I recommend FASTER to everyone"**; and *80 MICRO* (April, 1982, p. 40): **"If you...would like a significant increase in the run-time speed, then buy FASTER."**

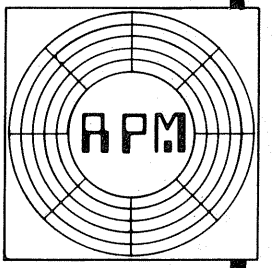
FASTER runs on the TRS-80 Models I and III, 16-48K tape or disk, and all major operating systems. **$29.95**

"QUICK COMPRESS" takes only 276 bytes of memory, and removes the blanks and remarks from even the largest BASIC program in less than 3 seconds. It produces smaller, faster programs without altering their logic.

$19.95

SPECIAL: FASTER and QUICK COMPRESS: **$39.95**

ERRATIC DISK DRIVES?



You can avoid unnecessary disk errors and repair bills by using **RPM.** This easy-to-use program measures the rotational speed and fluctuations of your disk drives, and warns you if they are running **too fast, too slow, or unevenly.**

Incorrect or erratic speed is a common cause of unexplained disk errors and loss of data. RPM's documentation explains how to **detect** *and* **correct** these problems quickly and easily. As *80 MICRO* (April, 1982, page 41) said: **"If your drives have problems I recommend RPM before paying to get it repaired."**

RPM is supplied on diskette for the TRS-80 Models I and III. We suggest you order a copy *before* you need it.

$24.95

ORDER FROM YOUR LOCAL SOFTWARE DEALER, OR CALL NOW, TOLL-FREE:

(800) 824-7888, Operator 422
CALIF: (800) 852-7777, Oper. 422
ALASKA/HAWAII: (800) 824-7919

FOR TECHNICAL INFORMATION CALL:
(213) 764-3131, or write to us.

PROSOFT®

Dept. B, Box 560, No. Hollywood, CA 91603

128

TERMS: We accept VISA, MasterCard, checks, or even cash. Please add $2.00 shipping/handling within U.S.A. or Canada, and $5.00 overseas. C.O.D. charge is $2.00 in U.S. only. We ship within one day of receiving orders.

goal is to unscramble the array by rotating any slice or layer until each face of the larger cube is a solid color.

Unlike the plastic cubes, the computer game will record a player's last 255 moves and allows a player to undo or redo them. Positions can be saved to tape, two players can compete against the clock, and there is a variety of different color combinations possible. Use different colors to flag different break points in solving the cube.

 # 202

80-Grafix board

Now have high resolution graphics on your Model I or III. The 80-Grafix board gives any Model I computer an effective resolution of 384 by 192 pixels through the use of 128 programmable characters. Each character is made up of a 6 x 12 matrix of individually controlled dots. The Model III board has a 512 by 192 resolution, also made through the use of 128 characters, each of which is comprosed of an 8 x 12 dot matrix.

Once programmed, in BASIC or machine language, the characters can be displayed by PRINT or POKE statements. High resolution is enabled through the OUT command and is disabled whenever reset is pressed. The device will not interfere with existing software. Included with the board are over twenty programs and files for instructing and entertaining. The board also includes a hardware lowercase modification and requires no soldering for installation.

Available for tape or disk systems (please specify configuration and model) for $169.95 from Micro-Labs Inc., 902 Pinecrest, Richardson, TX 75080, (214) 235-0915. Texas residents add 5% sales tax.

 # 203

Spectrum Stick

The Spectrum Stick is a new joystick for the Color Computer. A hair-trigger fire button makes long sessions at the computer easy and it uses swivel-ball components for a smooth and true feel. Extra long cable and brushed aluminum knob for comfort and control. Its sturdy construction eliminates the worry about broken parts and a red LED indicator tells you whenever the computer is left on. It is available from Spectrum Projects, 93-15 — 86 Drive, Woodhaven, NY 11421, (212) 441-2807 for $39.95 plus $2.00 shipping and handling.

 # 204

Number Cruncher

Is a complete statistical package for the Models I and III and will soon be available on the Model II. Data is entered directly into a disk file through a data file manager. Over twenty-five transformations can be used for processing the data. The manager allows for editing, adding, or merging of the data.

Entered data can be analyzed by numerous procedures, including multiple regression, stepwise regression, correlation analysis, t-tests, scatter plots, components analysis, and a host of ANOVA programs. These statistician written programs are available on two single-sided disks. They will run on 32K, but 48K is recommended. The system is priced at $99.95 plus $3.00 shipping and handling. For more information contact Dr. Jerry L. Hintze, 865 East 400 North, Kaysville, UT 84037, (801) 546-0445.

 # 205

COPY-NOT

COPY-NOT is an external security program for BASIC software authors. It significantly modifies TRSDOS 2.3 and stores all "/BAS" compressed files in an encrypted form. COPY-NOT modifies DOS READY but still allows library execution and up to nine DOS sequence commands through use of its DO/JCL file. It does not affect available memory during execution, but does render BASIC * inoperative. It allows the software author to title each diskette and has an auto serial number feature for use when creating application disks. Stops piracy of software and gives software authors sole ownership of their work. For use with Model I disk systems. Available from HPB Vector Co., 130 Center St., East Stroudsburg, PA 18301, for $275.00.

 # 206

Handicapping Program

The KEL-CO Thoroughbred Handicapping System RS-5 was developed and introduced in the 1960's after an analysis of over 10000 races. It has evolved into a system that has both quantitive and qualitative aspects. You supply such facts as date, name of horse, number of starts, earnings, type of track, win place and show records, age, purse, date of workout and last race, and judgment calls regarding type of win the horse has had. The system notes prime and longshot bets as well as allowing for Daily Doubles, Exactas and Quinellas, and on or off track betting. There is also a version, RS-6, for harness racing and handicapping. Available in USA from Canella Sales Corp., 420 E. Genesee St., Suite 208, Syracuse, NY 13202, (800) 448-5713, for $200. NY residents add sales tax. In Canada, contact Equine Management & Innovations, P.O. Box 2214, Station D, Ottawa, Ontario K1P 5W4. Canadian price is $225 Cdn.

 # 207

Model 16 Spreadsheet System

FINANSWER+ is written in RM/COBOL and runs in 16-bit mode on the Model 16's 68000 CPU. The package includes menu-based processing, English language input, automatic saving and restoring of spreadsheets, password protection, sorting and graphics. It supports twelve spreadsheets per data file and up to 1200 figures per sheet. Consolidate spreadsheets for a total of up to 14400 figures. Use of the 16-bit processor gives high speed operation and rapid disk input. For $395. Maintenance after 90 days is $60 per year. Contact DATAMATE Co. Inc., 4135 South 100th East Avenue, Suite 101, Tulsa, OK 74145, (918) 664-7276.

 # 208

Expansion Interface for the Color Computer

This new expansion interface provides a Centronics compatible I/O port, 64K memory access circuit and expansion capability for up to seven additional peripheral cards. It requires no modification to the Color Computer and is compatible with the Radio Shack Disk System. It does not interfere with existing ROM or output ports. Access to a complete 64K is possible for 32K Rev-E Color Computer owners. Optional

aluminum chassis is available. The CX-2001A expander card is $139.95, the chassis (CX-3001A) is $39.95, and the ribbon cable (CX-2401A) is $29.95. Introductory package price is $189.95. Contact General Automation, 9600 Roosevelt Blvd., Suite 100-LL, Philadelphia, PA 19115, (215) 934-3758.

 # 209

ASG-III hard disk

The ASG-III hard disk plugs directly into the TRS-80 Model III expansion port and includes Winchester drive technology. It includes drivers for Model III LDOS (LDOS is not included) and is available in five or ten-megabyte capacity. Five-megabyte drive is $1795, ten-megabyte drive is $1995, and the DOSPLUS 4.0n hard disk operating system is $149. For further information, contact All Systems Go, 638 S. Dillard St., Winter Garden, FL 32787, (800) 327-6590. In Florida call (305) 877-2830.

 # 210

Macro Assembler and Editor on Disk for the Radio Shack Color Computer

MACRO-80c is a macro assembler, screen oriented editor and machine language monitor for the Radio Shack Color Computer disk system.

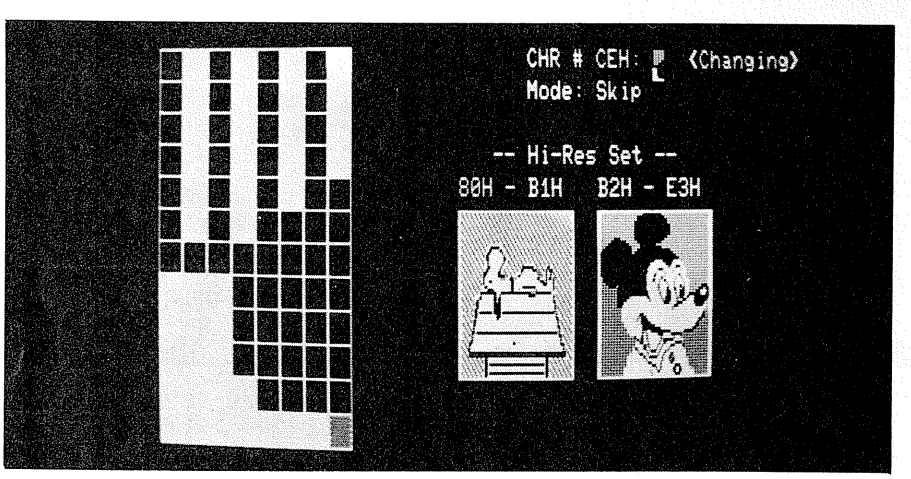
The Assembler features local labels, conditional assembly, page headers and symbol table cross-reference listings. The full standard 6809 instruction set is supported, along with many additional pseudo-ops. Input from multiple input files.

The text editor may be used on any type of text file, but has been designed specifically for quick and easy editing of assembly language source files.

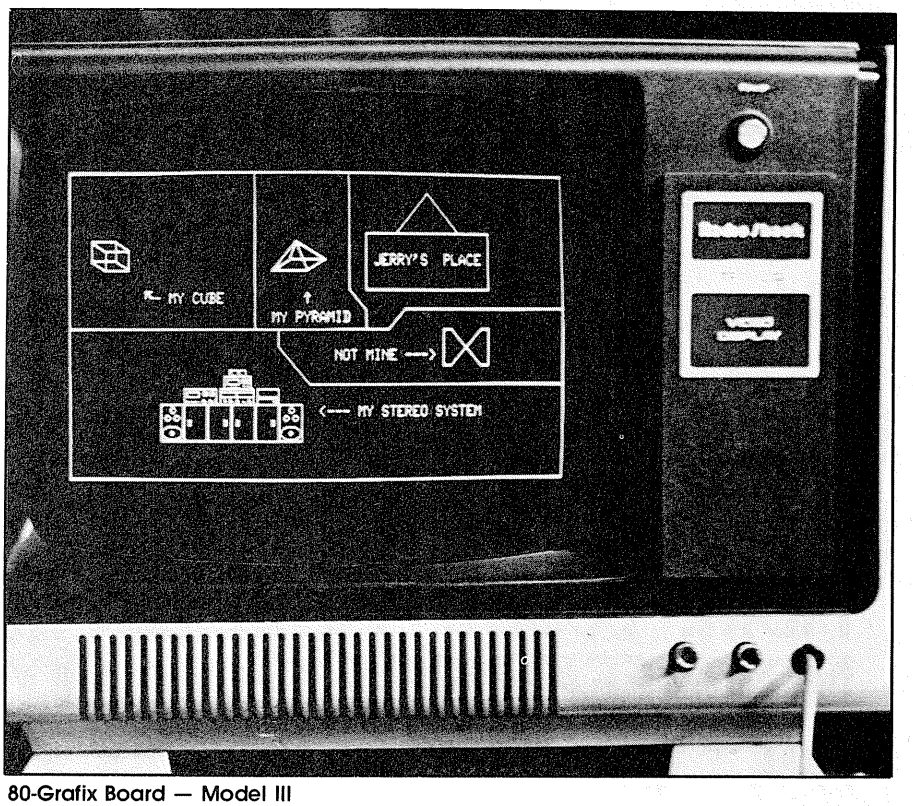
A machine language monitor is included, and memory may be dumped to a printer or a text file. Breakpoints may be set, blocks of memory set or moved.

MACRO-80c runs on a 16K or 32K computer, one Radio Shack disk drive is required, and multiple drives are supported. $99.95 from The Micro Works, P.O. Box 1110 Del Mar, CA 92014, (714) 942-2400.

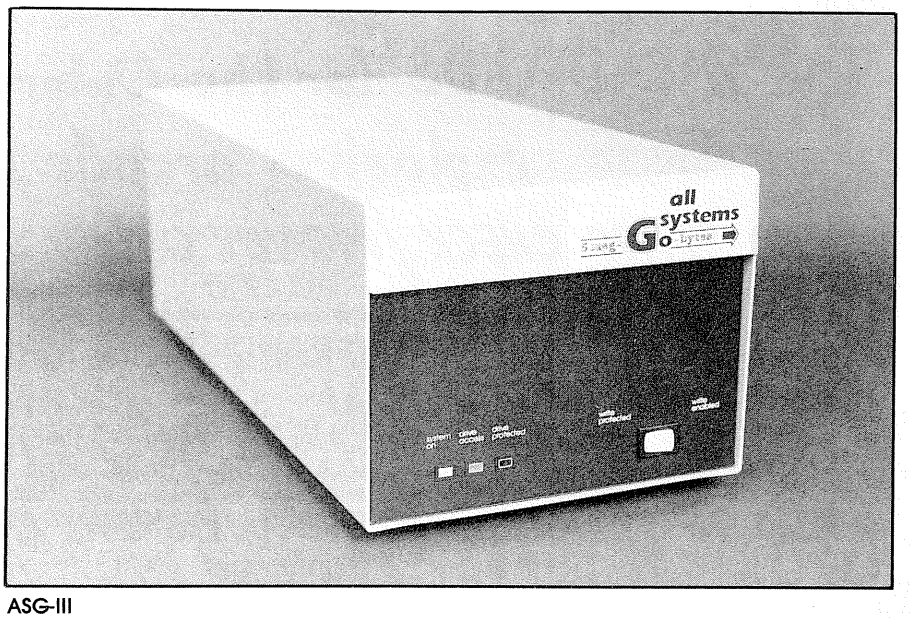
 # 211



80-Grafix Board — Model I



80-Grafix Board — Model III



ASG-III

80-U.S. Journal

Annual index for 1982

Advertiser index

RSN	Advertiser	Page
97	A-K Software	124
48	ALPS	81
49	ALPS	119
103	AR Systems	125
23	Aardvark-80	43
123	Abacus Associates	127
133	Access Unlimited	140, 141
34	Ace Computer Rentals	60
18	Adventure International	31
43	Adventure International	74
70	Algorix	61
125	American Computer	127
35	American Small Business Computers	63
55	Anitek Software Products	89
25	Apparat, Inc.	45
81	Arctic Computer Systems	120
115	Armadillo International Software	126
89	Aspen Ribbons, Inc.	120
★	Astro Star Enterprises	119
41	Avalon Hill Microcomputer Games	71
38	B. T. Enterprises	68
96	BAP$ Software	124
111	Barclay Whyte Associates	125
51	Binary Devices	86
110	Bock, Eugene & Associates	125
132	Business Division, The (A Division of Scott Adams, Inc.)	139
83	CDC	120
99	CDC	124
11	CLOAD Magazine, Inc.	17
101	CRB Microtools	124
76	Cer Comp	111
10	Chromasette Magazine	16
112	Clough, Eric	126
117	Comnet Software Associates	126
67	CompuKids Magazine	105
39	Computer House	69
122	Computer Label Company	127
64	Computer Plus	97
131	Computer Shack	138
66	Computer Shopper	101
56	Computer Trader	89
14	Computing Teacher, The	23
95	Convert-A-Disk	124
37	Cosmopolitan Electronics Corp.	67
★	D&M Software	83
74	Data Associates	114
119	Data Mania, Inc.	126
113	Dental Computer Newsletter	126
126	Disk 'n Data	127
114	EAP Company	126
86	East Texas Color Computer Club	120
44	Eigen Systems	76
★	80-NW Books	39, 54
★	80-U.S. Journal	124, 125
53	Electronic Specialists, Inc.	88
7	Epson America, Inc.	12, 13
116	Excellonix	126
2	Fantastic Software	3
87	Fielding, Don	120
54	Fink, William	88
118	Glenn/Cliff Associates	126
★	Gooth Software	41
22	HPB Vector Company	37
71	Hacks	111
5	Hexagon Systems	9
85	Holman D-P Service	120
107	Holman D-P Service	125
63	Holmes Engineering, Inc.	95
33	Howe Software	59
12	IJG	19
17	IJG	29
29	IJG	53
50	IJG	85
127	Ilume Design	127
137	Interpro Corporation	50
65	J&M Systems, Ltd.	99
6	J. F. Consulting	11
57	Jessup Company, The	90
31	Jimscot, Inc.	56
45	KWIK Software	77
105	Kutten, L. J., attorney	125
19	LNW Research Corporation	33
47	Langley-St. Clair Systems, Inc.	79
80	Lawyer's Microcomputer, The	118
136	Leading Edge, The	144
46	Lemons Tech Services	77
4	Lindbergh Systems	8
27	Lobo Drives International	49
1	Logical Systems, Inc.	2
15	Logical Systems, Inc.	25
106	MDFS, Inc.	125
32	MISOSYS	57
26	Marymac Industries, Inc.	46
60	Micro Architect, Inc.	91
124	Micro Grip, Ltd.	127
20	Micro Management Systems, Inc.	35
121	Micro Moonlighter Newsletter	127
104	Micro Technology	125
13	Micro-Systems Software, Inc.	21
130	Micro-Systems Software, Inc.	137
8	Micro-Labs, Inc.	14
92	Microsette Company	122
★	Nepenthe Programs	127
100	New Classic Software	124
61	Nocona Electronics	93
62	Oasis West	95
69	One Stop Software	109
52	Orion Instruments	87
93	PMC Software	123
94	PMC Software	123
40	Perry Oil & Gas, Inc.	70
21	Pioneer Software	36
24	Powersoft	44
72	Powersoft	113
75	Powersoft	115
78	Powersoft	117
129	Programmer's Guild, The	132
3	Prosoft	7
128	Prosoft	129
★	Publisher's Exchange	127
84	Quatrk Computer Systems, Inc.	120
82	REM Industries, Inc.	120
68	REMarkable Software	107
42	Radio Shack	72, 73
135	Radio Shack	143
58	Rainbow, The	90
98	Realty Software Company	124
77	Rimes Computer Products	116
120	Scientific Engineering Laboratories	126
102	Skeberdis, Pete	124
134	Soft Sector Marketing, Inc.	142
88	Softlex Games	120
36	Software Options, Inc.	65
28	Specialized Software, Inc.	51
9	Standard and Poor's Corp.	15
73	Sugar Software	114
91	TOPS Programming Enterprises	121
30	Think Software, Inc.	55
59	3G Company, Inc.	91
109	Toucan Software	125
16	Trend Software	27
90	Triple D	121
79	Western Operations	118

★ **Contact these advertisers directly**

Bulletin board

This bulletin board space is available free to individuals with single or unusual items for sale or trade, and for other announcements of interest to the general readership of this magazine. 80-U.S. Journal reserves the right to reject any commercial advertising in this section and suggests using our display advertising for that purpose.

These notices are free of charge and will be printed one time only on a space available basis. Notices will be accepted from individuals or bona fide computer user clubs only. All announcements must be typed, contain 75 words or less and include complete name and address.

CLEARANCE SALE: 1977 TRS-80 Model I 16K with good keyboard, 1982 Microtek MT-32C Expansion Interface with 32K for a total of 48K, 1981 Exatron Stringy Floppy with starter kit and extra wafers, 1981 Epson MX-70 with cable. Enough software to overwhelm the average human. Includes dust covers, system desk, and manuals. Best haggle. A true classic. Jon Waples, 70 Ives Road, E. Greenwich, RI 02818. (401) 885-1763.

OCTUG: The Orange County California TRS-80 User Group meets the third Sunday of the month from 1:00 to 4:00 p.m. in Hashinger Hall of Chapman College. Call (714) 638-7889 for voice information or (714) 530-8226 for computer bulletin board. Write P.O. Box 5278-95, Santa Ana, CA 92704 & 0728 for more information.

METRO TRS-80 User's Group meets second Wednesday of the month after working hours. Contact G. Mueden, 310 West 106 Street, New York, NY 10025 or phone (212) 222-8751 for further information.

HELP Needed: 1) An inexpensive way to get lowercase on a PMC-80 computer (PMC-80 wants $75). 2) A patch to allow Exatron's "Type Right Secretary" to work with an Okidata M-80 printer. 3) A Stringy Floppy patch for the latest cassette version of Microsoft's EDTASM+. Does anyone have or know where I can get my answers? Contact Wayne King, 36 Sickles St. #2D, New York, NY 10040, phone (212) 567-3395.

For Sale: CP/M for Model III combined with 80 by 24 screen. Hurricane Compactor 1 and Compactor 4. Misc. CP/M programs included. $300, call Larry (408) 475-2733 or write to 2853 Park Ave., Soquel, CA 95073.

WANTED: Software house to represent me and sell my programs. Also, agent to handle book on BASIC programming — and two other subjects. Have spreadsheet forecasting and Dungeons and Dragons DM programs to market. Willing to sell to individuals for $24.50 each, ppd. Larry Lipton, Harrington, Maine 04643-0134.

ANYBODY know how to, or which ROM to change in an NEC-8023-A printer to be able to print Radio Shack graphics? Please contact Leslie F. Rudolph, 3370 S.E. Bielmeier Rd., Box 41, Port Orchard, WA 98366.

Highest Bidder Gets 'Em: *80-U.S. Journal* from July '79 thru Aug '82, missing Mar/Apr '80. Radio Shack's Microcomputer News from May '79 thru Aug. '82 (missing July '79, June '80, Feb. '81 and June '82). Will notify highest bidder. R. Mueller, 1608 Edgemont Ave., Bristol, TN 37620.

TRS-80 Users Group of Eastern Massachusetts meets in Weston on the second Wednesday of the month. Contact Matthew W. Slate, 42 Oak Hill Road, Sudbury, MA 01776 or phone (617) 443-3327 for more information.

M I C R O T E R M

More and more hardware and communications services are allowing speeds up to 1200 baud. Soon, some may be going faster than that. Today's terminal software simply can't keep up. But now there is an alternative. Micro-Systems Software introduces MicroTerm, the high speed terminal.

Model III MicroTerm will communicate, without insertion of null characters, at 4800 baud. Guaranteed. No cop-outs, no question. MicroTerm is so fast that you can exit from the terminal to the main menu, adjust video width, open the buffer, turn on the printer, or any one of dozens of other functions, and return to the terminal mode **without missing a thing!**

MicroTerm continues to input from the RS232, even while at the main menu. This is the only terminal capable of such an astounding feat. MicroTerm offers you most of the features that "Brand X" smart terminals have, plus it gives you: • Ultra high baud rate operation (up to 9600 in certain cases). • Input while at menu. • Easy to use translation tables. • Easy to use phone number listings. • Maximum auto dial support — most major brands. • Direct file transfer companion program included at no extra cost (compatible with DFT). • DOS commands from menu without exiting program. • Over 34K of capture buffer (in a 48K TRS-80). • Can be set to automatically dial telephone and transmit buffer at preset time without **any** operator intervention.

And many, many more great features, MicroTerm is so fast you must see it to believe it. The various menus are displayed so fast, they seem to jump out at you. Status of various functions can be displayed and altered in split seconds.

For the computerist who wants the ultimate, state-of-the-art terminal software, there is no other choice.

MicroTerm retails for $79.95, but registered DOSPLUS owners can purchase it for only $59.95. $20.00 off the retail price! MicroTerm comes complete with the terminal program, the direct file transfer program, some standard translation tables, and documentation.

Don't delay, order yours today! Specify when ordering: Model I or III and whether you want it on 40 or 80 track media. Requires a 16K TRS-80 with one disk drive. We recommend 48K for serious communications work. MicroTerm will be available beginning June 30, 1982.

MICRO-SYSTEMS SOFTWARE, INC.

4301-18 Oak Circle
Boca Raton, FL 33431
Telephone: (305) 983-3390
800-327-8724

130

— COMPUTER SHACK —

★ BEST BUY ON DISKS ★

BASF has taken one of the best disks and made it even better. The quality is so good that BASF guarantees each disk for not one year, not five years, but for LIFE. If anything ever goes wrong you can return it to us or to BASF and it will be replaced at no charge. Can you believe MAJOR brand name, LIFETIME guarantee, FAST shipment and at these prices.

35/40 Track Single Density 1 side ... $19.95
35/40 Track Double Density 1 side... $21.95
35/40 Track Double Density Flippy ... $29.95
The following come in special Computer Shack packaging.
35/40 Track Double Density 2 sides.... $29.95
77/80 Track Double Density 1 side... $25.95
77/80 Track Double Density 2 sides.... $29.95

All of the above come in Boxes of 10 with BASF labels, hub rings and Tyvek sleeves. All disks have one write protect notch and one read hole, except the Flippy's which have two.

Disk Savers (vinyl sleeves) 20 for **$6.00**
Colored Disk box (holds 10 disks) ... $2.95@
5 Colored Disk boxes (red, blue, green, yellow, and brown asst. colors).... **$14.50**

FAMILY TREE

Excellent family genealogy program works on both the Model I or Model III. Along with the normal documentation we send a manual with over 200 pages of instruction on how to research your family tree. You can get three different printouts, a family tree or a family report of each person, or a birthday printout. Sort on Birthdays. The Nebraska State Genealogical Society tested our program and said "Your program will help form a valuable data source that may be accessed an used in different ways. The printouts are excellent."
Disk or Tape **$29.00**

JOVIAN

Newest game by Dunlevy and Frayer. This is Cyborg with a space theme! 9 screens of action simultaneously. Featuring WRAP AROUND ACTION". Most exciting spaceship type game out for the TRS-80.
Only ..... **(19.95/24.95)**

ARACHNID PLUS

Three exciting first class **arcade** games for the TRS-80. A special value pack including a real time arcade game, an exciting car race game, and a strategy game. All excellent games for only ..... **(19.95/24.95)**

SUPER DIRECTORY

The Best Catalog program on the market. One program reads Multidos, NewDos 80, Dos Plus, LDos and TRS-DOS. Automatic density recognition! Automatic track count. Reads almost anything. Superfast sort routines! (Sorts on 5 different fields). Super fast find routines (five different find routines including a string search)
ONLY **$39.95**

NEW!
MONTHLY BILL SYSTEM

This is a new program for the businessman who wants to send a monthly or quarterly bill to the same people every billing period. (Landlords, garbage man, etc.) The program will send out a bill to each person (you can set the frequency, monthly, quarterly, etc.). It then has a very easy way of inputing your paid customers. This will help you to keep track of who is past due. Gives you printouts of your financial condition at any time.
Disk only. **$149.95**
We also have a Special program for Auto, and Home Insurance agents, Call for complete details.

Small Business Programs

CHECKING ACCOUNT Mod 1 or III, 48k disk $39
Excellent check writing program for small businessman or for personal use. Menu operated for easy use. Has Screen editor, 99 catagories or expenses. Sorts on payee, Check number, or date. It can print your checks on the printer.

BILLING SYSTEM Mod. I or III, 48k disk $39.00
A billing system written by a businessman for his own use. It is simple, fast and easy to use. Has a screen editor for quick error correction. Prints out invoices, then will send a monthly bill to each customer. It will add interest or carrying charges. This is not a complicated accounts receivable it is a simple but effective way to keep track of who owe's you money for the company that doesn't do a lot of credit business.

E BASIC

There are five major **Dos's** on the market all have there good points. Some folks like one, some like another, but every single reviewer has said that MULTIDOS had the BEST BASIC. Some would stop here, but not Vernon Hestor. He now brings us EBASIC, a new innovative state of the art basic. It makes Graphics and Sound EASY. This will give the basic programer more power than ever! Here are some of the new basic commands: print(a,b), input(a,b), line input(a.b), print$, call, sort, labeling, array read, array let, cound, shape, circles, cubes .... All of this and you still have over 38K for your basic programs. EBASIC is only $29.95 for MultiDos owners.

MULTIDOS

MULTIDOS got bigger and better. New EASY ZAP, New TAPE/CMD, and the New DISK DRIVE TIMER. All of this and its still the cheapest full Dos going. **$99.95**

Z DOS

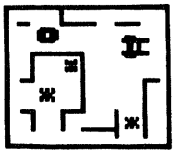
Z Dos the Dos for the person who does not want to spend a hundred dollars or more for an improvement over TRS-DOS. For only $39.95 you will double your fun. For only $39.95 you can get what reviewers in three different magazines said was the BEST BASIC for the TRS-80.
Buy MultiDos at the list price of $99.95 and you can take 20% off of any or all of the following.
E Basic $29.95 ..... Super Directory $39.95
Aerocomp Doubler $149.00

SUPER DOS

If you are using TRS-DOS, on the Model III, you can now take out some of its biggest drawbacks. SUPERDOS will make a few automatic zaps to your copy of TRS-DOS and it will boot up instantly! It will give you error messages in English. It has a short directory, and on the long directory it will pause to give you a chance to read the screen.
Only. **$19.95**

CYBORG

This is the biggest blockbuster to come along since the first BIG FIVE games hit the market. A real TRS-80 classic. The first game for the TRS-80 that features **WRAP AROUND GRAPHICS**". This exciting game occurs in an interstellar space station. This game features graphics that are super fast and move smoothly in all FOUR directions. Rated #1 three months in a row.
ONLY ..... **(19.95/24.95)**



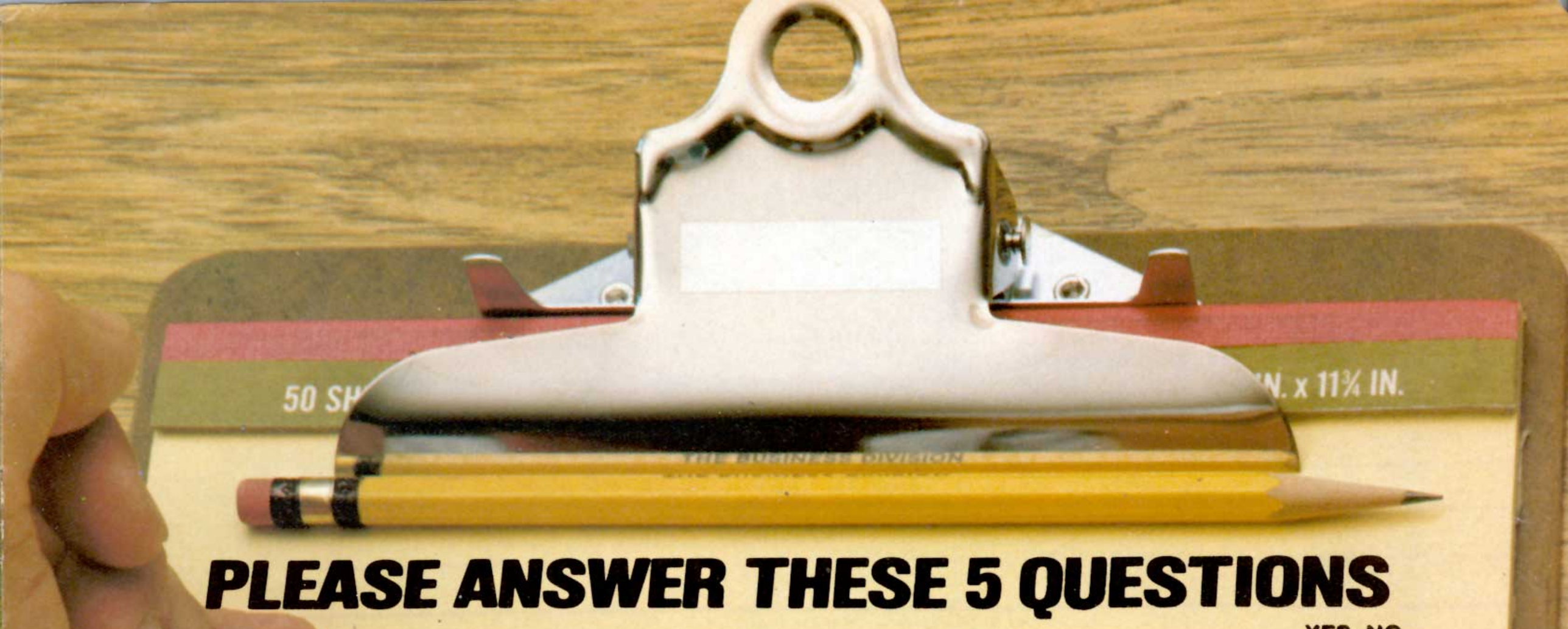
DEMON SEED

The greatest graphics effects yet in a real time arcade game for the TRS-80. Birds of various sizes drop down on you. A gian space ship must be destroyed. A great game based on the arcade game Phoenix".
Only ..... **(19.95/24.95)**

COMPUTER SHACK

1691 Eason ● Pontiac, Michigan 48054
Info: **(313) 673-8700** ● Orders: **CALL TOLL FREE (800) 392-8881**

Master charge and VISA OK. Please add $3.00 for shipping in the U.S.A. - $5.00 for Canada or Mexico - Proper postage outside of U.S. - Canada - Mexico.
Dealers: We are distributors for all items in this ad. Write for our catalog and price list.



PLEASE ANSWER THESE 5 QUESTIONS

	YES	NO
1. Do you have information that must be kept organized and accessible?	☐	☐
2. Do you ever need to perform statistical analyses?	☐	☐
3. Would you like to have a tool that will allow you greater flexibility in managing your own or your company's money?	☐	☐
4. Could you use a mail list program that will be easy to use, maintain an unlimited number of names, allow you great sorting flexibility, and even interact with a word processor?	☐	☐
5. Do you own or have access to a TRS-80 microcomputer?	☐	☐

If you answered "yes" to any of these questions, we can be of assistance to you. Our Maxi Series of applications programs are designed to give you maximum versatility with a minimum of hassle. These programs were created for business use, but you'll find yourself using them for personal applications as well.

Thorough support *— Of course, each program comes with in-depth, user-oriented documentation, and is menu-driven to make it easy to use. When necessary, the Maxi programs are compatible with each other, and, whenever pertinent, are interactive with the major word processing and spreadsheet programs published by other manufacturers. Also, we maintain a telephone support line to provide you with any assistance you might require.*

Maxi Manager *by Dale Kubler*

Maxi Manager is a remarkable data base manager. Its fast machine language sort complements its large data storage capacity. The sophistication of its data entry, management, and printing capabilities makes Maxi Manager a versatile tool for many applications.

The program now includes Maxi Utility, which allows you to rescue files on diskettes that have been damaged by excessive wear or misuse and lets you expand, add, or delete fields from an existing data base. Let the unmatched capabilities of Maxi Manager handle your data management!

Model I & Model III. Minimum 1 disk drive required 012-0196 $149.99

Maxi Mail *by Dale Kubler*

Maxi Mail is a powerful mail list management system that:

1. Is easy to use.
2. Has virtually unlimited storage capacity.
3. Interfaces with the major word processors to generate form letters and other text.
4. Prints mailing labels up to four-across in any format desired.
5. Has virtually unlimited coding capabilities with thirteen fields of information for each record.

Maxi Mail is the most sophisticated user-oriented mail list program available.

TRS-80 Model III only. 2 disk drives required 012-0148 $99.95

Maxi Cras *by Dale Kubler*

(Check Register Accounting System)

Maxi Cras is a system that will computerize check writing, recording and analysis for business and personal finance. The system features 223 income and expense accounts, each of which will handle an unlimited number of transactions. Extensive register and report printing capabilities make Maxi CRAS an indispensible tool for managing money effectively.

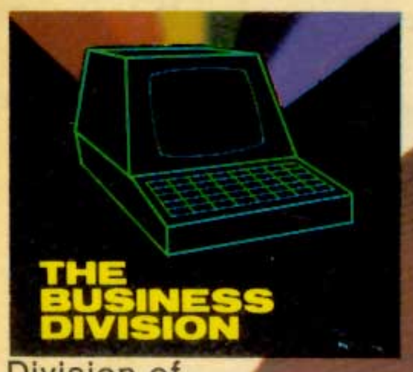
Model I & Model III. Minimum 2 drives required 012-0145 $99.95

Maxi Stat *by David Walonick*

Maxi Stat is the most useful statistical analysis package on the market today. It was developed to allow maximum flexibility in designing customized analysis. Maxi Stat handles the three main components of statistical analysis:

1. Complete menu-driven codebook creation and editing.
2. Menu-assisted data entry.
3. User-created control files to describe the statistical analyses to be performed and printed out on the variables of your choice.

Model I & Model III. Minimum 2 drives required 012-0153 $199.95

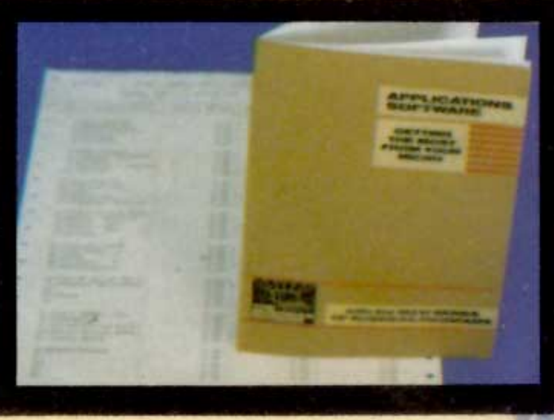


A Division of
Scott Adams, Inc.

132

Send $1.00 for our 16 page booklet "Getting The Most From Your Micro" All 16 pages are packed with indepth explanations and printout samples from the Maxi Series of applications programs.

THE BUSINESS DIVISION
BOX 3435
LONGWOOD,FL 32750
(305) 830-8194



ANNOUNCING THE NEW
ACCESS UNLIMITED

FREE

FREE! DOSPLUS 3.4 disk operating system with any Percom or Access Unlimited Drive for the Model III***

We have just added approximately $100,000.00 worth of brand new business and game software.

Call or write for our new FREE catalog.

Save $$ on our most popular items!

Percom Data Separator (reg. $29.95)	now **$23.95**
Screens for Models I*, II*,III Green, Lt. Blue, Dark blue, and Amber. Bronze for color video (reg. $24.95)	now **$13.95**
Head Cleaning Kit (reg. $29.95)	now **$19.95**
Drive Numbering Tabs, pkg. 0-3 (reg. $4.50)	now **$3.95**
Flip-N-File 5¼" (reg. $39.95)	now **$18.95**
Flip-N-File 8" (reg. $54.95)	now **$23.95**

AFD 100-40-track single-density drives $275.00

NEW! 5¼" Soft Sector Diskettes W/Hub Ring

"BASF 5¼" Lifetime Warranty Reg. $44.95
Now $22.90 SSDD Box of 10

ATHANA 1 yr Warranty **$19.90** SSSD Box Of 10

Anti-Glare Masks—Now $12.95

8" Diskettes from $22.75

Other Diskettes at Equal Savings

Flip 'N File Boxes—Holds 50 Diskettes†
5¼" Reg. $34.95 Sale: **$19.95** 8" Reg. $49.95 Sale: **$29.95**

LIBRARY CASES—Holds 10 Diskettes Sale **$2.95 ea.**

Bare Drives for "IBM PC"—Internal or External — $269.95 ea.

Percom TFD 100 Mod. 1 Drive
Reg. $439.00 Ea. Now $278.00 ea.
OR Two for $540.00. Only 100 left.

DOS + 3.4 Reg. $159.00. SALE.
FREE DOS + 3.4 with any Percom or AU Drive for the Mod. I* or III-*
Model III* Drive Reg. $499 Now $449.
Add Ons $344 Single and $628 Double

Anti-Static Mats—Colors: Russett, Blue & Gold, Natural Brown, & Golden Brown.

3′ x 5′	Reg. $75.60	Sale: $57.00
4′ x 6′	Reg. $120.90	Sale: $91.00
4′ x 8′	Reg. $161.30	Sale: $122.00

Typing Easels—$14.95

SAVE $50.00 on fully-tested PERCOM Disk Drive for Model III*
Reg. $499.00. Now $449.00

Percom's internally-mounted drives, with their widely-acclaimed disk controllers, are completely pre-tested with a 48 hour Burn-in. So, you know it works right when you get it. Choose single or double-density storage capacity.

A First-Drive System includes the four-drive disk controller, your choice of one or two drives, power supplies, cables and a fully-illustrated, easy-to-understand owner's manual.

OVER $500,000 IN NEW COMPUTERS, PERIPHERALS AND SUPPLIES. NEW PRINTERS "CENTRONICS"—"C'ITOH"—"OKIDATA"—"STAR"—"TALLY". VALUES TO $3500.00, NOW $325.00 to $1695.00. Call for latest price.

Make sure it's done right. Let us install your First-Drive System.

Purchase your Model III* Drives from us and for $39.95 plus shipping, you can have our experienced professionals perform a 48-hour burn-in of your Model III* computer, install the drive system, and check out your expanded system to make sure everything works correctly. Call for pricing of complete Model III* system with 2 to 4 Access Unlimited or Percom drives.

Get the best! Our own fully-tested drives* for as low as $275.00 *Bare Add-on.

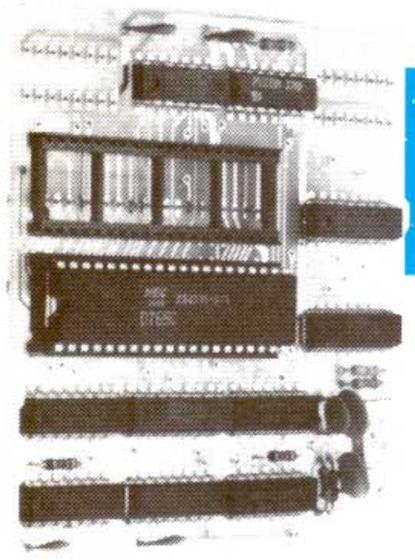
Every Access Unlimited Floppy and Flippy Disk Drive is electrically and mechanically tested, then burned-in for 48 hours under operating conditions. The signed test list in your shipping carton is proof of that.

Our floppy disk drives offer you either single or double density. With double-density, it stores up to 364 Kbytes, depending on the format. And you can use both sides of the diskette on our flippy drives.

SATISFACTION GUARANTEED! If you're not completely satisfied with your Disk Drive, return it within 15 days for a full refund. All our drives have a comprehensive 90-day limited warranty.

• LIMITED TIME OFFER/LIMITED QUANTITIES •
Prices subject to change without notice.

MICRO SHOPPING CENTER



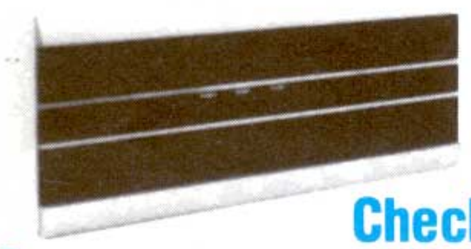
SAVE $$ 3 WAYS! now $184.95!

Upgrade your Model I* to Double Density

Percom's proven Doubler II™ with DOSPLUS 3.4** — a $309.95 value, now $155.00!

The completely assembled and fully-tested Percom DOUBLER II™ is ready to go to work when you get it. All you have to do is plug it in. It can store over 4 times more on one side of a diskette than a standard Tandy Model 1* drive. It reads, writes, and formats either single or double-density. Comes with a DOSPLUS 3.4** disk-operation system. Runs TRSDOS*, NEWDOS**, Percom OS-80™, and other single-density software. (You can change to double-density at any time.) Uses Move 1-2-3 utility ($29.95) or other appropriate software to read Model III* programs on your Model I*. Has on-card high performance data separation and write precompensation circuitry.

Doubler II™	$159.95
DOSPLUS 3.4D**	150.00
Total Value	$309.95
Now only	$155.00
You SAVE	$154.95



From **$1,995 (5 MB)**

TRS-80, "IBM-PC", ATARI

Check out the benefits of Access Unlimited Hard Disk Drives

- ✓ can be used for Model II*, Model III* computers.
- ✓ 5, 10 Megabyte and larger drive units.
- ✓ lets controller handle up to 4 drives.
- ✓ works along with existing floppy disk drives.
- ✓ includes host system support software.
- ✓ 15-day FREE trial period.

SAVE on an Arrick Quick Switch‡

Changes a TRS-80* printer port or a peripheral between computers instantly and easily. Available for RS-232 and Centronics. Plugs included.

Now from $99.95. Cables from $31.95.

LAST MINUTE CLOSE-OUT

TFD 100 40 track floppy disk drives – only 200 left reg. 439.00 – now 278.90 or 2 for $540.00.

"PAGEMATE" — $15.95

" " Reg. Trademarks
The above prices do not include state taxes.

Enjoy 1-stop shopping and BIG savings

Order your disks, software, supplies, and accessories from Access Unlimited. We offer low prices on automatic Datamarc*** Sheet and Envelope Feeders for Radio Shack*, QUME, Ricoh, Diablo Daisywheel, and Dot Matrix Printers. And the more you buy at one time, the more you can save.

For more information, call (214) 340-5366 Monday through Friday; (214) 690-0207 after 5 p.m. and Saturdays.

To order or for FREE literature, call TOLL-FREE

VISA **1-800-527-3475** master charge

Ordering Information

Order by phone or by mail. We accept Visa, MasterCard, cashier's checks, certified checks, and money orders. With personal checks, allow additional time for bank clearance. Your bankcard will not be charged until your order is shipped. On orders over $1,000, we pay freight (surface only) and insurance; please add $3.00 shipping and handling under 50 lbs. Over 50 lbs., add $5.00 for orders under $1,000.00. Texas residents add 5% sales tax. Allow 2 to 4 weeks for delivery.

EXPLANATION OF SYMBOLS
* Trademark of Tandy Radio Shack Corp.
†† Trademark of Apparat Corporation
TM Trademark of Access Unlimited
† Trademark of Percom Data Company, Inc.
** Trademark of Micro-Systems Software, Inc.
‡ Trademark of Arrick Products Company
*** Trademark of Datamarc, Inc.
x Trademark of Digital Research

☐ Please send me a FREE catalog. I'm not ready to order at this time.
☐ YES, I'm taking advantage of your Sales prices.

Name _____
Company Name _____
Address _____
City _____ State _____ Zip _____
Phone Number ( ) _____

Quantity	Item	Unit Price	Subtotal
		Subtotal	
		State Sales Tax (Texas residents only)	
		handling charge	
		Total	

Check one:
☐ payment enclosed ☐ Visa ☐ MasterCard*

☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

*If MasterCard, numbers above name: ☐☐☐☐

Expiration Date: ☐☐ — ☐☐

Authorized signature, if charged

ACCESS UNLIMITED

DEPT. C/401 N. Central Expwy. #600/Richardson, Texas 75080
Tel. 1-800/527-3475 214/340-5366
214/690-0207 — Sat. and Evenings Only

OUTHOUSE

Is There Nothing Sacred?

DISK VERSION TALKS

JOYSTICK Compatible*

Don't Miss This Outstanding Original Arcade Game!

SSM SOFT SECTOR MARKETING, INCORPORATED
6250 Middlebelt • Garden City, MI 48135 • 800-521-6504/(313)425-4020

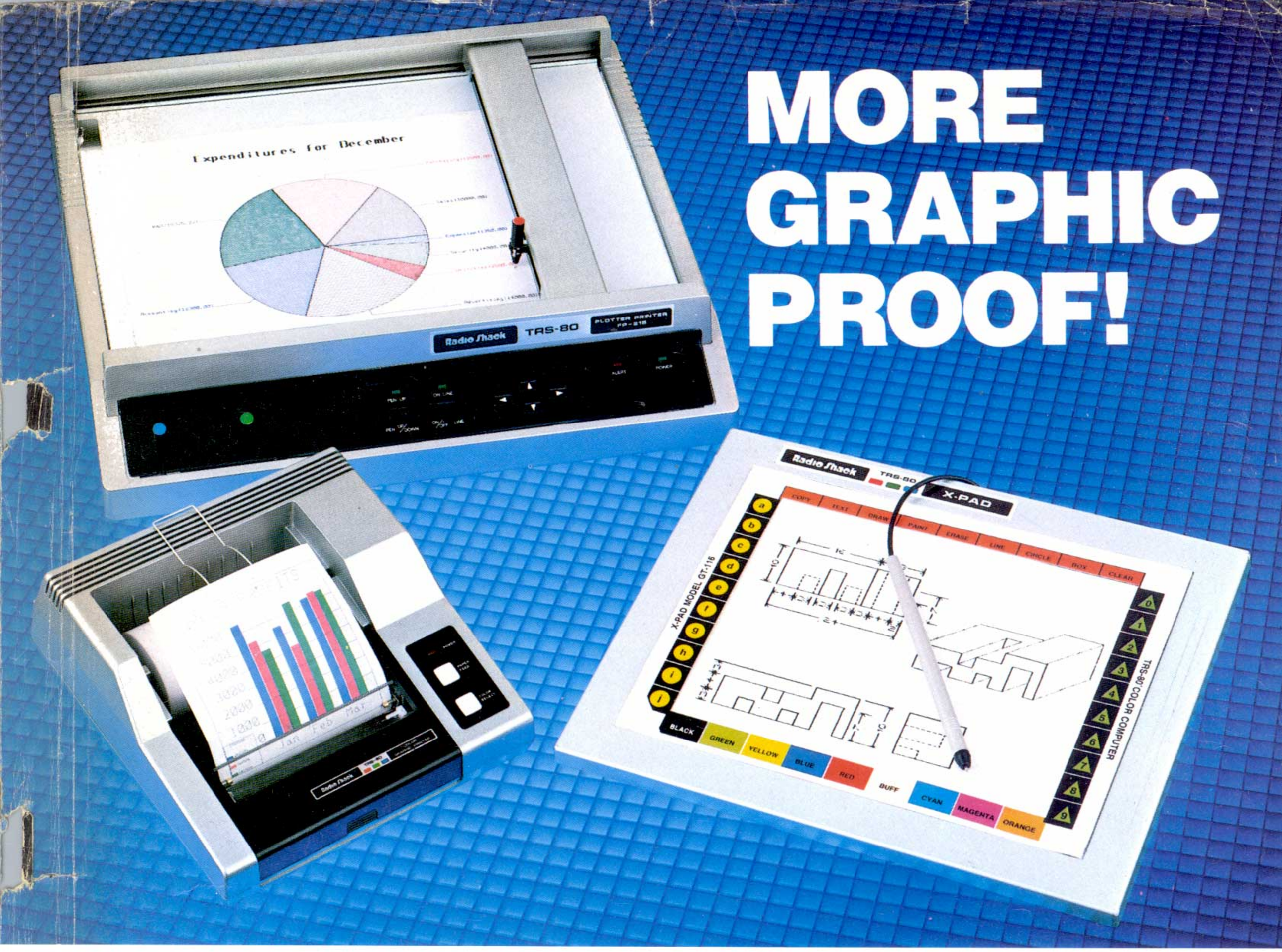
Prices per Game: TRS-80 16K Level II Mod I/Mod III Cassette $15.95
TRS-80 32K Level II Mod I/Mod III Diskette $19.95

10% discount for 2 items, 15% for 3 or more. Please add $2.50 per order for postage & handling, Michigan residents add 4% sales tax. Outside USA (except Canada) please add $10.00 per order for postage & handling.

Talking and sound effects are playable through the cassette AUX plug. High scores are automatically saved after each game on disk versions. *Compatible with Trisstick.
Call or write for our complete catalog.

MasterCard VISA AMERICAN EXPRESS

The Quality Continues . . .



MORE GRAPHIC PROOF!

Radio Shack's TRS-80® is #1

These new peripherals let you explore an exciting dimension of your TRS-80—*Graphics!* They're three more reasons Radio Shack's TRS-80 is the best supported microcomputer in the world!

New! FP-215 Flatbed Plotter/Printer. This intelligent, 4-color, single-pen graphics device draws charts and graphs using BASIC ASCII command codes and parameters, and prints text horizontally or vertically using the BASIC LPRINT command. Parallel and Color Computer-compatible serial interfaces. Cat. No. 26-1193, $995.00.

New! CGP-115 Color Graphics Printer. Ultra-compact—yet produces beautiful graphics in red, blue, green and black on standard 4½″ roll paper and prints alphanumerics at 40 or 80 characters per line. Built-in commands simplify drawing and plotting. Parallel and Color Computer-compatible serial interfaces. 26-1192, $249.95.

New! GT-116 Graphics Tablet. Instant graphics at the touch of a pen! What you draw or trace is read into your TRS-80 Color Computer. Compatible with Extended BASIC graphics commands. Menus and other options simplify data input. 26-1196, $349.95.

See Them All at Radio Shack—Your Graphics Headquarters. Discover how to turn your TRS-80 computer into a complete graphics system. Stop by your nearby Radio Shack Computer Center, store or participating dealer to get a "hands on" demonstration of these peripherals. Be sure to ask about our six-pen color plotter and our 11 × 17″ digitizer, too.

Radio Shack®

The biggest name in little computers™

A DIVISION OF TANDY CORPORATION

Send me your free TRS-80 Computer Catalog today!

Mail To: Radio Shack, Dept. 83-A-175
300 One Tandy Center, Fort Worth, Texas 76102

NAME _____

ADDRESS _____

CITY _____ STATE _____ ZIP _____

PHONE _____

Retail prices may vary at individual stores and dealers. Special order may be required at some stores.

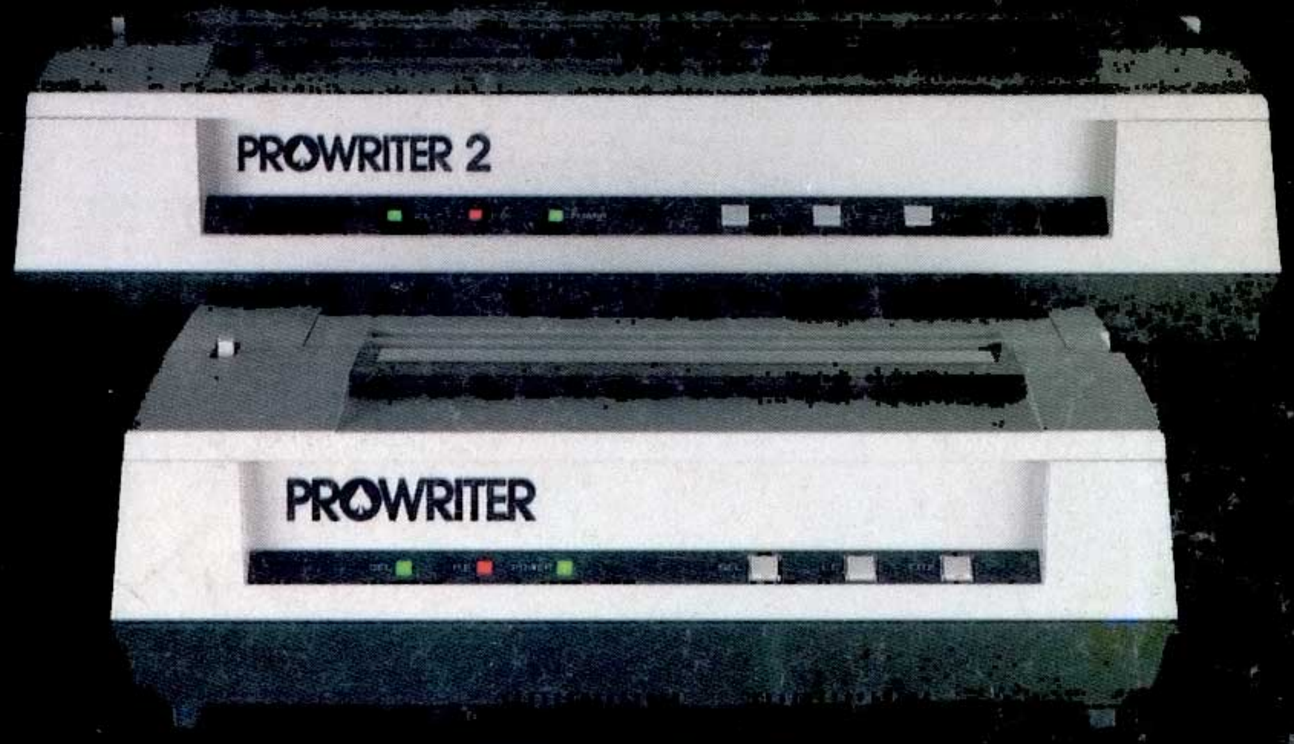
THE LEADING EDGE IN PRINTERS

ONE GREAT LINE. ONE GREAT WARRANTY.

Finally, there's one full family of printers that covers every business or word processing application—all from C. Itoh, a company known for packing more product into less price; and all distributed exclusively by Leading Edge, a company known for searching out and providing that very thing. Which means that one call to one source can get you any printer, any time you need it, for any purpose. All backed by a full years' warranty from Leading Edge. (Try *that* on any other line of printers.)

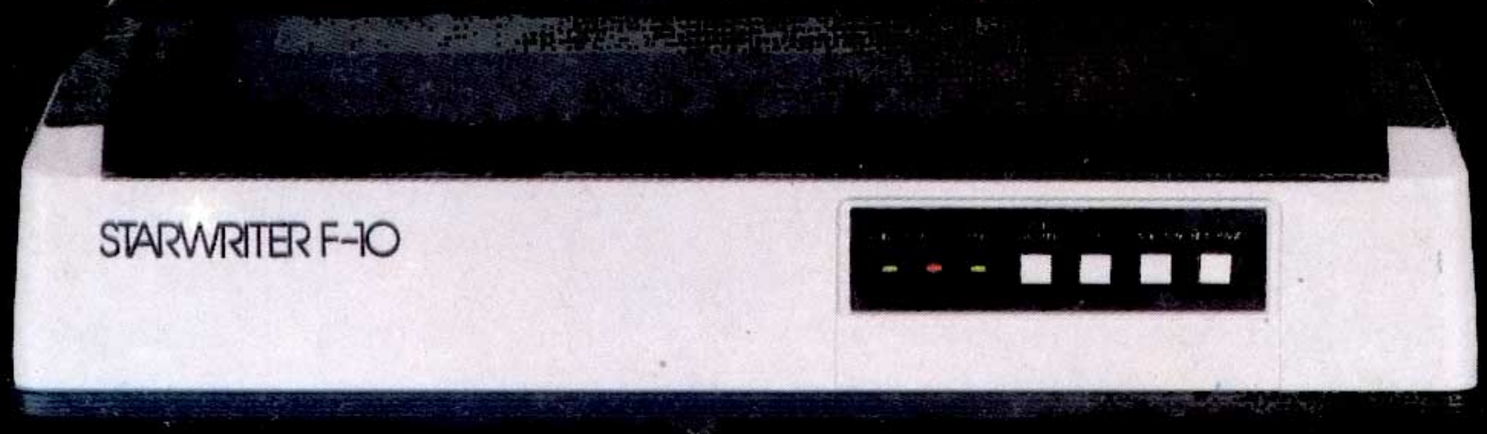
THE PRO'S.

The Prowriters: business printers—and more. The "more" is a dot-matrix process with more dots. It gives you denser, correspondence quality copy (as opposed to business quality copy, which looks like a bad job of spray-painting).
Prowriter: 120 cps. 80 columns dot matrix compressable to 136. 10" carriage. Parallel or serial interface.
Prowriter 2: Same as Prowriter, except 15" carriage allows full 136 columns in normal print mode. Parallel or serial interface.



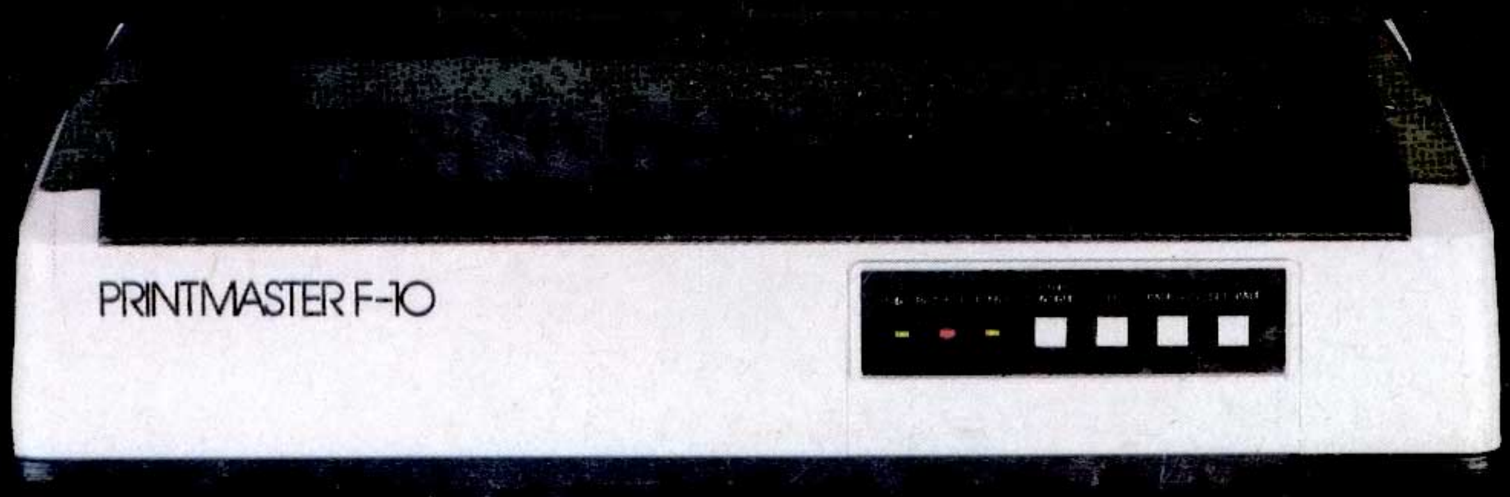
THE STAR.

The Starwriter F-10. In short (or more precisely, in a sleek 6" high, 30-pound unit), it gives you more of just about everything—except bulk and noise—than any other printer in its price range. It's a 40 cps letter-quality daisy-wheel with a bunch of built-in functions to simplify and speed up word processing. It plugs into almost any micro on the market, serial or parallel.



THE MASTER.

The Printmaster F-10. Does all the same good stuff as the Starwriter except, at 55 cps, the Master does it faster.



Distributed Exclusively by Leading Edge Products, Inc., 225 Turnpike Street, Canton, Massachusetts 02021.
Call: toll-free 1-800-343-6833; or in Massachusetts call collect (617) 828-8150. Telex 951-624.

#136