February 1984 USA $4.00

80micro

A WAYNE GREEN PUBLICATION

the magazine for TRS-80* users ®

Young Programmers
The Best of the Brightest

New and Improved: The Creator Part II

Let Your Micro Count the Votes

Your TRS-80 Learns to Spell

C•Notes For the Model 100

*TRS-80 IS A TRADEMARK OF RADIO SHACK, A DIVISION OF TANDY CORP.

0 74470 65947 02

80micro 2/84 #49 A

For users of Apple, IBM, TRS/80, Atari, Commodore, Texas Instruments, and other brand name computers:

Here's the easiest way to buy quality diskettes at discount prices

Now you can get error-free double density diskettes by IBM, Control Data, Maxell and Verbatim delivered to your door. For some of the lowest prices around.

You save because we ship huge volumes of magnetic media in boxes of 10.

To order, use this form.
For even faster service, call toll-free.

1-(800)-FLOPPYS or
1-(800)-521-5700
Michigan • 1-800-482-4770
Canada • 1-800-265-4824
Alaska/Hawaii • 1-800-821-9029

ALL MAJOR CREDIT CARDS ACCEPTED
Shipping & handling F.O.B. Southfield
Transaction Storage Systems Inc., Southfield, MI TSS

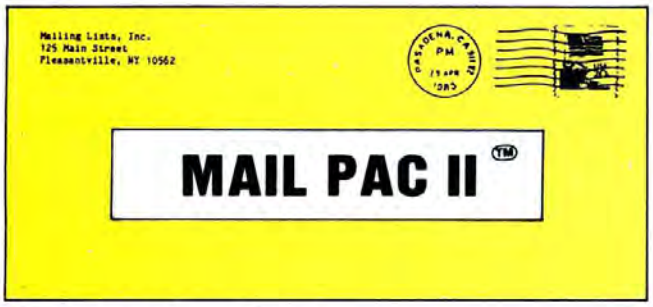
EXPECT A MIRACLE

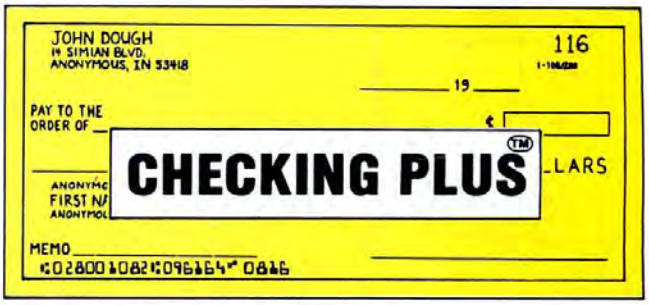
Size	IBM	Qty.	Verbatim®	Qty.	Control Data	Qty.	maxell	Qty.
5-¼" SS	23.90		23.90		19.90		28.90	
5-¼" DS	34.90		34.90		29.90		41.90	
8" SS	24.90		—		28.00		—	
8" DS	39.90		—		29.90		—	
Sub Totals	$_____		$_____		$_____		$_____	

Name _____ Title _____
Company _____
Address _____
City _____ State _____ Zip _____
☐ AMEX ☐ MASTERCARD ☐ VISA
Card # _____ Exp. Date _____
Shipping and handling 1-6 boxes add $4.00 per order
7 or more add $6.00 per order. TSS
Clip and mail today to: Transaction Storage Systems, Inc., 22255 Greenfield Road, Southfield, MI 48075

ANYONE CAN USE OUR SOFTWARE!

MAIL PAC II and CHECKING PLUS represent a new generation of computer software. Our software doesn't even need a manual! Anyone can quickly learn to use either package by following simple, explicit on-screen instructions. But, for those of you who insist -- we've included a complete user's manual as well.





MAIL PAC II FEATURES:

High Capacity --

Stores from one to one million names in Zip Code, Numerical or Alphabetical order. The only limitation is your disk storage space.

Flexibility --

Prints your mailing list on 1, 2, 3, or 4-across labels (with up to 5 user-defined lines on each label) or as a compact, user-designed directory. The record length is completely user-defined, and each field within each of your records is completely variable (allowing storage of any number of characters for any particular address entry). Devote fields to telephone numbers, codes, or even special messages related to each particular name on file.

Powerful --

A built-in word processor allows you to create personalized form letters for each address on your list (or just a particular group of addresses).

Ease of Use --

Create new mailing lists, review existing lists, handle changes of address, delete cancelled names, sort lists, and purge duplicate names from your files. Complete on-screen instructions tell you in plain english exactly how to accomplish all of these tasks.

MAIL PAC II $99.95

CHECKING PLUS FEATURES:

Complete Check Register --

Checking Plus stores your entire check register in a disk file, and then uses the data to balance your account, track your expenses, and help you make budget projections. Review the entire checkbook, enter checks, deposits, fees and adjustments, mark outstanding checks when paid, and get an instant cash balance at any time. All data can be viewed onscreen or printed out in report form.

Tax Preparation --

Storage of monthly and yearly totals and other important information aids in income tax preparation, for your personal use or for your accountant.

Handles the Details --

Store names and addresses for frequently written checks, then print checks to fit standard window envelopes, eliminating the need for extra typing. You can even store any comment, explanation or other message (up to 255 characters) related to a particular check.

Automatic Monthly Bill Payments --

Enter amounts and names of payees for all of your fixed-cost monthly payments, and then sit back while the system automatically prints checks.

CHECKING PLUS $99.95

Both Programs Require 48K And At Least One Disk Drive (2 Drives For Higher Storage Capacities). And All Hard Disk Systems Are Supported.

FREE--TRS-80 Mod I,III & 4 programs supplied on DosPlus (minimum system). Complete DosPlus also available.

FOR YOUR TRS-80™ • APPLE™ • IBM PC™ • NEC™ • OSBORNE™ • XEROX™ • KAYPRO™ • TELEVIDEO™ • ZENITH™ • SANYO™
DEC™ • TI PROFESSIONAL COMPUTER™ • SUPERBRAIN JR.™ • EPSON™ • Any CP/M™ Computer

CP/M-based Computers must be equipped with Microsoft BASIC (MBASIC or BASIC-80)

OUR SOFTWARE CATALOG

H & E Computronics, Inc., has mailed more than 1 million software catalogs since 1978. Send $2 for our new 64-page catalog today! (We also send you our catalog FREE with every order). DEALER INQUIRIES WELCOME

30-DAY MONEY BACK GUARANTEE

*** ALL PRICES & SPECIFICATIONS SUBJECT TO CHANGE ***
DELIVERY SUBJECT TO AVAILABILITY

H&E COMPUTRONICS, INC.

50 N. PASCACK RD., SPRING VALLEY, N.Y. 10977

24 HOUR ORDER LINE
(914) 425-1535

ADD $3.00 FOR SHIPPING IN UPS AREAS
ADD $4.00 FOR C.O.D. OR NON-UPS AREAS
ADD $5.00 TO CANADA & MEXICO
ADD PROPER POSTAGE OUTSIDE OF U.S., CANADA & MEXICO

NEW TOLL-FREE ORDER LINE
(OUTSIDE OF N.Y. STATE)
(800) 431-2818

VISA MasterCard

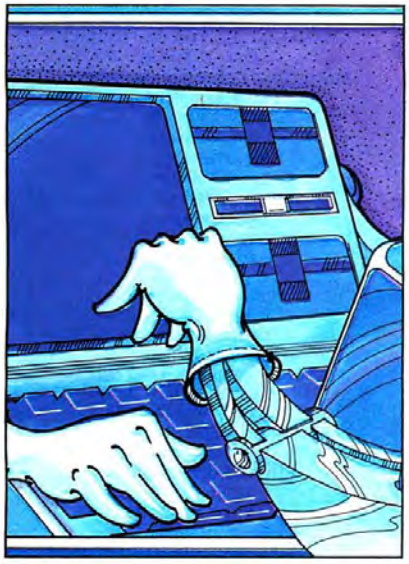
9

PUBLISHER/PRESIDENT
Wayne Green
VICE PRESIDENT/GENERAL MANAGER
Debra Wetherbee
VICE PRESIDENT/FINANCE
Roger Murphy
ASSISTANT TO PRESIDENT/VP
Matt Smith
ASSISTANT TO VP/FINANCE
Dominique Smith
DIRECTOR OF MARKETING & SALES
David Schissler
DIRECTOR OF CIRCULATION
William P. Howard
603-924-9471
RETAIL & NEWSSTAND SALES MANAGER
Ginnie Boudrieau
1-800-343-0728
DIRECTOR OF ADVERTISING
Stephen Twombly
603-924-7138
Sales: Betty Butler,
William J. Smith
Ad Coordinator: Mary Hartwell
PUBLIC RELATIONS
James Leonard

80 Micro (ISSN -0744-7868) is published 12 times a year by The Wayne Green Publications Group, 80 Pine St., Peterborough, NH 03458. Phone: 603-924-9471. Second class postage paid at Peterborough, NH, and additional mailing offices. Subscription rates in U.S. are $36 for one year, $55 for two years, and $75 for three years. In Canada and Mexico $45—one year only, U.S. funds drawn on a U.S. bank. Nationally distributed by International Circulation Distributors. Foreign subscriptions (surface mail), $55—one year only, U.S. funds drawn on a U.S. bank. Foreign subscriptions (air mail), please inquire. In South Africa contact *80 Micro*, P.O. Box 782815, Sandton, South Africa 2146. All subscription correspondence should be addressed to *80 Micro*, Subscription Department, P.O. Box 981, Farmingdale, NY 11737. Please include your address label with any correspondence. Postmaster: Send form -3579 to *80 Micro*, Subscription Services, P.O. Box 981, Farmingdale, NY 11737.

80 Micro is a member of the CW Communications/Inc. group, the world's largest publisher of computer-related information. The group publishes 42 computer publications in 18 major countries. Nine million people read one or more of the group's publications each month. Members of the publication group include: Australia: *Australasian Computerworld, Micro Magazine*; Argentina: *Computerworld/Argentina*; Brazil: *DataNews, MicroMundo*; Denmark: *Computerworld/Danmark, MikroData*; France: *Le Monde Informatique*; Germany: *ComputerWoche, MicroComputerWelt, PC Welt*; Italy: *Computerworld Italia*; Japan: *Computerworld Japan*; Mexico: *Computerworld/Mexico*; Norway: *Computerworld Norge, MikroData*; People's Republic of China: *China Computerworld*; Saudi Arabia: *Saudi Computerworld*; Spain: *Computerworld/Espana, MicroSistemas*; Sweden: *ComputerSweden, MikroDatorn, Min Hemdator*; United Kingdom: *Computer Management, Computer Business Europe*; United States: *Computerworld, HOT CoCo, InCider, InfoWorld, Micro Market World, Microcomputing, PC World, 80 Micro, RUN.*

Problems with Subscriptions: Send a description of the problem and your current and/or most recent address to: *80 Micro*, Subscription Department, P.O. Box 981, Farmingdale, NY 11737.
Problems with Load 80 Circulation: Address correspondence to Lori Eaton, 80 Pine St., Peterborough, NH 03458.
Problems with Advertisers: Send a description of the problem and your current address to: *80 Micro*, Rt. 101 & Elm Street, Peterborough, NH 03458, ATTN.: Rita B. Rivard, Customer Service Manager. If urgent, call 1-800-441-4403.
Change of Address: Send old label or copy of old address and new address to: *80 Micro*, P.O. Box 981, Farmingdale, NY 11737. Please give eight weeks advance notice.
Microfilm: This publication is available in microform from University Microfilms International. United States address: 300 North Zeeb Road, Dept. P.R., Ann Arbor, MI 48106. Foreign address: 18 Bedford Row, Dept. P.R., London, WC1R4EJ, England.
Dealers: Contact Ginnie Boudrieau, Retail Sales Manager, *80 Micro*, Pine St., Peterborough, NH 03458. (800) 343-0728.



page 68

page 186

page 50

On the Cover

Features

Reviews

C•Notes / *for the Model 100*

Departments

page 200

EDITORIAL DIRECTOR
WAYNE GREEN PUBLICATIONS
Jeffrey D. DeTray

EDITOR-IN-CHIEF
Eric Maloney
MANAGING EDITOR (EDITORIAL)
Peter E. McKie
MANAGING EDITOR (PRODUCTION)
Deborah M. Sargent
NEWS EDITOR
Eric Grevstad
REVIEW EDITOR
Lynne M. Nadeau
NEW PRODUCTS EDITOR
S.F. Tomajczyk
ASSISTANT EDITORS
Amy Campbell
Steven Casey
Susan Gubernat
Robert L. Mitchell
TECHNICAL EDITORS
Bradford N. Dixon
Amee Eisenberg (Load 80)
Mare-Anne Jarvela
Beverly Woodbury
EDITORIAL DESIGN MANAGER
Susan Gross
EDITORIAL DESIGNER
Philip Geraci
LAYOUT EDITORS
Joan Ahern, Maurelle Godoy,
Susan Hays, Laura Landy,
Judy Oliver, Phyllis Pittet
PROOFREADERS
Peter Bjornsen
Harold Bjornsen
Robin Florence
EDITORIAL ADMINISTRATION
Carole Macioci

PRODUCTION
Director: Nancy Salmon
Lahri Bond, Michael Ford, Marjorie Gillies,
Anne Rocchio, Lynne Simonson,
Kenneth Sutcliffe;
Film Production: Donna Hartwell,
Theresa Verville, Robert M. Villeneuve;
Ad Coordinators: Patricia Bradley, Paula Ramsey;
Assistant: Jean Southworth
Advertising Production: Jane Preston,
Fiona Davies, Bruce Hedin, Scott Philbrick
PHOTOGRAPHY
Supervisor: Thomas Villeneuve;
Sandra Dukette, Nathaniel Haynes,
Laurie Jennison, Sturdy Thomas
TYPESETTING
Supervisor: Sara Bedell;
Darlene Bailey, Marie Barker, Prem Gongaju,
Lynn Haines, Cynthia Letourneau,
Kimberly Nadeau, Debbie Nutting,
Lindy Palmisano, Heidi Thomas, Sue Weller
DESIGN
Creative Director: Christine Destrempes
Manager: Joyce Pillarella
Design Consultant: Beth Krommes;
Assistant: Sarah Werninger
Administrative Assistants: Susan Donohoe,
Patrice Scribner
Chief Copywriter: Steve Tripp
Copywriters: Louis Marini, Gail Morrison,
Dale Tietjen

Cover by Erick Ingraham

The left bracket, [, replaces the up arrow used by Radio Shack to indicate exponentiation on our printouts. When entering programs published in *80 Micro*, you should make this change.

80 formats its program listings to run 64-characters wide, the way they look on your video screen. This accounts for the occasional wrap-around you will notice in our program listings. Don't let it throw you, particularly when entering assembly listings.

Article submissions from our readers are welcomed and encouraged. Inquiries should be addressed to: Submissions Editor, 80 Pine Street, Peterborough, NH 03458. Include an SASE for a copy of "How to Write for *80 Micro*." Payment for accepted articles is made at a rate of approximately $50 per printed page; all rights are purchased. Authors of reviews should contact the Review Editor, 80 Pine Street, Peterborough, NH 03458.

You can count on 3M diskettes. Day after day.

Just like the sun, you can rely on 3M diskettes every day. At 3M, reliability is built into every diskette. We've been in the computer media business for over 30 years. And we've never settled in. We're constantly improving and perfecting our product line, from computer tape and data cartridges to floppy disks.

3M diskettes are made at 3M. That way, we have complete control over the entire manufacturing process. And you can have complete confidence in the reliability of every 3M diskette you buy.

Look in the Yellow Pages under Computer Supplies and Parts for the 3M distributor nearest you. In Canada, write 3M Canada, Inc., London, Ontario. If it's worth remembering, it's worth 3M diskettes.

3M hears you...

3M

Radio Shack Presents
THE NEW TRANSPORTABLE

The Computer for Today's Upwardly Mobile Society

If you're headed for success, our newest TRS-80 can help you get there in record time. Model 4P is a compact, disk-based desktop computer with a big difference: it has a handle. So wherever your work may take you, you can get a handle on those tricky scheduling problems, ever-changing sales projections, and last minute reports. It's like taking your office with you.

Easy to Use, Easy to Tote

The Model 4P works anywhere there's AC. It has the power and versatility to function as your office desktop computer. Or take the Model 4 to a client's office to calculate budget projections. Present graphic information in a conference room. Finish up a contract proposal at home, or electronically send back the latest figures to the home office—from your hotel room! With our wide array of expansion options, the number of uses is limitless. And when you're ready to move on, slip the keyboard into the sturdy plastic case, snap on the protective cover and go. The Model 4P weighs just 26 pounds, and it's small enough to stow in overhead luggage racks on planes, trains or buses.

The Completely Self-Contained System

We don't sacrifice features for portability, either. The Model 4P comes standard with two 184K disk drives and 64K internal memory. You also get a full 80-character by 24-line 9″ display and an electric typewriter-quality keyboard with numeric keypad. In addition, both printer and communications interfaces are included.

Expand as Your Needs Grow

The Model 4P can be expanded as your work demands. Because it's also a powerful desktop computer, the Model 4P has a large data storage capability—add up to four hard disk drives for an extra 20 megabytes of storage! You can also increase the 64K internal memory to 128K. And with the 640 x 240 pixel high-resolution graphics upgrade (26-1126, $249.95), you can create sophisticated business graphs, tables, charts and more.

Stay in Touch—Anywhere, Anytime

Install Model 4P's internal modem board (26-1084, $149.95) and you can access the huge data bases of computerized information services by phone. The latest information of all types—sports, weather, general and financial news—is only a few keystrokes away. Read movie reviews, shop and bank from your home, or do research with an electronic encyclopedia. In addition, you can communicate with your office mainframe so you can access vital business information whenever you want.

Choose from an Extensive Library of Ready-to-Run Software

Simply add ready-to-run programs for word processing, accounting, spreadsheet analysis, data base management and many other professional applications. In addition to running the entire selection of TRS-80 Model III/4 disk software, you can add the optional CP/M Plus™ operating system (26-2216, $149.95) to use thousands of additional ready-to-run programs. Because Model 4P is compatible with so much software, you're sure to find programs ideally suited for you. So you won't have to pay for software features you never use, or for software that doesn't provide all the features you need.

The Transportable Computer For You

Compare the TRS-80 Model 4P with other transportables and see why you get more for your dollar. The Model 4P's standard features—large disk storage, full-size screen, typewriter-quality keyboard, and huge software base—give you a complete workstation to increase your productivity both in the office and out.

TRS-80® MODEL 4P COMPUTER

Like All TRS-80's We Stand Behind The "4P" Computer 100%

Tandy Corporation is a solid two-and-one-half billion dollar company with over 6,600 stores nationwide—and still growing. In addition to a wide selection of software and a complete line of accessories, we offer businesses a 37-month lease for computer systems valued over $1500. And when you buy a TRS-80, you're assured of after-the-sale support. We offer "Carry-In" and "On-Site" service at over 300 company-operated repair centers nationwide. And with just a phone call, our Computer Customer Services Department will answer any TRS-80 related question you might have. In addition, we publish a monthly TRS-80 magazine that's packed with operating tips, product news and more.

Learn All About Model 4P Today

Stop by your nearest Radio Shack Computer Center, participating store or dealer and find out where Model 4P will take you. And learn more about the leasing, training, service and support that every TRS-80 owner can depend upon.

Radio Shack®

The biggest name in little computers®

A DIVISION OF TANDY CORPORATION

500

1799.00

26-1080

Commercial Lease Available For Only $65 Per Month (Plus Applicable Use/Sales Tax)

Prices apply at participating Radio Shack stores and dealers. CP/M Plus is a trademark of Digital Research, Inc.

Send for your free TRS-80 computer Catalog today.
Mail To: Radio Dhack, Dept. 84-A-904
300 One Tandy Center, Fort Worth, Texas 76102

NAME
COMPANY
ADDRESS
CITY
STATE
ZIP
TELEPHONE

Tandy's Model 2000: No Cloning Around

The gold dust hadn't settled at Comdex in Las Vegas last November before TRS-80 users were engaged in a vigorous debate: Shouldn't the Model 2000 have been IBM PC compatible?

Tandy bills the Model 2000 as an "ultra-high performance MS-DOS system." In other words, it will run software that conforms to MS-DOS conventions. It will not run software written for the IBM PC.

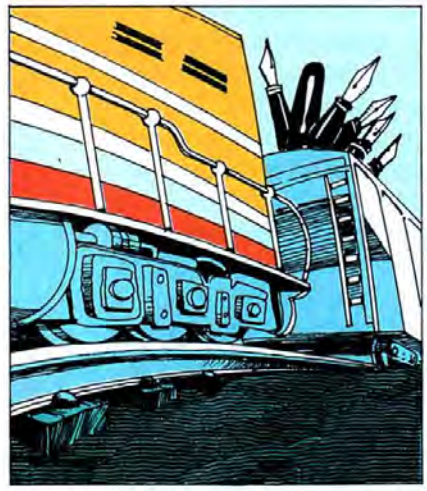
Of some 100 IBM PC programs tested by Tandy on the 2000, about half ran. The other half presumably used the IBM PC hardware, and gave the 2000 indigestion. This second group included such heavyweights as Lotus 1-2-3, the Easy series from Info Sys Unlimited, the Star series from MicroPro, and the Visi series from VisiCorp.

Tandy has often been taken to task for its sins of omission, and rightfully so. They introduced the Model 16 before a decent operating system was available. They announced CP/M for the Model 4 and then didn't deliver.

But this time, Fort Worth made the right decision. And make no mistake about it: the decision was deliberate, one that Director of Merchandising Ed Juge has called "religious."

Anybody can make an IBM clone. In fact, just about anybody has. What Tandy delivered was a high-powered 16-bit machine that takes advantage of MS-DOS's popularity, but that can potentially give MS-DOS a new dimension. Its speed, graphics, memory, and ease of use should make for a computer that will surpass the IBM PC in performance. And ultimately, making a better product is what business is all about.

Don't worry about software. The PFS series, Word, MultiMate, dBase II, and Multiplan are proven products. The Ovation integrated software package Tandy will offer has tremendous potential. And if the 2000 enjoys any success whatsoever, other MS-DOS manufacturers will make the necessary



modifications to their programs.

The Model 2000 has yet to be sufficiently road-tested for final judgments to be made. We're running it through its paces, and will be sharing our results with you in the next few months.

But whether it lives up to its press clippings or turns out to be a disappointment, the 2000 should be evaluated on its merits, not according to how IBM-compatible it is. To do so would be to give IBM final sanction as the only standard in micros, a prospect both distasteful and disdainful.

Comments on Comdex

Some people have noted the Tandy TRS-80 Model 2000's name: the Radio Shack logo has been dropped. Obviously, Tandy has come to the realization that many businesses don't like to have a piece of equipment in the office that comes from the Shack. But there is a precedent: Fort Worth sells through their OEM program under the Tandy logo....

The 80186 has emerged as a chip to be reckoned with, but at Comdex it appeared mostly in multi-user systems such as the Mad 1, the North Star Dimension, the Onyx, and the DBS 16. ACT-North America President Christopher J. Buckham, when asked why his company's Apricot used the 8086 instead, commented, "80186 computers are not mass-market machines." Indeed, word has it that Intel may not be able to produce the chip in sufficient quantity, which might mean that some Model 2000 buyers will have to wait in line....

What would you get if you crossed a Model 2000 with a Model 100? The Magnum, from Dulmont Electronic Systems in Australia. It's an 8-pound, 80186 lap computer that would make the average Model 100 owner drool with envy. It comes with a 16-line by 80-column flip-up screen, 128K of ROM, up to 256K of CMOS RAM, up to 256K in ROM packs, MS-DOS 2.0, and Basic 86. Its ROM-based software includes a word processor, a spreadsheet, and a planner/diary. Dulmont hopes to sell the machine in the U.S. for about $2,000. It remains to be seen whether they can successfully market the computer here....

And Finally...

In the wake of the Model 2000 emerges an interesting question: What next? Tandy has several options, including a 2000 at-home model (similar to the PCjr) or a 2000 transportable. But first, they're going to have to address the growing competition for the Model 100. Tandy is still ahead in the portable market by virtue of being first with the most, but being a leader in this business is tenuous, as Tandy itself proved with the Model I. A revamped Model 100 is a good bet, sometime in mid-1984.

We've got several feature articles on the 2000 in the works for the next few issues. We will, of course, give the machine a complete review. We'll also bring you a pair of in-depth articles on MS-DOS and GW-Basic, and we'll give you the lowdown on just exactly what MS-DOS and IBM software the 2000 will and won't run. In the future, we'll take a closer look at the Intel 80186 and explore the impact the 2000 will have on the rest of the Tandy line. ■

The Answer is...
NEWSCRIPT!™

THE WORD PROCESSOR FOR BUSINESSMEN AND PROFESSIONALS

With ongoing support directly from us

A FEW OF NEWSCRIPT's 200 STANDARD FEATURES:

- FORM LETTERS WITH MERGING OF NAMES AND ADDRESSES
- GIVES SUPERB APPEARANCE TO YOUR FINAL DOCUMENTS
- COMPREHENSIVE MANUAL WITH TUTORIAL AND EXAMPLES
- CENTERING, TOP/BOTTOM TITLES, INDENTS, PAGINATION
- UNDERLINING, BOLDFACE, DOUBLE-WIDTH, ITALICS†
- SUB/SUPER SCRIPTS, RIGHT-JUSTIFIED PROPORTIONAL†
- CREATES TABLE OF CONTENTS, SORTED INDEX
- "LEGAL" LINE NUMBERING
- SCREEN GRAPHICS, SPECIAL PRINTER SYMBOLS†
- SEARCH/REPLACE GLOBALLY OR WITHIN LINES, COLUMNS
- BLOCK MOVE, COPY, DELETE, INSERT. FILE MERGES
- AUTOSAVE, WHOOPS, DIRECTORY, KILL
- SUPPORT FOR ALL LISTED PRINTERS IS INCLUDED ** (NO PATCHES INVOLVED) **
- SUPPLIED READY-TO-RUN ON "TINY" DOSPLUS
- ALSO RUNS UNDER NEWDOS/80, LDOS, MULTIDOS, TRSDOS

NEWSCRIPT 7.1:	**$124.95**
Mailing Labels Option:	**29.95**
Special: NEWSCRIPT + LABELS:	**139.95**
Daisywheel Proportional Option:	**49.95**
"Pencil"/"Scripsit" File Convertor:	**24.95**
NEWSCRIPT Manual & Reference card only:	**29.95**
Electric Webster + Correction Feature:	**149.50**
Hyphenation Feature for Electric Webster:	**49.95**
Grammatical Feature for Electric Webster:	**49.95**
Dotwriter 3.0:	**79.95**
Dotwriter + Letter Utilities:	**99.95**
6.0 PLUS-Micro Systems Model 4 utility pak	**49.95**

REQUIRED CONFIGURATION:

48K TRS-80, MAX-80, LNW, or compatible, with one or more disk drives. Specify Model I or Model III.

† some features work only if your printer has the mechanical capability.

** Daisy Wheel Proportional is an extra-cost option.

BUILT-IN SUPPORT FOR MOST POPULAR PRINTERS†, INCLUDING:

Anadex, Brother, Centronics. C.Itoh, Diablo, Epson, Gemini, Microline, NEC, Prowriter, Qume, Radio Shack (LP 1-8, DW2, DMP-410, DWP 200-2100), Smith Corona, Teletype, Typewriter, anything compatible with any of these, and many others, parallel and RS-232.

SPECIAL AVAILABLE OPTION: Right-justified proportional for Diablo, F-10, Qume, Spinwriter, etc. Requires "Daisywheel Proportional" Option plus NEWSCRIPT.

REVIEWERS AND USERS AGREE:

"NEWSCRIPT" is the best word procesor I have seen . . . unsurpassed in printer control . . . no other TRS-80 word processor can match its ability to format text . . . its editor is fast, easy, and powerful." *(80 MICRO, Oct. 1982)*

"Your phone information system and the prompt and courteous staff that you provide to help your clients . . . are worth the cost of the system." *(V.H.H.)*

"Better than cold beer on a hot day!! Thank you!!" *(R.S.)*

"What a program. So easy to learn and easier to use. I waited too long before ordering!" *(P.J.M.)*

". . . takes the TRS-80 to a new level of text handling . . . very user-friendly . . . superb documentation, adaptability to many printers and operating systems . . . a standard against which other TRS-80 word processing programs will be judged." *(SOFTSIDE, Dec. 1982)*

". . . ongoing support second to none, with superb documentation." *(80 U.S. Journal, Feb. 1982)*

TO ORDER, CALL NOW, TOLL-FREE: (800) 824-7888, Operator 422

For orders, information, or names of nearby dealers: (213) 764-3131, or write to us.

Order from your Software dealer or from:

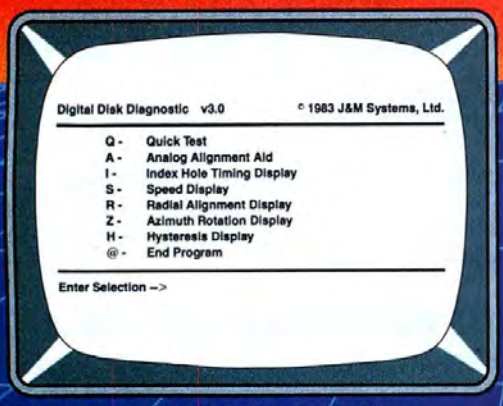
PROSOFT® ✓98

Dep't. C, Box 560 No. Hollywood, CA 91603

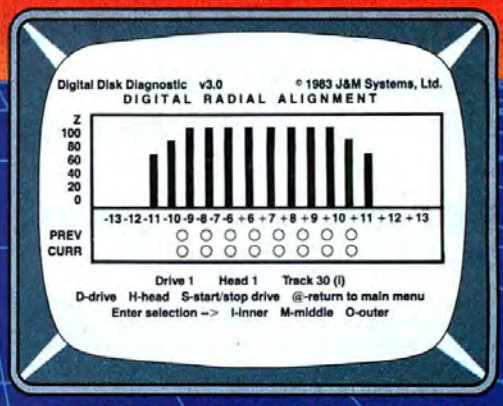
TERMS: VISA, Mastercard, checks, money orders, COD. No P.O.'s accepted. Most orders shipped within 24 hours. Please add $3.00 for surface UPS in U.S.A., or $6.00 for UPS Blue Label. Add $6.00 in Canada, $15.00 overseas air shipment, 6½% sales tax in California.

DISK DRIVE ANALYSIS PROGRAM

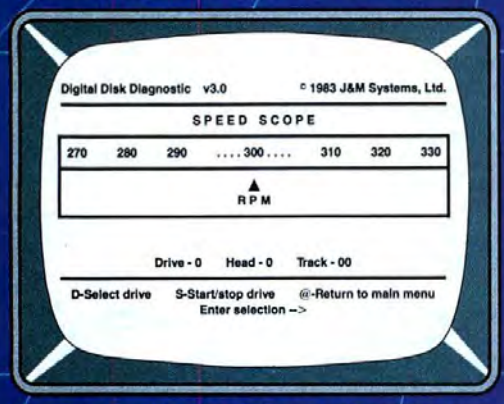
. . . A UNIQUE APPROACH TO DISK RELIABILITY!



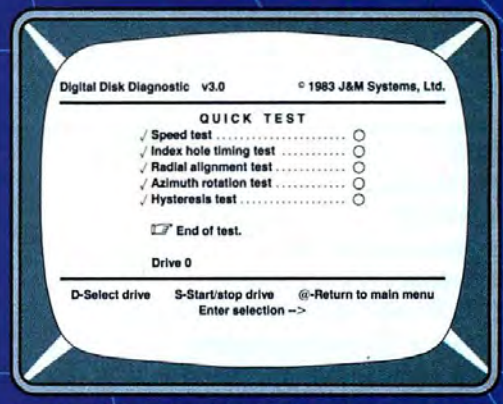
Select any one of seven tests to perform preventive maintenance or to isolate problems. Simple, single-letter commands make DDA easy to use! Use DDA to align the head, adjust the index hole detector, or adjust the speed.



Use the DDA Radial Alignment Test to check the head alignment of your drives. No need for an oscilloscope or other expensive test equipment!



Check the motor speed of your drives. Or, you can even use the Speed Test to adjust the drive speed. No need for any test equipment!



Use the Quick Test to quickly and automatically test five of the most important performance parameters of your drive. Monitor your drives for long term drift. Isolate problems quickly and automatically!

PROTECT YOUR DATA.

Now you can make sure your data is being recorded properly by the use of the revolutionary *Disk Drive Analysis Program (DDA)!*

The *Disk Drive Analysis Program* from J & M Systems, tests your disk's performance and calibration without any additional equipment! It measures your disk's performance and displays it on your screen.

This is the most comprehensive disk diagnostic program available for your TRS-80 microcomputer. You can even adjust drive alignment while watching the display!

Spot problems BEFORE they endanger your data! If you own a disk drive, you NEED the *Disk Drive Analysis Program!*

Now Available For The TRS-80 Microcomputer.

J & M SYSTEMS IS THE DRIVING FORCE!

J & M SYSTEMS, LTD. ⌐101

137 UTAH NE · ALBUQUERQUE, N.M. 87108 505/265-1501

Scientific Discovery

Congratulations on your October science issue. Karl Sarnow's "Molecular Matters" (p. 100) was particularly worthwhile—it is unusual to see this caliber paper published in a general-audience magazine. Keep up the good work!

Thomas T.S. Huang
Department of Chemistry
East Tennessee State University
Johnson City, TN

Caveat Emptor

A lot of TRS-80 software advertised as usable on the Models I, III, and 4 often turns out to be a Model III version.

I know that a Model III version of a program runs in Model III mode on my Model 4, but when I order a Model 4 version of a program I expect it to fully utilize the features of my Model 4.

I will not buy a Model III program if the Model 4 version is available. This borders on false advertising, and if the trend continues, Model 4 software buyers will not know what they are getting.

Marvin E. Decker
San Antonio, TX

DOSPLUS 3.5

While I enjoyed John B. Harrell's review of DOSPLUS 3.5 (October 1983, p. 160), he failed to mention a few important features that DOSPLUS 3.5 lacks.

DOSPLUS 3.5 does not contain the 8-inch disk driver included in version 3.4. Also, the manual's example for setting aside memory is BASIC −M61000. I have a program that calls Basic as follows: BASIC −M53248. I then run a user program that zeros out

memory from this address to FFFF hexadecimal. The program displays the MEMORY SIZE message but doesn't return to my original program. DOSPLUS 3.4 runs the same program without failure.

Finally, DOSPLUS 3.5 contains various patches on disk but gives no information on how to install or use them.

John F. Reedich Jr.
Port Orange, FL

Pillow Piracy

Eric Maloney's comments about whether *80 Micro*'s programs are in the public domain (Side Tracks, October 1983, p. 6) were undoubtedly correct, but are removed from reality. I am reminded of those tags on cushions and mattresses that say "Do not remove under penalty of law."

Let's examine the implications of Maloney's statement that only the subscriber/purchaser can use programs published in *80 Micro*. The license accompanies title to the magazine, so when I give my old magazines to a computer club, I must destroy my programs that came from those magazines.

Who, among the members of the club, may then use the programs? An institution holds the title to the magazine. Can just one person use each program, or can they all use it?

Did you warn your librarian that her borrowers must never use any of the programs in *80 Micro*? Does the FBI get into this, or does interstate transport of pirated programs have to occur first?

If I use only one program, may I tear it out and give the rest of the magazine to a friend? If I throw an old issue away and a computer freak salvages the magazine and makes second use of it, who is liable for copyright violation? Should I run my old magazines through a paper shredder?

It is foolish to threaten enforcement of unenforceable laws. I realized that truth the first time I tore the tag off my pillow.

Edward M. Roberts
Glen Head, NY

Seal Me

Are any of your staff old enough to remember the *Good Housekeeping* Seal of Approval?

80 Alert serves a useful purpose, but have you considered the clout you could wield with your advertisers? Readers could be reasonably assured that a product works, and advertisers would have the integrity of *80 Micro* behind them.

A. Ray Crawley
Ooltwah, TN

At this point, there are simply too many products on the market for us to do the kind of thorough critique you suggest. The responsibility for policing the industry rests with the industry itself.—Eds.

INPUT

Pascal Blues

Where, O where is my Pascal? I like *80 Micro*, and I know that Basic is popular, but Pascal is a more structured language. Why don't you try to start a column dedicated to Pascal, or better yet, a Pascal magazine?

Rob Williamson
Hacienda Heights, CA

Look for our new Pascal column in May.—Eds.

Dismal Dept.

I am watching—with a tear in my eye and memories of the past—as *80 Micro* slowly but surely changes into a hardware magazine.

Only a few months ago your magazine was filled with glorious full-page color ads from major software manufacturers and dealers. Where are they now? What is causing this software slowdown?

Obviously software piracy and slow sales have made it impossible for sellers to afford the current advertising rates. They simply can't afford to advertise and not even break even on the cost of their ad.

I cannot offer a solution, although lower ad rates and new products would help. In order to survive, my Model III needs love and support from software authors and manufacturers.

Bob Krotts
Kettering, OH

Name Calling

Why don't you change the name of your magazine to *TRS-80 Models I, III, and 4*?

Curtis L. King
Knoxville, TN

It's too hard to type.—Eds.

Lost in Space

We have a long way to go in personal programming if Brian Durell has to sound apologetic over his style ("Brief Exposures," October 1983, pp. 236–241). With his spacing, indentations, and plentiful remarks, he has produced one of the most intelligible listings I've ever seen in a microcomputer publication.

Sure, it takes more memory and print space, but the increased value to readers makes it worthwhile.

Victor G. Feser
Bismarck, ND

Unfortunately, such listings are longer, and take space from other articles. We opted for more articles.—Eds.

Tough Guys

You should be tougher in your reviews of computer products. More effective use of charts and graphs and other aids for comparison would be of vital interest to your readers.

Bernard S. Korotkin
Bowling Green, KY

You'll be seeing more visual aids in the near future.—Eds.

80 ALERT

Occasionally, 80 Micro *receives letters from readers who have had difficulties with our advertisers. Most of the time, these problems are resolved to the satisfaction of all parties, but some problems appear to be insoluble.*

As a service to readers and advertisers alike, 80 Alert will pinpoint distributors who cannot be reached, by readers or by our advertising department, for customer service. Anyone who has current information about a manufacturer or distributor mentioned in the column is welcome to write and update our data.

Victor Andrews, president of Software Central (P.O. Box 247, Westland, MI 48185), has informed *80 Micro* that Software Central went out of business as of Oct. 31, 1983. Andrews will handle all correspondence mailed to the above address.

80 Micro has received a number of complaints regarding advertisements for *Briefcase Portable* magazine (560 South Hartz Ave., Suite 447, Danville, CA 94526). David Gourley has told *80 Micro* he was experiencing trouble producing the magazine, but that he expected the first issue to be out by Dec. 6, 1983. Readers with questions should call Gourley at 415-820-8149. Due to the number of complaints received, we have cancelled advertisements for *Briefcase Portable*.

Meta Technologies Corp. (26111 Brush Ave., Euclid, OH 44132) has gone out of business and will no longer market AIDS-III. Softrends Inc. (26111 Brush Ave., Euclid, OH 44132) says that it will market an upgraded version of the database management program in the near future.

80 Micro has received complaints from readers that Omikron Systems (1127 Hearst St., Berkeley, CA 94702) has not filled orders or issued refunds. Omikron informed *80 Micro* that it was issuing refunds and expected to be shipping within a few weeks. Customers who contacted *80 Micro* had received refund checks by press time.

Computex/World Wide Data Systems has filed for bankruptcy. Customers seeking refunds should contact Rod Hardie at Huges, Watters, Askanase & Redford, Attorneys at Law, Suite 2153, 1100 Milam Building, Houston, TX 77002. Customers in need of technical assistance should contact Greg Taylor, Medcomp Data Systems of Texas, 15502 Old Galveston Road, Suite 112, Webster, TX 77598.

"THE RESULTS ARE IMPRESSIVE..."
—Dennis Kitsz, 80 Microcomputing: 12/82

Langley-St. Clair's* *Soft·View*™

Replacement CRT's

eliminates the strobe, flicker and fatigue from TRS-80's.™

Now you can upgrade your monitor with the new medium persistence green or amber phosphor tube.

State-of-the-art systems such as IBM™ and Apple III™ do not use the less costly "P4" B&W display tube because it is actually intended for TV viewing and its rapid strobes (60 times per second) cause irritating eye fatigue.

No amount of "green plastic" will solve this problem. But the new ***Soft·View*** CRT display tube from Langley-St. Clair will.

- Available in slow decay Green or medium decay "European Amber" (the standard in Europe)
- Made with Lead/Strontium impregnated glass that stops X-ray emission.
- Of high-contrast face glass that also stops most U.V. radiation.
- Available in frosted glass with extra Anti-Glare benefits.
- Easily installed...comes with pre-mounted hardware.
- Warranted for one full year against manufacturing defects or tube failure.
- The finest quality double-dark glass phosphor fields to produce dramatic contrast.
- Ideal for Word Processing and Programming, yet fast enough for Games and Graphics.

LSIS *Soft·View*™ CRT'S

☐ #GN42 Green Phosphor	$79.95
☐ #GN42G Green Phosphor w/Anti-Glare	$89.95
☐ #OR34 Amber Phosphor	$89.95
☐ #OR34G Amber Phosphor w/Anti-Glare	$99.95
also available:	
☐ #R22G Red Phosphor w/Anti-Glare	$139.95
☐ #B22G Blue Phosphor w/Anti-Glare	$139.95

Plus: $7.00 for packing and UPS Shipping
$17.00 for Overseas, Parcel Post or UPS Blue Label
Add Sales Tax where applicable.
(Inquire about the CRT's we have available for many other computer models)

For MasterCard and Visa Orders only, call
800/221-7070 (in N.Y. call 212/989-6876)

Langley-St. Clair Instrumentation Systems, Inc.
132 West 24th St., New York, N.Y. 10011

Actual unretouched photo.

462

*World's largest supplier of upgraded replacement CRT's.

Quality you expect, at a price you don't.

BECK DOUBLE DENSITY DISKETTES

SINGLE SIDED **$2.19** ea. / **$2.79** ea. DOUBLE SIDED

Our message to you is simple. If you like the quality of Dysan, Verbatim, 3M, et al, you'll like the quality of Beck soft sector, 5¼" flexible diskettes. The only major difference is cost. We're less expensive. In fact, a lot less expensive.

Why does Beck cost less?

Our philosophy is: Excellent quality and reliability, at a cost that beats the jackets off other diskettes. We can do it because we (1) put our money into the product, not mega-marketing schemes and fancy packaging; and (2) sell our money-saving 25-diskette pack to you direct via a toll free order line, so you get fast, door-to-door service efficiently.

When you buy Beck, you've got the best. Beck Quality. Beck Reliability. And, of course, Beck Price.

1D, soft sector 5¼" diskette $2.19 each
2D, soft sector 5¼" diskette $2.79 each

For IBM, Apple, TRS and 97% of popular microcomputers.

What about quality and reliability?

At Beck, our success as a diskette manufacturer depends upon our ability to provide you with a fully reliable, quality diskette – every time. For that reason we take no shortcuts. You get the best because we are committed to excellence. Every diskette is manufactured to very strict quality standards. We test and retest 21 times throughout the manufacturing process to insure compliance with no less than 42 rigid specifications. We make sure you get the very best – a 100% certified, 100% error free diskette.

Our satisfaction money-back guarantee and full 7 year warranty are proof of our commitment to excellence and confidence in our product.

271 (in New Hampshire call 924-3821) **Door to Door in 48 hrs.**

Order Now Toll Free

VISA MasterCard COD'S CASH ONLY

Corporate Accounts Welcome

1-800-BECK-MFG

Order Toll Free 1-800-232-5634. Available in 25 pack only, plus freight, Complete with hub reinforcing rings, Tyvek envelopes, color coded user labels, and nonmetallic write protect tabs. All Beck Diskettes meet or exceed ANSI specifications.

FEEDBACK LOOP / by Terry Kepner

Send any questions or problems dealing with any area of TRS-80 microcomputing to Feedback Loop, 80 Micro, *80 Pine St., Peterborough, NH 03458.*

Q: I have a Model I with Percom TFD40 and TFD100 drives. Whenever I attempt to back up a disk from the TFD40 to the TFD100 I get a DOS Error 4 (CRC Error During Read) on the TFD40. This also occurs when I try to transfer a file from the TFD40 to the TFD100, though not all the time. Whenever this happens I have to use a disk zap utility to correct the parity in order to make the disk usable again.

This error occurs no matter which drive I use as drive zero. In all cases I have the terminator resistor pack in drive zero. This error occurs in single-density with a Percom data separator installed and in both single- and double-density with a Percom Doubler II installed. It occurs with TRSDOS, NEWDOS Plus, and DOSPLUS 3.4D. It occurs no matter what brand disk I use, including new Verbatim double-density disks. I'm using a Percom two-drive cable that places drive zero at the end of the cable.

The TFD40 works fine when used by itself. Backing up or copying from the TFD100 to the TFD40 works perfectly every time. A single-drive back-up also works fine. It seems the error only occurs when I use both drives at the same time, with the TFD100 writing and the TFD40 reading.

I've used Floppy Doctor on both drives and never had any errors on either drive, although occasionally, when checking the motor speed on the TFD40, the message alternates between "Speed within +−33%" and "Speed Too Slow."

Percom has looked at the TFD40

several times and seems to think that nothing is wrong with the drive. (Stephen Milliken, Randolph, MA)

A: My first thought is to check the stepping speed configuration of your drives. Using DOSPLUS, type CONFIG. The DOS then indicates how your drives are set up. Using the CONFIG command changes the stepping time to a higher number. It could be that the TFD40 is slower at stepping from track to track than your TFD100 and that's what's causing the problem.

Other than that, I don't know what to suggest. You've already done everything I would've tried. Good luck.

Q: I've purchased a few cassettes from Load 80, but I prefer to get disks since I have a two-disk system. However, I have trouble knowing what to do with programs that are in assembler or source-code format. I purchased the disk version of EDTASM from Radio Shack, but find it very confusing. I don't want to be a programmer, but I would like to know how to type in a program using EDTASM.

I have downloaded programs from CompuServe in machine language, and can, on rare occasions, get them to work, even when I change the extension to CMD. Is there a simple way to deal with these programs from Load 80 and CompuServe? Is there a back issue that describes how to develop a module that loads and runs a source-code program? The example in the EDTASM manual is confusing.

I purchased a book from Radio Shack about Assembly language, but it applies to the Model I. I have a Model III. I also don't want to have to read an entire book just to learn how to load and run programs from Load 80 and CompuServe. I can use the TRSDOS Load command, but I can't get it to run. Run doesn't work in TRSDOS. The source program may be loaded, but I can't access it.

Trying to run something from Basic gets "Direct Statement in File" errors, and running it from TRSDOS by appending a CMD extension gets a "Not a Program File" error. Help! (William Nicholson, Wheaton, IL)

A: First you need to establish a few ground rules. A machine-language program takes control of your computer when you execute it. If you call a machine-language program from Basic, Basic gives up all control and passes it to the machine-language program. If the machine-language program works properly, control returns to Basic when the machine-language program is finished. If something goes wrong, the result is a complete reset of the computer, leaving you back at first base.

There are many ways to integrate a machine-language program with a Basic program. The most common method is to make the machine-language program a partner to the Basic program. When you go to Basic, you

answer the Memory Size question with an address. This tells Basic that it can't use any of the memory above that address for any purpose at all, that it's reserved for your direct use only.

Once you're in Basic, you usually use the System command to load the machine-language program into memory (the location you protected earlier), and then return to Basic to load the Basic program (all this assumes you're working in cassette Basic only). The Basic program must contain two POKEs to tell the operating system that a machine-language module is in memory and where its start address is. Now when the operating system reaches the A=USR instruction in your Basic program, it will transfer control from Basic to your machine-language program, which hopefully will return to Basic when it's through.

Another common method uses only one cassette-load operation: The machine-language routine and the Basic program are combined. Several subsequent methods are used to make this combination. One is to make a series of data statements in the Basic program that contain all the instructions of the machine-language program. When the Basic program is executed, the machine-language program is POKEd into place, the operating system is informed of the machine-language program, and you proceed on your way. This method is generally used only with small machine-language programs of a hundred or so bytes in length.

Another way is to make a string variable in the program the carrier of the machine-language program. This is very limited since certain machine-language instructions are interpreted by the operating system as line or data delimiters in the Basic program, prematurely cutting the program short.

A third method uses numeric variable arrays to form strings of machine-language instructions. This is the most complex method, but wastes the least amount of space.

All these methods try to accomplish the same purpose: combining Basic and machine-language programs to speed up the operation of Basic by assigning certain time-consuming tasks to machine-language routines.

If you have a disk system, it becomes a bit easier to make the machine-language and Basic programs cooperate. Since you want to load a machine-language program into memory and then load in a Basic program, you don't want to turn over control of the computer to the machine-language program just yet. Instead, you want the DOS to load the machine-language program into memory, and return control to you. This is the purpose of the DOS Load command. It takes the machine-language file specified (usually ending with the extension CIM), puts it in memory, and then returns to the DOS. Now you can load Basic, set Memory Size to protect the machine-language program you just loaded, and load the Basic program.

When you run the Basic program, tell the operating system that a machine-language program is available by using the DEFUSR(*n*) command, where *n* is the number of the routine (DOS lets you have up to 10 independent machine-language routines in memory simultaneously, but not at the same location). Now when Basic reaches a line in your program that says A=USR(n), Basic transfers control to the appropriate machine-language routine. If all goes well, control comes back to Basic and your program continues to operate.

By definition, you can't load machine-language programs with Basic commands (Basic commands are for loading and saving Basic programs or data). Therefore, attempting to load a machine-language program results in Direct Statement in File errors (machine-language programs don't have line numbers, and Basic expects to see line numbers after every carriage return character [CHR$(13)]. It also expects to see a carriage return at least every 240 characters during program loading). I hope this at least partially explains some of the strange things you see in hybrid machine-language/Basic programs.

Most of the instructions in your Model I machine-language book apply to the Model III. The only differences involve the printer and RS-232 ports, and certain ROM locations. Most of the book concentrates on the Z80 CPU native to both computers.

When you want to use a machine-language program without Basic, you use a different set of rules. Usually, these programs come in one of two formats: source code or object code. You can't execute source-code files with either DOS or Basic. You can only load or write them with an editor/assembler. Object-code programs execute only with either the DOS or the System command. Object-code programs are indicated by the extension CMD; source-code files by SRC. Just adding the extension CMD to a program doesn't make it an object-code program, however.

As far as EDTASM and Load 80 machine-language files are concerned, the procedure is as follows: Load EDTASM, and use the L file name command to load the source-code file you want. Then use the A command to assemble source code into an object-code program, and write the finished program to tape or disk.

If you're using tape EDTASM, you won't have any problems loading the Load 80 SRC files into it. However, there's a problem with disk EDTASM: The disk Load 80 SRC files require a conversion to load properly. While some may load and assemble, very few work correctly. The April 1983 *80 Micro* Reload 80 (p. 404) has a program that provides this conversion. If you don't have access to this issue, you can send a TRSDOS-formatted disk to *80 Micro*, attention to Reload 80, and they'll put the program on it for you.

Learning machine-language programming isn't easy, it requires study and hard work. However, if you only want to type in programs and use them, you can easily accomplish that with EDTASM, and without having to read a book to do it. EDTASM uses line numbers to keep track of program logic while you're writing it, just like Basic. Unlike Basic, however, when you want to execute the program, EDTASM assembles the source program and converts the mnemonic codes into machine codes, removing the line numbers, and assigning the machine code to memory locations for execution. During this assembly, EDTASM checks the program for errors in syntax and structure. Any error is reported immediately.

Many of the programs listed in magazines use one of two formats. The first is the format you see when you type in a program to EDTASM: a line number column, a column for a labeling subroutine, a column for a machine-lan-

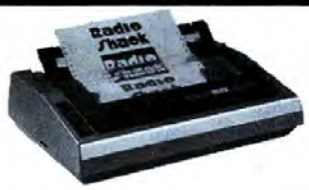
From Computer Plus to YOU . . .

PLUS *after* PLUS *after* PLUS

Model 100 8K $679
Model 100 24K $835

Color Computer II 16K $185
w/16K Ext. Basic $245

Model 4 16K $849
Model 4 64K
2 Disk & RS232 $1699



DMP120 $395
DMP200 $520

CoCo Drive 0 $329
CoCo Drive 1 $235

DWP210 $629

BIG SAVINGS ON A FULL COMPLEMENT OF RADIO SHACK COMPUTER PRODUCTS

COMPUTERS	
Color Computer	
64K ext. basic	$305
Model 4 Portable	
64K 2 Drives	1525
Pocket Computer 2	165
Pocket Computer 3	89
Pocket Computer 4	59
Model 12 64K 1 Drive	2699
Model 12 64K 2 Drive	3375
Model 16B 1Dr 256K	4249
Model 16B 2DR 256K	4915
MODEMS	
Hayes Smartmodem II	235
Hayes Smartmodem 1200	565
Novation Smartcat 1200	459
Novation J-Cat	125
R.S. AC-3	129
R.S. Modem I	89
R.S. Modem II	160

PRINTERS	
Silver Reed EXP500 D.W.	425
Silver Reed EXP550 D.W.	645
Daisy Wheel II	1745
DWP410	1159
DWP210	629
CGP115	159
CGP220 Ink Jet	545
DMP100	315
DMP420	735
DMP500	1219
DMP2100	1689
Gemini 10X	315
Delta-10	515
Gemini 15X	399
Prowriter	375
Prowriter II	649
Okidata	CALL
Epson	CALL
DISK DRIVES	
Model III/4 Drive 0	515
Model III/4 Drive 1	195

Model III/4 Drive 2	289
Model III/4 Drive 3	259
Primary Hard Disk M12	2689
Primary Hard Disk MIII	1799
ETC.	
CCR-81 Recorder	52
Cassette Tapes (10 Pk)	9.95
16K Ram Chips	25
64K Ram Chips	75
Model 4 64K Upgrade Kit	140
M100 Bar Code Reader	89
8K Printer Buffer	135
Printer Stand	25
Printer Cables	
Printer Ribbons	
Printer Paper	
Dust Covers	
Computer Books	

R.S. software 10% off list.
Send for complete listing
of brand name software.

CALL TOLL FREE
1-800-343-8124

- *LOWEST POSSIBLE PRICES*
- *BEST POSSIBLE WARRANTY*
- *KNOWLEDGEABLE SALES STAFF*
- *TIMELY DELIVERY*
- *SHOPPING CONVENIENCE*

VISA · MasterCard

computer plus

P.O. Box 1094
480 King Street
Littleton, MA 01460

SINCE 1973

IN MASSACHUSETTS CALL (617) 486-3193

18

TRS-80 is a registered trademark of Tandy Corp.

guage mnemonic, another column for an extended machine-language mnemonic, and then a column for comments. The second format is the same as the first, except that it is preceded by two additional columns, the first giving the starting address of the machine code generated for that line, the second giving the machine code generated by that line's mnemonic instructions.

The editor/assembler generates and prints second format during assembly of the source code to object code.

Magazines prefer that all machine-language programs be submitted in source-code format so that they can assemble the programs and see if they contain any errors.

CompuServe is another problem. Transmitting machine code over phone lines can lead to problems since CompuServe uses some machine-language codes as control-code instructions to your terminal program. To get around this, machine code is usually sent in hexadecimal-ASCII code. To use a CompuServe program requires that you convert it from this code to normal machine code. For more information about that I'll have to refer you to one of the Radio Shack SIGs on CompuServe. One of their experts can give you better instructions than I.

Q: I agree with J.B. Harrell ("Fortran Breakout," July 1983, p. 186) that Microsoft's Fortran 80 and Macro 80 make a powerful team. It's disappointing they haven't received more attention in the magazines. As a newcomer, however, I'm puzzled by the problem of how to create a relocatable subroutine library.

Mr. Harrell's USRLIB functioned perfectly when linked with Breakout, but when I attempted to use a few of Mr. Harrell's subroutines with a Fortran program of my own I found a surprising result. The L80 picked up all the subroutines, not just the undefined globals from the main program. With a small program this is clearly a minor problem, but with large programs the memory wasted could be significant. The manuals don't give a clue as to how to avoid this problem. (Roger Curran, Warrington, PA)

A: I seem to recall reading that the system was designed to operate in that manner, the idea being that the larger programs would use more of the subroutines, not wasting the space. And the waste wouldn't matter on small programs.

Of course, even I can think of some reasons to avoid integrating the whole library into the program, but I don't know enough about that package to actually help you. Does anyone else have a solution?

Q: I hope you can answer a couple of questions from a beginner. I have a Model III with two Percom dual-headed drives. I have tape Scripsit and put it on disk. My question is: Is there a patch or simple way to store and retrieve files onto disk? I really don't want to buy disk Scripsit when the version I have suits my needs.

My second question concerns programs in Assembly language. I have tape EDTASM, but can't seem to get any of *80 Micro*'s programs to run when I load them from Load 80. First I load EDTASM, then type L file name, A (for assemble), W (for write), go to SYSTEM mode, load target tape, and press the enter key when the asterisk prompt appears. Invariably, nothing happens. What am I doing wrong? (J.E. Guffey, St. Peters, MO)

A: Trying to alter tape Scripsit to work with disk storage is not a job for a beginner. I have heard of people who've made the transition, but it usually took a lot of time and work. I think it would be more cost-effective, in terms of your time, to buy the disk version.

You're also using the wrong procedure for assembling machine-language programs. First load EDTASM, then use the L file name command to load the source-code file into EDTASM. Then type A for assemble. EDTASM should prompt you to ready your tape for the object code. When you press the enter key, EDTASM writes the object code to tape. Go to Level II Basic and type SYSTEM. At the asterisk prompt, type the file name used for the object-code file and load the tape into memory. When the prompt reappears, type the slash (or type /### where ### is the execution address of the program) and press the enter key. The program should begin to execute.

EDTASM's W command writes a source code duplicate of the program in EDTASM's buffer, which can only be used by EDTASM for working on the program. The object-code file (A command) creates a file capable of being executed by the Z80 CPU of the Model III.

Q: For the last few months I've been using a second-hand Model I (48K with expansion interface and one disk drive) and the Radio Shack Line Printer VII for word processing.

As you probably know, the lowercase set for the LP VII has no true descenders, and requires special programming to produce graphics characters (making normal screen dumps of graphics characters very difficult). I'd also like to add at least one user-defined sign (an English pound sign), for use with Scripsit as a replacement for a rarely used symbol.

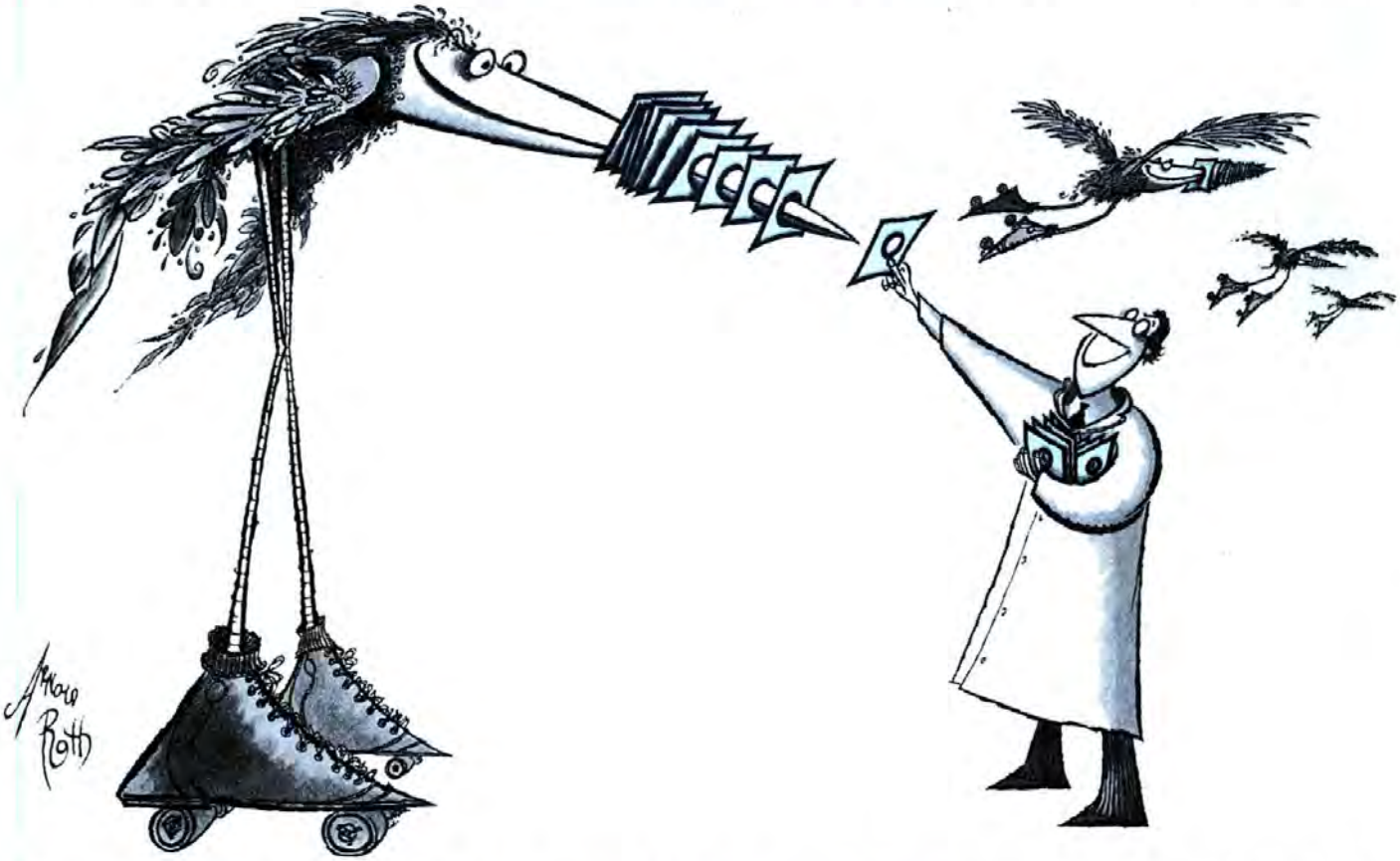
Do you know of a program that will overcome these problems by setting up a protected character set in the computer's memory, and then print all text in the graphics mode? A spooler routine would be useful, but not essential, and compatibility with TRSDOS 2.3 would be advantageous.

Alternatively, does anyone sell a hardware modification for the printer that will give a better character set? (Marcus Rowland, London, England)

A: What you want is no small task. Designing a new character set for the LP VII would take a good amount of time, and about 2K of RAM. Just the design, in bit graphics, of the new characters would take almost 1K. The driver software would probably take just as much room since it would have to intercept the Scripsit printer driver and substitute the appropriate bit-image character. Of course, such a driver would make possible things like underlining, boldfacing, super- and subscripting, and other special functions.

Replacing the character-generator chip in the LP VII would be possible if someone has a redesigned character set for it. Otherwise you will have to design and burn-in your own character set, using the appropriate hardware

Smallware™



Our software is making a name for itself.

Smallware. That's what we've named our unique software designed for microcomputers. Smallware offers much more than ordinary software: high quality, customer support and a complete product line. You can buy software anywhere. But for the special features of Smallware, The Small Computer Company is your one and only source.

The Small Computer Company is known to many as the company who developed the filing system software Profile® II, Profile Plus and Profile III Plus for Radio Shack; and filePro,™ our CP/M® version.

Now, whether you're a microcomputer end-user, dealer or manufacturer, you can order our Smallware directly from us.

Introducing two of the best utilities you'll ever find for Radio Shack computers: QUIKBACK™ WITH FORMAT and PROPACK™ for the Models II and III.

QUIKBACK WITH FORMAT, as its name implies, is a quick backup program. It whizzes you through a format and backup in less than 1½ minutes.

But it's even more useful as an error-checker. Ever found yourself with a diskette you suddenly can't read? Quikback lets you get around those nasty little error spots and recover your data. When we tested the program, we found that we could even punch holes in the diskette (we don't recommend *you* do this, however) and *still* recover most of the data.

Quikback with Format runs with any TRSDOS 2.0a diskettes, on the TRS-80 Models II and 12.

PROPACK lets BASIC programmers customize their Profile Plus and Profile III Plus systems. The programs you write with Propack are shorter, easier to write and faster-running. Propack also includes such benefits as indexed access to Profile data and BREAK confirmation. The manual even includes sample subroutines and functions to use in your own programs.

Propack is available for the Radio Shack Model II/12 or Model III. Please specify the version required when ordering.

To order, call (800) 847-4740 outside New York state. For further information, or to order in New York state, call (212) 398-9290.

The Small Computer Company, Inc.

230 West 41st Street, Suite 1200, New York, New York 10036

80 readers' choice micro 1982

245

tools. As yet, I don't know of any second-source character sets for the LP VII.

Q: I have a cassette-based Model III with a Daisy Wheel II printer. I remember reading, some months ago, about a program to use the printer as a typewriter, but I can't find the article.

Also, I'm considering upgrading to a disk system and I'm concerned about using a non-Radio Shack drive. I don't know anything about the inner workings of the computer, and living overseas (no computer repair center, and import taxes are a problem, even on things sent out for repair), I'm worried that if I do install a non-Radio Shack drive and it doesn't work, I'll be stuck without a computer. Any advice? (Monica Beukenkamp, San Jose, Costa Rica)

A: I can't find the article either, but it's a rather simple program. Just use INKEY$ to input characters into a character string, which prints when the number of characters reaches a preset line length.

The problem with using the printer as a typewriter is that the printer doesn't print one character at a time: It prints only when its character buffer is full, or when it receives a carriage return. So you can't use it to fill out forms because you can't get the printhead to space over before you start typing letters or numbers.

From what I've seen here as comments from readers with problems, the non-Radio Shack disk systems are as reliable as the Radio Shack systems. I haven't reviewed or used any of the upgrade kits myself, so I can't recommend any particular one. Even if you install a disk system and have problems, you can always disconnect the drives, ship them out, and continue to use your cassette system.

Q: I have a Model I and an Okidata 82A printer. I'm tired of printing on the perforations and would like to modify the Basic LPRINT and LLIST commands so I can get automatic perforation skip-over every 60 lines printed.

How do I modify NEWDOS so that I obtain a POKE 14312,12 command every 60 lines? (David Ansley, Hall, NY)

A: If you're using NEWDOS, and not NEWDOS80 version 1.0 or 2.0, then your only choice is to write a supervisor program that takes control away from the ROM line printer routine.

For LPRINT, this is fairly easy: Write a Basic subroutine that calculates the number of lines printed each time you use the printer. That is, store whatever you want printed in a string variable, and go to this subroutine (which counts how many lines' worth of information are in that string and which adds it to a counter before printing the string). When you reach the line limit, the routine sends a CHR$(12) to form feed the printer.

Intercepting LLIST is essentially done with the same technique, but done in machine language.

If you don't want to do the work yourself, Ramparts (Box S-8, Greenfield, NH 03047, 603-924-9406) sells a product called KVP for $24.95, which gives NEWDOS a Forms command, letting you control line length, page length, and automatic indentation for subsequent lines of any line that exceed one full line in length.

Q: I have a problem with my new disk system and printer: I can't back up my machine-language tapes to disk. It seems the addresses of the programs in memory are too low to be dumped to disk since TRSDOS 2.3 won't accept any address below 7000 hex. (J. S. Bellefontaine, New York, NY)

A: Your problem is rather simple to solve. On your TRSDOS disk is a program called Tapedisk. This DOS program is designed to load non-disk machine-language tapes into memory and transfer them to disk. This puts your tape programs on disk for execution with the disk drives. The programs aren't altered in any way and won't load or save data to the disk drives. In fact, once you execute one of these programs by typing its name at the DOS prompt, the computer ignores the DOS system until you reset it.

Another choice is to buy one of the machine-language monitor programs with tape and disk commands, and use it to transfer the tapes for you.

An important consideration is whether or not the system programs disable the disk interrupts. If the programs don't disable the interrupts, then it's possible for the programs to crash during operation by having a disk interrupt signal from the system-clock trigger a disk reboot. If the programs contain a Disable Interrupts command, this won't be a problem.

Q: I recently completed writing a Basic program on my new Model 4 which I believe could be of interest to special interest groups. It works well and does all things I had hoped for during conception, but it's far too slow to be useful. I think the problem is its string pool organization. I've been told that the problem would be cured, and the program would run much faster, if it were compiled into machine language.

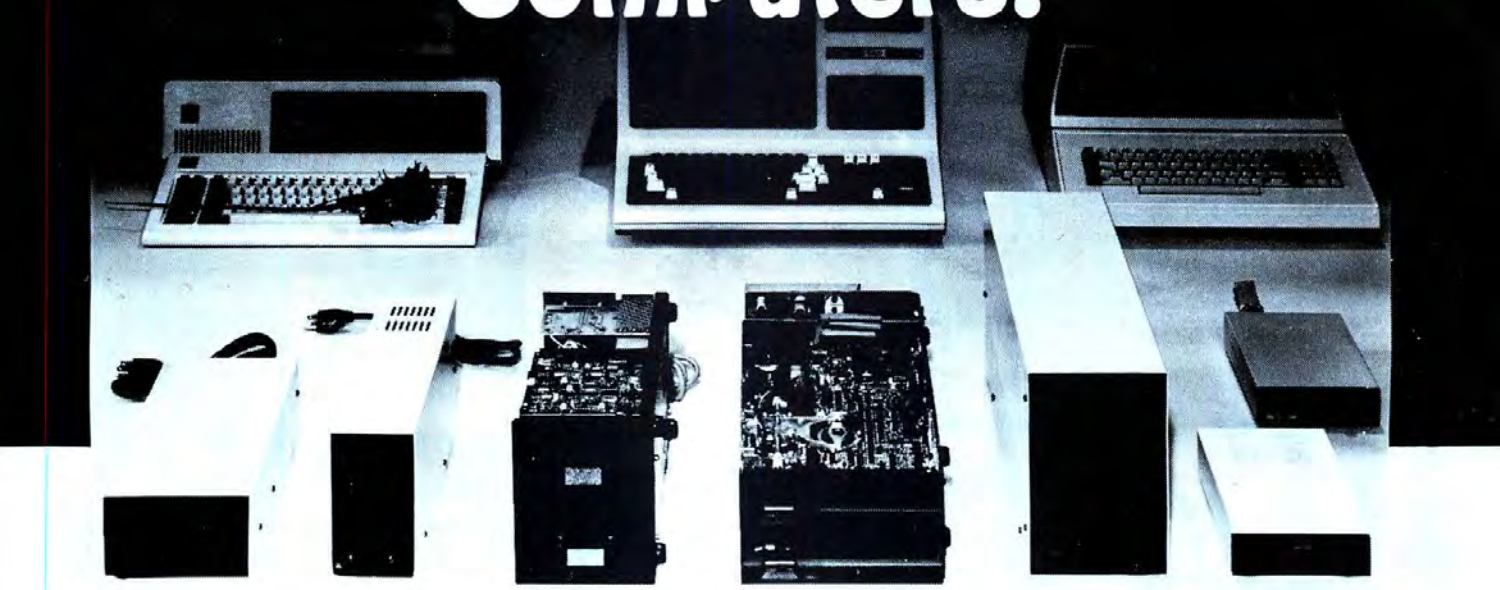
I don't understand what a Basic compiler really does, and I can't find one for Model 4 TRSDOS 6.0. Is there a program that will convert my Basic programs into machine language that can be operated by a Basic language programmer who knows nothing about machine language? If there is, will it solve the string pool reorganization problem, as well as speed up the Basic program? And where can I get such a program? (Dwain Hankins, Casper, WY)

A: Normally, Basic takes your program one line at a time and converts it into machine-code instructions so it will know what to do. After it converts and executes a line, it loads the next line and ignores the previous converted code. So if you branch back to a line previously executed, Basic converts it to machine code again. At any given instant, only one line of your Basic program is in machine code.

A Basic compiler takes a Basic program and converts it into direct machine-code instructions. The converted program is then saved as a machine-language program. From this

Continued on p. 234

Who sells low-cost high performance disk systems for TRS-80, IBM and other Popular Computers?



We Do!

We bring together name brand disk drives along with premium power supplies and cases to offer you the highest quality disk drive systems available on the market today. The reliability of our peripherals combined with excellent service assures you'll get the best value for your dollar. **CALL US!**

☎ **TOLL FREE**
1-800-321-3552

BARE DRIVES

Manufacturer	Model	No. of Heads	No. of Tracks	Full or Slimline	Price
Tandon*	5¼" TM100-1	1	40	Full	$169.00
Tandon*	5¼" TM100-2	2	40/40	Full	239.00
Tandon*	5¼" TM100-4	2	80/80	Full	319.00
MPI*	5¼" Model 51	1	40	Full	179.00
MPI*	5¼" Model 52	2	40/40	Full	239.00
MPI*	5¼" Mdl 501	1	40	Slimline	179.00
MPI*	5¼" Mdl 502	2	40/40	Slimline	239.00
MPI*	5¼" Mdl 902	2	80/80	Slimline	329.00
MPI**	8" Mdl 41	1	77	Slimline	399.00
MPI**	8" Mdl 42	2	77/77	Slimline	499.00

FOR TRS-80, HEATH, ZENITH, IBM & OTHERS

Single full size 5¼" case & power supply ... add $54.00
Dual thin line 5¼" case and power supply ..... add $69.00
Dual full size 5¼" case & power supply .. add $89.00
Case & Power Supply for 1 full size or 2 Thin line 8" drives ..... add $189.00

MICRO DATA SUPPLIES

22295 EUCLID AVE. | A SUBSIDIARY OF THE COMPUTER WAREHOUSE, INC. OF OHIO | EUCLID, OHIO 44117

WE'RE GIVING MAIL ORDER A GOOD NAME . . .

You no longer have to sacrifice quality products, personal service, or fast delivery for rock-bottom prices. With **MICRO DATA SUPPLIES,** you can have it all. And it's only a phone call away. Looking forward to your call.

Russ Knotts • President

CALL OUR NEW TOLL-FREE NUMBER
1-800-321-3552

NEWDOS/80 Version 2.0

FLOPPY DOCTOR INCLUDED

Model I and Model III

The hottest Disk Operating System is now available in its latest version. This is the ONE from Apparat, Inc., the people whose systems have made the TRS-80 the reliable computer.

$119.95

Let your TRS-80™ Test Itself With

THE FLOPPY DOCTOR & MEMORY DIAGNOSTIC

DISK DRIVE & MEMORY DIAGNOSTIC PROGRAM

by David Stambaugh

A complete checkup for your MODEL I or MODEL III. THE FLOPPY DOCTOR-Version 3 completely checks every sector of single or double density 35-, 40-, 77-, or 80-track disk drives. Tests motor speed, head positioning, controller functions, status bits and provides complete error logging. THE MEMORY DIAGNOSTIC checks for proper write/read, refresh, executability and exclusivity of all address locations. Includes both diagnostics and complete instruction manual.

For MODEL I $24.95
For MODEL III $29.95

MODEL I DOUBLE DENSITY PACKAGE

Everything you need to convert your TRS-80 Model I to run double density. Complete with software hardware, and instructions, installs in minutes with no soldering, wiring or cutting.

Dosplus V 3.4 w/LNDOUBLER 5/8 .. $199.00
LNDOUBLER 5/8 Board w/o dos ... $169.00

16 K RAM SPECIAL

4116, 200ns. Guaranteed one full year $17.95

MODEL IV — 64K MEMORY EXPANSION

GUARANTEED 1 FULL YEAR - SET OF 8

4164 RAM CHIPS $69.95

w/PAL CHIP to go from 64 to 128K $84.95

RETURNS: Must have authorization number, obtained at (216) 481-1600. Unauthorized returns will be refused & damaged goods will be refused. All returns subject to 15% restocking fee. No returns after 30 days.

DISKIT III™

MODEL III & IV DISK UPGRADE

FREE

Floppy Doctor with the purchase of any DISKIT III $30.00 VALUE!

Features
- Gold Plated Edge Connectors
- Switching Power Supply
- Supports 5" or 8" Drives
- 40 80 Track Supported
- Single Dual Head Supported
- Metal Disk Drive Brackets
- All Hardware and Cables
- 1 Hour or Less for Installation
- 100% Compatible
- No Soldering Needed
- 180 Days Warranty on Controller

DISKIT III w/o Drives **$219.00**
DISKIT III w/one Tandon 100-1 40 Track Drive **$385.00**
DISKIT III w/two Tandon 100-1 40 Track Drive **$555.00**
DISKIT III w/two Tandon 100-2 40/40 Dual Drive **$695.00**

PRINTERS New Lower Prices

New! Silver Reed EXP Daisy Wheel	$429.00
New! Delta 10 by Star Micronics	$499.00
Gemini 10X by Star Micronics	$299.00
Gemini 15X by Star Micronics	$ CALL
New! STX-80 "Thermal" by Star Micronics	$179.00
Epson RX-80	$319.00
Epson FX-80	$ CALL
Epson FX-100	$ CALL
Okidata 82A	$379.00
Okidata 83A	$649.00
Okidata 92	$439.00
Okidata 93	$719.00

ACCESSORIES

Graftrax Plus	$ 65.95
MX-80 Ribbons	$ 7.95
MX-100 Ribbons	$ 18.95
Epson Service Manual MX80/100	$ 34.95
TRS-80 Model I & III	
10 ft. Parallel	$ 19.95
10 ft. Parallel cable extensions	$ 19.95

ELECTRIC PENCIL Version II

Model I and Model III

An expanded version of the critically acclaimed original word processing system! Includes all features of Version I plus many new extensions.

Disk Version $79.95
Tape Version $69.95

ELECTRIC WEBSTER

"Cadillac" of Spelling Checkers 80 Micro 9/82.

- 50,000 WORD DICTIONARY
- FAST and ACCURATE - No other Spelling Checker comes close!
- INTEGRATED - Proofs and corrects from within most popular word processing programs!
- SMART - Finds and displays correct spelling!
- HYPHENATES automatically - inserts discretionary hyphens with 100% accuracy (optional).
- COMPLETE - One step proofing system.

Electric Webster w/correcting feature $129.95
Hyphenation feature $49.95
Grammatical feature $39.95
COMPLETE SYSTEM all Four Programs $199.00

NEWSCRIPT 7.0

NEWSCRIPT is the versatile TRS-80 word processing program. It supports "smart" printers like the Epson, Okidata, NEC, C.Itoh, Spinwriter, Centronics 739 and more.

NEWSCRIPT $109.95

THE HOME ACCOUNTANT

The #1 best selling program for home and small business accounting is now available for TRS-80 Model 3. Handles up to 99 accounts, five checkbooks, multiple income accounts, and can split transactions to any number of accounts. Prints net worth statements, income statements, as well as custom designed reports. Displays a bar chart and trend for any selected category. It is easy to use and yet provides all the power you'll need. (By Continental Software, requires Model 3, 2 drives, 48K.) $62.95

MICRO DATA SUPPLIES

MICRO DATA SUPPLIES
A SUBSIDIARY OF THE COMPUTER WAREHOUSE, INC. OF OHIO
22295 EUCLID AVE. EUCLID, OHIO 44117

NEW PRODUCT! ORGANIZE & PROTECT

ROLLTOP 100 Disk File™
Model #RT100

Twice the capacity (100 - 5¼" Disks) of the leading "flip top" file. But it takes no more desk space! An outstanding design that combines contemporary styling with the elegance of a rolltop enclosure and a textured buff plastic body. It includes 10 diskette dividers and anti-skid feet $36.00
Locking Model #RT100L $46.00

MISC.

Avery Tabulables
5000 - 3½ X 15/16" $15.95
Fan Fold Paper
9½ X 11" 18 lb. White - 3000 ct. $29.95
14⅞ X 11" 18 lb. White - 3000 ct. $39.95
5¼" File Box for 75 Diskettes $19.95
LABLMAKER Kit w/ 400 1 up Labels $29.95
CASSLABEL Kit w/ 200 Casette Labels and 200 1 up Labels $34.95

MODEL III & IV RS-232 Communication

DIRECT REPLACEMENT FOR RS-232, FULLY TESTED & BURNED IN, EASY INSTALLATION, 120 DAY WARRANTY.
RS-232..... 78.95

SPRINTER

Double Your Speed

Speed-up cuts computer operation time in half, saves time and money. Fast4 MHz Z80B CPU included, installs in 15 minutes with no soldering or cutting.

SPRINTER III for MOD III $89.95
SPRINTER I for MOD I $89.95

wabash® SUPER SALE!

AS LOW AS **$1.59**

5¼" SOFT SECTORS BULK SINGLE SIDE SINGLE DENSITY W/HUB RING 100% CERTIFIED 1 YR WARRANTY **$1.59***	5¼" BOX OF 10 **LOGO PRODUCT** SINGLE SIDE SINGLE DENSITY W/HUB RING 100% CERTIFIED 2 YR WARRANTY **$18.95**
5¼" SOFT SECTORS BULK SINGLE SIDE DOUBLE DENSITY W/HUB RING 100% CERTIFIED 1 YR WARRANTY **$1.89***	5¼" BOX OF 10 **LOGO PRODUCT** SINGLE SIDE DOUBLE DENSITY W/HUB RING 100% CERTIFIED 2 YR WARRANTY **$22.95**
5¼" SOFT SECTORS BULK DOUBLE SIDE DOUBLE DENSITY W/HUB RING 100% CERTIFIED 1 YR WARRANTY **$2.79***	5¼" BOX OF 10 **LOGO PRODUCT** DOUBLE SIDE DOUBLE DENSITY W/HUB RING 100% CERTIFIED 2 YR WARRANTY **$31.95**

*Now Get High Quality at a Low Price Manufactured by a Major Disc Company For MDS Without Their Name on Diskettes.
*Minimum order 20 diskettes with Tyvek envelope and storage shipping box
*Quantity discounts - 100 deduct 3% 1000 deduct 5%, 10,000 deduct 10%

BOOKS

OTHER MYSTERIES
TRS-80 DISK $19.95
Microsoft Basic Decoded $24.95
The Custom TRS-80 ..... $24.95
Basic Faster & Better $25.95
How To Do It On The TRS-80 ..... $24.95
TRSDOS 2.3 Decoded $24.95
The Custom Apple $24.95
Machine Language Disk I/O $25.95
CP/M Primer $15.95

ELECTRONIC PROTECTION DEVICES

The PLUM 3-way EMI-RFI AC Power Line Noise Filter (wall outlet) $44.95
The LEMON - 6-way Surge Suppressor for AC Power Lines (wall outlet) $44.95
The LIME - same as LEMON w/5 ft. cord and on-off switch $79.95
The PEACH - 6-way Surge Suppressor, EMI-RFI Line Filter (wall outlet) $89.95
The ORANGE - same as LIME w/EMI-RFI Filtering PLUM & LIME Combined $129.95

Verbatim

*23.95

DATALIFE
5¼" 1S/DD (MD 525-01) box of 10 ..... $26.95
5¼" 2S/DD (MD 550-01) box of 10 ..... $39.95
5¼" 2S/4D (MD 557-01) box of 10 ..... $51.50
8" 1S/DD (FD 34-8000) box of 10 ..... $43.95

3M

*21.95

5¼" 1S/DD/SS — box of 10 ..... $23.95
5¼" 2S/DD/SS — box of 10 ..... $35.95
***With Purchase of 10 Boxes ONLY**

Dysan CORPORATION

5¼" 1S/DD/SS — box of 10 ..... $35.95
5¼" 2S/DD/SS — box of 10 ..... $44.95

ISN'T IT TIME YOU SCORED . . .WITH SCOREPAC!™

20 Paragon Plain Janes 1S/SD $38.95
20 Paragon Golds 1S/DD $46.95
20 Paragon Golds 2S/DD $56.95

LNW

TRS-80 Mod. I Expansion

$349.00

- RS232c serial I/O
- Parallel printer port
- Gold-plated connectors
- Floppy disk controller
- Full 32k 200ns RAM
- Real time clock
- 6 month warranty
- Heavy steel case
- Thousands of users
- Works with any DOS 100%

NEW! MODEL IV FEATURES ON YOUR MODELS I & III FOR 1/2 THE PRICE!

BUT ONLY WITH . . .

The HOLMES VID-80

80 CHARACTER VIDEO INCLUDES:

Easy plug-in installation inside case

Software patches available for many programs

Improved Graphic Resolution

80 X 24 or 64 X 16 selectable modes.

CP/M 2.2.

112K EXTENDED MEMORY . . . ADD $100.00

(CP/M 3.0 4th quarter '83)

ALL of these features for a SPECIAL price

$ CALL

PRICES SPECIFICATIONS & OFFERINGS SUBJECT TO CHANGE WITHOUT NOTICE

MAIL ORDER PRICES ONLY

Call 1-800-321-3552
MONDAY thru SATURDAY
IN OHIO AND OTHER INQUIRIES (216) 481-4993

WE ACCEPT
- Visa
- MasterCard
- Checks
- Money Order
- C O D

MOST SCHOOL, GOV'T. AND FORTUNE 1000 CO. P.O.'s ACCEPTED ADD 3% Serv. Chg. Min. order $50

ADD $3.00 FOR SHIPPING & HANDLING
$6.00 Extra for C O D Orders
Ohio Residents add 6.5% Sales Tax

Which Chip Is Which?

Today's MS-DOS and tomorrow's 80186.

edited by Eric Grevstad

Things were easier when personal computers either had Z80's or 6502's, but today's micros carry a baffling variety of CPU's or microprocessors. Intel Corp. of Santa Clara, CA, has a family of four chips, found in machines ranging from IBM's PC and PCjr to Tandy's new Model 2000; the chips offer wildly different specs, nearly identical names, and—at least to some extent—the ability to run the same software.

This makes for powerful computers and confused owners. For instance, though home computer buyers rarely care about chip counts and input/output (I/O) buses, a few PCjr owners will learn that their computer uses the same Intel 8088 microprocessor found in IBM's PC. They may even pick up enough jargon to tell friends they have a "16-bit computer," more impressive than 8-bit micros. They'll be wrong.

The 8088 is not strictly a 16-bit chip. It does process data in 16-bit registers, but it's limited to an 8-bit bus; in effect, its operations are fast but its I/O is slow. Its big brother, the 8086, is a true 16-bit device with a 16-bit bus.

And both are obsolete. Intel's new single-user flagship, the 80186, is not only a true 16-bit chip but a whole CPU board in itself: an enhanced 8-MHz 8086, plus a clock generator, two direct memory access channels, a programmable interrupt controller and timer/counters, and other support chips—all on one integrated circuit, which not only replaces as many as 20 other chips in an 8086-based system, but runs twice as fast.

To micro manufacturers, the 80186 is cause to rejoice: drastically reduced size, weight, power requirements, and cost, with increased performance—and compatibility with existing 8086/8 software.

Tandy, of course, chose the 80186 for its state-of-the-art Model 2000. Australia's Dulmont Electronic Systems launched its briefcase-sized 80186 Magnum (256K RAM, 128K ROM including MS-DOS, 16-by-80 LCD display) at the Fall 1983 Comdex, and other manufacturers are jumping on the Intel bandwagon.

And, just as there are two versions of the older chip, Intel has two versions of the newer: the 80186 and the 80188. It was originally rumored that the mysterious Peanut would use the 80188—an integrated chip with an 8-bit interface, less expensive than the 80186 and allowing the use of cheaper 8-bit peripherals.

But IBM stayed with the 8088. Why is a matter for speculation. *InfoWorld*'s John Dvorak, citing a Santa Ana, CA, industry newsletter and "gossip rag," reports that, in writing MS-DOS, Microsoft programmers overlooked Intel's documentation and used some 8088 vector addresses, reserved by Intel for future hardware functions.

Although such a slip would be Microsoft's and not Intel's fault, market factors would oblige an angry Intel to redesign the 80188, rather than have Microsoft belatedly change MS-DOS and render thousands of current programs unusable. This, if you believe the rumor, is why PCjr uses the old 8088, and why an 80188 machine hasn't appeared yet.

(The other IBM rumor is that Big Blue is tired of sharing the world with Microsoft, and will eventually drop MS-DOS in favor of a proprietary operating system. This prediction's been around for over a year, but new hardware that won't run MS-DOS might give IBM a chance to make its break.)

Meanwhile, companies like Tandy and Dulmont are counting on being able to use today's software in their 80186-based systems. It's certain the Model 2000 will expand Tandy's software library—even if MS-DOS programs should require minor changes to run on the 80186, publishers are a lot more likely to make those than to write 8-bit TRSDOS versions—but program crashes could give the new machines a stumbling start.

This, according to Model 2000 Product Line Manager Don White, is why Tandy sent back its first batch of 80186's. "[Compatibility] is not a problem for us in that we are not using the earlier version of the chip, which

"If you violate the operating system, you're going to have trouble."

did have problems with MS-DOS," White told *80 Micro*.

"We found out when we got the machines in and did some testing that there were problems, which is why we sent them back and are using the latest version," White said. "There have been several different [80186] masks and we're only using the last one."

Tandy's software product planner, Doug Dillhoff, gave more details: "The first [80186] had a Move String function that wasn't working. That was no go, because our MS-DOS linker uses [that function], so we sent them back. That's one reason we had a little bit of delay, because Intel wasn't able to deliver the new versions as fast as we would have liked."

Though not every IBM PC program will run on the Model 2000, Dillhoff said, fears of MS-DOS incompatibility are groundless: "The big thing is that we don't use 80186 instructions, so we don't use any of those [reserved] vectors. We do require that people go through the BIOS for everything for documented calls, so programs that go directly to video will have problems, since our video memory's not in the same place as IBM's.

"If you violate the operating system, you're going to have trouble. Things are becoming more standard [as MS-DOS becomes the norm], so we stayed within the operating system," Dillhoff said.

"We've been doing some tests and seeing what will run, and the only trouble is if people go directly to memory. Otherwise there's no problem, and of course [programs] run a lot faster."

Looking to the future, neither Microsoft nor anyone else is talking about MS-DOS 3.0, the Bellevue, WA, company's long-awaited multitasking system—Microsoft's answer to Digital Research Inc.'s Concurrent CP/M, which lets users run several applications programs at once. On Nov. 10, however, Microsoft announced Windows (due in April 1984), a window management extension to MS-DOS 2.0 that competes with VisiCorp's new VisiOn. Radio Shack is among the manufacturers listed as supporting Windows.

As for hardware, the 80186 won't be the new chip on the block forever. Intel already markets the multi-user 80286, which includes a pair of 8086's and is two or three times faster than the Model 16's Motorola 68000, and plans a frighteningly powerful CMOS 32-bit 80386 for late 1984 or 1985. ■

—E.G.

PCjr: There They Go Again

Fair performance, whopping price.

In 1981, IBM introduced its Personal Computer to titanic publicity and divided opinions: The PC had the industry's biggest name behind it, and its 16-bit 8088 chip was more impressive than its 8-bit competitors'. But its performance was only adequate, it had an awkward, nonstandard keyboard, and its price—even for a stripped-down model—was far above its rivals'.

Even so, the PC not only defied but outsold other desktops. On Nov. 1, 1983, Big Blue repeated its successful strategy.

After a year of mounting, if not hysterical, press speculation, IBM unveiled the "Peanut," formally named PCjr—an 8088-based home computer that can run many of its big brother's MS-DOS programs, but otherwise flies in the face of the year's trend to more powerful, less expensive low-end micros.

IBM offers two PCjr models, each consisting of a 25-ounce keyboard and a 9-pound system unit that connects to a color monitor or TV set. The base model offers 64K of RAM, a cassette interface for bulk storage, and two cartridge slots for ROMpack programs; its suggested retail price is $669.

The enhanced version, which IBM expects to account for 90 percent of PCjr sales, has 128K RAM, an 80-column display (twice as wide as the base model's), and comes with both cartridge slots and one 360K double-sided disk drive. It costs $1,269.

The disk-based Junior runs under PC-DOS (IBM's name for MS-DOS) 2.1, an update of version 2.0. But not all MS-DOS 1.1 or 2.0 programs may be compatible with the PCjr—besides DOS differences, some 1.1 or 2.0 software requires more than 128K RAM.

While PCjr owners may not be able to bring their 1-2-3 disks home from the office, they will have access to a large library of software. Besides four cartridge-based games ($35 each), IBM announced PCjr versions of 20-odd disk programs, ranging from Bumble Games and Juggles' Butterfly to PFS:File, Time Manager, and EasyWriter. Peanut's color and graphics capabilities are better than the PC's, but no match for Commodore or Atari machines' game animation.

Even allowing for the attraction of MS-DOS, PCjr prices seem outrageous. The 64K model is more than three times the current price of the Commodore 64 ($200), and the 128K one-disk Junior costs about twice Coleco's Adam—which offers 80K RAM, a fast tape storage device, and a daisy-wheel printer for $600 ($750 after Jan. 1).

Also, some of PCjr's specifications are bizarre. Rather than expand the system's skimpy cassette Basic (a fuller Basic is available on cartridge for $75), IBM opted to dedicate some of its 64K ROM to a cartoon tutorial called Keyboard Adventure, similar to the "Apple Presents Apple" disk included with the IIe.

The PCjr's sole innovation is its infrared keyboard, which runs on four AA batteries and permits cordless operation from up to 20 feet away (keyboard cords, $20 each, are required with two or more Peanuts in one room, lest signals get scrambled).

On the other hand, as the *New York Times*' Andrew Pollack wrote, "As reporters first caught sight of the keyboard at a demonstration, there were gasps of dismay." Unlike the full-travel keyboards of the Adam, C64, or 64K Color Computer, Junior's has rubberized chiclet keys, unsuitable for long periods of typing.

The inadequate keyboard, in fact, may be an indication of PCjr's awk-

Will families flock to PCjr?

ward market position: as a 128K MS-DOS system, Peanut is competing with the original PC as much as with Commodore or Atari. Obviously, a functionally identical PCjr would stop sales of the more costly PC overnight.

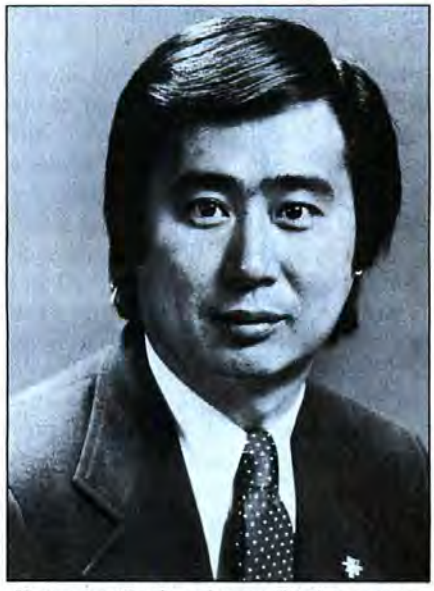
Hence, the new machine has an inferior keyboard and, at least officially, no potential for expansion—128K and one disk is as far as IBM will go, though PCjr's architecture seems open to outsiders who'd like to add a second drive or up to a megabyte of RAM. Also, if family history is any guide, buyers can expect less expensive PCjr clones—Compaq Juniors or Eagle Juniors, perhaps.

At this writing, though, they can't even expect a PCjr. Though the Nov. 1 announcement attracted network TV news and front-page newspaper coverage, Juniors will not reach IBM's 1,100 dealers until the first quarter of 1984. Even then, a press release admitted, "Initial supplies will be limited and may not be sufficient to meet expected demand."

What that means, cynics declared, is that Big Blue was not only announcing the year's most awaited product at breathtaking prices, but relying on its clout to freeze everyone else's Christmas sales—hoping that families would wait to buy a $1,269 PCjr in March, rather than a $200 Commodore in December.

The more generous interpretation was that IBM was, as the release said, bracing for high demand. It's been rumored since summer that Teledyne Inc. of Irvine, CA, which assembles the PCjr to IBM's specifications, has a warehouse full of units ready to ship, but—judging from the PC's skyrocketing success—a warehouse might not be enough.

Dave O'Connell, IBM's PCjr product manager, summed it up for reporters: "We are going to sell as many of these things as we can possibly make." ■



Yamagata: Tandy welcomes IBM's company. (Radio Shack photo)

Mixed Peanut Reviews

Rivals and writers react to PCjr.

Now that months of second-guessing the PCjr's specs are over, industry watchers are trying to predict its success. No one believes an IBM machine will flop, but the new machine's price and performance kept critics from getting enthusiastic.

The Seybold Report on Professional Computing put it bluntly: "[The PCjr] is a surprisingly modest machine, embodying relatively little innovation. It is priced much higher than the systems against which it will be sold. If the system is a success it won't be deserved based on the technology or features; rather, it will be because of the IBM name. If it were not for [that] name, we could write the machine off as an interesting but overpriced also-ran."

Many onlookers, however, feel that IBM's name is exactly what the topsy-turvy home market needs—that PCjr will not only pep up the field, but stabilize it, reassuring buyers who've watched the chaos of Commodore price cutting and TI and Atari losses.

Mark Yamagata, Radio Shack's director of merchandising for personal computer products, told *80 Micro*, "I think it's been said by everybody that IBM is going to give some credibility to the whole industry. We don't consider ourselves a failure, but people have been seeing what's happened to TI, Atari, and others, and it's nice to see someone with some credibility coming into the business. It'll be good company for us."

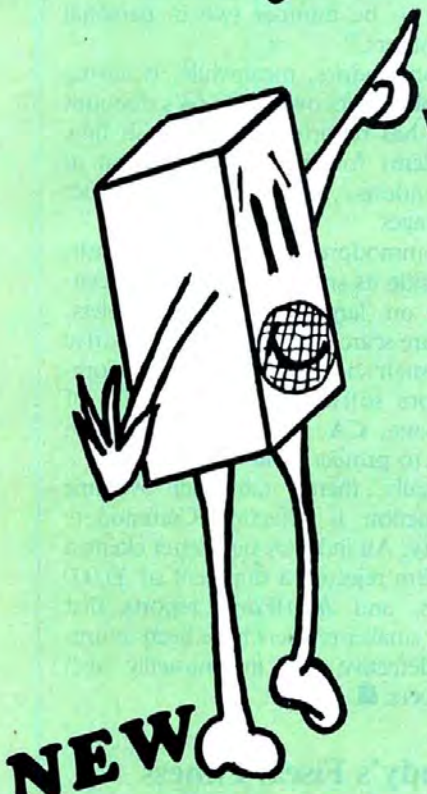
As for PCjr's price, Yamagata is charitable: "Well, their unit is 16-bit equipment, to be fair to them. This is their first venture into the home computer business, and I'm sure they probably have a fair price point."

Others expect the PCjr to sell well, but not to dominate its field as the PC does. Talmis' analysts say, "The suggestion that IBM will take over the home computer market is similar to suggesting that Porsche will take over the auto market. Everyone might want one, but few would buy one to teach their teenager to drive."

Though some might mutter about Volkswagen keyboards at Porsche

LOOK WHO'S TALKING

NEW DEMO HOTLINE (212) 296 0399



VS 100

VOICE SYNTHESIZER FOR MODEL I, III, IV

- Unlimited vocabulary.
- Automatic inflection.
- Proven VOTRAX technology.
- We use the famous VOTRAX SC01 phoneme synthesizer. 4 programable pitch level.
- Built in audio amplifier with volume control.
- You can add voice to any basic program in minutes.
- Super efficient: one single line in Basic will produce a full sentence!
- Works with any speaker (or add $5.95 for handsome speaker module).
- Price breakthrough: same performance as units costing hundreds of dollars.
- Voice editor will help you create unlimited number of words, sound effects, etc.
- Hundreds of applications—now cost effective in education, robotics, speech therapy, monitoring, games, aid to handicapped, security, prompting... .
- Fully assembled and tested, 90-Day warranty.
- Ready to plug in and talk.
- As usual our 15-day money back guaranty protects you.

Toll Free Order Line
800·221·0916
Orders Only. NY & Info call (212) 296-5916. Hours: 9-5 E.S.T.

SPEAK ENGLISH AUTOMATICALLY!

TALKER 1.4 TEXT TO SPEECH CONVERTER - Unlimited vocabulary including: words, letters, symblols and numbers up to 99,999,999,999,999! Talking is as simple as: PRINT ★ "I talk!" Machine language DISK or TAPE: only $19.95

VS 100 Order # C201 for MODEL I
C203 for MODEL III or IV

Price includes Synthesizer Module, Power Supply, Instructions, Editing and Demo Software.
Speaker Module not included, add $5.95
Text to Speech Software $19.95

$69.95

SUPER CONTEST
Win $1000 + Royalties for the best talking game
+ Win $1000 for the best "serious application" software

Clubs: Contact us at (212) 296-5916 if you would like to arrange for a product presentation. Votrax is a registered trademark of Votrax.

Toll Free Order Line 800-221-0916

ORDERS ONLY, INFORMATION AND N.Y. ORDERS CALL (212) 296-5916. HOURS: 9 - 5 E.S.T.

800·221·0916

ALPHA Products 17

79-04 Jamaica Ave., Woodhaven, NY 11421 ☎ *(212) 296-5916*

ADD $3.00 PER ORDER FOR SHIPPING AND HANDLING.
WE ACCEPT VISA, MASTERCARD, CHECKS, M.O.
C.O.D. ADD $3.00 EXTRA.
NY RESIDENTS ADD SALES TAX.
OVERSEAS, FPO, APO; ADD 10%.
DEALER DISCOUNTS AVAILABLE.

"With the PC, IBM gambled that it could sell machines on the strength of its name alone."

prices, Talmis' logic is generally accepted. Few expect the PCjr, for instance, to wipe out Commodore overnight; the 64 has a sizable installed base, which is growing as fast as Commodore can ship the $200 machines.

In fact, Morrow Inc. President George Morrow told *InfoWorld* that Commodore "is going to make mincemeat of these people [IBM]." The PCjr's "toylike" specs and Commodore's no-holds-barred discounting, the desktop manufacturer said, would leave IBM as a high-priced novelty in the low-end arena.

But if its price puts Commodore safely in a sub-IBM part of the market, you can't say the same for Apple. Many analysts expect PCjr to spell trouble for the IIe, now selling for $1,795 with a monitor and one disk drive. Apple dealers are already offering several bundled packages with different peripherals and software, and further IIe discounts seem likely.

Apple II inventor Steve Wozniak told *InfoWorld* he wasn't worried: "We're solidly entrenched. We have a ton of software."

Nevertheless, everyone expects the PCjr to leave its mark on the industry. International Resource Development's Joan de Regt told *ISO World* that Commodore would fall from first place, with 25 percent of 1983's home computer market, to third place, with 15 percent of 1984's. Coleco's Adam, she predicted, would take second place, and the winner—seizing 20 percent of the market in its first year—would be PCjr.

Future Computing Inc. Chairman Egil Juliussen agreed, noting, "In its first full year of production, the IBM PC got 18 percent of the office computer market. The PC Junior could do the same."

Still, as the *Seybold Report* concluded, "With the PC, IBM gambled that it could sell machines on the strength of its name alone. It won that gamble. Now, with the PCjr, it is making the same gamble again." ■

—E.G.

One Down, Two to Go?

TI, Apple, and Commodore problems.

Compared to IBM's introduction of the PCjr, Texas Instruments' decision to leave the home computer business went almost unnoticed. The Austin, TX, firm officially threw in the towel on Oct. 28, ending a two-year saga of optimistic sales forecasts and profit-slashing price cuts with the words, "It became clear that fourth-quarter demand wouldn't be sufficient to prevent large additional losses. In order to limit further financial drain on TI we have made the decision to withdraw."

While TI's last hurrah, slashing its half-million remaining 99/4A's to $49, upstaged rivals' Christmas sales, the company kept its seat at the micro table with the well-received Professional Computer, reportedly outselling all but IBM's and Compaq's entries in the MS-DOS market.

A similar shift, focusing marketing attention on a different machine, may await Apple Computer Inc. when its McIntosh debuts in early 1984. Not only does PCjr pose a threat to the IIe, IBM may have dealt a death blow to Apple's Lisa.

Shortly before Peanut's debut, Big Blue quietly announced two new desktops, the 3270 PC and the PC XT/370, with, respectively, terminal capability and the ability to run most IBM 370 mainframe software.

Priced competitively with Lisa, which already trails the PC in sales and which still lacks mainframe communications, the new versions may shut Apple out of the office market.

Still, the Cupertino, CA, orchards aren't barren yet. The IIe should receive a hard disk and Lisa-type windowing software soon, and Chatsworth, CA's Rana Systems has announced a box with an Intel 8086 and two disk drives that will let the popular Apple run MS-DOS programs. And McIntosh, though the latest rumors describe it as a $2,400 desktop instead of a $1,200 home micro, could give Apple a happy new year.

"If Apple did nothing, it would be devastated," Future Computing Inc. Chairman Egil Juliussen told *ISO World*. "But Apple has plenty of room to change things. They'll have to make a lot of changes, but they'll continue to be number two in personal computers."

Commodore, meanwhile, is having troubles of its own. The 64's discount price has reportedly led to cash flow problems for the company, even as stupendous sales cause product shortages.

Commodore is rumored to be pushing aside its smaller dealers to concentrate on large chains; nevertheless, 64's are scarce, and the 1541 disk drive is in such short supply that one Commodore software company, HES of Brisbane, CA, may produce its own drive to protect its investment.

Finally, there's talk that overtime production is affecting Commodore quality. An industry newsletter claimed the firm rejected a shipment of 30,000 drives, and *InfoWorld* reports that some smaller retailers have been returning defective 64's in unusually high numbers. ■

Tandy's Fiscal Fitness

While Apple's third-quarter profits declined 73 percent and Texas Instruments pulled out of the home computer market after half-billion-dollar losses, Radio Shack's cash registers keep ringing.

According to Tandy Corp.'s annual report, fiscal 1983 net sales were $2.48 billion, up 22 percent from last year. Net income was $278.5 million ($2.67 per average common share), up 24 percent. And Tandy's retail sales outlets increased from 8,518 to 8,868.

Except for CB radios, walkie-talkies, police band scanners, and PA systems, every class of Radio Shack merchandise posted increased sales in the year ending June 30, 1983. Microcomputers and software led the way, with sales up 34 percent from last year.

The TRS-80 line now accounts for

THE ALPHA JOYSTICK -30%

NOW ONLY $28.00

SPECIFY MOD I OR III

DON'T WAIT!

Limited time special

Works with all joystick games.

Model IV: order model III version

$59.95 NEWCLOCK-80

Wouldn't it be nice if your computer could always boot up with the right time and date and then stay accurate. Newclock-80 will enhance your Model I or III system with powerful clock/calendar/timer functions.

Using LSI (large scale integration) and custom circuits, Newclock-80 provides MO/DATE/YR, HR:MN:SEC plus AM/PM and day of week and even takes care of leap years! It continues to keep time and date with quartz accuracy when the computer is turned off or experiences a power failure. A single battery lasts over 2 years.

Compatibility: Newclock-80 is compatible with any operating system, including DOSPLUS, NEWDOS, LDOS. With its fully decoded circuitry it will work with any other hardware you may own. Bus expanders are available.

Installation is very simple, no tools, no disassembly, no soldering. Just plug it in, that's all. There is no power supply or messy cable. Newclock-80 plugs into the rear of the keyboard ❸ or side of the Exp. Int. ❷. Model III Newclock fits the 50 pin card edge (underneath) ❶

The Software: Newclock-80 is as easy to use as it is to install. -"SET", a Basic program, is used only once to set the time and date and select 12 or 24 hour format. -"TIMESTR", also in Basic, patches your computer "TIME$" function to read Newclock-80. It also adds "TIME$" to keyboard-only systems, a short routine is simply "poked" into low memory.

Newclock-80 uses 12 ports (176 to 188): 6 for the time, 6 for the date. The data is conveniently stored in decimal form, no conversion is needed. You can read or modify any digit using simple Basic "INP" and "OUT" statements.

No risk trial. Order your Newclock-80 today, see how easy it is to install and operate then decide within 30 days if you want to keep it. If for any reason you are not delighted with its quality and performance, you may return it for a prompt and courteous refund.

Your unit will come complete ❹ with software on tape, detailed instructions, handy reference card, and a 90 day warranty. Specify Model I or III. Software is also available on disk: add $5.
Lithium battery (not included) available from RADIO-SHACK (#23-162) or add $1.50 to your order.

Thanks to outstanding engineering and efficient manufacturing, ALPHA Products is once again able to offer a great product at a surprising price. Order your Newclock-80 at no obligation today.

Toll Free Order Line

800·221·0916

Orders Only. NY & Info call (212) 296-5916. Hours: 9-5 E.S.T.

ALPHA Products

79-04 Jamaica Ave., Woodhaven, NY 11421 ↙17 (212) 296-5916

ADD $2.50 PER ORDER FOR SHIPPING AND HANDLING.
WE ACCEPT VISA, MASTERCARD, CHECKS, M.O.
C.O.D. ADD $3.00 EXTRA.
NY RESIDENTS ADD SALES TAX.
OVERSEAS, FPO, APO: ADD 10%.
DEALER DISCOUNTS AVAILABLE.

34.6 percent of Tandy warehouse shipments, followed by audio equipment and tape recorders (18.2 percent) and electronic parts, test equipment, and batteries (13.2 percent).

Radios, phonographs, and TV sets slipped from 9.4 to 8.6 percent of Tandy business. Telephones and intercoms climbed from 6.5 to 8 percent, with more expected as Radio Shack increases its Telephone Center campaign.

A closer look at Tandy's micro sales reveals that the Models I/III/4 represented 28.1 percent of TRS-80 shipments. The Models II/12/16 trailed with 21.4 percent, followed by printers (16.5 percent), Color Computers (9.8 percent), and software (9.2 percent). Other equipment contributed 11.9 percent of TRS-80 totals; the Model 100 hadn't been around long enough to boost portable and pocket computer sales beyond 3.1 percent.

Looking ahead, Fort Worth admits that slugging it out with IBM, Apple, Pioneer, and Panasonic isn't easy: "The consumer electronics business is highly competitive. . . .The products which compete with [Radio Shack's] are manufactured by numerous domestic and foreign manufacturers and many of them carry nationally recognized brand names or private labels. Many of the Company's competitors have financial resources equal to or greater than the Company's resources."

Nevertheless, the report blandly concludes, "Management believes that among the factors that are important to its competitive position are price, quality, service, and the wide selection of electronic products carried at conveniently located retail outlets."

And, at least as of the first quarter of fiscal '84, Tandy's good fortune seems to be holding. Figures released for that period show consolidated sales of $583.4 million and net income of $59.6 million, up 11 percent from a year ago.

Finally, there's a vote of confidence from John Gantz, editor of the *Tech Street Journal*, in the Nov. 14 *InfoWorld*: "Tandy is not only a financial fat cat—its current assets dwarf its current liabilities, and the company has close to $3 cash per share in the bank—but it is uniquely situated among personal computer vendors.

"With more than 8,500 stores, it is probably the only [micro] vendor that really understands retailing. By making money in the distribution side of the business, it insulates its margins.

"Tandy's stock, close to an all-time low, looks like a bargain. And it would be one at almost twice the price."

Stores Within Shacks?

It wasn't long ago that TRS-80 owners could buy software from anyone they wanted, as long as it was Radio Shack. Recently, however, Tandy's drawbridge has been coming down, with LDOS, CP/M Plus, Multiplan, and PFS:File ending the TRS-DOS monopoly in Radio Shack stores and catalogs.

But that's nothing compared to an experiment now being conducted by Tandy and PC Telemart, an independent software distributor based in Fairfax, VA. Under a pilot project, PC Telemart is putting information booths and ordering facilities, similar to catalog sales desks in department stores, inside 12 Computer Centers around Washington, DC—giving TRS-80 buyers access to thousands of non-Tandy programs.

According to *ISO World*'s Steven Burke, Radio Shack has "signed a preliminary agreement" allowing PC Telemart "to sell software not owned, licensed, or labeled by Tandy in Radio Shack stores."

The PC Telemart kiosks, Burke reports, will use a microcomputer to "provide a detailed listing and description of software available for Tandy machines." The distributor must test each program offered to make sure it runs on TRS-80's, and will be responsible for after-sale support.

Customers will order directly from PC Telemart, which will deliver programs to their homes or businesses. The agreement will increase Radio Shack store traffic and, presumably, provide some revenue from space rental for sales desks; however, Fort Worth reportedly will not share in the distributor's profits.

Tandy officials stress that the Washington plan is currently only experimental. Director of Personal Computer Merchandising Mark Yamagata told *80 Micro*, "That's just a test market in the Washington, DC, area. We'll decide [whether to expand the program] once we have more information."

PC Telemart spokesmen, meanwhile, claimed that the project would increase Radio Shack customers' choices from 300 to 3,000 software packages—and that was referring to CP/M, before news of the Model 2000 and Tandy's entry into the booming MS-DOS market. Whether Tandy or PC Telemart makes more money, it looks like the joint venture will profit TRS-80 owners the most.

1983: Desktops

A year after *Time*'s "Machine of the Year" story, the Yankee Group has belatedly named 1983 "the Year of the Desktop Computer." Worldwide, the Boston, MA, analysts say,

Live from Las Vegas

The Model 2000, released at Comdex/Fall's halfway point, easily upstaged a raft of 8088-based PC copies and portables (including entries from big names Sperry and ITT). Next month, Pulse Train reports on the industry's and Las Vegas' biggest convention, and on the few entries that managed to share Tandy's spotlight.

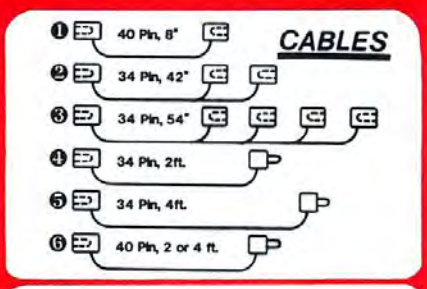
Incidentally, Tandy's news blackout held almost intact until the 2000's Caesars Palace press debut on Nov. 30. The *New York Times* had a story that day and the Dec. 12 *InfoWorld* broke the story as did January's *80 Micro*, but *Micro MarketWorld*'s Nov. 28 issue declared, "[Tandy's MS-DOS] system, possibly to be called the Model 14, will be introduced before April. It will undoubtedly be built by taking the Model 12 and adding a new board with an 8088 microprocessor."

NOW MODEL I AND MODEL III!

Now Model III users can take advantage of the ALPHA I/O system too. Our new MOD III/I BUS CONVERTER allows most port based Model I accessories (such as our ANALOG-80, INTERFACER 2 and INTERFACER-80) to connect to the Model III bus. MOD III/I BUS CONVERTER, complete with all connectors, only $39.95.

PRINTSWITCH

Have 2 printers on line at all times and select printer 1 or 2 by means of a conveniently located switch. End the problem of constantly plugging and unplugging printer cables. PRINTSWITCH is a compact module that plugs onto the parellel printer port of your TRS-80 and provides an edge connector for each of your two printers. It works with any two types of printers: dot matrix, daisy wheel, plotters, TRS-80 converted selectrics, etc. Assembled, tested, ready to use with connector and instructions. For Model I or III (please specify). ONLY . . $59.00



SUPERIOR QUALITY REPLACEMENT & EXTENSION CABLES

Highest quality cable and high force, gold plated contacts ensure the utmost in connection reliability.

- 1 KEYBOARD TO EXPANSION INTERFACE . . . . . $21.
- 2 DISK DRIVE CABLE FOR 1 OR 2 DRIVES . . . . . $32.
- 3 DISK DRIVE CABLE FOR 3 OR 4 DRIVES . . . . . $45.
- 4 DISK DRIVE CABLE EXTENDER . . . . . $22.
- 5 PRINTER CABLE EXTENDER . . . . . $24.
- 6 40 PIN BUS EXTENDER — 2 ft. . . $22 4 ft. . . $24.

Custom cable configurations are also available. Call us.

YOU ASKED FOR IT: "EXPANDABUS" X1, X2, X3 AND X4.

CONNECT ALL YOUR TRS-80 DEVICES SIMULTANEOUSLY on the 40 pin TRS-80 bus. Any device that normally plugs into the keyboard edge connector will also plug into the "EXPANDABUS". The "X4" is shown with protective covers (included). The TRS-80 keyboard contains the bus drivers (74LS367) for up to 20 devices, more than you will ever need. Using the E/I, it plugs either between KB and E/I or in the Screen Printer port. Professional quality, gold plated contacts. Computer grade 40 conductor ribbon cable. X2 . . $29. X3 . . $44. X4 . . $59. X5 . . $74. Custom configurations are also available, call us.

ANALOG-80: A WORLD OF NEW APPLICATIONS POSSIBLE.

8 DIGITAL MULTIMETERS PLUGGED INTO YOUR TRS-80!!! Measure Temperature, Voltage, Current, Light, Pressure, etc. Very easy to use: for example, let's read input channel #4. 10 OUT 0,4 'Selects input #4 and also starts the conversion 20 A = INP(0) 'Puts the result in variable "A" Voila! Specifications: Input range: 0-5V to 0-500V. Each channel can be set to a different scale.
Resolution: 20mV (on 5V. range). Accuracy: 8 bits (.5%). Port Address: jumper selectable. Plugs into keyboard bus or E/I (screen printer port). Assembled and tested. 90 day warranty. Complete with power supply, connector, manual . . . . . $139.

TIMEDATE 80: REAL-TIME CLOCK/CALENDAR MODULE

Keeps quartz accurate time for 3 years on 2 replaceable AAA batteries (not included). Gives MO/DATE/YR, DAY of WEEK, HR:MIN:SEC and AM/PM. Features INTELLIGENT CALENDAR and even provides for Leap Year. This compact module simply plugs into rear of Keyboard or side of Expansion Interface (may be shipped inside E/I). Includes cassette software for setting clock and patching to any DOS (including NEWDOS 80 2.0). Optional "Y" connector allows for further expansion. For Model I. Fully assembled and tested. Complete with instructions and cassette. ONLY $95.00. "Y" option, add $12.00.

GREEN SCREEN WARNING

IBM and all the "biggies" are using green screen monitors. Its advantages are now widely advertised. We feel that every TRS-80 user should enjoy the benefits it provides. But **WARNING:** all Green Screens are not created equal. Here is what we found:

- Several are just a flat piece of standard colored Lucite. The green tint was not made for this purpose and is judged by many to be too dark. Increasing the brightness control will result in a fuzzy display.
- Some are simply a piece of thin plastic film taped onto a cardboard frame. The color is satisfactory but the wobbly film gives it a poor appearance.
- One "optical filter" is in fact plain acrylic sheeting.
- False claim: A few pretend to "reduce glare". In fact, their flat-and shiny surfaces (both film and Lucite type) ADD their own reflections to the screen.
- A few laughs: One ad claims to "reduce screen contrast" Sorry gentleman but it's just the opposite. One of the Green Screen's major benefits is to increase the contrast between the text and the background.
- Drawbacks: Most are using adhesive strips to fasten their screen to the monitor. This method makes it awkward to remove for necessary periodical cleaning. All (except ours) are flat. Light pens will not work reliably because of the big gap between the screen and the tube.

Many companies have been manufacturing video filters for years. We are not the first (some think they are), but we have done our homework and we think we manufacture the best Green Screen. Here is why:

- It fits right onto the picture tube like a skin because it is the only **CURVED** screen **MOLDED** exactly to the picture tube curvature. It is Cut precisely to cover the exposed area of the picture tube. The fit is such that the static electricity is sufficient to keep it in place! We also include some invisible reusable tape for a more secure fastening.
- The filter material that we use is just right, not too dark nor too light. The result is a really eye pleasing display.

We are so sure that you will never take your Green screen off that we offer an unconditional money-back guaranty: try our Green Screen for 14 days. If for any reason you are not delighted with it, return it for a prompt refund.

A last word: We think that companies, like ours, who are selling mainly by mail should:•list their street address•have a phone number (for questions and orders)•accept CODs, not every one likes to send checks to a PO box•offer the convenience of charging their purchase to major credit cards.

How come we are the only green screen people doing it?

Order your **ALPHA GREEN SCREEN** today . . . $12.50

DISK DRIVE EXTENDER CABLE, FREE YOUR MINI-DRIVES.

End the daisy-chain mess once and for all. Fits all mini-drives: Percom, Aerocomp, Shugart, Micropolis, MTI, Vista, Pertec, Siemens, BASF. Easy to install: just remove the drive cover, plug in the EXTENDER CABLE and replace the cover.

Now you can change and move your drives without disassembly. Keep the cover on and the dust out. High reliability gold plated contacts, computer grade 34 conductor cable. Tested and guaranteed.

Get one for each drive . . . . ONLY $8.95

INTERFACER-80: the most powerful Sense/Control module.

- 8 industrial grade relays, single pole, double throw isolated contacts: 2 Amp. @ 125 Volts. TTL latched outputs are also accessible to drive external solid state relays.
- 8 convenient LEDs constantly display the relay states. Simple "OUT" commands (in basic) control the 8 relays.
- 8 optically-isolated inputs for easy direct interfacing to external switches, photocells, keypads, sensors, etc. Simple "INP" commands read the status of the 8 inputs. Selectable port address. Clean, compact enclosed design. Assembled, tested, 90 days warranty. Price includes power supply, cable, connector, superb user's manual . . . . $159.

ALPHA Products

✓17

ADD $2.50 PER ORDER FOR SHIPPING AND HANDLING
ALL ORDERS SHIPPED FIRST CLASS MAIL.
WE ACCEPT VISA, MASTER CHARGE, CHECKS, M.O.
COD: ADD $2.00 EXTRA.
QUANTITY DISCOUNTS AVAILABLE.
N.Y. RESIDENTS ADD SALES TAX.

79-04 Jamaca Ave., Woodhaven, N.Y. 11421 ☎ Info and order: (212) 296•5916

3.3 million micros were shipped, compared to 1.5 million in 1982. Total sales rose from $5.4 million to $10.5 million, even though the average price of a system fell from $3,825 to $3,375. (U.S. prices are about $400 lower.)

Of the 2.3 million desktops sold stateside, Apple led the way with approximately 700,000, to IBM's 550,000 and Tandy's 335,000. Hewlett-Packard and DEC trailed with 250,000 and 110,000 respectively.

Where are these machines going? Seven of 10, Yankee says, are sold to businesses ranging from Fortune 1000 offices to independent professionals. Scientific or engineering users and high-end home computer buyers each claim 10 percent, and educational institutions take 5 percent.

"Between 1983 and 1984," Yankee analysts predict, "technology will continue to drive desktop computers' price-performance." In other words, micros will keep getting better—hard disks and high-resolution color graphics will become cheaper, and 256K RAM "will become the standard offering"—as prices keep shrinking. The Group "estimates that prices will fall 13.5 percent annually between 1980 and 1986."

For 1987, Yankee forecasts over 10 million desktops sold at an average price of $1,700, bringing the U.S. installed base to over 24 million units.

1983: Grand Totals

If the Yankee Group numbers look impressive, remember that they're only for high-end desktop systems. Counting "personal computers," including home micros, Future Computing Inc. has some even bigger figures.

In 1983, the Richardson, TX, market watchers say, U.S. retail sales of micro hardware and software exceeded $10 billion. IBM led the top ten with $1.5 billion in PC sales, and Apple and Tandy tied for second with about $1.1 billion in sales each. Commodore moved into fourth place, moving $800 million worth of VIC's and 64's.

Ranked fifth through tenth were Hewlett-Packard, Texas Instruments, Atari, and three newcomers—DEC, Victor, and TeleVideo, who pushed Zenith, Altos, and Osborne from the list.

The 1983 numbers are formidable, but they're just the beginning. "Only 7 percent of U.S. office workers and under 10 percent of U.S. households" had computers by year's end, FCI admits; by 1990, the firm's *Personal Computer Industry Report* anticipates micros for 40 percent of office workers and two-thirds of American homes. In short, "the personal computer industry is still in its infancy."

"Sales of personal computers will surpass those of minicomputers by 1985 and mainframe computers by the end of the '80s," the report states, while hardware and software sales will pass the $40 billion mark by 1988. ■

END BYTES

Two 2000's

● A week before Tandy's Comdex unveiling of the top-secret Model 2000, Monroe Business Systems unwittingly provided the year's biggest **COINCIDENCE:** The Morris Plains, NJ, firm introduced a near-identical micro (128K RAM, 640-by-400 resolution, MS-DOS), joining the wave of manufacturers using Intel's 80186. Monroe's name for its machine? System 2000.

● Buying the house brand: According to Talmis, "Home **SOFTWARE** is dominated by hardware manufacturers." Seven micro makers' programs, the research firm says, accounted for 37 percent of 1983's $1.2 billion in home software sales. Microsoft, VisiCorp, and MicroPro teamed for another 10 percent, while Sierra On-Line, Adventure International, and Broderbund claimed 4 percent. The remaining 49 percent of the pie was divided by all other software vendors.

● There was a **HALLOWEEN** treat for Atari owners on CompuServe: In the first message from an employee of Warner Communications Inc.'s struggling subsidiary, public relations man Bill Cabeche went on AtariSIG to reassure users that "WCI and Atari remain deeply committed to the home computer industry and continue to analyze the market very carefully." WCI Vice President Geoff Holmes, Cabeche said, would become a CompuServe regular to answer worried Atarians' questions.

● Another sign that Atari's making a comeback: Schools can now receive Atari micros and Verbatim disks for proofs of purchase from Post **CEREAL BOXES**. The year-long promotion, "Catch on to Computers," will also sponsor "computer learning festivals" in 10 U.S. cities.

● Turning from home and school to the office, Talmis reports that **SECRETARIES** are more computer literate than their bosses. Only 52 percent of companies that provide computer training teach managers or executives to use micros, but 69 percent train clerical workers.

● Oklahoma's **MODEM** blues (see *80 Micro*, November 1983, p. 285) may be over. An Oklahoma City *Oklahoman* story quoted on CompuServe says that the $50 monthly tariff for telecommunications users will end with AT&T's Jan. 1 divestiture; Southwestern Bell's $14 basic service charge will entitle users to connect "anything they like," so long as it's FCC approved, to their phone lines.

● Last month, End Bytes reported that **COMDEX** had grown from two to three shows a year with the addition of Comdex/Winter (April 5–7 is winter?) in Los Angeles. Now The Interface Group has added Comdex in Japan, slated to debut at the Tokyo International Trade Center on March 26–28, 1985.

● It's not news that at least three publishers (led by Wayne Green Inc.) plan IBM PCjr **MAGAZINES**, but one new entry is more specialized still: *MUM*, the MicroPro Users' Monthly, a Larkspur, CA, journal for users of WordStar and other MicroPro software.

● And, with **BOOK** publishers like Dell and Harcourt Brace Jovanovich starting computer magazines, a software superpower is getting into the book business. Microsoft Corp.'s newest division, Microsoft Press, plans 30–35 titles a year, focusing on company products such as Multiplan and MS-DOS. The books will be distributed by Simon and Schuster. ■

Yes, I want to subscribe to 80 MICRO. Send me 12 issues for only $24.97. I understand that with payment enclosed or credit card order I will receive a free issue making a total of 13 issues for $24.97.

☐CHECK/MO ☐MC ☐VISA ☐AE ☐BILL ME

CARD# _____ EXP. DATE _____

SIGNATURE _____

NAME _____

ADDRESS _____

CITY _____ STATE _____ ZIP _____

Canada & Mexico $27.97/1 yr. only US Funds.
Foreign Surface $44.97/1 yr. only US Funds drawn on US Bank.

Please allow 6-8 weeks for delivery

80 MICRO•PO Box 981•Farmingdale, NY 11737

342F8B

Yes, I want to subscribe to 80 MICRO. Send me 12 issues for only $24.97. I understand that with payment enclosed or credit card order I will receive a free issue making a total of 13 issues for $24.97.

☐CHECK/MO ☐MC ☐VISA ☐AE ☐BILL ME

CARD# _____ EXP. DATE _____

SIGNATURE _____

NAME _____

ADDRESS _____

CITY _____ STATE _____ ZIP _____

Canada & Mexico $27.97/1 yr. only US Funds.
Foreign Surface $44.97/1 yr. only US Funds drawn on US Bank.

Please allow 6-8 weeks for delivery

80 MICRO•PO Box 981•Farmingdale, NY 11737

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 73 PETERBOROUGH, NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE

Wayne Green Inc.
80 MICRO®
Subscription Department
PO Box 981
Farmingdale, NY 11737

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 73 PETERBOROUGH, NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE

Wayne Green Inc.
80 MICRO®
Subscription Department
PO Box 981
Farmingdale, NY 11737

Survival Kit for the 80's

Don't program without it.

Technology in transition. Changes in hardware, software, disk systems, printers and modems—the list goes on. It's all happening so fast. How do you survive in the fast-paced computer world?

With the help of 80 MICRO, the largest single source of information for your TRS-80*. It's required reading for those who want to discover:

- •TUTORIALS—To help you become a better programmer whether you're a beginner or an old hand.
- •UTILITY PROGRAMS—Designed to let you get the most from your system.
- •NEW APPLICATIONS—In science, business, and the home.
- •NEW PRODUCT REVIEWS—Straightforward evaluations of state-of-the-art hardware and software that give you the facts before you buy.
- •NEW GAMES—Scores of innovative new games offering a real challenge and lots of fun.

With 80 MICRO in hand you won't just survive. You'll prosper in the Age of the Computer.

Subscribe today. Use the coupon below, the attached card, or call toll free **1-800-258-5473. In NH call 1-924-9471.**

Yes, I want to subscribe to 80 MICRO. Send me 12 issues for only $24.97**. I understand that with payment enclosed or credit card order I will receive a free issue making a total of 13 issues for $24.97.

☐CHECK/MO ☐MC
☐VISA ☐AE ☐BILL ME

Card# _____

Signature _____ Exp. Date _____

Name _____

Address _____

City _____ State _____ Zip _____

Canada & Mexico $27.97/1 yr. only US Funds.
Foreign Surface $44.97/1 yr. only US Funds drawn on US Bank.
Please allow 6-8 weeks for delivery.

*TRS-80 is a trademark of Radio Shack, a division of Tandy Corp.
**This price voids all previous offers.

342F8

80 MICRO • PO Box 981 • Farmingdale, NY 11737

Catching an Improvement

I've made an improvement to Smith Harris' "Catching Rays" program (October 1983, p. 256). In line 1530 the program uses the number 0.39782 radians for the value of the obliquity of the ecliptic. Actually this number, which is slowly getting smaller, won't reach that value until approximately 6000 AD.

The correct present value is 0.409106 radians. This compares somewhat favorably with the value found on p. 93 of the article (0.409191 for 1950). The "Catching Rays" program can be made to function until 4000 AD with the following quadratic function:

```
Obliquity = 0.413648 - 2.12972E - 06*Year
- 7.99123E - 11*Year*Year.
```

Hugh Fairman
334 Springbrook Trail
Sparta, NJ 07871

Bar Code Modifications

I found the bar code articles in the November 1983 issue very interesting. Not having an Epson printer, however, I had to change some of the programs to work on my C.Itoh ProWriter. Here's a sample, using Davey Thornton's "Bars and Stripes Forever" program (p. 104), of how to make such a modification.

First, the print subroutine is changed as follows:

```
770  LPRINTCHR$(27)"P";CHR$(27)
     "T14";CHR$(27)">";
780  QZ$ = STR$(N6)
     :QZ$ = "0000" + RIGHT$(QZ$,
     LEN(QZ$) - 1)
     :PT$ = RIGHT$(QZ$,4)
790  FORM = 1TO8
800  LPRINTCHR$(27);"S";PT$;
820  K = 0
830  FORJ = 1TON1
840  IFK = 0THENK = 1ELSEK = 0
850  FORI = 1TO(R(J)*C2 + C1)*2
855  QP = 127*K
```

```
860  IFPEEK(14312) < > 59THEN860ELSE
     POKE14312,QP
870  NEXTI
880  NEXTJ
890  LPRINT
910  NEXTM
915  LPRINT
920  LPRINTCHR$(27)"A* "Q$" *";
     CHR$(27)"<";STRING$(3,13)
940  RETURN
950  END
```

Lines 770, 780, 800, 920, and 940 are changed to the C.Itoh's graphics commands. Lines 855 and 860 are used instead of LPRINT CHR$(127*K) because the Model I's printer driver doesn't pass some of the codes to the printer. Line 850 is changed to get better print quality, but only if you also change line 620 to: 620 N6 = N6 + (R(I)*C2 + C1)*2:NEXTI. This allows maximum print density.

After some experimentation, I found that the program printed the best-looking bar code when I replaced line 250 with 250 C1 = 2:C2 = 2 and deleted line 260. I also changed line 340 to accommodate the maximum length of 22 characters: 340 IFLEN (Q$) > 22THENPRINT "STRING TOO LONG REENTER":GOTO 310.

David Tenney
241 Randolph Road
Morgantown, WV 26505

Model 4 Hints

Here's how you can get your Model 4, when running on Model III TRSDOS 1.3, to operate at the higher (4MHz) clock speed. At the beginning of your Basic program, type: X = PEEK(16912):X = XOR64:POKE 16912,X.

Also, rather than getting your Model 4 to make sounds via JCL commands, you can make it generate tones in Basic by using the Sound command. The structure of the command is: SOUND tone,duration. The tone may be any value of zero through 7, and duration may be any value of zero through 31 (ex. SOUND 4,22).

Paul Sventek
10052 Harwin Drive
Houston, TX 77036

Upgrading COS

After using Richard Castor's machine-code Save and Load routines ("Cassette Operating System," March 1983, p. 226), I upgraded my system to 48K and found that I needed to put the machine-code routines into higher RAM. Since the modification proved to be difficult, I thought I'd give some assistance to others trying to do the same thing.

The modifications load the routines into RAM above 65421 and change two addresses in each routine to accommodate the jumps.

- Delete line 3.
- Change line 4 to: POKE 16561, 141:POKE 16562,255. This sets the memory at 65421.
- In line 6, change both 127s to 255. This is necessary because of jumps to specific addresses within the machine-code Save routine.
- Change line 7 to: FOR I = - 108 TO - 84: READ X: POKE I,X: NEXT I. This puts the Save routine in 65428–65452.
- Change the 127s to 255s in line 8. This is necessary because of jumps to

specific addresses within the machine-code Load routine.

- Change line 9 to: FOR I = −83 TO −59: READ X: POKE I,X: NEXT I. This puts the Load routine in 65453–65477.
- Replace line 850 with: POKE 16526,148: POKE 16527,255. This points to the first byte of the saved routine.
- Delete lines 855 and 860.
- Replace line 950 with: POKE 16526,173: POKE 16527,255:CLS. This points to the first byte of the Load routine.
- Delete lines 955 and 960.

Hugh Lochrane
USDELMC(IMS)
Box 91
APO, NY 09667

Beyond AIDS-III

Robert Fiorelli's enhancements to his AIDS-III program (Input, August 1983, p. 12) are quite good, but they present a slight drawback. The ASCII code for the up-arrow (91) command key is beyond the range of the 32 (space) and 122 (lowercase "z"). Therefore, you not only lose the ability to return to a previous field when

Continued on p. 40

DEBUG

Killing Bugs

Table 1 (Sort Logic) in my "Sorting in Place" article is incomplete (October 1983, p. 172). Subset 1 under the Unsorted List column should consist of C, H, and A; Subset 2: H, B, and G; Subset 3: E, D, and F.

I also found two bugs that prevent the sort from working properly or mix data in the resulting sorted files. To correct these bugs, first add three characters to the end of line 50 so that it reads:

```
50 IFFB=<S1+1THENB2=
1ELSEB2=FB-S1
```

Second, add an LSET to the middle of line 10030 so that it reads:

```
10030 Y=CVI(RIGHT$(S$(W),2)):
IFY=ZTHENT=Y:GOSUB13000:
LSETF1$=D1$:LSETF2$=D2$:
PUT1,J:MID$(S$(Y),1,2)=MKI$(0):
GOTO10050
```

George Reardon
1450 Ranchero Drive
Sarasota, FL 33582

Don't Drop Your Letters

I enjoyed James Blatt's "In Search of..." article (October 1983, p. 244), but I found a serious bug in the program that causes letters to drop out under certain conditions.

The program works by picking a random spot and testing to see if it can insert the word. It checks the first spot to see if A$(H,V) = 0 or whether the ASCII value of A$(H,V) equals the ASCII value of the current position of W$(n). If the condition is met, then it throws two flags and R$ = A$(H,V).

The program makes A$(H,V) equal the ASCII value of the current position of W$(X). It increments the position count and repeats the test for the next position of W$(X). If the test fails, the program decrements the counter and replaces the original value of A$(H,V) by recalling the temporary variable R$.

Now comes the problem. If two or more letters match, the program drops the first match and replaces the proper letter in the array with zero. This is done in each of the insertion routines. To fix this problem, make the following changes:

- In lines 3050, 3120, 3190, 3260, 3330, and 3400 change E = 0:F = 0: R$ = "0" to FOR W = 0 TO 10: E(W) = 0: F(W) = 0: R$(W) = "0": NEXTW: W = 0.
- In lines 3075, 3145, 3215, 3285, 3355, and 3425 change E = H: F = V: R$ = A$(H,V) to E(W) = H: F(W) = V: R$(W) = A$(H,V).
- In lines 3110, 3180, 3250, 3320, 3390, and 3460 insert W = W − 1 at the beginning of each line and change IF E = H AND F = V THEN A$(H,V) = R$ to IF E(W) = H AND F(W) = V THEN A$(H,V) = R$(W).
- Finally, in lines 3070, 3140, 3210, 3280, 3350, and 3420 insert W = W + 1 at the end of each line.

Jerry Feldstein
648 Hill St.
Santa Monica, CA 90405

Graftrax Fix

I have discovered an error in my "Graftrax Art Palette" article (November 1983, p. 140). The section titled "Running the Program" should be ignored. If followed, it will mess up programs 3 and 4.

I programmed the listings' statements for user key-in after first loading Program Listing 1. To save memory, I frequently used the edit function to push past the 240-character limit imposed by ASCII formatting used in merging programs. Saving Program Listings 3 or 4 in ASCII format deletes all statement line code and command word characters beyond the 240-character limit.

Francis Kalinowski
16 North Alder Drive
Orlando, FL 32807

A Calculating Mistake

I've detected and fixed a bug in Robert Fiorelli's CALCS-III program ("Inside AIDS-III—Part II," April 1983, p. 168). The program doesn't generate totals for the fields specified. I discovered that line 11721 is not following the flow to the total print subroutine (lines 11800–11870), because the dump flag (DP = −1) is set to zero, although the total print flag (TP) is set to minus one.

I fixed this problem by adding a DP = −1 to line 11120, just after the TP = −1:

```
11120 IF ZT>0 THEN FOR I=1 TO ZT:
J=ZT(I): ZT#(J)=ZT#(J)+ZF#(J): NEXT
I: TP=-1: DP=-1
```

Marcio Ehrlich
Praia de Botafogo, 340 gr. 210
22250-Rio de Janeiro-RJ
Brasil

GREAT PROGRAMS AT GREAT PRICES...

PROGRAMS SO GOOD
THAT WE GUARANTEE YOU'LL LIKE THEM!

We choose every program we sell based on quality, and for a very good reason; we GUARANTEE satisfaction with every sale. We take all the guesswork out of mail-order purchases by allowing YOU to be the judge of the product that you buy. Over the last year we've had an incredible 99.5% satisfaction rate, unmatched in the software industry. As well, every program you buy from us is fully supported by both the manufacturer and JMG Software. You can't lose!

SUPREME RULER and SUPREME RULER PLUS

(By JMG.) In this simulation/strategy game you become leader of a struggling country and must deal with your economy, your population, and your army. From 1 to 9 players can each take a country and compete, or you may have computer-controlled countries. A great game!
16K / 32K CASS $18.50 32K DISK $20.50.

(By JMG.) A 48K Extended version of SUPREME RULER. We could take up many pages describing it, but instead we'll quote one of our users: "I have hundreds of dollars in game software from Epyx, Avalon Hill, SSI, etc. RULER is the best of the lot. I love it!!" - G.C., Oneida, NY.
48K, either CASSETTE or DISK $26.50.

STARFIGHTER

(By Adventure International.) An incredible program that combines strategy with the realism of a spacecraft simulator. You get a sharp, out of the cockpit view, and you have very realistic controls at your command.
16K CASSETTE $24.50 32K DISK $26.50.

FLIGHT SIMULATOR

(By Sublogic.) This program turns your TRS-80 into a realistic flight simulator with an out-the-cockpit view of your surroundings. Gives an amazing 3-D view of your "world", with mountains at your side and a runway to land on, as well as other landmarks.
32K DISK $29.50

JMG IS ALSO THE PLACE FOR UTILITIES...

LDOS 5.1

(By LSI.) The number one operating system for TRS-80 Models I and III. Chosen by Radio Shack as a sophisticated alternative to TRSDOS, LDOS is loaded with features, from type-ahead to Model I & III compatibility of data disks, and much more. Supports Model I double density, and non-Radio Shack disk drives as well. Comes with an excellent manual.
Model I or III DISK $114.50.

EDAS 4.1

(By Misosys.) EDAS is the most advanced Editor/Assembler available for TRS-80s. It is a must for anyone doing serious assembly language work, and it is also ideal if you are just starting; as you become more proficient, you will find more use for the advanced features available. EDAS supports assembly from disk, macros, and MUCH more. Model I/III version requires LDOS; TRSDOS 6 for Model 4 version.
Model I/III (LDOS) or Mod 4 (TRSDOS6) $89.50.

SUPER UTILITY PLUS

(By Powersoft.) An extremely useful utility package that combines over 50 functions into one program. It supports almost every DOS, both Model I and III. With so many utilities combined into one in SUPER UTILITY PLUS, this program is a great value.
MODEL I/III DISK $64.50

DISASSEMBLER III

(By Misosys.) An advanced disassembler with many features and extensive capabilities such as direct disassembly from CMD files, full label generation, data "screening" for text areas, and more. Output to screen, printer, tape, or disk. Runs under most DOSes (incl. TRSDOS), on Models I, III and 4.
Model I, III and 4 DISK $34.50

TO ORDER: Please specify Model of computer, Disk or Cassette, amount of memory you have, program(s) wanted, and method of payment. We accept Check, Money Order, VISA, MasterCard, or COD. Please enclose $2.50 for shipping costs. We also accept orders by phone.

OTHER PROGRAMS AVAILABLE:

Z CHESS III	(Adventure Int'l)	DISK $26.50	CASS $24.50
GALACTIC TRILOGY	(Adventure Int'l)	DISK $34.50	
PROJECT OMEGA	(Adventure Int'l)	DISK $22.50	
CAPTAIN 747	(LSI)	DISK $27.50	CASS $29.50

OUR GUARANTEE:

We sell only top-quality software. If, however, you are unsatisfied with a product you have purchased from JMG, you may return it (in good condition) within 10 days for a refund, less a $2.50 handling charge for programs under $50 ($5.00 for programs over $50). We also ask that you send us a letter stating the reason for your return.

JMG SOFTWARE INTERNATIONAL

THE FINE PRINT:

Shipping Charges are $2.50 in North America, $5.00 overseas.
All prices in U.S. Funds.
Please allow 1 to 2 weeks for Personal Checks to clear.
Add $1.50 for COD Charges.
Model III programs will also run on Mod 4 in III mode.

P.O. BOX 598
FALLS STATION,
NIAGARA FALLS, N.Y.
U.S.A. 14303

OR

710 UPPER JAMES ST.
HAMILTON, ONTARIO
CANADA L9C 2Z8
(416) 389-6086

126

See List of Advertisers on Page 211

Continued from p. 38

you enter data, but you also get a useless bracket on the screen.

To fix this, search for line 180 in AIDS-III, line 950 in MAPS-III, line 16 in CALCS-III, and line 9070 in MERGE-III. Then simply change IF IC<CL OR IC>CH THEN... to IF IC<CL OR IC=91 THEN...

Marcio Ehrlich
Praia de Botafogo, 340 gr. 210
22250-Rio de Janeiro-RJ
Brasil

Blast 'Em on the III

I've found a way to change John Beringer's "Star Guard" game (August 1981, p. 116) so that the subroutine, which causes the screen to appear to shake when the defending spaceship is hit, works on the Model III as it does on the Model I.

To make this modification, change line 7600 to:

```
7600 FORA1=1TO10:OUT236,4:
FORA2=1TO5:NEXTA2:OUT
236,0:FORA2=1TO5:NEXTA2:
NEXTA1:PRINT@896,CHR$(30);
```

Robert Pillischafske
2811 Wheaton Ave.
Overland, MO 63114

Model III One-Liner

Here's a nice one-liner: If you sequentially number your Model III disks and insert them sequentially into drive 1 in response to the prompt, this program produces a comprehensive directory of all disk files on any printer. Also, it's easy to modify the program to include system and invisible files.

```
10 FOR X=1TO50: LPRINT:
LPRINT "DIRECTORY FOR DISK
ETTE NO. ";X: CMD"Z","ON"
:CMD "D:1": CMD"Z","OFF":
INPUT "CHANGE DISKETTE
IN DRIVE 1; PRESS ENTER TO
CONTINUE; BREAK TO END";C:
NEXT.
```

Thomas Longstaff
39 Pleasant St.
Waterville, ME 04901

Looking for Help

Our school district is trying to find a serial printer driver for Scripsit on the Model III. Presently, all our printing is done with a Scripsit printer driver on an old Model I. Can someone help us adapt the Model I driver for the Model III?

John Robbins
Monterey School District
P.O. Box 1031
Monterey, CA 93940

I'd like to hear from anyone who can tell me where I can buy Microsoft's EDTASM Plus on either tape or disk.

T.E. Bennett
425 Leisure World
Mesa, AZ 85206

Can someone tell me where I can get a new print head for my Line Printer IV (Centronics parallel interface) that costs less than the $200 Radio Shack wants to charge me?

Julie Petersen
1245 Toledo St.
Bellingham, WA 98226

I'm searching for software that will let me generate straightforward graphs, bar charts, pie charts, and so forth with the simple input of data that is "non-programmer" friendly. Does anyone have any recommendations where I can find this elusive program?

Robert Brickhouse
724 N.W. 61st Terrace
Ft. Lauderdale, FL 33317

I have a Model III and I recently bought a Xerox 615 Memorywriter. I'm wondering if anyone makes an interface for this electronic typewriter. Does anyone know?

John Rack
10425 South Kenton Ave.
Oak Lawn, IL 60453

I have a Model III with the high-resolution graphics board. I recently bought Radio Shack's Pascal and found that I'm unable to use the graphics subroutines that are supplied on the board (Line, Paint, Put, etc.). Has anyone written or translated these routines meant for use with Fortran, into a form that I can call from Pascal?

David Maharry
Wabash College
Computer Center
Crawfordsville, IN 47933

Can anyone supply me with an Assembly listing of a relocatable lowercase driver that works with Phillip Van Praag's lowercase modification ("Lower Cost Lowercase," April 1981, p. 228)? All the ones I've found so far print lowercase characters when I press the shift key.

Robert Moquin
125 Des Tulipes
La Baie, Quebec G7B 4G4
Canada

New User's Groups

80 Micro *frequently receives information about user's groups from all parts of the country. The list below contains current information about the groups; it is arranged in alphabetical order by state and province.*

Contra Costa County
TRS-80 User's Group
984 Hawthorne Drive
Walnut Creek, CA 94596
415-939-5285 or 415-932-8856

Model II User's Anonymous
P.O. Box 523
Southbridge, MA 01550
Contact: Don Palmerino

Dearborn TRS-80 User's Group
18037 Breezeway
Fraser, MI 48026
313-459-9787
Contact: Paul Sockow, President

International Adventure
User's Group
84 Camberley Crescent
Brampton, Ontario L6V 3L4
Canada
Contact: Maurice Dow

Three Rivers User's Group
6530 Covington Road
Fort Wayne, IN 46804

NUTRI-CALC (TM)

AT $350 NUTRI-CALC™ WAS A HOT SELLER . . . SO WE IMPROVED THE PROGRAM, ADDED TO THE DATA BASE, AND DROPPED THE PRICE!

The reason is simple. We want EVERYONE who has access to a computer to use our professional nutritional analysis software.

NUTRI-CALC – DESIGNED BY HEALTH PROFESSIONALS TO LET YOU:

- Analyze diet, menus, and recipes
- Calculate ideal weight and caloric consumption
- Set up schedule for weight loss or gain
- Use interactively to plan exercise, diet and menus

SOME OF THE MANY PEOPLE AND PLACES USING NUTRI-CALC INCLUDE:

- Home Computer Users
- Health Clubs
- Natural Food Stores
- Hospitals
- Clinics
- Elementary and High Schools
- Colleges
- Nursing Homes
- Dieticians
- Physicians
- Nutritionists
- Spas
- Athletic Teams
- Government Health Agencies
- Food Companies
- Restaurant Chains

NUTRI-CALC is the most powerful nutritional analysis package available for desktop microcomputers. Consider these features:

- Comes with a 900 food DATA BASE including popular fast foods with 18 common nutrients per food taken from USDA listings
- Space for 100 additional foods of your choice
- SUMMARY ANALYSIS displayed on the screen in NUMERIC or GRAPHIC format, or directed to a PRINTER for paper copy
- Permits the user to ADD, UPDATE, or MODIFY individual food items, recipes, or menus and ALL nutrient values in the data base
- SEARCH the data base for selected food items or for KEYWORDS, by food CATEGORY from 16 different groups

EASY TO USE PROGRAM, menu driven, self-prompting. Insert the disk and run! NO operating system, additional hardware, or special training is required. Even someone with NO computer experience can be doing computerized nutritional analysis IN MINUTES.

NUTRI-CALC is available on TRS-80 Models II, III (48K), 4, 12 and 16 (requires 1 disk drive). Also on Apple, IBM, Altos and CP/M based computers with 8" disk drives and at least 48K RAM.

ALL FOR THE INCREDIBLY LOW PRICE OF $129!

CALL TODAY TO PLACE YOUR ORDER. NOW IS THE TIME TO MOVE UP TO PROFESSIONAL COMPUTERIZED NUTRITIONAL ANALYSIS.

Quantity and educational discounts available.

Price does not include shipping.

MasterCard VISA

COD OR MONEY ORDER

NUTRI-CALC (tm) is a trademark of PCD Systems, Inc.

PCD Systems, Inc.
P.O. Box 277, Dept. EM 14
Penn Yan, N.Y. 14527
(315) 536-7428 433

See List of Advertisers on Page 211

Imagine.... a letter quality printer compatible to your TRS 80

Introducing the JUKI MODEL 6100

Letter Quality, Daisy Wheel Printer/Driver Software Specifically Developed For Use With The TRS-80™ And SuperSCRIPSIT™ !!!

$699.00

All control codes must be preceded by a <CLEAR>

UNDERSCORE	<CLEAR> <->	UNDERLINES EVERYTHING
BOLD PRINT	<CLEAR> <->	GEE IT LOOKS BOLDER
SHADOW PRINT	<CLEAR> </>	THIS SHADOW IS WILD HU?
1/2 LINEFEED F	<CLEAR> <+>	I LOVE SuperSCRIPSIT(TM)
1/2 LINEFEED R	<CLEAR> <->	$Ca(OH)_2 + H_2SO_4 \rightarrow CaSO_4$
PAUSE PRINTER	<CLEAR> <?>	I HATE PAUSING.
TOP OF FORM	<CLEAR> <>	

Now you can enhance your TRS-80™ Model III word processing system with the JUKI 6100 letter quality printer and driver software package.

Totally compatible with the TRS™ Model III when used with the SuperSCRIPSIT™ word processor, the Model 6100 is a sophisticated printer with all the advanced features of a higher priced unit. And the JUKI Model 6100 printer driver software package can be added to the TRS 80™ Model 4 system as well.

Utilizing a 100 character daisy wheel drop-in system, the Model 6100 prints bi-directionally at 18 cps, has 10/12/15 pitch, proportional spacing, and a 2k buffer memory expandable to 8k. Operating on a linear induction motor for accurate positioning, the logic seeking Model 6100 performs superscript, subscript, bold/shadow printing, double strike, underlining and has graphic capabilities.

Also, the Juki printer is lightweight and conventional in design, has low noise level, is easy to maintain, and features a self-testing function. And the printer driver software is available on diskette for either parallel or serial interface.

But the dream is not over...with a low price tag of $699.00 and driver software diskette for less than $25.00 you'll have top quality performance for a remarkably affordable price.

JUKI® AMERICA

521

JUKI INDUSTRIES OF AMERICA, INC. CONTACT YOUR JUKI DISTRIBUTOR FOR THE DEALER LOCATION MOST CONVENIENT FOR YOU:

ACORN DATA PRODUCTS
7304-L South Alton Way
Englewood, CO 80112
303/779-6644
Serving:
MT, WY, CO, UT, NM

BUTLER ASSOCIATES, INC
82A Winchester Street
Newton, MA 02161
617/964-5270
Serving:
ME, NH, VT, CT, RI, MA

COMPUTER SERVICES INTERNATIONAL CORP.
560 Sylvan Avenue
Englewood Cliffs, N.J. 07632
201/569-6300
Serving: METRO NY, E. PA, NJ

GENTRY ASSOCIATES, INC.
7665 Currency Drive
Orlando, FL 32809
305/859-7450
Serving:
TN, NC, SC, MS, LA, AL, GA, FL

INFORMATION SYSTEMS, INC.
2420 E. Oakton Street, Unit K
Arlington Heights, IL 60005
312/228-5480
Serving:
WI, IL, MN, IA, MO, NE, ND, KS, SD

INTERNATIONAL BUSINESS SYSTEMS CENTER
7023 Little River Turnpike
Annandale, VA 22003
703/750-3882
Serving: MD, DE, DC, VA

OSSMANN COMPUTER TECHNOLOGIES, INC.
6666 Old Collamer Road
E. Syracuse, NY 13057
315/437-6666
Serving: UPSTATE NY

SIGMA DISTRIBUTING
2110 116th Ave. N.E.
Bellevue, WA 98005
206/454-6307
Serving:
WA, OR, ID, AK

SOUTHERN MICRO DISTRIBUTORS
8708 Royal Lane
Irving, TX 75063
214/258-6636
Serving: TX, OK, AR, LA

STAR-TRONIC DISTRIBUTOR CO.
23976 Freeway Park Drive
Farmington Hills, MI 48024
313/477-7586
Serving: MI, IN, OH, KY, PA, W. PA, WV

TECHNOLOGY MARKETING CORP.
2300 Valley View Lane
Suite 109
Dallas, TX 75234
214/243-7994
Serving: TX, OK, AR, LA

VITEK
930-G Boardwalk Avenue
San Marcos, CA 92069
619/744-8305
Serving:
S.CA

WESTERN MICRO TECHNOLOGY
10040 Bubb Road
Cupertino, CA 95014
408/725-1660
Serving: N.CA, NV, AZ

NATIONAL HEADQUARTERS:
JUKI INDUSTRIES OF AMERICA, INC.
DA DIVISION
299 Market Street
Saddle Brook, NJ 07662
201/368-3666

WEST COAST:
JUKI INDUSTRIES OF AMERICA, INC.
CALIFORNIA DIVISION
3555 Lomita Boulevard
Torrance, CA 90505
213/325-3093

The Juki 6100: A Printer with Personality

by David Dalton

edited by Lynne M. Nadeau

The Juki 6100 might be the best daisy-wheel printer in its price range. It's smart, looks durable, and has plenty of features, including true proportional spacing. It comes with a 2K buffer and is compatible with the Diablo 630. The print quality is excellent.

Photo. The Juki 6100 daisy-wheel printer.

Description

This is a big printer: 20½ inches wide, 18 inches deep, and 6 inches high (see Photo). It weighs 31 pounds. Strong rubber feet cut down on vibration, and the foam-lined case reduces noise. The Juki 6100 is one of the quietest daisy-wheel printers I've seen, rated at 62 decibels. It won't shake the table.

A 2K data buffer is standard and expandable to 8K of memory. Memory chips list for $28.02 for each 2K. A serial interface is available as an option for $59, and the bidirectional tractor lists for $149.

The Juki uses Triumph-Adler daisy wheels, available in many typewriter stores for about $20. You can get these wheels in more than 20 type styles in 10, 12, and 15 characters per inch (cpi) as well as proportional and foreign language sets.

The Juki 6100 prints at 18 characters per second (cps). It's bidirectional and logic-seeking, which makes it faster.

This printer uses a linear motor. You'll find no belts and pulleys inside. A magnetic field across a heavy metal rail moves the carriage. Juki says this gives better print positioning and longer service.

You'll need IBM Selectric II ribbons, which cost $3.50 or less for a single-strike film ribbon. The fact that daisy wheels and ribbons are available from a variety of suppliers, usually locally, should make this printer easy to own.

One Courier daisy wheel and one ribbon are included. The printer comes with a 90-day warranty. Juki's 13 regional distributors handle service, and they promise a fast turnaround on repairs.

The Juki distributor in my area received my check on a Monday, and the printer arrived the following Thursday morning. I had it up and running in a few minutes without a hitch. It connects to your TRS-80 with the standard Radio Shack cable. The 25-page manual gives clear instructions on unpacking and setting up the printer, installing the daisy wheel and ribbon, and so on.

You need to set 10 DIP-switch pins under the lid. The printer comes with all the pins in the off position.

I had to move two switches: one to tell the printer that I was using cut sheets rather than continuous-feed paper and another to tell it to trigger an automatic line feed on receiving a carriage return. The manual clearly explains the function of these switches, and they are readily accessible.

Features

This printer does everything I want a printer to do. To load a sheet of paper, place it behind the roller and pull the paper-bail lever forward. The paper-bail lever trips a switch. The machine makes a soft whir and pulls the paper in automatically, always stopping at just the right spot.

Nine times out of 10, the paper goes in straight and needs no adjustment. If the paper does need straightening, you release a lever and straighten it as you would on a typewriter.

The word processor I usually use is Scriplus, Powersoft's upgrade to Scripsit. Because Scriplus lets you embed printer control codes in text, You can obtain all the Juki's features except proportional spacing. Any word processor would work well with the Juki as long as it permits embedded control codes.

The printer's front panel has four switches and three lights. A slide switch selects 10, 12, or 15 pitch and proportional spacing. Three membrane switches are marked Reset, Pause, and Form Feed. The lights are marked Power, Ready, and Check.

Assume you want to print a document that's two pages long. The Juki prints the first page and ejects it, then waits for you to insert the next sheet of paper. When the paper is in place, press the reset switch to resume printing.

If the two pages contain fewer than

2,048 characters of data (more if you order a bigger buffer), your computer doesn't have to wait for the printer to finish. And even without an automatic sheet feed, printing a long document isn't too tedious. You don't even have to touch the platen knob because a motor turns the platen.

I do feel that the reset switch should be labeled Resume, because I expect a reset switch to clear the buffer and return all the printing parameters to their defaults. But the Juki desperately wants to protect your data. You can't clear the buffer with the reset switch. You can clear the buffer only by sending control codes from the computer, or by turning the printer off and back on again.

A Juki technician told me it's not harmful to turn the machine off and on, but I don't like it. And sending the control code might not be easy if you have a file in your word processor.

The technician, by the way, was easily accessible by phone and very helpful, as was an executive in Juki's main office who answered some of my questions. I think you can expect good support from Juki.

Control Codes

The manual lists almost 50 special control codes. You execute them by sending an ESC, or ASCII 27, followed by a code.

One special function is a graphics mode, which I have not yet explored. The Juki 6100 spaces horizontally as little as 1/120 inch and vertically as little as 1/96 inch. It also feeds the paper backward. You'd need a lot of help from your software to take advantage of the graphics mode.

Other control codes set the tabs, margins, and page length. You can turn automatic underlining and bold printing on or off.

A shadow print option gives a nice effect. The printer types the character, then offsets the print head 1/120 inch and types it again. Shadow print is very nice for highlighting text.

You can also use control codes to execute all the functions handled by the hardware switches. The manual does a good job of explaining the control codes, and enough of them exist to give you full control over the printer.

Juki's advertising does not mention that this printer's control set is compatible with the Diablo 630. But they say they've tested Diablo software drivers with the Juki and the printer works fine. Juki has prepared a printer driver for SuperScripsit, and Juki distributors supply it.

Proportional Spacing

With the Juki 6100, you can have proportional type just by buying a proportional daisy wheel. However, printing proportional type and justifying proportional type are two different matters.

Not all word processing programs support justification of proportional type. SuperScripsit and Newscript do. I tested Newscript with the Juki, using the Diablo-compatible option, and it worked. But for proportional justification with Newscript you must buy Newscript's daisy-wheel proportional option.

I found that SuperScripsit works with the Juki, using the Radio Shack Daisy Wheel II driver, as long as you don't try to use proportional mode or try anything too fancy. I recommend that you get the Juki software driver if you plan to use SuperScripsit, or you're not taking full advantage of this printer.

Evaluation

I am very satisfied with this printer and recommend it to anyone in the market for a daisy wheel. I studied three other daisy-wheel printers under $800 and came to the conclusion that the Juki 6100 was by far the best choice.

It's quiet, pleasant to use, and has a good service network. Wheels and ribbons are cheap and easy to find. It even has a nice personality. When you press the reset switch, the Juki 6100 whirs and clicks much like R2D2. ■

Juki 6100

★★★★½

Juki Industries of America Inc.
299 Market St.
Saddle Brook, NJ 07662
Centronics Parallel Model $699

Easy to use?	★★★★½
Good docs?	★★★★½
Well-made?	★★★★★
Does the job?	★★★★★

More Model 4: 6.0 Plus

by John B. Harrell III

6.0 Plus is a powerful enhancement package to the Model 4's TRSDOS 6.0 operating system. It thoroughly augments the Model 4's already strong disk operating system, making the combination extremely powerful. If you're satisfied with TRSDOS 6.0 and don't need DOSPLUS compatibility, this feature-loaded package is a great buy for a little money. I recommend it as an addition to every Model 4 owner's library.

The utilities provided are standard DOSPLUS 3.5 utilities that work exactly as specified (see Table 1). 6.0 Plus also provides enhancements to the Basic interpreter that let you access features currently found in most Model I/III DOSes but missing from the Model 4's MBasic (see Table 2).

6.0 Plus comes with a distribution disk that contains all the DOS enhancements and TRSDOS 6.0. The disk is ready to boot and back up, something I highly recommend. The distribution disk doesn't have the Basic interpreter enhancements installed yet, but the accompanying manual provides instructions on doing so.

The manual is well prepared and easy to read. It clearly and concisely explains each of the DOS's extensions, using good examples to illustrate their functions.

The Enhancements

Other than changing the system calls to the Model 4's SVC calls, I suspect that little was done to change 6.0 Plus's utilities from their DOSPLUS counterparts. This has the advantage of providing utilities that are relatively free of bugs.

Each utility listed in Table 1 is commonplace today and exists in several other forms under different operating

systems. However, never has so much power been bundled into one package for such a low price. Micro-Systems Software could sell DISKZAP and DISKDUMP alone for the price of this package.

Utility	Description
DIRCHECK	Checks the integrity of the disk's directory, reports errors, and can automatically correct most errors.
DISKZAP	Provides capability of displaying, modifying, copying, verifying, and printing sectors or tracks, and formatting individual tracks on the disk.
DISKDUMP	Provides features similar to DISKZAP for files.
Map	Provides a file-by-file allocation map of the disk.
Restore	Reclaims files that you have removed from the disk. Restore is subject to certain limitations.

Table 1. DOS utilities.

Feature	Description	
DI pln,nln	Deletes the present line number (pln) and inserts the Basic statement at the new line number (nln).	
DR pln,nln	Deletes the present line number and inserts the Basic statement at the new line number. Renumbers all references to the old file.	
DU pln,nln	Duplicates the line at the present line number to the location specified by the new line number.	
Shorthand	*Immediate Commands:*	
	Up arrow	Lists the preceding line of the program.
	Down arrow	Lists the next line of the program.
	Shift/up arrow	Lists the first line of the program.
	Shift/down arrow	Lists the last line of the program.
	;	Lists the last line of the program.
	/	Lists the last line of the program.
	.	Lists the current line of the program.
	,	Edits the current line of the program.
	Abbreviated Statements:	
	A	Auto
	D	Delete
	E	Edit
	G	GOTO
	I	Input
	K"	Kill
	L or L"	List
	N	Name
	R or R"	Run
	S"	Save
	!	System
"REF",opt,opt,	Performs a cross-reference of selected Basic information.	
"SR",sexp,rexp	Performs global search and/or replace.	
"SORT"	Sorts any data type Basic array.	
INPUT @	Allows easy input of string data from anywhere on the screen and complete control over input format.	
Label Addressing	Offers full use of names as labels within Basic programs. Names can include Basic keywords.	
OPTION <c>	Option S allows entry of Basic programs in the format compatible with Model III Basic. Option L restores the interpreter to the normal Model 4 mode.	
"RESOLVE"	Reads a Basic source text and resolves all named references into line numbers.	
Error Messages	Adds a detailed error message display.	

Table 2. Basic interpreter enhancements.

The Basic interpreter enhancements are powerful. Microsoft Basic (MBasic) is already renowned as the industry standard for microcomputers. To add to this superior interpreter, the routines listed in Table 2 change the interpreter to operate in the manner of the major DOSes produced for the Models I and III.

Other Features

In addition to many favorite shortcuts, 6.0 Plus also provides other options. The machine-language sort facility is the most powerful. It allows sorting of up to 10 key arrays in ascending or descending order specified individually by array.

You can also specify up to 30 tag-along arrays. These arrays don't participate in the sort comparisions, but the elements tag along with the key array elements whenever the program must make a swap.

For example, assume that you have a videotape movie-title manager with the following data structure: The variable TI$ contains the movie title, FM$ contains the tape format, NU contains the tape volume number, MY contains the movie number on a tape, and CA$ contains the type of movie. The following Basic code sorts the movie list into alphabetical order categorized by tape format:

```
10 MAX = <number of movies>
20 DIM TI$(MAX),FM$(MAX),NU(MAX),
MV(MAX),CA$(MAX)
   ...
900 SYSTEM "SORT",MAX,FM$(1)+TI$,
NU,MV,CA$
```

One feature of the Basic enhancement patches the error display capabilities. The changes aid immensely in troubleshooting a program. The full-length error message appears followed by the program line with an arrow pointing to the offending part of the statement.

The most incompatible facet of MBasic is its requirement for delimiting keywords with blanks or other suitable delimiters. This is disturbing because of the overwhelming quantity of Model III software that you must now relegate to the slow Model III mode. Compressed Basic programs aren't portable under MBasic.

6.0 Plus has two extra alternatives

to the Option statement, S and L. They let you place the Basic interpreter in the long (Model 4) mode or the short (Model III) mode. When in the short mode, the program automatically inserts blanks around the keywords in any program line it reads. If you save a program to disk using the A option under Model III Basic and reload it with MBasic using the Options S mode, the program is already converted to run on the Model 4.

The Option statement doesn't automatically change PRINT@ addresses, tab values, or PEEK/POKE statements to the correct values. You still have to do this by hand, but this function removes the majority of the conversion effort.

Conclusion

I have used this system thoroughly and I could do no better for the money spent.

I noticed only one problem with the enhancements to the Basic interpreter. When you're in the Option L mode and enter a code segment such as the code on the left below:

Keyboard Input	Internal Program
10 FOR I=1 TO 10000	10 FOR I=1 TO 10000
20 NEXT I	20 NAME EXT I

the interpreter converts it to the code listed on the right.

This is one of the disadvantages of the shorthand notations in Table 2. The Micro-Systems Software enhancement sees the special abbreviated statement if it is the first character of a program statement, and automatically converts it to its appropriate keyword format.

The 6.0 Plus software enhancements are powerful. If you remember to avoid situations such as the one described above, you'll have no problems with this excellent product from Micro-Systems Software. ■

6.0 Plus

★★★★☆

Micro-Systems Software Inc.
4301-18 Oak Circle
Boca Raton, FL 33431
Model 4
$49

Easy to use?	★★★★☆
Good docs?	★★★★☆
Bug-free?	★★★★★
Does the job?	★★★★★

MicroEditor II Versus Scribe

by Eric Grevstad
***80 Micro* staff**

The Model 100's Text program is admirable for putting words on the screen, but it's rotten at putting them on paper. The portable's unformatted, single-spaced Print and Shift Print commands have inspired a dozen patch programs, ranging from simple print formatters to elaborate word processors.

Scribe, one of the former, is reasonably powerful and inexpensive; MicroEditor II, one of the latter, is practically unusable.

Dear A$

MicroEditor offers two sophisticated features. One, the ability to print a series of form letters, works nicely. Insert underline (shift/minus key) characters in text where appropriate, and the program replaces them with words stored in an auxiliary file and read as Basic data statements.

However, MicroEditor's other good idea—a help menu for those who lose the manual—isn't much help. Guardian consists of two screens giving vague advice, but few instructions ("Use Search & Replace for quick editing").

The Search and Replace function is worse than Text's built-in Find and Paste commands; as soon as you replace an item, the program returns you to the menu. You must reinitiate a search and specify the item again, rather than proceed to its next occurrence.

And MicroEditor is awkward to use. Instead of simply pressing the enter key, you must end paragraphs with Grph E, which is almost impossible to type with one hand. This keeps different paragraphs on the same screen line—extremely puzzling to LCD readers.

The main menu's cursor only moves down, even if you press the up-arrow key, and you return to the top after every option. After viewing Status, which lets you set pagination, page length, and left and right margins, you must scroll through the menu again to the next choice, Print.

MicroEditor II sends copy to either the printer or LCD, and offers justification, line numbering, and page headings, but you're still stuck with single-spaced text.

Worst of all, the program takes a whopping amount of memory. It's big by Model 100 standards anyway (4,664 bytes), but it also prints a Text file by copying it, very slowly, into its own memory (CLEAR 9500 in the first line).

I have a 24K Model 100; repeated tries to use MicroEditor with as many as 10,000 bytes free, not counting the program or text file, produced out-of-memory errors. In effect, MicroEditor II is a 15K program.

A Modest Success

Scribe, on the other hand, might even be useful to 8K Model 100 owners. It's only 2,031 bytes long, and prints files from cassette as well as from RAM—allowing infinitely long printouts, since it reads one line from tape at a time and lets you chain files with different page numbers.

While it works well with daisy-wheel printers, Scribe shines with dot-matrix machines. Its rather homely (cryptic abbreviations and all caps) format menu supports double-strike and emphasized as well as normal printing. Its embedded command vocabulary lets you send printer codes for these or other functions, such as underlining and italics, from within text.

Scribe's embedded codes are much more powerful than its stripped-down menu. You can set and alter top, bottom, and left margins and line width; indent text; start a new page; center or skip a line or a number of lines; and turn justification on or off by adding commands such as .ce or .lm12 to your prose.

Except for line spacing—though, unlike MicroEditor, Scribe lets you

Shape your TRS-80 to communicate with any computer you want.

NEWS FLASH!
OMNITERM wins first place for data communications software!

Omniterm is the most flexible, powerful terminal program you can buy. Omniterm lets you adapt your TRS-80 to communicate with 99.9% of the world's computers. Your company's mainframe, for example. Or any other personal computer, time-sharing computer, or communications service.

Omniterm overcomes incompatibilities in screen formats, baud rates, character sets, control codes and file transfer protocols. Seven complete translation tables let you change any character, for complete compatibility of all input and output devices. Omniterm is so flexible, users have even set up their ASCII-coded systems to communicate with EBCDIC-coded systems.

You can send all ASCII characters, even those that aren't on your keyboard. Reformat your screen to neatly accommodate any line length. Run your printer while you're sending or receiving data. And even review data that's scrolled off the top of the screen.

Omniterm's well-thought-out design makes it easy to use. You can get a status display of all functions while on line to tell you what's going on, and make any changes at the same time. You can create a special file of your settings to make it easier next time. You also get X/Y cursor control, single keystroke sign-on and auto-dialing. Even a phone directory. And lots more.

You don't have to be a computer expert to use Omniterm. Just spend a day with what the reviewers call "the best manual in the business." Then if you need help, just call, write, or contact us via CompuServe, Delphi, or Source.

Omniterm is the proven terminal program. The program thousands of people have used successfully. And the one the editors call the "top program available" (Byte, 80-Micro, Infoworld, etc.)

Omniterm comes complete with sample setting files, conversion utilities, a practical text editor, seven translation tables, and a 76-page manual with index.

Available at leading dealers, or prompt shipment on direct phone and mail orders. Order Omniterm, for a super-smart TRS-80 that's putty in your hands.

Only $95 for TRS-80 Models I, III or 4 (32K memory, one disk minimum). $175 for Models II, 12 or 16 (64K memory minimum). (In Mass., add 5% sales tax.) MasterCard, VISA, and C.O.D.

IBM PC version coming soon.

DEALER INQUIRIES INVITED.

Telephone: (617) 852-0233
CompuServe: 70310,267
Source: TCA818
Delphi: Lindbergh

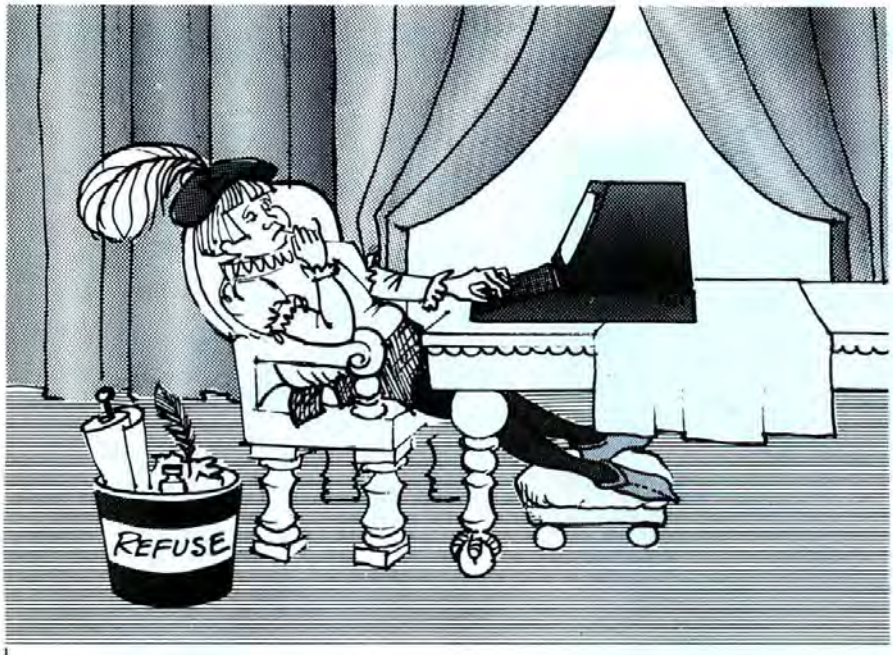
Omniterm

Lindbergh Systems

49 Beechmont St., Worcester, MA 01609

135

NOW AVAILABLE FOR THE IBM PC!



choose single, double, or whatever at startup—you can create or change nearly any formatting effect while you write. You can even turn Scribe's formatting off (.fo off), which is the only way to make the program recognize a carriage return without a special command.

This is the quirk that took me longest to learn: With formatting on, pressing the enter key has no effect. You must press the enter key, type .br or .sk (to break for a new printing line or skip a line between paragraphs), and press the enter key again.

The manual, otherwise adequate, doesn't specify that it's best to use .fo off for, say, the date and address at the top of a letter, and then muddle through with paragraph commands.

Also, it's safer to use the space bar rather than the tab key to indent the first line of a paragraph. Scribe indents properly, but counts a tab as only one character, letting the first line run past the other lines' right margins.

Given these foibles, though, Scribe performs swiftly and skillfully. Users without tractor-feed printers can order a pause for paper insertion between pages.

Finally, there's Scribe version 2.0, released at press time: Not only is the manual a little better, but the new program supports headers atop pages as well as centered numbers at bottom.

A new embedded command, .ke, lets you pause printing while entering text directly from the keyboard—not as elegant as MicroEditor's auxiliary file, but more than adequate for addressing a few form letters (the opening menu lets you tell Scribe to print a file several times). Used alone, the new feature turns your 100 and printer into a typewriter, if you can make a mental note to press the enter key every time you reach the proper line length.

Three Choices

Neither of these programs is the best Model 100 word processor I've seen; that would be Portable Computer Support Group's Write+, which combines copious embedded commands with a do-file full of changeable defaults (see review, December 1983, p. 65). Compared to Scribe, though, Write+ is a little big (4K), and available only as part of a $90 package.

If you have, perhaps, a 32K machine and must print many copies of a short form letter, MicroEditor II might be acceptable. Otherwise, Scribe offers more features in less memory, is much more flexible once you learn its idiosyncracies, and at $25 is a bargain. ■

MicroEditor II

★★

Alphaware Inc.
2100 S. Boulevard
Edmond, OK 73034
Model 100
$49.95

Easy to use?	★★☆☆☆
Good docs?	★★★☆☆
Bug-free?	★★★★☆
Does the job?	★☆☆☆☆

Scribe

★★★★

Chattanooga Systems Associates
P.O. Box 22261
Chattanooga, TN 37422
Model 100
$24.95 + $2 handling

Easy to use?	★★★½☆
Good docs?	★★★½☆
Bug-free?	★★★★☆
Does the job?	★★★★☆

Good Words, Cheap: The Word Machine

by R. Walter Steur

The Word Machine is the Model I/III word processing sleeper of the year! If you're longing for a full-function word processor but can't justify the cost, The Word Machine is your solution. Pel-Tek makes it possible to have full-feature convenience at a reasonable cost.

Overview

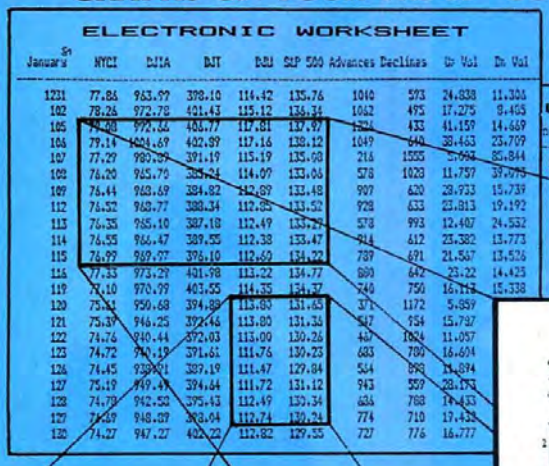
The Word Machine is a line-oriented, machine-language program intended for preparing letters and short documents. Pel-Tek distributes the program on a nonsystem disk that appears to be compatible with most DOSes.

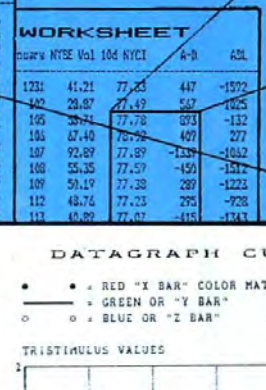
Continued on p. 236

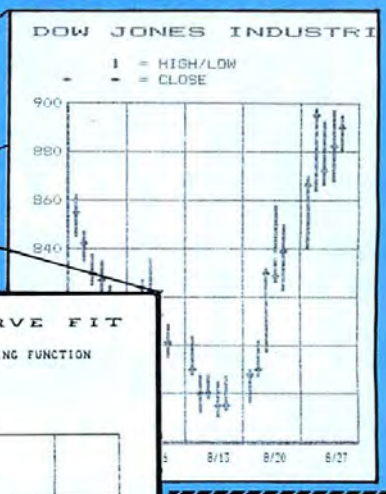
DATAGRAPH™

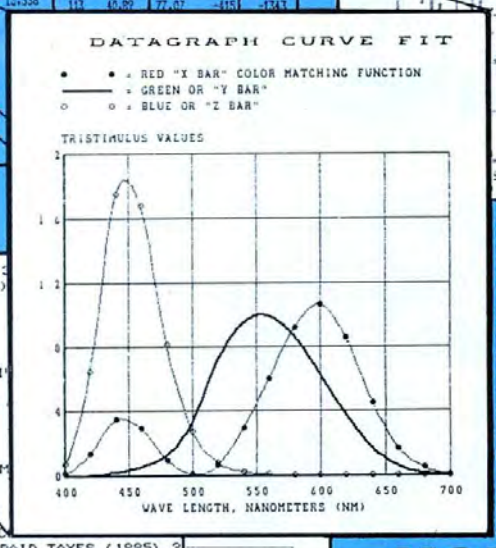
PRINTER GRAPHICS PROGRAM

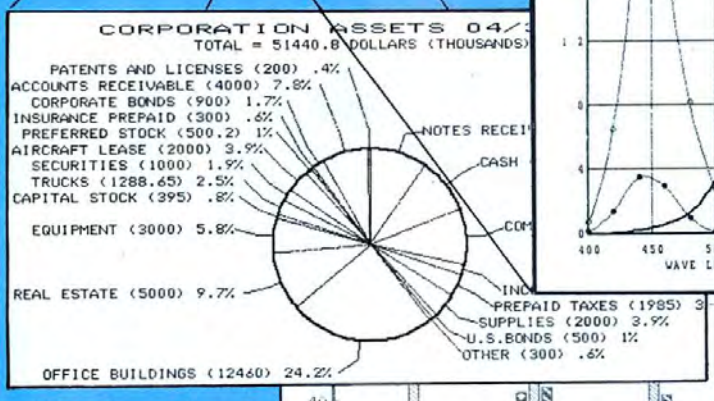
TRANSFORM YOUR VISICALC™ FILES INTO HIGH-RESOLUTION CUSTOM GRAPHS ON YOUR TRS-80™ COMPUTER AND GRAPHICS PRINTER.

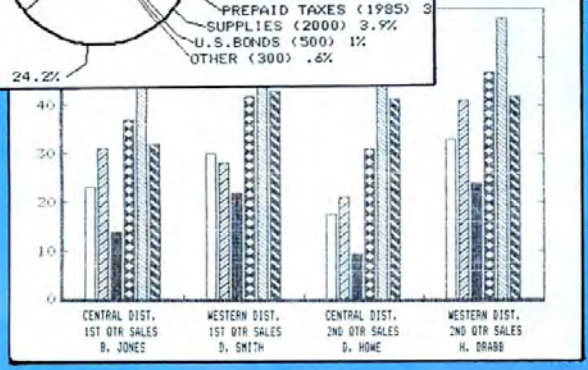












NEW! MODEL II/12/16 VERSION

MODEL I, III
$79.95

PIE CHART OPTION: $34.95

MODEL II/12/16
$129.95*
***INCLUDES PIE CHARTS**

* HIGH RESOLUTION - 60 x 72 data Points/inch. * LARGE DATA CAPACITY - 1000 Input Data Points per graph. * SELECTABLE GRAPH SIZES - From 1" sq. to 7" x 24" * STANDARD DATA SOURCE - Plots Data from VISICALC or USERS OWN PROGRAMS using the DIF™ Standard Format. * GRAPH FEATURE SELECTION - Fill out Pre-formated Form on VISICALC screen or in users own program. * MINIMAL ENTRY REQUIREMENTS - Enter only name of Datafile and location therein of data to be plotted. * MULTIPLE FUNCTION GRAPHS - Plots over 10 Data Sets per graph. * DATA SYMBOLS - Plots data with user composed symbol shapes. * DATA INTERPOLATION - connects data points with user composed line shapes. * LINE/SYMBOL LIBRARY - Plots each Data Set with different line/symbol shape chosen from 12 line library. * CUSTOM LINES AND SYMBOLS - Has interactive screen-graphics program for composing symbol shapes. * AUTO SCALING - Selects scale values for ease of graph interpretation. User adjustable Mantissa Table. * GRID SELECTION - Prints selectable number of vertical and horizontal grid lines. * CALENDAR SCALE - Optionally prints names of month on horizontal scale. * CURVE SELECTION - Can Mix Scatter, Line, Curve-Fit, Stairstep, Bargraphs, Pie Charts. * OPTIONAL MIN/MAX VALUES - Extends graph beyond the values of the Data Sets. * DATA SET DESCRIPTIONS - Prints text descriptions of each Data Set in graph legend. * TEXT ENTRYS - Prints graph title, axis labels, and date on graph. * USER FRIENDLY - Checks validity of input data and displays cause of errors. * COMPLETE DOCUMENTATION - Comprehensive 75 page Users Manual with examples covering data preparation, graph feature entry, composing lines and symbols, and technical notes.

USER REQUIREMENTS

COMPUTER
- TRS-80 MODEL I, III 48K
- TRS-80 MODEL II, 12, 16
- LNW80, LNWII, MAX-80

DOS
- TRSDOS 1.3, 2.3, 2.0, 4.2
- NEWDOS, NEWDOS/80
- DOSPLUS 3.4/5, LDOS 5.1

DISK DRIVES
- SINGLE DRIVE
- DUAL DRIVE (PREFERRED)

GRAPHICS PRINTER:
- MX-80 GRAFTRAX, OR GT+
- MX-100, FX 80/100
- LP VIII, DMP 200-2100, 120
- NEC 8023 A-C, C.ITOH 8510
- IDS 460/560, 480, 80/132
- OKIDATA 82/83, 92/93, 84
- GEMINI 10/15

TO ORDER: Send check, purchase order, or request for COD shipment. Specify Computer and Printer Type. Include $2.50 for postage and handling. Calif. residents add 6% tax.

MICRO SOFTWARE SYSTEMS • MICROPLOT, INC.

DEALER INQUIRIES WELCOME

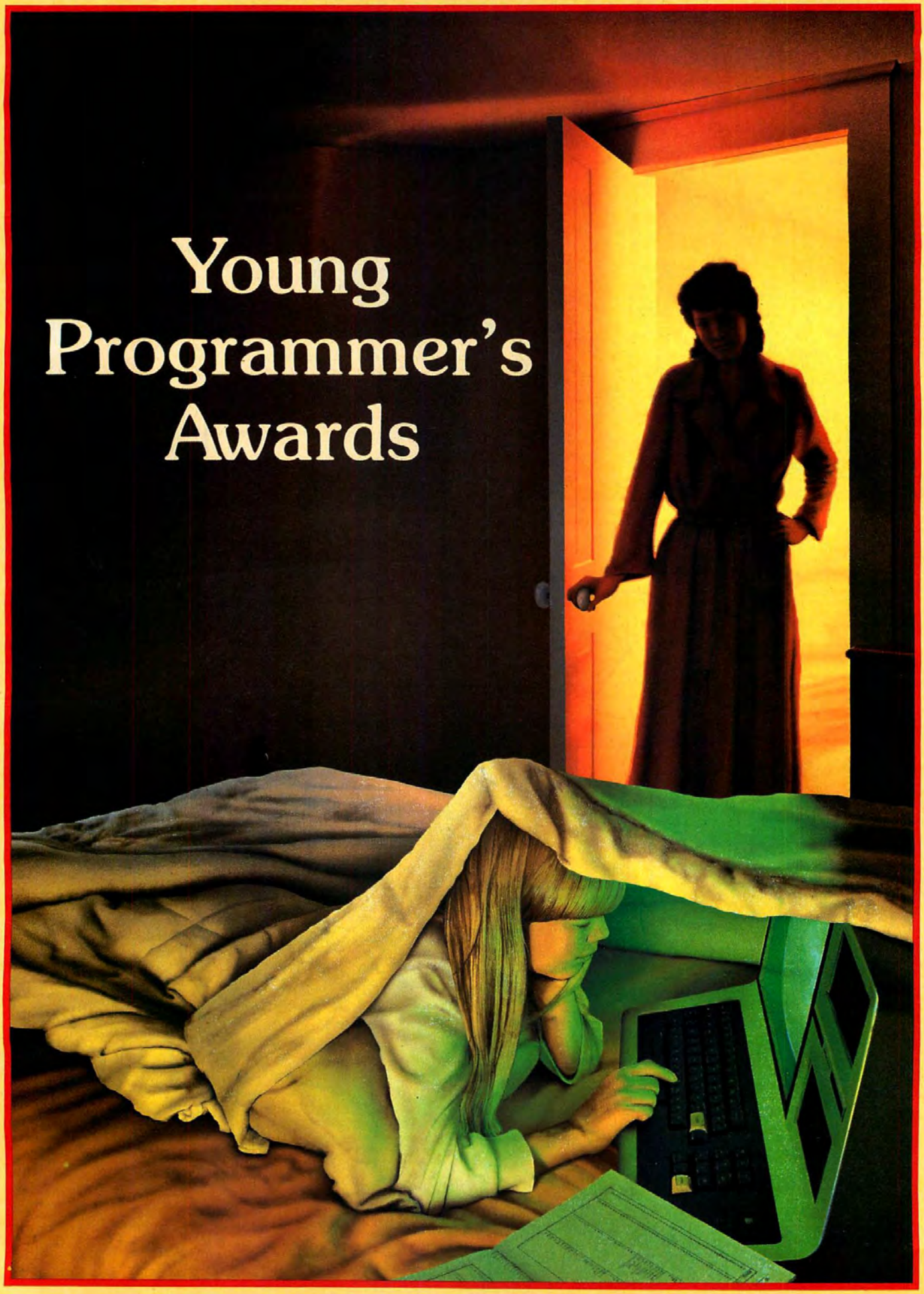
1815 SMOKEWOOD AVE. • FULLERTON, CA 92631 • (714) 526-8435

526

VISA/MASTERCARD ACCEPTED

VISA MasterCard

Young Programmer's Awards



Winning entries from our young programmers transport you to Broadway, outer space, and then back to Basic.

Besides breaking into the Pentagon's mainframes and depositing money in Swiss bank accounts via overseas telecommunications, kids are writing programs for the TRS-80: games, utilities, graphics—and entering them in our second annual Young Programmer's Contest.

We selected winners from a field of nearly 200 entrants. No easy task, even for our technical staff, who characterized some of the programs as the most sophisticated they'd seen—in any age group.

The judges evaluated all entries on the basis of the following criteria:

Programming elegance. We looked at individual programming techniques: use of advanced commands, including shortcuts in otherwise linear program flow, overall programming logic, helpful comments within the listings.

Documentation. Here we expected a brief form of how-to manual: well-organized, and written to anticipate questions. Left alone with computer and documentation, could the user run the program without difficulty?

Originality. We asked ourselves: Have we seen this kind of program before? And if so, does the programmer handle a familiar concept in a unique and surprising way?

Error-trapping. We wanted user input so well defined that our deliberate attempts to subvert or crash the program failed.

Usefulness. Was a game fun? Could we anticipate using a utility or technique in our own programming?

The Winners' Circle

The grand prize winner is 16-year-old Stephen Roth's Play-Byte, a menu-driven machine-language program. Booting up Play-Byte turns you into an impresario on the Model III: You create stage sets for a play, then draw characters, program their movements, and get the computer to memorize your script.

What distinguishes this entry is that it provides endless possibilities for developing new programs within the main program—computer plays opening and closing—as you like it.

Get Lost!, a CoCo maze program for two players, unique in its double-screen presentation, won first prize for Steve Francis in the 15–18 category.

Illustration by Erick Ingraham

While games were in the majority of entries, young programmers are writing—and winning with—utilities and other applications as well. A case in point is 14-year-old Brian Craft's SINSTEP, first place winner in his age group. His machine-language subroutine steps through Basic programs to make changes at any point during execution.

Mark Kennedy's Model III Adventure Sampler garnered a blue ribbon in the 11 and under category. The scenarios of this adventure game set in outer space include spaceship breakdowns and threatening aliens to contend with.

Second prize in the 15–18 category is Joseph Goldberg's Electronic Inkwell, a user-friendly word processor whose functions rival those of Scripsit. Nathaniel Koch's Pilot+ for the Model II, second prize winner in the 12–14 group, adds commands to a new computer language to make it even more powerful. Eric Bailey, age 11, created a Model 100 bar graph that won him second prize in his category.

Dungeon of Death, a Model III adventure game comprising both mazes and dialogue routines netted third place for Mike Erickson, 17, in the oldest group. Fourteen-year-old Scott Bradley's Haunted Mansion, another third place winner, is a high-resolution thriller for the CoCo. And Tyler Kim took third prize in the 11-and-under category with a flight simulation game for the Model III.

Honorable mention goes to the following: Lloyd Kupchanko, 17, for Fourth Dimension, an arcade-style machine-language game; Raul Acevedo, 14, for his Key Art graphics program; Kim Skidmore, 13, for her matchmaking entry that transformed the parameters of social life at *80 Micro*; and Frank Conley, 11, for his Mad Libs game.

Originally, we'd intended to publish all the winning entries in this issue, but space limitations permit us to publish only three of the highest scorers. Space permitting, we'll publish other winners in future issues.

Further, when we first advertised the rules and regulations for this year's contest, we were still publishing programs for the Color Computer. Since then, we've suspended our CoCo coverage so that, though we've awarded prizes to CoCo entries, we're unable to print the Color Computer winners.

Already, the deadline for our third annual Young Programmer's Contest is less than a year away. To the current winners: Congratulations, and try again. To all other programmers under 18: Go for it! —*S.G.*

Grand Prize

The Play's the Thing

by Stephen Roth

Play-Byte is a menu-driven Model III Assembly-language program that lets you stage plays on your computer screen. I modeled my animated computer play on a real play. First you design backgrounds, construct characters, and determine moves for the characters. Then you manipulate these backgrounds, characters, and moves using the program I wrote (see Program Listing 1). The end result is an animated play.

Play-Byte's main menu appears when you first load the program into memory. It allows you four choices:

1) define characters, background or moves;
2) construct the play;
3) view the action; or
4) return to TRSDOS.

Options 1 and 2 present you with submenus (refer to the Figure.)

Defining Characters, Backgrounds, and Moves

Main menu selection 1 lets you draw the backgrounds, characters, and moves. Choose from the submenu which of these you want to draw, and give it a file name when the program prompts you.

Program Listing 1. Play-Byte, part 1.

LOAD 80

```
00100           ORG     7000H
00110 ;         DISPLAY MAIN MENU
00120 BEGIN     CALL    01C9H
00130           LD      A,0C3H
00140           LD      (41C1H),A
00150           LD      A,3
00160           LD      (9300H),A
00170           LD      HL,LISNUA
00180           LD      (41C2H),HL
00190 INPUT     LD      A,1
00200           LD      (POS),A
00210           LD      A,(PLAG12)
00220           OR      A
00230           JP      NZ,INPT1A
00240           LD      HL,3C00H+147
00250           LD      DE,MMENU
00260           CALL    OPTION  ;DISPLAY MENU
00270           LD      HL,3C00H+147+192        ;MOVE CURSOR
00280           CALL    CURSOR  ;WAIT FOR USER TO MOVE CURSOR
00290           CALL    01C9H
00300           LD      A,(POS)
00310           PUSH    AF
00320           LD      A,1
00330           LD      (POS),A
00340           POP     AF
00350           CP      1       ;WAS FIRST OPTION SELECTED
00360           JR      NZ,INPUT1
00370 ;         IF YES THEN DISPLAY ANOTHER MENU
00380           LD      HL,3C00H+150
00390           LD      DE,MMENU1
00400           CALL    OPTION
00410           LD      HL,3C00H+150+192
00420           CALL    CURSOR
00430           LD      HL,3C00H+598
00440           LD      DE,NAME
00450           LD      A,8
00460           LD      (LEN),A
00470           CALL    BOX     ;GET NAME OF BCK/CHR/MV
00480           LD      HL,(SCRIN)
00490           LD      A,(HL)
00500           CP      '-'
00510           JP      Z,INPUT5
00520           LD      A,(POS)
00530           CP      1       ;IS BACKGROUND
```

Listing 1 continued

In the Background mode, you can draw up to 15 sets. Use the arrow keys to move the cursor. Keys 1, 2, or 3 adjust the speed of the drawing cursor. Press the enter key to draw, and the clear key to erase or move the cursor without drawing. The break key allows you to erase the screen, continue the drawing, or store the background in memory and return to the main menu (refer again to the Figure).

In the Character mode you can define a maximum of 20 players. Enter the name of the character. If you define more than one character, the program asks you for a base character (a previously created character to add to, subtract from, or use as a guide for constructing a new character). If you don't want to recall a base character, press the enter key, and you can create any new character on the clean matrix.

The Move option gives you up to 10 moves in your play. Each move can take up to 125 steps. You need the background and the character as a basis so you know where to define the steps. Enter the character name when prompted. In this mode you can move the character around with the arrow keys; it's a lot of fun!

Move commands are the same as those in the Background mode, except that dashes show where the steps are. Dashes, however, don't show up in the final play. Be careful erasing dashes; always erase starting at the last step you made.

The Key Box

Model III
48K RAM
Assembly Language
Editor/Assembler
Disk Drive

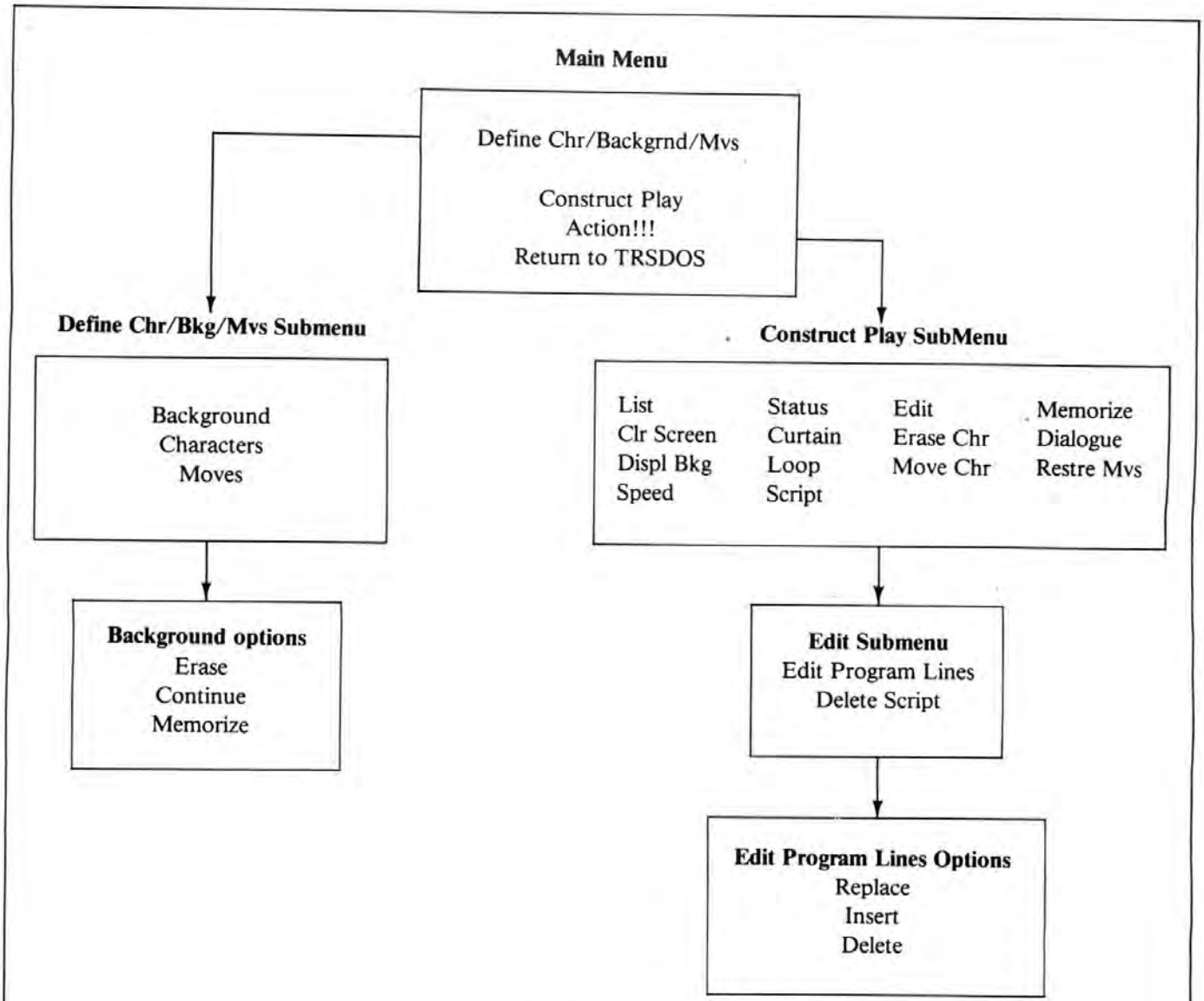


Figure. Play-Byte menus.

Directing the Play

Main menu option 2, Construct the Play, lets you animate the action. A submenu of 14 options appears when you enter this option (see the Figure). All Construct subcommands, except for five, are commands that you can use in the program (or play) you construct.

You can enter the commands described below as program statements. When you're prompted for a background, character, or move, the names are in menu-like format. Press the enter key when the cursor is on the choice you want.

The Clear Screen statement does just that.

The Curtain statement displays the background you request as though the stage curtain is opening. You have to see it to believe it.

The Dialogue statement prints a line of the play's script on the top of the screen. Each time you use this statement it prints the next line of text. If no more text is left, nothing appears.

The Display Background statement displays the background of your choice.

The Loop statement lets you jump to a line for a specified amount of loops. You cannot have more than one loop to the same line.

The Move Character statement is the most important statement of all. It moves the character along the path you previously defined. You're asked for the name of the character and the move that you want to use. Each time you use this statement, it moves the character only one step along the path you defined. This is why it's important to define the number of moves along the path in order to loop to the line containing the Move statement the correct number of times (see Status).

The Restore Moves statement restores the pointer for the move you specify, letting you start a character's move over again.

The Speed command lets you specify the execution rate. Enter a value from 1-9, with 1 being the fastest. You may have as many of these statements throughout the program as you want.

The Erase Character statement erases the character on the screen. It asks you which move the character uses. Use this command when two characters overlap one another. In addition, you can use it when the character finishes making its moves.

Four additional commands execute at the end of the program: List, Status, Edit, Memorize, and Script.

The List command lists the program you entered. It lists eight lines at a time; press any key to continue the listing.

The Status command lists the names of all the backgrounds, characters, and moves you've defined. It also shows the number of steps for each specific move,

Listing 1 continued

```
00540           JP      Z,DEFB1
00550           CP      2       ;IS IS CHAR
00560           JP      Z,DEFC1
00570           CP      3       ;IS IT MOVE
00580           JP      Z,DEFM1
00590 ;         DISPLAY CONSTRUCTION MENU
00600 INPUT1    CP      2
00610           JP      NZ,INPUT2
00620 INPT1A    PUSH    HL
00630           LD      HL,3C00H+147
00640           LD      DE,MMENU2
00650           CALL    OPTION
00660           LD      HL,3C00H+147+173
00670           LD      A,1
00680           LD      (FLAG9),A
00690           CALL    CURSOR
00700           XOR     A
00710           LD      (FLAG9),A
00720           POP     HL
00730           LD      A,(FLAG12)
00740           CP      1
00750           JP      Z,INPT1B
00760           LD      A,(POS)
00770           LD      HL,(EPROG)
00780           CP      1
00790           JP      Z,LIST
00800           CP      2
00810           JP      Z,STAT
00820           CP      3
00830           JP      Z,EDIT
00840           CP      4
00850           JP      Z,MEMOR
00860           PUSH    AF
00870           PUSH    HL
00880           LD      DE,PRGEND       ;IS PROGRAM LONG ENOUGH
00890           LD      HL,(EPROG)
00900           RST     18H
00910           JR      Z,INPT1D
00920           POP     HL
00930           POP     AF
00940 INPT1B    LD      A,(POS)
00950           CP      5
00960           JP      Z,CLR
00970           CP      6
00980           JP      Z,CURT
00990           CP      7
01000           JP      Z,DELET
01010           CP      8
01020           JP      Z,DIALOG
01030           CP      9
01040           JP      Z,DISPL
01050           CP      10
01060           JP      Z,LPLIN
01070           CP      11
01080           JP      Z,MOVCHR
01090           CP      12
01100           JP      Z,RESTOR
01110           CP      13
01120           JP      Z,SPEED
01130           CP      14
01140           JP      Z,SCRIPT
01150 INPT1C    XOR     A
01160           LD      (FLAG9),A
01170           LD      A,1
01180           LD      (POS),A
01190           PUSH    HL
01200           CALL    01C9H
01210           POP     HL
01220           JP      INPT1A
01230 INPT1D    POP     HL
01240           POP     AF
01250           JR      INPT1C
01260 ;         ACTION1
01270 INPUT2    CP      3
01280           JP      Z,RUN1  ;RUN THE PROGRAM
01290 INPUT3    JP      402DH
01300 INPUT4    POP     IX
01310 INPUT5    XOR     A
01320           LD      (FLAG9),A       ;RESET FLAGS
01330           LD      (FLAG10),A
01340           LD      (FLAG12),A
01350           CALL    01C9H
01360           JP      INPUT
01370 ;         MOVE CURSOR
01380 CURSOR    LD      A,(POS)
01390 WAITA     LD      A,(HL)
01400           LD      (MCHR),A
01410           LD      A,143
01420           LD      (HL),A
01430 WAIT0     LD      (COORD),HL
01440 WAIT      CALL    49H     ;WAIT FOR USER INPUT
01450           CP      0DH
01460           RET     Z
01470           CP      1
01480           JR      Z,INPUT4
01490           CP      10
01500           JR      Z,LOW   ;MOVE CURSOR DOWN
01510           CP      91
01520           JR      Z,HIGH  ;MOVE IT DOWN
01530           CP      9
01540           JP      Z,SIDER ;MOVE IT TO THE RIGHT
01550           CP      8
01560           JP      Z,SIDEL ;MOVE IT TO THE LEFT
01570           JR      WAIT
01580 LOW       LD      DE,64
```

Listing 1 continued on p. 166

and displays all the script lines you entered.

The Memorize utility lets you save and load your program to disk. If any DOS errors occur, the screen displays them. Press the enter key to return to the main menu if such an error occurs.

The Script command lets you enter a play script into a buffer. You may terminate each line by pressing the enter key. Press the break key to return to the Construction menu. Each time you use this utility, the program places the script that you enter at the end of the buffer.

A fifth command, the Edit command, lets you edit both the program and the script using its own submenus—Edit Program Lines and Delete Script. The Edit Program Lines option gives you three choices—to replace, insert, or delete program lines.

Main Menu Options 3 and 4

The Action mode runs the program you wrote. Press the break key to interrupt the program and return to the main menu. When the program executes fully, press any key to return to the main menu.

The Return to TRSDOS mode returns you to TRSDOS READY. Once you exit the program, you can never retrieve the information unless you saved it to disk.

Tips

Play-Byte is too long to type into your editor/assembler, which is why I divided it into two listings. Load both assembled programs into memory, and dump them to disk by using this format:

DUMP NAME OF PROGRAM (START = 7000, END = 8690,TRA = 867C)

When you use a menu, move the cursor with the arrow keys and press the enter key when the cursor is on the option you want. Any time you press the break key while in a submenu, you return to the main menu.

When you assign names to backgrounds, characters, and moves, you can't enter two names with the same beginning. For example, don't name one character BOY1 and another BOY2. You can, however, name them as 1BOY and 2BOY. ■

Stephen Roth, age 16, lives at 7725 Silver Fox Drive, Youngstown, OH 44512.

MONTEZUMA MICRO

P.O. Box 2169
Camp Verde (Lizard Flats)
Arizona 86322

✓ 416

WE KEEP YOU RUNNING

COMPUTERS

	Model 4 with 2 drives & RS-232 64K Complete	1495
26-1067	16K Model 4, Pure Radio Shack	849
26-1068	1dr Model 4, Pure Radio Shack	1444
26-1069	2dr Model 4 64K RS232, Pure Radio Shack	1650
26-1080	**NEW!** Model 4 Portable	1529
26-3003	64K Color Computer	340
26-3011	MC-10 Micro Color Computer	99
26-3026	16K Color Computer 2	199
26-3027	16K Extended BASIC Color Computer 2	275
26-3590	Pocket Computer 3	84
26-3601	Pocket Computer 2	169
26-3650	Pocket Computer 4	59
26-3801	Model 100 8K Portable Computer	679
26-3802	Model 100 24K Portable Computer	849
26-4004	Model 12 2-Drive	2599
26-4005	Model 12 2-Drive	3199
26-6004	Model 16B 1-Drive	4249
26-6005	Model 16B 2-Drive	4928
26-6006	Model 16B with built-in 12Mb Hard Disk	5949
26-6050	DT-1 Terminal	599

LNW Model II Computer, 112K, RS-232, TRSDOS Compatible. Complete with CP/M 2.2 and lots of FREE Software 1488

ZORBA Portable, 2 DSDD drives, 80x25, 4Mhz, graphics, 7 green phosphor screen, RS-232, IEEE-488 & Centronics ports, with WordStar, MailMerge, SuperSort, CalcStar 1695

MODEMS

26-1084	Model 4P Modem Board	128
26-1173	DC Modem II	169
26-1174	Acoustic Coupler Modem	127
26-1175	DC Modem 1B	85
76-1005	DC-1200 High Speed Modem	595
76-1009	Auto Dial Attachment for DC-1200	127
	Anchor Mark I Modem, 300 Baud	89
	Password 300/1200 Baud Modem	399
	Hayes Smartmodem, 300 Baud	233
	Hayes Smartmodem, 1200 Baud	525

PERIPHERALS

	LNW Model I Expansion Interface, 32K & RS-232	329
	12" Green 18MHz NonGlare Video Monitor	87
	12" Amber 20MHz NonGlare Video Monitor	99
	Video cable for Model I keyboard-to-monitor	7
16-0230	13' Remote Control Cable Ready Color TV	285
16-0501	Model 10 Video Tape Recorder w/Remote	449
26-1130	Model 1, 3, 4 5MB Hard Disk Primary	1699
26-1131	Model 1, 3, 4 5MB Hard Disk Secondary	1529
26-1183	Model 100 Bar Code Reader	85
26-1208	CCR-81 Cassette Tape Recorder	51
26-1410	Model 100 Modem Cable	17
26-3007	Color Computer Carrying Case	17
26-3008	Color Computer Joysticks, Per Pair	21
26-3012	**NEW!** Deluxe Joystick/each	34
26-3022	Color Computer Disk Drive 0 Kit	339
26-3023	Color Computer Disk Drive 1, 2 or 3	237
	Same as above except our brand. Save $$$$	199
26-3024	Color Computer Multi-Pak Interface	153
26-3025	Color Computer Mouse	43
26-3029	Disk Drive 0 for the Color Computer 2	329
26-3503	PC 1 Cassette Interface	25
26-3508	PC 1 Carrying Case	13
26-3608	PC 2 Carrying Case	25
26-3612	PC 2 RS-232C Interface	169
26-3615	PC 2 4K RAM Module	33
26-3616	PC 2 8K RAM Module	118
26-3651	PC 4 Cassette Interface	33
26-3653	PC 4 1K Ram Module	16
26-3654	PC 4 Carrying Case	6
26-3804	Model 100 AC Adapter	5
26-3805	Model 100 Acoustic Coupler	34
26-3809	Model 100 Carrying Case	42
26-3816	Model 100 8K RAM Module	101
	Same as above except our brand. Save $$$$	79
26-4152	Model 2, 12, 16 12Mb Hard Disk Primary	2545
26-4153	Model 2, 12, 16 12Mb Hard Disk Secondary	2120
26-4160	Model 2 1-Disk Expansion Bay	674
26-4162	Model 2 3-Disk Expansion Bay	1379
26-4165	Model 12, 16 SlimLine 1-Drive Expansion	1104
26-4166	Model 12, 16 SlimLine 2-Drive Expansion	1783

FURNITURE

26-1305	Model 3 System Desk	152
26-1308	Universal Printer Stand	85
26-1324	Computer Table	60
26-1325	Platform	26
26-4301	Model 2 Desk	299
26-4303	Deluxe System Desk	229
26-4304	System Desk Drawer	111
26-4305	Deluxe Printer Stand	128
26-4306	Terminal Stand	153
74-0550	Docu-Rack Stand	51

PRINTERS & ACCESSORIES

26-1155	Quick Printer II	69
26-1158	Daisywheel II	1525
26-1190	Plotter	424
26-1191	Multi-Pen Plotter	1695
26-1192	GCP-115 Color Graphics Printer	169
26-1193	Flatbed Plotter/Printer	845
26-1195	Digitizer	381
26-1196	Color Computer Graphics Input Tablet	297
26-1250	DWP-410 Daisywheel Printer	1099
26-1251	DMP-400 140cps Printer	715
26-1252	DMP-500 220cps Printer	1099
26-1253	DMP-100 50cps Printer	339
26-1254	DMP-200 120cps Printer	594
26-1255	DMP-120 120cps Dual Mode Printer	424
26-1256	DMP-2100 160cps Dual Mode Printer	1695
26-1257	DWP-210 Daisywheel Printer	679
26-1260	Envelope Feeder for the Daisywheel II	1015
26-1261	TP-10 Thermal Matrix Printer	85
26-1266	CR-510 Computer Card Reader	1440
26-1267	DMP-420 140cps Triple Mode Printer	849
26-1268	CGP-220 Color Ink Jet Printer	594
26-1269	PTC-64 64K Printer Controller(Buffer)	212
	All $29 Printwheels	25
	All $39 Printwheels	33
26-1440	Single Bin Sheet Feeder	845
26-1441	Bi-Directional Tractor	144
26-1443	Bi-Directional Tractor	127
26-1447	Bi-Directional Tractor	208
26-1448	Single Bin Sheet Feeder	999
26-1455	Acoustic Cover	339
26-1459	Bi-Directional Tractor	204
26-1401	Model 1, M3, M4 Printer Cable	29
26-1408	RS-232C Cable	16
26-1409	Model 100 Printer Cable	13
26-4401	Model 2, M12, M16 Printer Cable	29
26-1490	10' RS-232C Cable	25
26-1491	25' RS-232C Cable	33
26-1492	50' RS-232C Cable	46
26-1493	100' RS-232C Cable	76
26-1495	RS-232C Cable Extender	25
26-1496	RS-232C Null Modem Adapter	25
26-1498	Parallel Printer Switch	102
26-1499	RS-232C Selector Switch	128
26-3505	PC 1 Printer	108
26-3591	PC 3 Printer	102
26-3605	PC 2 Printer	187
26-3652	PC 4 Printer	68

C.ITOH

8510	Prowriter 120cps, Friction/Tractor, Parallel	359
1550	Prowriter 2, 120cps, for Wide Paper, Parallel	699
F10-40	Starwriter, 40cps Daisywheel	1299
F10-55	Printmaster, 55cps Daisywheel, FAST	1499
F10	Bi-Directional Tractor, with Plastic Cover	210
F10	Single Bin Sheet Feeder	888
8600	Near Letter Quality, 180cps, 2-Color	899
GX-100	Gorilla Banana, 50cps	229
CX-4800	Printer Plotter, 4-Color	599

ANADEX

DP-9620	180cps Printer	1199
DP-9625A	200cps Triple Mode Printer	1299
WP-6000	285cps Near Letter Quality Printer	2099
DP-6500	500cps Super Trick Includes Tractor	2299
	Tractor Assembly for WP-6000	129
	Automatic Sheet Feeder for WP-6000/6500	925

We have dust covers for most every printer we sell. Be sure to order one with your new printer to keep it like new for years.

MONTEZUMA'S REVENGE

This month we offer the Holmes VID-80 with CP/M 2.2 and 112K RAM for the Model III. Now you can be the CP/M King and/or Queen on your block and save $125 by installing it yourself. Complete instructions included.

See the Holmes ads in 80 Micro

Retail Price $524.50 **YOUR COST ONLY ... $ 399**

SOFTWARE

	NEW CP/M 2.2 by Montezuma Micro For The Model 4	199
	NEW MBASIC for the Model 4	149
26-0310	TRSDOS 2.3 For The Model 1	12
26-0312	TRSDOS 1.3 For The Model 3	12
26-0313	TRSDOS 6.0 For The Model 4	12
26-1507	Stockpack	42
26-1510	Trendex	51
26-1512	**NEW!** Target PlannerCalc	85
26-1515	**NEW!** pfs file	106
26-1516	**NEW!** pfs report	85
26-1518	**NEW!** pfs file for the Model 4	107
26-1520	**NEW!** Model 4 VisiCalc	212
26-1521	VisiCalc Business Forecast	85
26-1530	**NEW!** Model 4 Multiplan	169
26-1540	**NEW!** Model 3 General Ledger	169
26-1541	**NEW!** Model 3 Accounts Receivable	169
26-1543	**NEW!** Model 3 Payroll	169
26-1552	General Ledger	85
26-1553	Inventory Control I	85
26-1554	Accounts Payable	128
26-1555	Accounts Receivable	128
26-1556	Disk Payroll	169
26-1557	Concrete Take-Off	49
26-1558	Business Mailing List	85
26-1559	Manufacturing Inventory Control	169
26-1560	Fixed Assets	68
26-1562	Profile	68
26-1563	Scripsit	85
26-1564	Mailgram	34
26-1565	Microfile	85
26-1566	Model 1 VisiCalc	85
26-1568	Medical Office Systems	254
26-1569	Model 3 VisiCalc, Enhanced Version	169
26-1579	Real Estate	85
26-1580	Project Manager	85
26-1581	Personnel Manager	85
26-1582	Time Manager	85
26-1584	Checkwriter 80	85
26-1585	Business Checkwriter	127
26-1588	Videotex Plus	42
26-1589	MICRO/Courier	127
26-1590	SuperSCRIPSIT	169
26-1591	Scripsit Dictionary	126
26-1592	Profile Plus	169
26-1593	Profile Plus LD/HD Version	254
26-1594	Desktop/Plan-80	169
26-1595	**NEW!** SuperSCRIPSIT For The Model 4	169
26-1596	**NEW!** Scripsit For The Model 4	85
26-1597	Business Graphics Pak	148
26-2010	Mod 3 BASIC Programming Course	25
26-2011	EDAS, Tape Version	25
26-2012	Assembly Language Development Course	126
26-2013	EDAS, Disk Version	29
26-2014	Mod 3 Disk Course	25
26-2017	**NEW!** Assembly Language Course, Tape	34
26-2018	**NEW!** Assembly Language Course, Disk	59
26-2022	**NEW!** Power Tool	43
26-2023	**NEW!** Dot Plot	17
26-2150	Introduction to BASIC	99
26-2200-1	Model 1 Model 3 FORTRAN	85
26-2201	Model 1 FORTRAN	85
26-2203	COBOL	169
26-2204	Compiler BASIC	126
26-2205	PILOT	85
26-2206-7	COBOL Runtime Disk	25
26-2208-9	Compiler BASIC Runtime Disk	17
26-2210	**NEW!** BASCOM	169
26-2211	**PASCAL**	**212**
26-2212	**NEW!** PASCAL For The Model 4	169
26-2213	Model 1 LDOS	109
26-2214	Model 3 LDOS	109
26-2216	CP/M Plus For The Model 4	126
26-GOOD	The Good CP/M for the Model 4(MM)	199
26-2217	CBASIC For The Model 4 (Requires CP/M)	85
26-2220-23	Videotex	26
26-2224	Videotex Compuserv Kit	16
26-2709	Color Computer PILOT, Tape	51
26-2710	Color Computer PILOT, Disk	68
26-2718	PILOT III	101
26-2721	Color Computer LOGO, Disk	85
26-2722	LOGO ROM Pack	43
26-3019	Diagnostic ROM Pack	17
26-3030	**NEW!** OS-9 For The Color Computer	59
26-3036	**NEW!** BASIC-09 For The Color Computer	85
26-3821	**NEW!** Model 100 Learning Lab	25
	ALL Color Computer GAMES 25% OFF Catalog Price	
	ALL PC SOFTWARE 30% OFF Catalog Price	
26-4501	General Ledger I	135
26-4502	Inventory Management System I	135
26-4503	Payroll	270
26-4504	Accounts Receivable	199
26-4505	Accounts Payable	199
26-4506	Mailing List	44
26-4507	Mailing List II	101
26-4508	Medical Office Systems	637
26-4509	Manufacturing Inventory Control	637
26-4510	Versafile	59
26-4511	VisiCalc	254
26-4512	Profile II	152
26-4513	Job Costing	126
26-4514	Order Entry	169
26-4515	Profile II Plus	254
26-4516	Profile Training Guide	58
26-4517	Profile Plus Upgrade	102
26-4520	Time Accounting	424
26-4521	VisiCalc	254
26-4525	Electronic Broker	850
26-4531	SCRIPSIT 2.0	254
26-4532	SCRIPSIT Utility Disk	109
26-4534	SCRIPSIT Dictionary	169
26-4536	SCRIPSIT Plotter Driver	42
26-4540	Statistical Analysis	85
26-4545	Litigation Support	254
26-4550	**NEW!** Business Graphics Anylsis Pak	212
26-4554	Accounts Receivable	107
26-4555	Menu Generator	33
26-4556	Profile Forms	106
26-4557	Profile Archive	42
26-4558	Profile Prosort	126
26-4559	Prolook	169
26-4560	WESTLAW	211
26-4580	MultiPlan, Model 2 & Model 12 Version	212
26-4601	General Ledger	424
26-4602	Inventory Control System	254
26-4603	Payroll	509
26-4604	Accounts Receivable	424
26-4605	Accounts Payable	424
26-4607	Order Entry/ICS	424
26-4608	Sales Analysis	254
26-4621	Personnel Search	254
26-4701	FORTRAN	254
26-4702	EDAS	169
26-4703	COBOL	254
26-4704	COBOL Run-Time Disk	34
26-4705	Compiler BASIC	169
26-4706	Compiler BASIC Run-time Disk	26
26-4707	COBOL Generator	850
26-4708	Report Generator	254
26-4710	Program Editor	67
26-4711	**NEW!** PASCAL MT+	361
26-4712	Assembly Language Development System	211
26-4713	EDAS 1	43
26-4714	ReformaTTer	212
26-4716	BiSYNC 3780	850
26-4718	Term Administration	???
26-4721	Videotex For The Model 2, 12, 16	42
26-4725	**NEW!** BASCOM BASIC Compiler	169
26-4730	BiSYNC 3270, Version 2	1270
26-4740	CP/M Plus	211
26-4742	CBASIC	84
26-4802	Inventory for the Hard Disk	339
26-4831	SCRIPSIT for the Hard Disk	339
26-4834	Dictionary for the Hard Disk	169
26-4835	SCRIPSIT for Thinline Drives	339
26-4910	TRSDOS 2.0	21
26-4912	TRSDOS 4.2	21
26-6100	COBOL Development System	254
26-6101	COBOL Run-time Diskette	34
26-6105	TRSDOS 1.6	34
26-6201	General Ledger	509
26-6203	Payroll	594
26-6204	Accounts Receivable	509
26-6205	Accounts Payable	509
26-6207	Order Entry/ICS	509
26-6208	Sales Analysis	339
26-6209	Job Costing	169
26-6302	Inventory	339
26-6309	Manufacturing ICS	722
26-6401	XENIX Development System	637
26-6455	COBOL	599
26-6457	MBASIC	254
26-6480	MultiPlan, Model 16 & XENIX	296
	Electric Pencil 2.0, M1, M3	69
	Blue Pencil, M1, M3	69
	Red Pencil, M1, M3	69
	HEXSPELL 2	80
	HEXMAN	60
	Modem 80, A Powerful Communications Program	35

CP/M

	CP/M 2.2 for the Model 4. The **VERY BEST ONE**	199
	MBASIC for the Model 4	149
	P & T CP/M 2.2m Floppy, M2-12-16	180
	P & T CP/M 2.2m Hard, M2-12-16	225
	All **MICROPRO** Software Available. Call For Details.	

BOOKS

26-2102	Model 1 Level II Manual	5
26-2103	Model 1 Technical Manual	13
26-2109	Model 3 Technical Manual	8
26-2110	Model 4 Technical Manual	21
26-2111	Model 3 DOS Manual	6
26-2112	Model 3 BASIC Instruction Manual	5
26-2114	Sourcebook, New Edition	4
26-2115	Newsletter Book 1980	4
26-2117	Model 4 DOS Manual	15
26-2240	Newsletter Book 1981	8
26-3191	Color Computer Manual	5
26-3192	Extended Color Computer Manual	
26-4920	Model 2 Operators Manual	26
26-4921	Model 2 Technical Reference Manual	26
26-4922	Model 12 Operating Manual	59
26-6040	Model 16 Operators Manual	64
26-6041	Model 16B Operating Manual	72
62-2084	TRS-80 Pocket BASIC Handbook	6
	TRS-80 Disk & Other Mysteries	18
	Microsoft BASIC Decoded	24
	BASIC Faster & Better	24
	BASIC Faster & Better Demonstration Disk	15
	BASIC Faster & Better Library Disk	15
	BASIC Disk I/O Faster & Better	24
	The Custom TRS-80	24
	TRSDOS 2.3 Decoded	24
	Machine Language Disk I/O	24
	How Do It On The TRS-80	24
	TRS-80 Beginners Guide	19

UPGRADES

	16K RAM, 200 nsec, 1 Full Year Guarantee	16
	64K RAM, 150 nsec, 1 Full Year Guarantee	64
26-1122	Model 4P 64K RAM Kit with PAL	134
	Same except ours. Includes a genuine PAL. Save $$$$	89
26-1123	Model 3 to Model 4 Upgrade Kit	719
26-1125	Model 3 Hi-Resolution Board	314
26-1126	Model 4 Hi-Resolution Board	214
26-1127	Model 4 Drive 0 Kit	551
	Same as above except our brand. Save $$$$	369
26-1143	Model 1 Double Density Controller	127
	Famous AEROCOMP DDC Double Density Controller	99
26-1145	Model 1 RS232 Kit	85
26-1148	Model 3 RS232 Kit	95
	Same as above except our brand. Save $$$$	69
26-1162	Model 3 Drive 0 Kit	550
	Same as above except our brand. Save $$$$	369
26-1163	Radio Shack 40 Track Disk Drive	203
	Same as above except 40 track Tandon. Save $$$	169
26-4104	Model 2 Hi-Resolution Board	424
26-4105	64K RAM Board	339
26-4163	8" Disk Drive	510
26-4167	8" Sim Drive	359
	Holmes Mod 1, Mod 3 4Mhz Speedup Mod	89
26-3013	16K RAM For MC-10 Color Computer	42
26-3016	Color Computer Low Profile Keyboard	34
26-3018	Color Computer Extended BASIC ROM	85
26-6010	Model 2/12 to Model 16 68000 Kit	1299
26-6017	Model 12 Card Cage	169

We maintain an extensively equipped service facility with a large stock of parts. Give us a call for that hard-to find item. We'll try and provide what you need.

CABLES

We manufacture a great number of different cable assemblies to connect most anything to anything else. Call us with your specific needs. We might have it on the shelf. All at discount prices, of course.

RIBBONS

We have more ribbons than you do, so please call us when you need some. Way too many types to list here. They're cheeeep too.

SUPPLIES

5.25" SSDD Diskettes, Pack of 10	19
5.25" DSDD Diskettes, Pack of 10	25
8" SSDD Diskettes, Pack of 10	32
8" DSDD Diskettes, Pack of 10	49
5.25" Flipsort, Holds 75 Disks	19
8" Flipsort, Holds 50 Disks	24
8.5" x 11" Tractor Paper, Good 20 lb. Stock, 2900 Sheets	22
14" x 11" Tractor Paper, Good 20 lb. Stock, 2900 Sheets	33
5.25" Head Cleaning Kit	9
8" Head Cleaning Kit	9

BUY FROM US RIGHT NOW!

Our large inventory can not be listed completely. Please call if you do not see what you want. Chances are we have it. Because of the time lag in magazine advertising, prices are subject to change without notice. Price reductions will be passed along automatically. Your company and/or personal checks are welcome and will not delay your order. We use TeleCheck. Please observe these requirements and your order will be shipped without delay. The check must be drawn on a USA or Canadian bank and payable in US Dollars. It must be a bank printed check and contain your street address and telephone number. The signature must EXACTLY match the name printed on the check. Thats all there is to it. In addition, we accept American Express, Mastercard, Visa, Cashiers Checks, Wire Transfer, and we also ship COD. COD's and motor freight shipments may require a deposit. Some special items require deposits. ALL COD's require cash or cashiers check on delivery. Company and/or personal checks can NOT be accepted in payment of COD shipments. Credit cards are not charged until we ship your order. Shipping charges are not included in our prices unless otherwise indicated. No State Sales Tax collected on out-of-state shipments or software. If you order by 5PM your order will be shipped the NEXT DAY if stock is on hand. Due to the nature of this business there are NO REFUNDS ON SOFTWARE. REPLACEMENTS WILL BE PROVIDED FOR DEFECTIVE SOFTWARE ITEMS, provided we are notified within ten days of delivery of the merchandise. If you have a particular problem please call us for help and instructions.

WARRANTY

All items carry the original manufacturers warranty. We will be happy to mail you a copy upon request.

Ad 1083

TeleCheck

Makes checks as welcome as cash.
COMPANY AND PERSONAL CHECKS WELCOME.
Your order will be shipped immediately upon receipt of your check.

CALL TOLL FREE FROM ANYWHERE IN THE USA

800-527-0347

800-442-1310

VISA MasterCard AMERICAN EXPRESS

✓ See List of Advertisers on Page 211

Stepping Through Basic

by Brian Craft

12-14

It's 12:30 a.m. You're sitting 3½ inches from your computer's CRT. Your eyes are bloodshot and bulging, your mouth hanging open. Slowly you type RUN and press the enter key. Your eyes widen as you peer into the screen.

Suddenly your hand crashes down on the break key. You pound a command into the machine and hit the enter key with a force that makes the entire table shake. You lean back your head and scream, "BUT L ISN'T SUPPOSED TO BE 47!!" A light comes on in the neighbor's house; a face appears at the window, peering intently out into the darkness.

If you've ever encountered flaws in program logic this way, then I have a utility for you. SINSTEP (Program Listing 3), lets you single-step through Basic programs, display and change variables, and change the screen, among other things, at any point during execution.

SINSTEP requires a 32K Model III, a disk drive, and an editor/assembler. If you have 32K of memory, set the ORG statement in line 100 to 0B000 hexadecimal (hex), the EQU statement in line 110 to 0BBFF hex, and the EQU statement in line 120 to 0B7FE hex. With 48K RAM, the program runs as is.

Using the Program

Assemble the program as SINSTEP/CMD. To run it, go into Basic. Set the memory size to 61440 (45046 for a 32K system). Load the program with CMD"L","SINSTEP/CMD". Type DEFUSR=&HF000 (&HB000 for 32K), and X=USR(0). The system should appear to be locked up. To return to Basic Ready, depress the F key. Now run or type in the program you want to debug, then use SINSTEP as follows.

Because the program keeps track of two different screens—a normal screen and one displaying variables, line numbers, and so on—you can debug a program without affecting the appearance of the display. In the list of commands (see Table 1) use those preceded by one asterisk only in the data screen mode, those with two stars only in the normal screen mode, and those not starred in either mode.

When the SINSTEP program is in memory and you've initialized it with X=USR(0), commands typed in the immediate mode won't operate properly unless you hold down the F key. To break a program when using SINSTEP, hold down F and press the break key.

The Key Box

Model III
32K RAM
Disk Basic
Assembly Language
Editor/Assembler

Program Listing 3. SINSTEP.

LOAD 80

```
00050 ;SINGLE STEP BASIC UTILITY
00060 ;       BY BRIAN CRAFT
00070 ;
00080 ;       INITIALIZATION.
00090 ;
00100         ORG     0F000H          ;NICE STARTING PLACE.
00110 NORMS   EQU     0FBFFH          ;LOCATION OF NORMAL SCREEN.
00120 DATAS   EQU     0F7FEH          ;LOCATION OF DATA SCREEN.
00130         LD      HL,INIT         ;SET DOS EXIT AT
00140         LD      (41C5H).HL      ;41C5H TO JP INIT.
00150         LD      HL,DATAS        ;CLEAR DATA AND
00160         LD      DE,DATAS+1      ;REGULAR SCREENS
00170         LD      BC,800H         ;(STORED FROM F7FFH
00180         LD      (HL).32         ;TO FFFFH).
00190         LDIR                    ;
00200         RET                     ;GO BACK TO BASIC.
00220 INIT    PUSH    AF              ;SAVE REGISTERS.
00230         PUSH    BC              ;
00240         PUSH    DE              ;
00250         PUSH    HL              ;
00260         LD      HL,(4020H)      ;SAVE CURSOR POSITION.
00270         PUSH    HL              ;
00280 START   LD      A,(SCREEN)      ;IF NOT DATA SCREEN THEN
00290         CP      1               ;SKIP OVER PRINT LINE
00300         JP      Z,KEY           ;AND VARIABLES ROUTINE.
00310 ;
00320 ;       PRINT CHOSEN VARIABLES
00330 ;
00340         LD      A,63            ;SET CURSOR POSITION.
00350         LD      (4021H),A       ;
00360         LD      A,250           ;
00370         LD      (4020H),A       ;
00380         LD      HL,LOGO         ;DISPLAY AUTHOR'S LOGO.
00390         CALL    DISPL           ;
00400         LD      A,61            ;SET CURSOR POSITION.
00410         LD      (4021H),A       ;
00420         LD      A,64            ;
00430         LD      (4020H),A       ;
00440         LD      HL,BUFF4        ;HL=EMPTY BUFFER.
00450         LD      BC,DISV         ;BC=LIST OF VARABLES.
00460 LOOPA   LD      A,(BC)          ;GET CHARACTER.
00470         LD      (HL).A          ;TRANSFER TO BUFFER.
00480         CP      32              ;IF IT IS A BLANK
00490         JR      Z,ERR           ;THEN SKIP OVER IT.
00500         CP      0DH             ;IF IT IS A CARIAGE RETURN
00510         CALL    Z,PRINT         ;THEN PRINT THE VARIABLE.
00520         INC     HL              ;ADD 1 TO POINTERS.
00530 ERR     INC     BC              ;
00540         CALL    33H             ;PRINT CHARACTER.
00550         LD      A,(DISP+1)      ;IF AT END OF VARIABLE
00560         CP      B               ;LIST THEN JP TO LINE
00570         JR      NZ,LOOPA        ;(DISP=END OF LIST
00580         LD      A,(DISP)        ;POINTER).
00590         CP      C               ;
00600         JR      NZ,LOOPA        ;IF NOT THEN LOOPA.
00610         JP      LINE            ;
00620 PRINT   LD      A,'='           ;DISPLAY AN "=".
00630         CALL    33H             ;
00640         PUSH    BC              ;SAVE LIST POINTER
00650         LD      (HL),';'        ;ADD A ";" TO END
00660         INC     HL              ;OF BUFFER.
```

Listing 3 continued

Be sure not to exit a program while in the data mode. If you do this, the program tries to display the current line when you type in a command in the immediate mode. Though this doesn't crash the system, it makes some commands behave strangely and makes a mess of the screen.

When I decided to write this program the first problem was how to make the Basic interpreter stop after each instruction. This proved to be an easy task. As the Basic interpreter works, it calls a keyboard routine at 358 hex. This keyboard routine calls a DOS exit at 41C5 hex. TRSDOS uses this exit for some unknown, apparently useless, reason—removing it has no effect on execution.

The first two lines of the SINSTEP program reroute the DOS exit to F000 (0B000 for 32K RAM systems). Thus, every time the interpreter scans the keyboard, it jumps to the SINSTEP program. The rest of the program consists mainly of reading the keyboard and calling various ROM locations. ■

Write to Brian Craft, age 14, at 2021 Lincoln, Emporia, KS 66801. Please include a self-addressed, stamped envelope for a reply.

Command	Description
F	Fast Execution command. Makes the program operate as close to normal speed as possible.
I	Single Instruction command. Makes the program execute one Basic statement.
R	Restart Program command. Makes the program execute a Run command.
S	Switch-screens command. Alternates between normal and data screen modes. In the data mode the program prints any variables you specify and the current line number. If you switch to the data screen, the program doesn't display the current line until you encounter a new line.
**A	Puts you in the Alter Screen mode. A cursor appears in the upper left-hand corner of the screen. To move it, use the arrow keys. To change the character over which the cursor appears, press the clear key or space bar. Pressing the space bar adds 1 to the ASCII value of the cursor location. Clear does the same at a much faster rate. To exit the Alter Screen mode, press the enter key.
*C	Change Variable command. The program asks for a variable name and a new value. After you enter both, it sets the specified variable to the value you entered. Enclose the value in quotes if you change a string variable. You can enter a variable name instead of a new value. This sets the first variable equal to the second variable.
*V	Display Variable command. The program asks for a variable name, then displays the specified variable.
*D	Constant Display command. The program asks for a variable name, then displays the specified variable constantly as long as you're in the data screen mode.
*M	Change Memory Location command. The program asks for a memory location and a new byte to put in the location.
*J	Jump command. The program asks for a line number, then executes a GOTO to that line number.
*N	New Screen command. Clears the data screen.

Table 1. SINSTEP Commands.

✓ See List of Advertisers on Page 211

Listing 3 continued

```
00670           LD      (HL),':'        ;
00680           LD      HL,BUFF4        ;SET HL TO BUFFER.
00690           LD      A,(HL)          ;LD A WITH FIRST CHARACTER.
00700           CALL    206FH           ;PRINT VARIABLE.
00710           LD      HL,BUFF4-1      ;RESET BUFFER POINTER.
00720           POP     BC              ;RESTORE LINE POINTER.
00730           RET                     ;GO BACK FOR NEXT VARIABLE.
00740 ;
00750 ;         PRINT LINE NUMBER AND LINE.
00760 ;
00770 LINE      LD      A,60            ;SET CURSOR POSITION.
00780           LD      (4021H),A       ;
00790           XOR     A               ;
00800           LD      (4020H),A       ;
00810           INC     SP              ;MOVE SP AROUND THE CURSOR
00820           INC     SP              ;POSITION STORED EARLIER.
00830           POP     HL              ;SET REGISTERS TO THEIR
00840           POP     DE              ;ORIGINAL CONTENCE.
00850           POP     BC              ;
00860           POP     AF              ;
00870           PUSH    AF              ;
00880           PUSH    BC              ;
00890           PUSH    DE              ;
00900           PUSH    HL              ;
00910           DEC     SP              ;MOVE SP BACK TO
00920           DEC     SP              ;CORRECT POSITION.
00930           LD      A,(HL)          ;GET NEXT CHARACTER IN LINE.
00940           CP      ':'             ;IF IT IS A ':' THEN DON'T TRY
00950           JR      Z,KEY           ;TO PRINT A LINE # OR LINE.
00960           OR      A               ;IF IT IS NOT A 0 THEN
00970           JP      NZ,BACK         ;THERE IS AN ERROR, RETURN.
00980           INC     HL              ;IF NEXT TWO CHARACTERS
00990           LD      A,(HL)          ;ARE 0 WE ARE AT THE END
01000           INC     HL              ;OF THE PROGRAM,
01010           OR      (HL)            ;RETURN.
01020           JP      Z,BACK          ;
01030           INC     HL              ;GET LINE NUMBER INTO
01040           LD      E,(HL)          ;DE.
01050           INC     HL              ;
01060           LD      D,(HL)          ;
01070           EX      DE,HL           ;GET LINE # INTO HL.
01080           LD      BC,(40A2H)      ;SAVE CURRENT LINE
01090           PUSH    BC              ;POINTER.
01100           LD      (40A2H),HL      ;SET CURRENT LINE POINTER.
01110           PUSH    DE              ;SAVE DE.
01120           CALL    0FAFH           ;PRINT LINE NUMBER.
01130           LD      A,' '           ;PRINT A SPACE.
01140           CALL    32AH            ;
01150           POP     DE              ;RESTORE DE.
01160           EX      DE,HL           ;RESTORE HL.
01170           INC     HL              ;POINT TO NEXT CHARACTER.
01180           CALL    2B7EH           ;PUT LINE IN WORK AREA.
01190           LD      HL,(40A7H)      ;POINT TO WORK AREA.
01200           CALL    2B75H           ;DISPLAY LINE.
01210           POP     BC              ;RESTORE
01220           LD      (40A2H),BC      ;CURRENT LINE POINTER.
01230           LD      A,30            ;ERASE TO END OF LINE.
01240           CALL    33H             ;
01250 ;
01260 ;         CHECK KEYBOARD. JUMP TO OR CALL ROUTINES.
01270 ;
01280 KEY       LD      A,(3801H)       ;CHECK FOR "F".
01290           CP      64              ;
01300           JP      Z,BACK          ;
01310 WAIT      CALL    KBSCAN          ;WAIT FOR CHARACTER.
01320           JP      M,WAIT          ;
01330           CP      'I'             ;CHECK FOR "I".
01340           JP      Z,BACK          ;
01350           CP      'S'             ;IF NOT "S" THEN TO NEXT1
01360           JR      NZ,NEXT1        ;
01370           LD      A,(SCREEN)      ;SWITCH SCREEN POINTER.
01380           XOR     1               ;
01390           LD      (SCREEN),A      ;
01400           CP      1               ;IF SWITCHING TO NORMAL
01410           JR      Z,NORM          ;THEN TO NORM.
01420           CALL    VDS             ;DISPLAY DATA SCREEN.
01430           POP     HL              ;RESTORE CURSOR
01440           LD      (4020H),HL      ;POSITION.
01450           POP     HL              ;RESTORE REGISTERS.
01460           POP     DE              ;
01470           POP     BC              ;
01480           POP     AF              ;
01490           JP      INIT            ;GO BACK TO BEGINING.
01500 NORM      CALL    VDS2            ;DISPLAY NORMAL SCREEN.
01510           POP     HL              ;RESTORE CURSOR
01520           LD      (4020H),HL      ;POSITION.
01530           POP     HL              ;RESTORE REGISTERS.
01540           POP     DE              ;
01550           POP     BC              ;
01560           POP     AF              ;
01570           JP      INIT            ;GO BACK TO BEGINING.
01580 NEXT1     CP      'R'             ;IF "R" THEN CALL
01590           JP      Z,RUN           ;RUN.
01600           PUSH    AF              ;SAVE CHARACTER.
01610           LD      A,(SCREEN)      ;IF SCREEN NOT NORMAL
01620           CP      1               ;THEN TO NEXT2.
01630           JR      NZ,NEXT2        ;
01640           POP     AF              ;RESTORE CHARACTER.
01650           CP      'A'             ;IF "A" THEN CALL
01660           CALL    Z,SCRN          ;SCREEN ROUTINE.
01670           JP      KEY             ;GO GET ANOTHER CHARACTEI
01680 NEXT2     POP     AF              ;RESTORE CHARACTER.
01690           CP      'C'             ;IF "C" THEN CALL
01700           JP      Z,VARIB         ;VARIB.
01710           CP      'M'             ;IF "M" THEN CALL MEM.
01720           CALL    Z,MEM           ;
01730           CP      'J'             ;IF "J" THEN JUMP TO
01740           JP      Z,JMPR          ;JMPR.
01750           CP      'V'             ;IF "V" THEN JUMP TO
01760           JP      Z,VARIBL        ;VARIBL.
01770           CP      'D'             ;IF "D" THEN CALL SET.
01780           CALL    Z,SET           ;
01790           CP      'N'             ;IF "N" THEN CALL
01800           CALL    Z,CLS           ;CLS.
01810           JR      KEY             ;GO GET ANOTHER CHARACTEI
01820 ;
01830 ;         KEYBOARD SCAN ROUTINE
01840 ;
01850 KBSCAN    PUSH    DE              ;SAVE DE.
01860           CALL    02BH            ;SCAN KEYBOARD.
01870           POP     DE              ;RESTORE DE.
01880           RET                     ;RETURN
01890 ;
01900 ;         DISPLAY DATA SCREEN
01910 ;
01920 VDS       LD      HL,3C00H        ;MOVE NORMAL SCREEN
01930           LD      DE,NORMS        ;INTO FBFFH.
01940           LD      BC,400H         ;
01950           LDIR                    ;
01960           LD      HL,DATAS        ;MOVE DATA SCREEN IN
01970           LD      DE,3C00H        ;FROM F7FEH.
01980           LD      BC,400H         ;
01990           LDIR                    ;
02000           RET                     ;RETURN.
02010 ;
02020 ;         DISPLAY NORMAL SCREEN
02030 ;
02040 VDS2      LD      HL,3C00H        ;MOVE DATA SCREEN
02050           LD      DE,DATAS        ;INTO F7FEH.
02060           LD      BC,400H         ;
02070           LDIR                    ;
02080           LD      HL,NORMS        ;MOVE NORMAL SCREEN IN
02090           LD      DE,3C00H        ;FROM FBFFH.
02100           LD      BC,400H         ;
02110           LDIR                    ;
02120           RET                     ;RETURN.
02130 ;
02140 ;         RETURN TO INTERPETER
02150 ;
02160 BACK      POP     HL              ;RESTORE CURSOR POSITION.
```

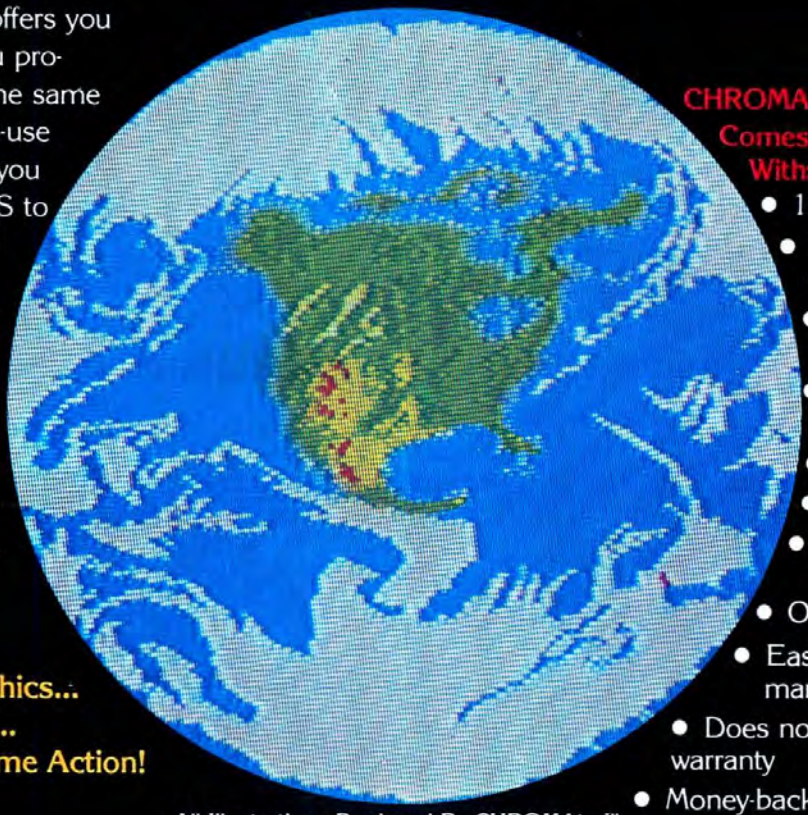
Listing 3 continued

ADD A WORLD OF COLOR TO YOUR TRS-80

MOD I & MOD III

CHROMAtrs™ MAKES IT HAPPEN

This powerful peripheral offers you 15 brilliant colors, lets you produce sensational effects the same day you plug it in! Easy-to-use "CHROMA BASIC" gives you 71 CHROMA COMMANDS to use in addition to regular BASIC. You can devise your own exciting games, plot points and lines, do 3-D rotations,translations, create a large range of sprite graphics, produce charts and graphs, and make great sound effects.



All Illustrations Produced By CHROMAtrs™

Now You Can Create Spectacular Color Graphics... Exciting Sound Effects... Paddle & Joy Stick Game Action!

CHROMAtrs™ Comes Complete With:

- 15 vivid colors
- High resolution graphics (256 x 192)
- 2 Atari joystick and paddle connectors
- 3-D animation using sprite graphics
- 16k display RAM
- Programmability in BASIC
- LOGO language subset on disk
- One complimentary game
- Easy-to-understand operating manual
- Does not affect Radio-Shack warranty
- Money-back guarantee

This is a quality product that can multiply the value of your TRS-80. Supplies are limited and prices subject to change. A word to the wise: Order TODAY!

ONLY $199!

(American Version. For European Version & accessories, see coupon below.)

I own a TRS-80 ☐ Model I ☐ Model III ☐ 16K ☐ 32K ☐ 48K

☐ CHROMA BASIC (Previous owners only)	$30
☐ CHROMAtrs assembled & tested, USA (With CHROMA BASIC)	$199
☐ CHROMAtrs assembled & tested, European (except France)	$230
☐ RF modulator with switch box	$25
☐ Mod 1 ribbon cable	$12
☐ Mod 3 ribbon cable	$14
☐ Cassette software or	(free with CHROMAtrs)
☐ Diskette software	(free with CHROMAtrs)
Subtotal	
N.Y.S. residents add 8.25% sales tax	
Shipping and handling (USA)	$7.50
TOTAL	

Check one:-( ) Check ( ) M.O. ( ) COD ( ) M.C. ( ) Visa

Account # _____ Exp. _____

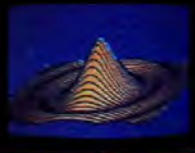
Name _____

Address _____

City _____

State _____ Zip _____




MICRO CONTROL SYSTEMS, INC.

(Formerly South Shore Computer Concepts)

1590 Broadway, Hewlett N.Y. 11557
Phone orders accepted (516) 569-4390

Listing 3 continued

```
02170           LD      (4020H),HL      ;
02180           POP     HL              ;RESTORE REGISTERS.
02190           POP     DE              ;
02200           POP     BC              ;
02210           POP     AF              ;
02220           RET                     ;RETURN TO INTERPETER.
02230 ;
02240 ;         DISPLAY MESSAGE ROUTINE
02250 ;
02260 DISPL     LD      A,(HL)          ;GET NEXT CHARACTER.
02270           OR      A               ;IF IT'S A 0 THEN RETURN.
02280           RET     Z               ;
02290           CALL    33H             ;DISPLAY CHARACTER.
02300           INC     HL              ;ADD 1 TO POINTER.
02310           JR      DISPL           ;BACK TO DISPL.
02320 ;
02330 ;         CHANGE A MEMORY LOCATION ROUTINE
02340 ;
02350 MEM       LD      A,0C9H          ;RESET DOS JUMP.
02360           LD      (41C4H),A       ;
02370           LD      A,64            ;SET CURSOR POSITION.
02380           LD      (4020H),A       ;
02390           LD      A,63            ;
02400           LD      (4021H),A       ;
02410           LD      HL,MEMES        ;DISPLAY "MEMORY
02420           CALL    DISPL           ;LOCATION:".
02430           LD      HL,BUFF         ;INPUT MEMORY LOCATION
02440           LD      B,5             ;INTO BUFF.
02450           CALL    5D9H            ;
02460           LD      HL,BYTE         ;DISPLAY "BYTE:".
02470           CALL    DISPL           ;
02480           LD      HL,BUFF2        ;INPUT BYTE INTO BUFF2.
02490           LD      B,3             ;
02500           CALL    5D9H            ;
02510           LD      HL,BUFF         ;CONVERT ADDRESS FROM
02520           CALL    1E5AH           ;ASCII TO BINARY.
02530           PUSH    DE              ;SAVE ADDRESS.
02540           LD      HL,BUFF2        ;CONVERT BYTE FROM ASCII
02550           CALL    1E5AH           ;TO BINARY.
02560           LD      A,E             ;PUT BYTE INTO A.
02570           POP     HL              ;GET ADDRESS.
02580           LD      (HL),A          ;LD ADDRESS WITH BYTE.
02590           LD      A,0C3H          ;SET DOS JUMP AGAIN.
02600           LD      (41C4H),A       ;
02610           RET                     ;RETURN.
02620 ;
02630 ;         CHANGE SCREEN ROUTINE
02640 ;
02650 SCRN      LD      A,0C9H          ;RESET DOS JUMP.
02660           LD      (41C4H),A       ;
02670           LD      HL,3C00H        ;HL=SCREEN ADDRESS
02680 LOOPB     LD      A,(HL)          ;WAIT FOR KEY.
02690           LD      (HL),191        ;BLINK CURSOR.
02700           LD      B,0             ;
02710 PAUSE     DJNZ    PAUSE           ;
02720           LD      (HL),A          ;
02730           CALL    2BH             ;
02740           JP      M,LOOPB         ;
02750           CP      91              ;CHECK FOR UP-ARROW,
02760           JR      NZ,NO1          ;JUMP IF NO.
02770           LD      DE,40H          ;MOVE CURSOR UP.
02780           SBC     HL,DE           ;
02790           LD      DE,3C00H        ;
02800           RST     18H             ;
02810           JR      NC.LOOPB        ;
02820           LD      DE,40H          ;
02830           ADD     HL,DE           ;
02840           JR      LOOPB           ;BACK TO LOOP.
02850 NO1       CP      10              ;CHECK FOR DOWN-ARROW,
02860           JR      NZ,NO2          ;JUMP IF NO.
02870           LD      DE,40H          ;MOVE CURSOR DOWN.
02880           ADD     HL,DE           ;
02890           LD      DE,4000H        ;
02900           RST     18H             ;
02910           JR      C,LOOPB         ;
02920           LD      DE,40H          ;
02930           SBC     HL,DE           ;
02940           JR      LOOPB           ;BACK TO LOOP.
02950 NO2       CP      8               ;CHECK FOR LEFT-ARROW,
02960           JR      NZ,NO3          ;JUMP IF NO.
02970           DEC     HL              ;MOVE CURSOR LEFT.
02980           LD      DE,3C00H        ;
02990           RST     18H             ;
03000           JR      NC,LOOPB        ;
03010           INC     HL              ;
03020           JR      LOOPB           ;BACK TO LOOP.
03030 NO3       CP      9               ;CHECK FOR RIGHT ARROW,
03040           JR      NZ,NO4          ;JUMP IF NO.
03050           INC     HL              ;MOVE CURSOR RIGHT.
03060           LD      DE,4000H        ;
03070           RST     18H             ;
03080           JR      C,LOOPB         ;
03090           DEC     HL              ;
03100           JR      LOOPB           ;BACK TO LOOP.
03110 NO4       CP      0DH             ;RETURN IF <ENTER>.
03120           JR      NZ,NO6          ;
03130           LD      A,0C3H          ;SET DOS JUMP AGAIN.
03140           LD      (41C4H),A       ;
03150           RET                     ;RETURN.
03160 NO6       CP      32              ;CHECK FOR SPACE.
03170           JR      NZ,NO5          ;JP IF NO.
03180           INC     (HL)            ;INC SCREEN LOCATION.
03190           JR      LOOPB           ;BACK TO LOOP.
03200 NO5       LD      A,(14400)       ;CHECK FOR CLEAR.
03210           CP      2               ;
03220           JR      NZ,LOOPB        ;TO LOOP IF NO.
03230           INC     (HL)            ;INC SCREEN LOCATION.
03240           LD      B,0             ;PAUSE.
03250 DLAY      DJNZ    DLAY            ;
03260           JR      NO5             ;LOOP.
03270 ;
03280 ;         GOTO ROUTINE
03290 ;
03300 JMPR      LD      A,0C9H          ;RESET DOS JUMP.
03310           LD      (41C4H),A       ;
03320           LD      A,64            ;SET CURSOR POSITION.
03330           LD      (4020H),A       ;
03340           LD      A,63            ;
03350           LD      (4021H),A       ;
03360           LD      HL,LINEN        ;DISPLAY "LINE NUMBER:".
03370           CALL    DISPL           ;
03380           LD      HL,BUFF3        ;INPUT LINE NUMBER INTO
03390           LD      B,5             ;BUFF3.
03400           CALL    5D9H            ;
03410           LD      A,0C3H          ;SET DOS JUMP AGAIN.
03420           LD      (41C4H),A       ;
03430           LD      A,1             ;SET SCREEN FLAG.
03440           LD      (SCREEN),A      ;
03450           CALL    VDS2            ;CHANGE SCREENS.
03460           POP     HL              ;RESTORE CURSOR.
03470           LD      (4020H),HL      ;
03480           POP     HL              ;RESTORE REGISTERS.
03490           POP     DE              ;
03500           POP     BC              ;
03510           POP     AF              ;
03520           LD      HL,BUFF3        ;POINT TO LINE NUMBER.
03530           JP      1EC2H           ;CALL JUMP ROUTINE.
03540 ;
03550 ;         CHANGE VARIABLE ROUTINE
03560 ;
03570 VARIB     LD      A,0C9H          ;RESET DOS JUMP.
03580           LD      (41C4H),A       ;
03590           LD      A,64            ;SET CURSOR POSITION.
03600           LD      (4020H),A       ;
03610           LD      A,63            ;
03620           LD      (4021H),A       ;
03630           LD      HL,VARI         ;DISPLAY "VARIABLE:".
03640           CALL    DISPL           ;
03650           LD      HL,BUFF4        ;INPUT VARIABLE NAME.
03660           LD      B,30            ;
```

Listing 3 continued

Listing 3 continued

```
03670           CALL    5D9H            ;
03680           LD      HL,VALU         ;DISPLAY "NEW VALUE:".
03690           CALL    DISPL           ;
03700           LD      HL,BUFF5        ;INPUT NEW VALUE.
03710           LD      B,247           ;
03720           CALL    5D9H            ;
03730           LD      BC.BUFF4        ;PUT BUFF 4 IN BUFF 6.
03740           LD      HL,BUFF6        ;
03750 LOOPC     LD      A,(BC)          ;
03760           CP      31              ;
03770           JR      C,FINIS         ;
03780           LD      (HL).A          ;
03790           INC     HL              ;
03800           INC     BC              ;
03810           JR      LOOPC           ;
03820 FINIS     LD      A,0D5H          ;ADD A "=".
03830           LD      (HL).A          ;
03840           INC     HL              ;
03850           LD      BC,BUFF5        ;PUT BUFF 5 IN BUFF 6.
03860 LOOPD     LD      A,(BC)          ;
03870           CP      31              ;
03880           JR      C,FINIS2        ;
03890           LD      (HL).A          ;
03900           INC     HL              ;
03910           INC     BC              ;
03920           JR      LOOPD           ;
03930 FINIS2    LD      A,58            ;ADD A ":".
03940           LD      (HL).A          ;
03950           LD      A,0C3H          ;SET DOS JUMP AGAIN.
03960           LD      (41C4H).A       ;
03970           INC     SP              ;SKIP OVER CURSOR
03980           INC     SP              ;POSITION.
03990           POP     HL              ;RESTORE REGISTERS.
04000           POP     DE              ;
04010           POP     BC              ;
04020           POP     AF              ;
04030           PUSH    AF              ;
04040           PUSH    BC              ;
04050           PUSH    DE              ;
04060           PUSH    HL              ;
04070           DEC     SP              ;SET SP BACK TO CORRECT
04080           DEC     SP              ;POSITION.
04090           LD      HL,BUFF6        ;POINT TO BUFFER.
04100           LD      A,(HL)          ;LD A WITH FIRST CHARACTER.
04110           CALL    1F21H           ;CALL LET ROUTINE.
04120           JP      KEY             ;JUMP BACK TO KEY.
04130 ;
04140 ;         DISPLAY VARIABLE ROUTINE.
04150 ;
04160 VARIBL    LD      A,0C9H          ;RESET DOS JUMP.
04170           LD      (41C4H).A       ;
04180           LD      A,63            ;SET CURSOR POSITION.
04190           LD      (4021H),A       ;
04200           LD      A,64            ;
04210           LD      (4020H).A       ;
04220           LD      HL,VARI         ;DISPLAY "VARIABLE:".
04230           CALL    DISPL           ;
04240           LD      HL,BUFF4        ;INPUT VARIABLE NAME.
04250           LD      B,30            ;
04260           CALL    5D9H            ;
04270           LD      A,B             ;ADD LENGTH OF VARIABLE
04280           ADD     A,L             ;NAME TO ADDRESS OF
04290           JR      NC.NOCARY       ;BUFFER.
04300           INC     H               ;
04310 NOCARY    LD      L,A             ;
04320           LD      (HL).':'        ;ADD A ':' TO END STATEMENT.
04330           LD      A,0C3H          ;SET DOS JUMP AGAIN.
04340           LD      (41C4H).A       ;
04350           INC     SP              ;MOVE PAST CURSOR POSITION.
04360           INC     SP              ;
04370           POP     HL              ;RESTORE REGISTERS.
04380           POP     DE              ;
04390           POP     BC              ;
04400           POP     AF              ;
04410           PUSH    AF              ;
04420           PUSH    BC              ;
04430           PUSH    DE              ;
04440           PUSH    HL              ;
04450           DEC     SP              ;PUT SP BACK
04460           DEC     SP              ;IN PLACE.
04470           LD      HL,BUFF4        ;POINT TO BUFFER.
04480           LD      A,(HL)          ;LOAD A WITH FIRST CHARACTER.
04490           CALL    206FH           ;CALL PRINT ROUTINE.
04500           JP      START           ;BACK TO START.
04510 ;
04520 ;         RUN ROUTINE
04530 ;
04540 RUN       POP     HL              ;RESTORE CURSOR POSITION.
04550           LD      (4020H).HL      ;
04560           POP     HL              ;RESTORE REGISTERS.
04570           POP     DE              ;
04580           POP     BC              ;
04590           POP     AF              ;
04600           LD      HL,RUND         ;POINT TO RUN DATA.
04610           LD      A,':'           ;
04620           RET                     ;RETURN TO INTERPETER.
04630 ;
04640 ;         SET VARIABLE FOR CONSTANT DISPLAY
04650 ;
04660 SET       LD      A,0C9H          ;RESET DOS JUMP
04670           LD      (41C4H),A       ;
04680           LD      A,63            ;SET CURSOR POSITION.
04690           LD      (4021H).A       ;
04700           LD      A,64            ;
04710           LD      (4020H).A       ;
04720           LD      HL,VARI         ;DISPLAY "VARIABLE".
04730           CALL    DISPL           ;
04740           LD      HL,BUFF4        ;INPUT VARIABLE NAME.
04750           LD      B,30            ;
04760           CALL    5D9H            ;
04770           LD      BC,(DISP)       ;GET END OF LIST.
04780           PUSH    HL              ;PUT BUFF4 INTO DE.
04790           POP     DE              ;
04800           LD      HL,DISP         ;HL=END OF LIST POINTER LOCATION.
04810 LOOPE     LD      A,(DE)          ;GET CHARACTER FROM BUFFER.
04820           LD      (BC).A          ;PUT INTO VARIABLE LIST.
04830           INC     DE              ;ADD 1 TO POINTERS.
04840           INC     BC              ;
04850           INC     (HL)            ;ADD ONE TO END OF LIST POINTER.
04860           CP      0DH             ;JUMP IF NOT A CARIAGE
04870           JR      NZ,LOOPE        ;RETURN.
04880           LD      A,0C3H          ;SET DOS JUMP AGAIN.
04890           LD      (41C4H).A       ;
04900           RET                     ;RETURN.
04910 ;
04920 ;         CLEAR SCREEN ROUTINE
04930 ;
04940 CLS       LD      A,28            ;OUTPUT A HOME CURSOR.
04950           CALL    33H             ;
04960           LD      A,31            ;CLS SCREEN.
04970           CALL    33H             ;
04980           RET                     ;RETURN
04990 SCREEN    DEFB    1
05000 MEMES     DEFM    'MEMORY LOCATION: '
05010           DEFB    0
05020 BUFF      DEFS    6
05030 BYTE      DEFM    'BYTE: '
05040           DEFB    0
05050 BUFF2     DEFS    4
05060 BUFF3     DEFS    6
05070 LINEN     DEFM    'LINE NUMBER: '
05080           DEFB    0
05090 VARI      DEFM    'VARIABLE:'
05100           DEFB    0
05110 VALU      DEFM    'NEW VALUE:'
05120           DEFB    0
05130 BUFF4     DEFS    30
05140 BUFF5     DEFS    247
05150 POINT     DEFB    0
05160 BUFF6     DEFS    278
05170 RUND      DEFM    ':'
05180           DEFB    8EH
05190           DEFM    ':'
05200 FLAG      DEFB    0
05210 LOGO      DEFM    'b.c.'
05220           DEFB    0
05230 DISP      DEFW    DISV+1
05240 DISV      DEFB    32
05250           END
```

Ground Control to Major John

by Mark E. Kennedy

11 and under

In Adventure Sampler, you play the character John R. Franklin, famous astronaut. You have just completed your mission to search for alien life, and you're on your way back to earth. The adventure in store, however, is more interesting than the total of your previous eight months of research put together!

Adventure Sampler requires a 32K RAM Disk Basic or 16K RAM Cassette Basic Model I or III.

At the start, you have a ship and crew, a laser pistol, a hook and rope, and an extra air tank for your spacesuit (Table 2 lists all the game's equipment). But on your way back to earth, your ship is damaged and you begin a long trek through space, trying to get home.

You encounter alien creatures, strange civilizations, junk yards, jails, and many other dangers. You collect different point values for accomplishing certain feats, such as retrieving the treasure or entering the second shuttle (see Table 3). The bonus scoring system has a possible 300 points.

The game uses no vocabulary, but rather presents you a number of choices. The wrong choice may lead to your death.

When typing in Program Listing 4, spacing is unimportant, although it makes the program more attractive. Also, you may adjust the GOSUB 80 commands to your taste. I number the program lines in increments of ten, so you can use the Auto command. The game uses the linear method of play, but the bonus points and graphics make this program unique. ■

Mark Kennedy, age 11, can be reached at 3709 Tumeric Lane, Bakersfield, CA 93309.

The Key Box

Models I and III
16K RAM Cassette Basic
32K RAM Disk Basic

Program Listing 4. Adventure Sampler.

LOAD 80

```
10 CLS:P=0
20 CLEAR9000
30 PRINT" ADVENTURE  SAMPLER
       BY MARK E.  KENNEDY
       INSTRUCTIONS(Y/N)"
40 INPUTX$
50 IFX$="Y"THEN70
60 IFX$="N"THEN150  ELSE30
70 GOTO 90
80 FORT=1TO1500:NEXT:RETURN
90 PRINT@345,"* * * ADVENTURE SAMPLER * * *":GOSUB80
100 PRINT@474,"* * BY MARK E. KENNEDY * *":GOSUB80   :CLS
110 PRINT@89,"HELLO THERE!"
120 PRINT@217,"WELCOME TO THIS ADVENTURE!
     IN IT YOU ARE KNOWN AS JOHN FRANKLIN,
   WORLD FAMOUS ASTRONAUT.YOU ARE LOST
 IN SPACE,(UNFORTUNATELY),AND MAY NEVER"
130 PRINT@473,"GET BACK!":GOSUB80   :GOSUB80
140 PRINT@537,"YOU HAVE A HOOK & ROPE,LASER PISTOL,
```


Listing continued

Available Objects
Laser rifle
Laser pistol
Air tank
Spacesuit
Treasure
Hook & rope

Table 2. Game objects.

Bonus Points	
Winning trial	2
Shooting guards	8, 20, 25
Entering shuttle	75
Getting treasure	100
Password	15
Experience	50
Finding second city	5

Table 3. Scoring System.

Listing 4 continued

```
            AND AN AIR TANK THAT IS ATTACHED TO
           YOUR SPACE SUIT.":GOSUB80
150 PRINT@729,"WELL,ON WITH THE ADVENTURE!":GOSUB80   :CLS
160 PRINT@64,"IT HAS BEEN 5 YEARS SINCE YOU LEFT EARTH IN YOUR
         SHUTTLE WITH YOUR 8 CREW MEMBERS.IN THAT AMOUNT
        OF TIME,YOU HAVE COMPLETED YOUR MISSION AND IT
      IS NOW TIME TO GO HOME"
170 GOSUB80   :GOSUB80   :CLS
180 PRINT@345,CHR$(23),"EMERGENCY!"
190 GOSUB80   :CLS
200 PRINT"YOUR SCREEN FLASHES ANXIOUSLY, AND YOU HEAR AN
      EXPLOSION BEHIND YOU. . . SHOULD YOU. . . .
210 PRINT"a) CALL ENGINEERING,THEY WILL TELL YOU WHAT HAPPENED
220 PRINT"b) KEEP FLYING,IT MAY HAVE BEEN NOTHING
230 PRINT"c) OR TRY TO LAND ON A PLANET,YOU SEE ONE BELOW ?
240 INPUT"  YOUR       CHOICE  ";A$
250 IFA$="A"GOTO360
260 IFA$="B"THENCLS:PRINT@480,CHR$(23)"B O O M !":GOTO290
270 IFA$="C"THENCLS:PRINT@480,CHR$(23)CHR$(166)CHR$(153):GOSUB80
 :CLS:GOTO300
280 ELSE240
290 GOSUB80   :CLS:PRINT@480,"YOU ARE BLOWN TO BITS!":GOSUB80   :C
LS:GOSUB2000:END
300 PRINT"C R A S H ! YOU'RE SHIP IS IN VERY BAD CONDITION! SUDDEN
LY,YOU  SEE A  SMALL ALIEN CREATURE COMING TOWARD YOU. . . .
     SHOULD YOU. . . ."
310 PRINT"a) SHOOT AT IT"
320 PRINT"b) OR TRY TO MAKE FRIENDS"
330 INPUT" YOUR  CHOICE ";A$
340 IFA$="A"THEN380
350 IFA$="B"THEN390  ELSE330
360 CLS:PRINT"ENGINEERING    REPORT
          ------------------
         BAD DAMAGE DUE TO SHORT
        METEOR  SHOWER  = "
370 PRINT"    WEAK  REACTOR
           DAMAGED  WING
          AND NO SHIELDS   ":GOSUB80   :GOSUB80   :GOSUB80   :CLS:
GOTO200
380 SHO=1:CLS:PRINT"IT IS IMMUNE TO THE LASER BLASTS!  ":GOSUB80
 :CLS:GOTO390
390 CLS:PRINT"THE CREATURE TOUCHES A BUTTON ON HIS BELT AND YOU AR
E STRUCK     DOWN IMMEDIATELY!":GOSUB80   :CLS
400 PRINT"WHEN YOU GET UP ,YOU ARE NO LONGER IN YOUR SHIP. YOU'RE
IN A     SMALL CELL.  SHOULD YOU. . . ."
410 PRINT"a)  TRY TO ESCAPE"
420 PRINT"b)  OR WAIT  ?"
430 INPUT"YOUR CHOICE  ";A$
440 IFA$="A"THENCLS:PRINTCHR$(23)"Y O U ' R E   C A U G H T !":GOS
UB80   :CLS:GOSUB2000:END
450 IFA$="B"THENCLS:PRINT"SOON, AN ALIEN COMES AND SHOWS YOU TO TH
E NEXT ROOM":GOSUB80   :CLS  ELSE430
460 PRINT"IT IS TIME FOR YOUR TRIAL":GOSUB80   :CLS:IFSHO=1THENPRI
NT"YOU ARE FOUND GUILTY OF ATTEMPTED MURDER. ":GOSUB80   :CLS:PRIN
TCHR$(23);"SENTENCE : D E A T H !";:GOTO80   :CLS:GOSUB2000:END
470 P=P+2:PRINT"YOU ARE INNOCENT : THEY SET YOU FREE !":GOSUB80
:CLS:PRINT"SHOULD YOU. . . . "
480 PRINT"a) GO EXPLORING"
490 PRINT"b) GO BACK TO YOUR SHIP"
500 INPUT"YOUR   CHOICE";A$
510 IFA$="A"THEN590
520 IFA$="B"THEN650  ELSE500
530 PRINT"UPON REACHING YOUR SHIP, YOU FIND THAT YOUR CREW IS DEAD
.  YOU   MUST CONTINUE ALONE. . . . SHOULD YOU. . . ."
540 PRINT"a) EXPLORE YOUR SHIP"
550 PRINT"b) OR LEAVE ?"
560 INPUT"  YOUR CHOICE ";A$
570 IFA$="A"THEN820
580 IFA$="B"THEN590  ELSE560
590 CLS:GOSUB80   :PRINT"AFTER A WHILE, YOU FIND WHAT REMAINS OF A
N ALIEN CITY. . . .        SHOULD YOU. . . . ."
600 PRINT"a)  EXPLORE THE CITY"
610 PRINT"b)  OR GO BACK TO YOUR SHIP"
```

Listing 4 continued

✓ See List of Advertisers on Page 211

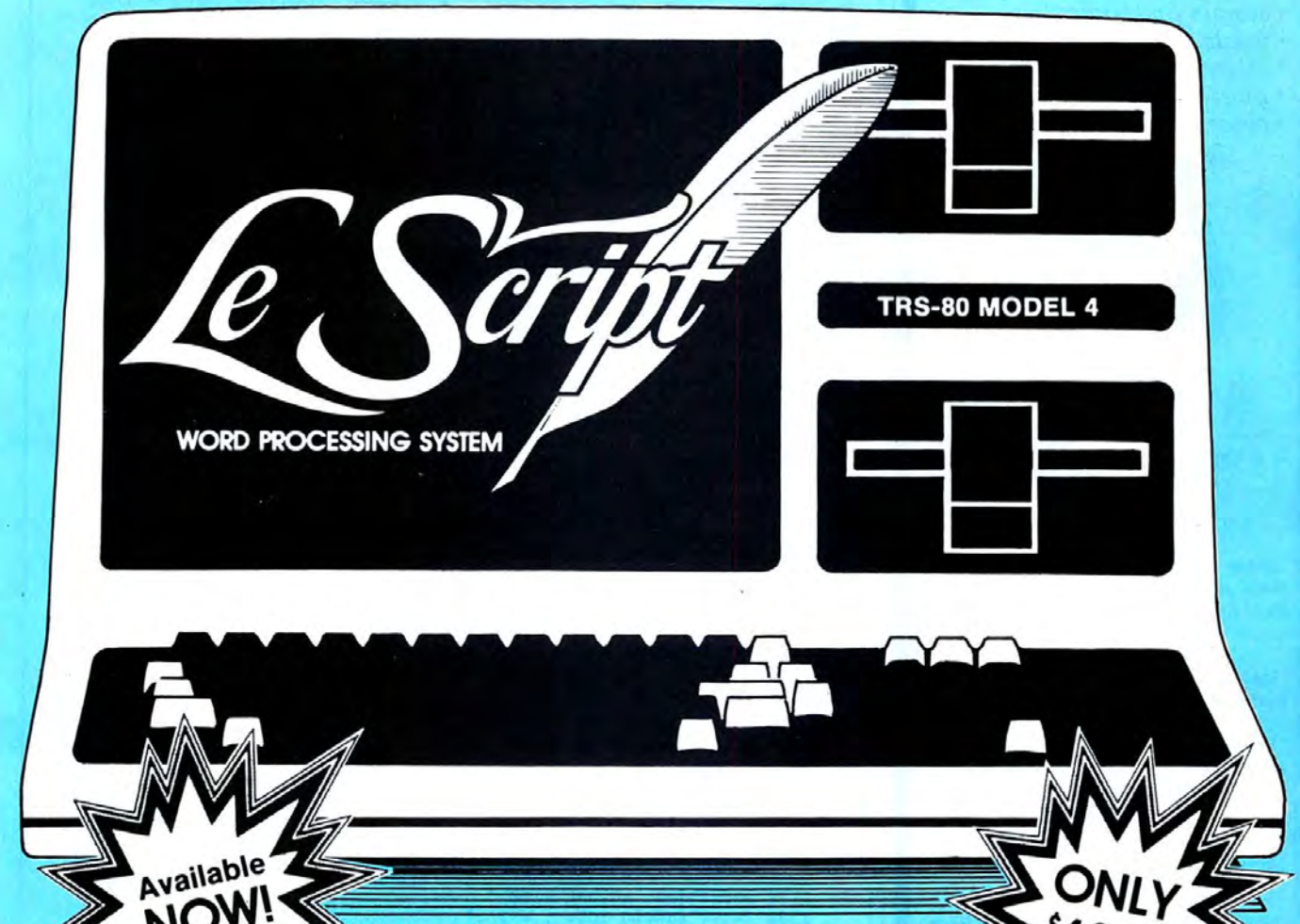
Listing 4 continued

```
620 INPUT"YOUR  CHOICE ";A$
630 IFA$="A"THEN740
640 IFA$="B"THEN530  ELSE620
650 CLS:PRINT"WHEN YOU GET TO YOUR SHIP, YOU GET A WARM WELCOME FR
OM YOUR       CREW MEMBERS.  BUT SUDDENLY. . .
              B  O  O  M  !"
660 PRINT"YOUR SHIPS MAIN REACTOR BLOWS OUT !  SHOULD YOU. . ."
670 PRINT"a)  LIMP AWAY"
680 PRINT"b)  CHECK THE DAMAGE"
690 PRINT"c)  OR ` HIT THE DECK ' ?"
700 INPUT"  YOUR        CHOICE  ";A$
710 IFA$="A"THEN590
720 IFA$="B"THENPRINT"THE DAMAGE IS TOO GREAT TO REPAIR":GOTO700
730 IFA$="C"THEN890  ELSE700
740 PRINT"YOU ARE IN THE CITY. YOU SEE A SMALL, ONE MAN, SHUTTLE.
      SHOULD YOU. . . ."
750 PRINT"a) ENTER THE SHUTTLE"
760 PRINT"b) OR LEAVE"
770 INPUT"YOUR CHOICE";A$
780 IFA$="A"THEN800
790 IFA$="B"THEN530  ELSE770
800 CLS:PRINT@480,CHR$(23)"YOU ESCAPE !":PRINT@600,CHR$(23)CHR$(18
3)CHR$(179)CHR$(187):GOSUB80   :CLS:GOSUB2000:END
810 GOSUB80   :CLS:PRINT@480," Y O U ' R E     D E A D !":PRINT:P
RINT,CHR$(183)CHR$(179)CHR$(179)CHR$(179)CHR$(187):GOSUB80   :GOSU
B80   :CLS:GOSUB2000:END
820 CLS:GOSUB80   :PRINT"YOU ENTER  YOUR SHIP ;":GOSUB80   :PRINT"
IT IS HEAVILY DAMAGED":GOSUB80   :PRINT" YOU FIND A  STRANGE LOOKI
NG  ALIEN  .  IT REACHES OUT  AT YOU ."
830 GOSUB80   :PRINT"SCREAMING, YOU RUSH OUT OF  YOUR SHIP ONLY TO
  RUN INTO A        MASSIVE, ALIEN GUARD . . .  SHOULD  YOU. . . .
."
840 PRINT"a)  TRY TO  FIGHT IT"
850 PRINT"b) OR TRY TO BE FREINDLY"
860 INPUT"YOUR   CHOICE ";A$
870 IFA$="A"THEN900
880 IFA$="B"THEN920  ELSE860
890 CLS:PRINT"IT  DOESN'T  WORK  !":GOSUB80   :CLS:PRINT@480,"Y O
U ' R E   DEAD !":GOSUB80   :GOSUB80   :CLS:GOSUB2000:END
900 P=P+8:CLS:PRINTSTRING$(63,176):PRINT:PRINT" ZZZAAPPP!!! ":GOSU
B80   :GOSUB80   :CLS:PRINT" YOU  DESTROYED HIM !":PRINT"YOU NOW H
AVE A LASER RIFLE !"
910 GOSUB80   :GOSUB80   :GOSUB80   :GOTO1220
920 PRINT" HE  GRABS YOU AND  DRAGS YOU AWAY":GOSUB80   :PRINT" AF
TER A WHILE, HE  COMES TO  A SMALL PLATFORM OVER-LOOKING       THE
 ALIEN CIVILIZATION. YOU ARE HIGH  IN THE MOUNTAINS. . .
   SHOULD  YOU. . . . ."
930 PRINT"a)  JUMP OFF"
940 PRINT"b)   FIGHT THE GUARD"
950 PRINT"c) YELL  FOR HELP"
960 PRINT"d)  OR LET THE GUARD KILL YOU"
970 INPUT"YOUR  CHOICE    ";A$
980 IFA$="A"ORA$="D"THEN890
990 IFA$="B"THEN1010
1000 IFA$="C"THEN1080 ELSE970
1010 PRINT" YOU FAIL. . . .
          HE  PUSHES  YOU  OFF THE  PLATFORM":GOSUB80   :GOSUB80
   :CLS:PRINT" YOU  LAND SAFELY  "
1020 GOSUB80   :GOSUB80   :GOSUB80   :CLS:PRINT" YOU ARE  IN A JUN
K  YARD . YOU  FEEL HEAVY BREATHING AT YOUR    BACK.  TURNING AROU
ND, YOU  SEE A  LARGE ALIEN . HE DRAWS  A       LASER RIFLE : HE
 AIMS  CAREFULLY."
1030 GOSUB80   :GOSUB80   :GOSUB80   :CLS
1040 ONRND(5)GOTO1050 ,1060 ,1050 ,1050 ,1060
1050 PRINT"HE MISSES !":GOTO1120
1060 PRINT"  HE  HITS YOU  !":GOSUB80   :GOSUB80   :CLS:PRINT@480,
"Y  O  U  '  R  E    DEAD !"
1070 GOSUB2000:END
1080 PRINT
1090 CLS:PRINT" NOBODY  COMES "
1100 PRINT" THE GUARD PUSHES YOU OFF  THE PLATFORM ."
1110 GOSUB80   :GOSUB80   :CLS:PRINT@480," YOU'RE   D  E  A  D ":G
OSUB80   :GOSUB80   :CLS:GOSUB2000:END
1120 GOSUB80   :GOSUB80   :GOSUB80   :CLS:PRINT" CAUTIOUSLY, YOU C
IRCLE AROUND HIM. . . .                          SHOULD YOU. . . .
. "
1130 PRINT"a) THROW  YOUR  HOOK & ROPE AT HIM"
1140 PRINT"b)  OR SHOOT  AT HIM "
1150 INPUT"YOUR CHOICE  ";A$
1160 IFA$="A"THEN1210
1170 IFA$="B"THEN1180 ELSE1150
1180 CLS:PRINTSTRING$(63,160):PRINT:PRINT" BLAM !     GOOD  SHOT A
CE !"
1190 GOSUB80   :GOSUB80   :CLS:PRINT" USING YOUR HOOK & ROPE  YOU
CLIMB UP THE  SIDE OF  THE MOUTAIN":GOSUB80   :GOSUB80   :CLS:GOSU
B80   :GOSUB80   :GOSUB80   :PRINT"ALMOST  TO  THE  TOP, YOU SLIP
 AND FALL !":GOSUB80   :GOSUB80   :CLS
1200 PRINT@480," YOU'RE   D  E  A  D !":GOSUB80   :GOSUB2000:END
1210 GOSUB80   :CLS:PRINT"IT  DOESN'T WORK !":GOTO1200
1220 GOSUB80   :CLS:PRINT" SHOULD YOU. . . . . ."
1230 PRINT"a)  GO EXPLORING, YOU MAY FIND  SOMETHING"
1240 PRINT"b) OR  QUIT WHILE YOU  ARE AHEAD"
1250 INPUT"  YOUR   CHOICE  ";A$
```

Listing 4 continued

See us at
SOFTCON
La Superdome
New Orleans
Tuesday-Thursday
February 21-23, 1984
BOOTH P-2063

TRS-80 MODEL 4 Word Processing Program



Available NOW!

ONLY $129.95

80x24 DISPLAY

- Works with **TRSDOS 6** or any **Model III DOS.**
- **90K TEXT BUFFER** when expanded memory is installed.
- Built-in **FORM LETTER** and **MAIL-MERGE** capabilities.
- Performs **PROPORTIONAL-SPACE** right-margin justification on over 60 different printers (all drivers included **FREE**).
- Same powerful editing features as **ZORLOF II,** with dozens more features added.
- Versions also available to run on **MODEL I, MODEL III, LNW-80, PMC-80,** and **MAX-80** (80x24 display).

LeScript costs $200 less than competing products matching its capabilities.

FREE
- 30-Day Support to Registered Owners
- Subscription to LeScript Newsletter

VISA MasterCard

CALL 1-305-259-9397

ANITEK

ANITEK SOFTWARE PRODUCTS
P.O. BOX 361136
MELBOURNE, FL 32936

141

FREE SHIPPING WITHIN THE U.S.; OUTSIDE THE U.S. ADD $10.00 FOR SHIPPING; FLORIDA RESIDENTS ADD 5% SALES TAX. ALL ORDERS PREPAID BY CHECK, MONEY ORDER, CREDIT CARD OR C.O.D.

See List of Advertisers on Page 211

Listing 4 continued

```
1260 IFA$="A"THEN1300
1270 IFA$="B"THEN1280  ELSE1250
1280 GOSUB80   :GOSUB2000
1290 END
1300 GOSUB80   :GOSUB80   :PRINT" AFTER  A WHILE ,  YOU COME UPON
 AN ALIEN CIVILIZATION"
1310 P=P+5:PRINT" SHOULD  YOU. . . . ."
1320 PRINT"a)  EXPLORE IT, YOU MAY FIND SOMETHING "
1330 PRINT"b)  LEAVE"
1340 PRINT"c) OR  QUIT"
1350 INPUT" YOUR  CHOICE    ";A$
1360 IFA$="A"THEN1400
1370 IFA$="B"THEN1390
1380 IFA$="C"THEN1280  ELSE1350
1390 CLS:PRINT" YOU LEAVE . . .  ":GOSUB80   :PRINT" YOU GET LOST
. . .":GOSUB80   :PRINT" Y O U   D I E ":GOSUB80   :GOSUB2000:END
1400 CLS:PRINT" E N T E R   C O D E ":GOSUB80   :PRINT" SHOULD  YO
U  SAY. . . ."
1410 PRINT"a) SESAME"
1420 PRINT"b)  SALEM"
1430 PRINT"c) OR  ALABASTER"
1440 INPUT" YOUR   CHOICE ";A$
1450 IFA$="A"ORA$="B"THENPRINT" WRONG ":GOTO1500
1460 IFA$="C"THEN1470  ELSE1440
1470 P=P+15
1480 PRINT" RIGHT !"
1490 PRINT
1500 GOSUB80   :CLS:PRINT" YOU ARE IN THE CIVILIZATION"
1510 PRINT"THERE IS A TIME WARP MACHINE HERE"
1520 PRINT"SHOULD YOU . . . . ."
1530 PRINT"a) GO TO THE FUTURE"
1540 PRINT"b) GO TO THE PAST"
1550 PRINT"c) OR PASS IT BY"
1560 INPUT"YOUR CHOICE  ";A$
1570 IFA$="A"THEN1600
1580 IFA$="B"THENP=0:GOTO820
1590 IFA$="C"THENPRINT"LEAVING THE MACHINE, YOU TRIP INTO AN OPEN
CHASM":GOSUB80   :GOSUB2000:END ELSE1560
1600 GOSUB80   :CLS:GOSUB80   :CLS:PRINT"YOU ARE STILL IN THE CITY
, BUT IT LOOKS DIFFERENT"
1610 PRINT"THE TOWN IS DESERTED. AND LOTS OF TREASURE IS THERE.
       ( DID SOMEONE SAY `TREASURE' ? )"
1620 P=P+100
1630 PRINT" GRABBING AN ARM LOAD OF TREASURE,
        YOU RE-ENTER THE TIME WARP MACHINE"
1640 GOSUB80   :GOSUB80   :GOSUB80   :GOSUB80   :GOSUB80   :CLS:GO
SUB80   :PRINT" BOY, THIS TOWN SURE LOOKS  FAMILIAR AGAIN"
1650 PRINT" JUST AHEAD, YOU SEE A SMALL SHUTTLE"
1660 PRINT"SHOULD YOU. . . . ."
1670 PRINT"a)  GO TOWARD THE SHUTTLE"
1680 PRINT"b) OR WALK AWAY"
1690 INPUT"YOUR   CHOICE ";A$
1700 IFA$="A"THEN1980
1710 IFA$="B"THEN1720  ELSE1690
1720 GOSUB80   :CLS:PRINT"YOU WANDER AROUND FOR A WHOLE HOUR"
1730 GOSUB80   :GOSUB80   :CLS:PRINT"YOU SEE MANY NEW THINGS"
1740 PRINT"SUCH AS SOME ALIEN MILITARY BASES (?), SHUTTLE HANGARS,
 AND THE ENTERTAINMENT CENTER "
1750 P=P+50
1760 GOSUB80   :GOSUB80   :CLS:PRINT"YOU SEE AN ALIEN GUARD COMING
 TOWARD YOU":PRINTCHR$(136)CHR$(172)CHR$(143)CHR$(172)CHR$(132)
1770 PRINT"(NOT ACTUAL SIZE)"
1780 PRINT"SHOULD YOU. . . . ."
1790 PRINT"a) SHOOT AT IT"
1800 PRINT"b) OR SURRENDER"
1810 INPUT" YOUR    CHOICE ";A$
1820 IFA$="A"THENCLS:PRINTSTRING$(63,191):PRINT"NICE SHOT!":GOTO18
40
1830 IFA$="B"THENPRINT"BAD CHOICE":GOSUB2000:END ELSE1810
1840 P=P+20
1850 GOSUB80   :GOSUB80   :CLS:PRINT"WALKING AWAY, YOU AGAIN NOTIC
E THE SHUTTLE"
1860 PRINT"IT IS A BEAUTIFUL SIGHT (IN GOOD CONDITION TOO)"
1870 PRINT"WHILE YOU STARE AT THE SHUTTLE, YOU FAIL TO NOTICE THE
GUARD     BEHIND YOU. . . .    HIS LASER  SHOT JUST MISSES YOU. .
."
1880 PRINT"SWINGING AROUND, YOU PULL OUT YOUR LASER RIFLE. . ."
1890 PRINT"SHOULD YOU. . . ."
1900 PRINT"a) FIRE AT THE GUARD"
1910 PRINT"b) OR RUN FOR THE SHUTTLE"
1920 INPUT"YOUR   CHOICE ";A$
1930 IFA$="A"THEN1950
1940 IFA$="B"THEN1980 ELSE1920
1950 PRINT"YOU MISS. ."
1960 PRINT" YOU START FIRING MADLY. . . HE IS HIT. . ."
1970 P=P+25
1980 PRINT"YOU RUN ABOARD THE SHUTTLE, START THE ENGINES (HOW YOU
DON'T     KNOW),AND BLAST OFF"
1990 P=P+75:GOSUB2000:END
2000 PRINT"YOU SCORED "P;"OUT OF A POSSIBLE 300 POINTS":IFP=300THE
NPRINT"EXCELLENT":RETURN
```

HOT CoCo

YES, I want a subscription at $24.97.

I understand that with payment enclosed or credit card order, I will receive a FREE issue making a total of 13 issues for $24.97.

☐MC ☐VISA ☐AE ☐CHECK/MO ☐BILL ME

Card#_____ Exp. Date_____

Signature_____

Name _____

Address _____

City_____ State_____ Zip_____

Canada & Mexico $27.97/1 year only, US funds drawn on US bank. Foreign surface $44.97/1 year only, US funds drawn on US bank.

• This offer expires March 31, 1984

342F8B

HOT CoCo • PO Box 975 • Farmingdale, NY 11737

Please allow 6–8 weeks for delivery.

HOT CoCo

YES, I want a subscription at $24.97.

I understand that with payment enclosed or credit card order, I will receive a FREE issue making a total of 13 issues for $24.97.

☐MC ☐VISA ☐AE ☐CHECK/MO ☐BILL ME

Card#_____ Exp. Date_____

Signature_____

Name _____

Address _____

City_____ State_____ Zip_____

Canada & Mexico $27.97/1 year only, US funds drawn on US bank. Foreign surface $44.97/1 year only, US funds drawn on US bank.

• This offer expires March 31, 1984

342F8B

HOT CoCo • PO Box 975 • Farmingdale, NY 11737

Please allow 6–8 weeks for delivery.

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 73 PETERBOROUGH NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE

Wayne Green Inc.
HOT CoCo
PO Box 975
Farmingdale, NY 11737

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 73 PETERBOROUGH NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE

Wayne Green Inc.
HOT CoCo
PO Box 975
Farmingdale, NY 11737

Some Like It HOT

HOT CoCo®

Guaranteed to make your TRS-80 Color Computer* sizzle with color, **HOT CoCo** magazine is informative, interesting, and *best of all* just for the Color Computer. You'll look forward to **HOT CoCo** month after month because it has something for everyone, from the novice right on up to the expert. **HOT CoCo** gives you:

- •PROGRAMMING TECHNIQUES & TUTORIALS—that promise to make you a superior programmer.
- •UTILITIES—to save you time and effort on all your routine tasks.
- •EXPERTLY WRITTEN COLUMNS—including *BASIC, GRAPHICS, FLEX and GAMES.*
- •HARDWARE & CONSTRUCTION—ideas on interfacing and enhancing to make building projects a breeze.
- •EDUCATIONAL APPLICATIONS—will stimulate and encourage imaginative thinking in your child.

PLUS

- •BUSINESS PROGRAMS—sure to make you a star at the office.
- •FEATURES ON COLOR APPLICATIONS—make your computer reach its full potential and get your money's worth from your machine.
- •BUYER'S GUIDES & PRODUCT REVIEWS—now you can stop running around comparing prices and products and start running your computer.
- •HOME & HOBBY APPLICATIONS—exciting ways to help your computer add enjoyment to your leisure time.
- •ANSWERS TO SPECIFIC QUESTIONS—it's like having your own private consultant—free!

Join in the color explosion with **HOT CoCo** today! Take advantage of our money saving offer, 12 issues for $24.97. A 13th issue is yours FREE with pre-payment (check or credit card). Use the attached order form, the coupon below, or call toll free 1-800-258-5473.
(In NH. call 1-924-9471)

YES! I want my Color Computer to sizzle with color. I understand that with payment enclosed or credit card order I will receive a free issue making a total of 13 issues for $24.97.

☐ CHECK/MO ☐ MC ☐ VISA ☐ AE ☐ BILL ME

card #_____ exp. date_____
signature _____
name_____
address _____
city _____ state _____ zip _____

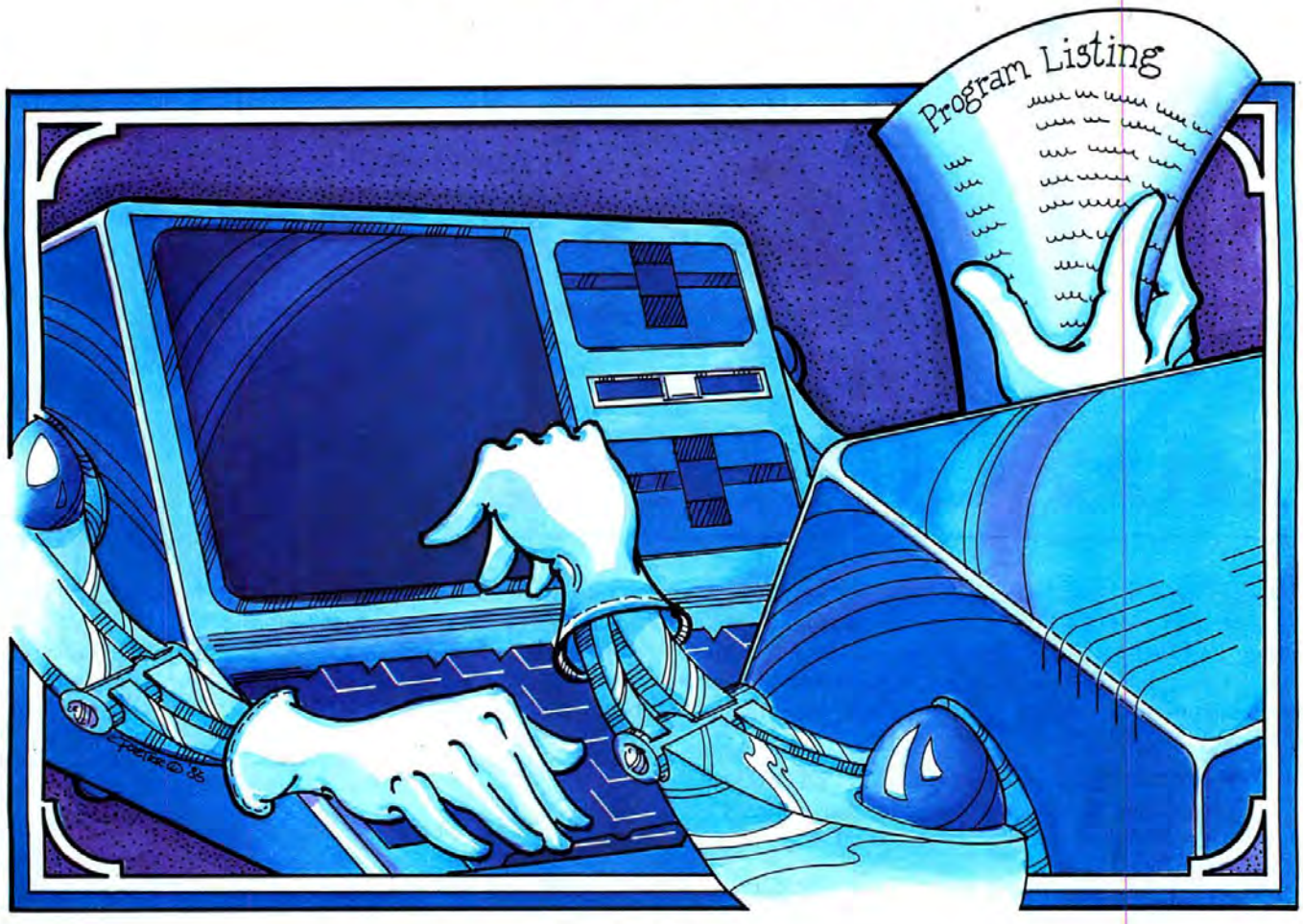
Canada & Mexico $27.97/1 yr. only, U.S. funds drawn on U.S. bank.
Foreign Surface $44.97/1 yr. only, U.S. funds drawn on U.S. bank.
Please allow 6-8 weeks for delivery.

*This price voids all previous offers.

HOT CoCo • PO Box 975 • Farmingdale NY 11737

* TRS-80 Color Computer is a trademark of Radio Shack, a division of Tandy Corp. 342F8

The Creator Is Back!



by Bruce Tonkin

And on the second day, Tonkin wrote new versions of the Creator and the Reportor, then placed the programs in the public domain.

I made my splash, but I'm not done (see "The Creator," *80 Micro*, January 1983, p. 74).

I'm going to continue my policy: to make good software available to everyone. This article contains the listings of the new Creator/Reportor, along with some useful utilities. Type them in, buy a copy from me, or get one from a friend.

It's fun to rock the boat. I still don't see other data base program generators coming down in price, so maybe I didn't rock it hard enough. Perhaps the Creator II will at least make some software manufacturers seasick.

The Creator II

Here are the improvements I made to the Creator II (see Program Listing 1):

1. Records can now span sectors—no more wasted disk space. If you use TRSDOS 1.3 or 2.3, you must enter Basic with the 3V response for the number of files, though. Check your manual for the correct syntax. DOSPLUS users should remember to enter Basic with the command BASIC-F:3 since DOSPLUS defaults to zero files.

2. The data file grows dynamically as you enter records. The initialization process in the old version took a while because all space in the file was preallocated. The new method shortens initialization by about 90 percent and makes record scan much faster. For one thing, initialization now allocates space only in the pointer file. As a side benefit, the data in the file remains in the order in which you entered it, at least until you

delete a record or update on the key field. Once you've performed either of those two operations, a future record can reclaim that space.

3. You can alter the key after generating your program or to use existing data files without retyping the data. You can even use foreign data files (those not generated by one of the Creator's programs).

4. You can include up to 47 fields in each record, 23 more than in the original. You can run generated programs using the 47 fields unless you use a large number of edits or memory-intensive routines (see the next item, for example). A record can also contain up to 256 bytes.

5. You can now display computed data along with your records. Only memory limits the number of displayable fields, and the display can include the result of any combination of any number of Basic operations—string, logical, or mathematical. Shades of VisiCalc! Of course, each individual computation is limited to a single line: about 15 operations for each of the 200 displayable items is probably a practical limit. Computed data takes up no space whatever in your record. Why waste the space? And you can print-format the computed data for a nicer display.

6. Your generated programs now have more sophisticated error-handling routines. Failure to initialize is now trapped out, for example. Not only that: the Creator warns you, before you've entered any data, if you've failed to initialize. The menu displays the current number of records in the file, including deleted records: if you add a record, reclaiming available space, the number of records in the file doesn't change.

7. Computational update lets you update a field by any of the four math functions (+,−,*,/), more, less, and/or by rounding the result of computations to any specified number of decimal places, up to Basic's limit of 16. You can tell your generated program to multiply a packed field by 1.23456 and round the result to two decimals, for example. Simply enter 1.23456*R02 for the update amount. Of course, the program recomputes and redisplays all computed data after any update. Unpacked data may not take advantage of this computational update feature.

8. You can scan, on alphabetic or numeric range, any field in the data file. Scan is much faster than in the original version. When you scan, the program asks if you want a delay after each record appears. If you don't, the scan may go so fast that only Evelyn Wood could keep up.

9. The prompt and operator answer can appear on the same line, making for a cleaner screen display.

10. The program no longer requires that you enter the key field and edit it against a lack of entry.

11. The generated routine contains only those routines necessary for the correct operation of your program. In other words, I've instructed the Creator to do a little code optimization. For example, if you don't use packed data, those routines are not included; if you don't use the Not Alpha edit, that routine is omitted as well.

12. When the program displays records, it strips most trailing blanks to use screen space more efficiently. I made the record display routine a little more attractive, too. If the screen fills when there is more to display, the display freezes and the program asks you to depress a key to see the rest.

13. I added a data type, too: packed half-precision (byte data for numbers in the range ±127). Now you can easily pack dates into three spaces, or store small numbers in one.

14. To save space, I put the commonly used routines in a library file rather than in the program itself. The library file looks like a Basic program, but if you alter it you must save it as an ASCII text file. This approach makes it easier for you hackers to add special-purpose routines or to modify the generated programs.

Modifications to the Reportor

I also added a number of enhancements to the Reportor (see Program Listing 2).

1. I've added a sort interface. When you run it, the generated program asks if you want to read an index file for a sorted report. If so, it reads one of your choice.

2. Your data file can be on any drive. The program asks for the drive number at the time you run a report.

3. You can use plain English names to define your report columns. The Reportor now permits commands such as "VALUE=PRICE*ON HAND". Choose any names you like, with length up to 255 bytes for each name. The names can include key words and punctuation (but not quotation marks), even other column names.

4. You can easily format columns as you like. The program asks whether you want to print numbers or letters in the

The Key Box

Models I and III
48K RAM
Disk Basic

Program Listing 1. The Creator II.

```
10 GOTO60
20 FX=0:IFLEN(CV$)<1 THENRETURN
30 FOR II=1 TO LEN(CV$):CV%=ASC(MID$(CV$,II,1)):IFCV%=34ANDFX=0THE
NFX=1:ELSEIFCV%=34ANDFX=1THENFX=0
40 IFCV%>90ANDFX=0THENCV%=CV%AND95
50 MID$(CV$,II,1)=CHR$(CV%):NEXT:RETURN
60 CLS:PRINTCHR$(23);"CREATOR/BAS: COPYRIGHT (C) 1983":PRINT"BY BR
UCE W. TONKIN":PRINT"34069 HAINESVILLE ROAD":PRINT"ROUND LAKE, IL
60073":PRINT"THIS WORK MAY NOT BE SOLD"
70 PRINT"WITHOUT THE EXPRESS WRITTEN":PRINT"PERMISSION OF BRUCE W.
 TONKIN.":PRINT"BUT MAY BE FREELY REPRODUCED.":PRINT"GIVE A COPY T
O A FRIEND!":PRINT"YOU MUST LEAVE THIS NOTICE":PRINT"IN PLACE AND
UNCHANGED!"
75 PRINT"PROGRAM MANUALS ARE AVAILABLE,":PRINT"FOR $10+$1 POSTAGE,
 FROM THE":PRINT"PROGRAM'S AUTHOR."
80 FOR I=1TO4000:NEXT
90 CLS
100 CLEAR0:CLEAR6000:PRINT"You may now change diskettes, if you wi
sh. However, please be":PRINT"sure you have the file called C451/L
IB on your drive 0.":PRINT"DEPRESS ANY KEY WHEN READY TO CONTINUE.
";
110 ANS=INKEYS:IF ANS="" THEN 110
120 CLS:PRINT"THIS IS THE CREATOR, VERSION 2.1. It will allow you
to generate":PRINT"a program that will create, update, and access
a hashed data":PRINT"file. First, you must tell me the name you wa
nt to give your":PRINT"program. ";
130 PRINT"YOU ARE LIMITED TO 8 ALPHABETIC CHARACTERS.":PRINT"PROGR
AM NAME= ";:LINE INPUT PN$:CV$=PN$:GOSUB 20:PN$=CV$:IF LEN(PN$)>8
THEN PRINT"TOO LONG.":GOTO130
140 FORI=1TOLEN(PN$):IFMID$(PN$,I,1)<"A"ORMID$(PN$,I,1)>"Z"THENPRI
NT"INVALID CHARACTER USED.":EX=1:I=LEN(PN$)
150 NEXT:IFEX=1THENEX=0:GOTO130
160 PRINT"The disk drives are numbered, 0 - 3. Which drive do you
want":PRINT"the data file on, WHEN YOUR PROGRAM IS RUN: ";
170 ANS=INKEYS:IF ANS="" THEN 170:ELSEIF ANS<"0"ORANS>"3"THENPRINT
:GOTO160
180 PRINT ANS
190 PRINT"What drive do you want your program written on: ";
200 BNS=INKEYS:IF BNS=""THEN200:ELSEIFBNS<"0"ORBNS>"3"THENPRINT:GO
TO190
210 PRINT BNS
220 PRINT"What drive do you want your program to be on, when you a
re":PRINT"running it: ";
230 CNS=INKEYS:IFCNS=""THEN230:ELSEIFCN$<"0"ORCNS>"3"THENPRINT:GOT
O220
240 PRINT CNS
250 ON ERROR GOTO 1930
260 OPEN"I",1,PN$+"/BAS"+":"+BNS:ERROR126
```

Listing 1 continued

column; if letters, it asks how many spaces to allow; if numbers, it asks how many digits to the right and left of the decimal point, whether commas should be used, and so on.

5. All headings are left-justified over character data, and right-justified over numeric data, making your report cleaner and easier to read.

6. The Reportor computes totals for all numeric columns automatically. You have only to specify whether you want them printed. You can use these totals to compute averages, for example.

7. You can choose any paper length.

8. When you run the report you can specify whether you want it printed to the screen, the printer, or both.

9. The Reportor uses a library file, R451/LIB, to make enhancements and alterations easier (see Program Listing 3). At last: hacker-friendly software!

"Creator II and Reportor handle errors better than do the originals."

By running the menu program with no file-type extension, you can run any program on the Creator II disk (see Program Listing 4).

Model I users will probably want to separate the files on the Creator II disk to free up some disk space; otherwise, no room is left. Those users should put Creator/BAS, C451/LIB (Program Listing 5), and C451MIN/LIB (Program Listing 6) on a system disk marked "Creator," and use that disk for creating programs. The other files should go on another system disk.

Both Creator II and Reportor run easier and handle errors better than do the originals. As a bonus, they both need only minor changes to run under

Listing 1 continued

```
270 KF$=PN$+"/KEY"+":"+AN$:DF$=PN$+"/DAT"+":"+AN$:PN$=PN$+"/BAS":O
PEN"O",1,PN$+":"+BN$:PRINT #1,"1 REM*PROGRAM NAME: ";PN$;"*":PRINT
 #1,"2 REM*DATA FILE NAME IS ";DF$;"*":PRINT #1,"3 REM*DATA FILE I
S ON DRIVE ";AN$;"*"
280 CLS:PRINT"What is the maximum number of records (per data disk
) you want":PRINT"to allow in your data file: ";:LINEINPUTMS$:MS=V
AL(MS$):IFMS<1ORMS>32767THEN280
290 PRINT"What is the record length (1 to 256 is permitted): ";:LI
NE INPUT RS$:RS=VAL(RS$):IF RS>256 OR RS<1 THEN 290
300 PRINT #1,"4 REM*NUMBER OF RECORDS=";MS;"*":PRINT #1,"5 REM*REC
ORD LENGTH=";RS;"*":Q$=CHR$(34):PRINT"Please input the title you w
ant displayed for your program:":LINE INPUT TI$
310 LN=LN+10:PRINT#1,LN;"CLEAR";10*RS+1000:LN=LN+10:PRINT#1,LN;"CL
S":LN=LN+10:IFRS<256 THENPRINT#1,LN;"OPEN";Q$;"R";Q$;",1,";Q$;DF$;
Q$;",";RS;":OPEN";Q$;"R";Q$;",2,";Q$;KF$;Q$;",2:FIELD#2,2 AS KP$"
320 IFRS=256THENPRINT#1,LN;"OPEN";Q$;"R";Q$;",1,";Q$;DF$;Q$;":OPEN
";Q$;"R";Q$;",2,";Q$;KF$;Q$;",2:FIELD#2,2 AS KP$"
330 LN=LN+10:PRINT#1,LN;"ONERRORGOTO25000":LN=LN+10:PRINT#1,LN;"DI
MF$(50),G$(50)"
340 PRINT#1,"32040 LSET KP$=MKI$(-1)"
350 IFRS<256 THENPRINT#1,"32045 FIELD#1,";RS;" AS DL$:LSET DL$=STR
INGS(";RS;",250):PUT1,1":ELSEPRINT#1,"32045 FIELD#1,255 AS DL$,1 A
S DX$:LSETDL$=STRING$(255,250):LSETDX$=CHR$(250):PUT1,1"
360 PRINT#1,"32050 FOR I=1 TO";MS:DIM F$(50)
370 CLS:PRINT"You may now choose, if you wish, an exit code or mes
sage. This":PRINT"is an entry the operator may use in place of a n
ormal entry":PRINT"in any field."
380 PRINT"If the operator makes this entry, the program will leave
 entry":PRINT"or update, and will return to the menu without addin
g to the":PRINT"file or altering any more information on the disk.
 Previous":PRINT"changes remain, though."
390 PRINT"Do you wish an exit code to be used (Y/N)? ";
400 OO$=INKEY$:IF OO$="" THEN 400:ELSECV$=OO$:GOSUB20:OO$=CV$:PRIN
TOO$:IFOO$<>"Y"ANDOO$<>"N"THEN390
410 IFOO$="Y"THENPRINT"What is your message or code?";:LINEINPUT O
T$
420 CLS:PRINT"Now we must describe the positions of the fields in
the data":PRINT"file. The fields may be smaller than you think nec
essary.":PRINT"Numbers can be PACKED --squeezed-- into less space
in the file":PRINT"than they take to print on the screen.";
430 PRINT" Whole numbers in the":PRINT"range -32767 to 32767 can b
e stored as packed integers, taking":PRINT"up only two spaces. Num
bers needing up to six digits of":PRINT"accuracy can be stored in
four spaces, and those needing up to"
440 PRINT"16 digits of accuracy can be stored in 8 spaces (double"
:PRINT"precision). Whole numbers in the range -127 to 127 can be":
PRINT"stored as packed half-precision numbers, and take only one"
450 PRINT"space. Please take care to allocate the correct number o
f":PRINT"spaces for each field. You must proceed from left to righ
t in":PRINT"your record as you allocate space. You have ":PRINTRS;
"spaces in each record, and up to";(47+RS+ABS(RS-47))/2;
460 PRINT"fields."
470 PRINT"TYPE IN 999 FOR THE NUMBER OF SPACES TO ALLOCATE WHEN DO
NE."
480 A=1
490 PRINT"FIELD #";A;"USES: ";:LINE INPUT F$(A):EX=EX+VAL(F$(A)):I
FEX>RS AND F$(A)<>"999"THENPRINT"EXCEEDS RECORD LENGTH. TRY AGAIN.
":EX=EX-VAL(F$(A)):GOTO490
500 IF F$(A)="999" THEN 560:ELSEIFVAL(F$(A))=0THENPRINT"INVALID.":
GOTO490
510 PRINT"Is this field size ok (Y/N)?";
520 EX$=INKEY$:IFEX$=""THEN520:ELSECV$=EX$:GOSUB20:EX$=CV$:PRINTEX
$:IFEX$="N"THENEX=EX-VAL(F$(A)):GOTO490
530 IF EX$<>"Y"THEN510
540 IF RS-EX<1THENA=A+1:GOTO560
550 PRINT"All right. You have";RS-EX;"spaces left in your record."
:A=A+1:GOTO490
560 LN=LN+10:PRINT#1,LN;"REM*SET UP FIELDS IN THE DATA RECORD*":A=
A-1:SL=LN:OF$="GOSUB36000":FORI=1TOA:PRINT#1,36000+I;"F%(";MID$(ST
R$(I),2);")="+";F$(I):NEXT
570 PRINT#1,"36000 CD%=0":PRINT#1,"55 DIM F%(";A;")":PRINT#1,"3681
0 FOR ZZ=1TO";A
580 LN=LN+10:PRINT#1,LN;"FORI=1TO";A;":G$(I)=";Q$;Q$;":NEXT:FC=0:C
LS:UF=0:GS=";Q$;Q$;":";OF$:LN=LN+10:PRINT#1,LN;"PRINT TAB(21);";Q$
;"PRODUCED BY THE CREATOR";Q$
590 LN=LN+10:PRINT#1,LN;"PRINT TAB(";(64-LEN(TI$))/2;")";Q$;TI$;Q$
630 LN=190:IF OO$<>"Y"THEN650
640 PRINT#1,LN;"PRINT";Q$;"To return to the menu from entry or upd
ate without entering or";Q$:LN=LN+10:PRINT#1,LN;"PRINT";Q$;"updati
ng your record, type in ";OT$;" for a field entry.";Q$
650 LN=LN+10:PRINT#1,LN;"ANS=INKEY$:IFANS=";Q$;Q$;"THEN";LN;":ELSE
 ON(INSTR(";Q$;"EeLlSsUuDdIiXx";Q$;",ANS)-1)/2+1 GOTO 1000,10000,3
5000,11000,12000,32000,38000,";LN+20:LN=LN+10:PRINT#1,LN;"GOTO";SL
:LN=LN+10:PRINT#1,LN;"CLOSE:NEW":DIMD%(50)
660 CLS:PRINT"Which field is the key field? Input the field number
:";:INPUT KF
670 PRINT#1,"35005 GET2,1:KP=CVI(KP$):FORK=1TOKP:RP=K:GET1,K":PRIN
T#1,"35010 IF F$(";KF;")>CHR$(249)THEN35990"
680 PRINT#1,"35000 CLS:INPUT";Q$;"Field number to scan (1-";A;")";
Q$;";NS:IFNS<1ORNS>";A;"THEN35000":PRINT#1,"35990 NEXTK:GOTO";SL
690 LN=1000:FOR QR=1TOKF-1:FT=FT+VAL(F$(QR)):NEXT:GOSUB700:GOTO810
700 CLS:PRINT"Now we must describe the input prompts the operator
will see,":PRINT"the edits to be performed, and the kind of data f
or each field.":PRINT"To do this, we will ask a series of question
s about each field."
710 PRINT"To the question KIND OF DATA, the possible responses are
:":PRINT"N.............numeric data, not packed, stored as charac
ters."
720 PRINT"PH.............packed half-precision, stored as one cha
racter.":PRINT"PI.............packed integer data, stored as two ch
aracters."
730 PRINT"PS.......packed single precision data, stored as 4 char
acters.":PRINT"PD.......packed double precision data, stored as 8
 characters."
740 PRINT"C.....................character data, stored as char
acters.":PRINT"To the PROMPT question, type the message the operat
or will see."
750 PRINT"To the BAD INPUT IF question, type as many edits as you
wish of":PRINT"the form: >999      <";Q$;"A";Q$;"      NOT NUMERIC
     NO ENTRY":PRINT"LENGTH>9       NOT ALPHA     CONTAINS";Q$;"E
";Q$;"      >=4      <>6"
760 PRINT"Note the quotes around alpha edits. Depress any key for
more:";
770 PZ$=INKEY$:IF PZ$=""THEN770
780 CLS:PRINT"To the question ERROR MESSAGE, type the message the
operator":PRINT"will see if that error is made. You may signal com
pletion of":PRINT"edits for any field by just depressing the enter
 key. To repeat"
790 PRINT"these directions, type HELP. NOW DEPRESS ANY KEY TO CONT
INUE:";
800 PZ$=INKEY$:IFPZ$=""THEN800:ELSEPRINT:RETURN
810 FORI=1TOA:I$=MID$(STR$(I),2):PRINT"FIELD #";I;" LENGTH=";:L=VA
L(F$(I)):PRINTL
820 P$="":PRINT"PROMPT:";:LINEINPUTP$:IFP$="HELP"THENGOSUB700:GOTO
820
830 INPUT"Should the prompt and reply be on the same line (Y/N):";
NY$:CV$=NY$:GOSUB20:NY$=CV$:IFCV$="Y"THENNY$=";":ELSENY$=""
840 K$="":PRINT"KIND OF DATA:";:IFL<>1ANDL<>2ANDL<>4ANDL<>8 THEN P
RINT"C":K$="C":ELSE LINEINPUTK$:CV$=K$:GOSUB20:K$=CV$:IFK$="HELP"T
HENGOSUB700:GOTO840
850 IFK$="PH"ORK$="PI"ORK$="PS"ORK$="PD"ORK$="C"ORK$="N"THEN870
860 PRINT"INVALID DATA TYPE: USE PH, PI, PS, PD, N OR C ONLY!":GOT
O840
870 IF (K$="PH"ANDL=1)OR(K$="PI"ANDL=2)OR(K$="PS"ANDL=4)OR(K$="PD"
ANDL=8)OR(K$="N")OR(K$="C")THEN890
880 PRINT"INCORRECT LENGTH FOR VARIABLE TYPE!":GOTO840
890 LN=LN+10:PRINT#1,LN;"PRINT";Q$;P$;Q$;NY$:BL=LN:IFK$<>"C"THENPR
INT#1,BL+1;"IFUF<>0ANDGS=";Q$;Q$;"THENGS=GS(";I$;")"
900 LN=LN+10:PRINT#1,LN;"LINEINPUTGS(";I$;")";:D%(I)=LN:IFOO$="N"T
HENPRINT#1," ":ELSEPRINT#1,":IFGS(";I$;")=";Q$;OT$;Q$;"THEN";SL
910 IFK$<>"PI"ANDK$<>"PH"THEN930
920 PRINT#1,LN+1;"IFUFTHENGOSUB40000":S1%=1
930 IFK$<>"PS"THEN950
940 PRINT#1,LN+1;"IFUFTHENGOSUB40500":S2%=1
950 IFK$<>"PD"THEN970
960 PRINT#1,LN+1;"IFUFTHENGOSUB40500":S2%=1
970 IFK$<>"PI"THEN990
980 LN=LN+10:PRINT#1,LN;"IF ABS(VAL(GS(";I$;")))>32767THENPRINT";Q
$;"NUMBER OUT OF RANGE. MUST BE FROM -32767 TO 32767.";Q$;":GOTO";
BL
990 IFK$<>"PH"THEN1010
1000 LN=LN+10:PRINT#1,LN;"IF ABS(VAL(GS(";I$;")))>127THENPRINT";Q$
;"NUMBER OUT OF RANGE. MUST BE FROM -127 TO 127.";Q$;":GOTO";BL
1010 IFK$="PI"ORK$="PH"THENLN=LN+10:PRINT#1,LN;"IFGS(";I$;")=";Q$;
"0";Q$;"THEN";LN+20
1020 IFK$<>"PI"ANDK$<>"PH"THEN1040
1030 LN=LN+10:PRINT#1,LN;"IFINT(VAL(GS(";I$;")))<>VAL(GS(";I$;"))T
HENPRINT";Q$;"MUST BE A WHOLE NUMBER.";Q$;":GOTO";BL
1040 BI$="":PRINT"Bad input if:";:LINE INPUTBI$:CV$=BI$:GOSUB20:BI
$=CV$:IFBI$=""THEN1260
1050 IF INSTR(BI$,"LENGHT")>0THENPRINT"MISSPELLED WORD":GOTO1040
1060 IFBI$="HELP"THENGOSUB700:GOTO1040
1070 PRINT"Error message:";:LINEINPUTEM$:IFEM$="HELP"THENGOSUB700:
GOTO1070
1080 PRINT"Is this a fatal error or a non-fatal error? A fatal err
or":PRINT"simply means the operator must try inputting this field
until":PRINT"the input is correct; a non fatal error means the err
or message"
1085 PRINT"will be printed as a warning, but the (possibly) errone
ous":PRINT"input will be accepted. Depress F if the error is FATAL
, or any":PRINT"other key, if not a fatal error: ";
1090 FE$=INKEY$:IFFE$=""THEN1090:ELSECV$=FE$:GOSUB20:FE$=CV$:PRINT
FE$:IFFE$="F"THENFE$=":GOTO"+STR$(BL):ELSEFE$=" "
```

Listing 1 continued

I don't think you can buy a better accounting package. I know you can't buy a cheaper one.

I'm Irwin Taranto. For the past five years, I've been developing my accounting systems, and they've gained a bit of reputation.

Now they're ready for almost any small computer: IBM PC and compatible machines, CP/M machines and the TRS-80. General Ledger, Accounts Payable, Accounts Receivable (Open Item or Balance Forward), Payroll/Job Costing and Inventory Control. Five interactive systems that handle the accounting for thousands of small businesses throughout the world. I also offer a Personal Accounting System.

Critical acclaim One magazine said my systems were "an impressive product at a very reasonable price." And that was when they cost three to four times as much as they do now.

Another magazine, in a general review of accounting systems, said that among all manufacturers, only Taranto and one other were "noted for their support."

Personal support The magazine didn't exaggerate. I think I offer the best support in the microcomputer industry. When you buy my systems, you also buy a phone number. If you have a problem, call and we'll fix it. If the problem's tough enough, I'll get on the phone and straighten it out myself.

A rock-bottom price I sell these systems for $99 each. You can buy the full interactive set for less than $500.

You might wonder how anything priced so low can be taken seriously. The answers are simple. I've sold thousands of these systems, and I paid off my development costs a couple of years ago. Also, I'm selling to you direct. I can bring the price down so low no other serious software can compete.

A free trial No software is worth buying until you've seen it work. So when you buy my systems, you get a demo disk wrapped separately from the actual software. Take out the demo, try it and get comfortable with it. Then, when you decide it's a good deal, unwrap the real disk and go ahead. If you're not happy with the demo for any reason at all, just pack up the unopened disk and send it back to me within 30 days. I'll refund the full $99.

But I'll be surprised. If there's anything better in the market, I haven't seen it yet.

Simple ordering Just call and tell us what you need. We'll ask you a couple of questions about your equipment and handle the whole transaction in one phone call.

Toll free: (800) 227-2868.
In California: (415) 472-2670.

Taranto
& ASSOCIATES, INC. ↙70

Post Office Box 6216, 121 Paul Drive, San Rafael CA 94903.

↙ See List of Advertisers on Page 211

CP/M or MS-DOS; the previous ones needed some surgery.

Cheapsort, Anyone?

The TRS-80 Sort/Merge utility (see Program Listing 7) requires a DOS that recognizes the CMD"0" function, which in turn sorts a single-dimensioned string array. It can sort up to 32,767 records on up to as many as 50 keys, ascending or descending, at the same time, regardless of your DOS version.

The keys can be string, byte, binary, or floating point. Floating point keys must be in Microsoft floating point format but can be of any length, from 3 to 8 or more bytes each. The sort, an adaptation of a sort I wrote for CP/M, runs under the standard TRS-80 Basic so it's not very fast, but I can't understand why something like this wasn't made available as public domain software a long time ago.

If it weren't for the garbage-collection time, the sort could be faster. If I can buy a copy of the Microsoft Basic compiler for the Model III TRSDOS, I may produce compiled versions of the sort that won't require the CMD"0" function and will sort a Model I disk in fewer than 10 minutes, if my times under CP/M are any indication.

On my Altos (4 MHz Z80 and CP/M 2.2), I was able to sort, merge,

Listing 1 continued

```
1100 IFINSTR(BI$,"NOT NUMERIC")THENS3%=1:GOTO1130
1110 IFINSTR(BI$,"NOT ALPHA")THENS4%=1:GOTO1140ELSEIFINSTR(BI$,"NO
 ENTRY")THEN1150
1120 IFINSTR(BI$,"CONTAINS")THEN1160ELSEIFINSTR(BI$,"LENGTH")THEN1
200ELSE1210
1130 LN=LN+10:PRINT#1,LN;"CD$=GS(";I$;"):E=0:GOSUB30000:IFE=1THENP
RINT";Q$;EM$;Q$;FE$:GOTO1040
1140 LN=LN+10:PRINT#1,LN;"CD$=GS(";I$;"):E=0:GOSUB31000:IFE=1THENP
RINT";Q$;EM$;Q$;FE$:GOTO1040
1150 LN=LN+10:PRINT#1,LN;"IFLEN(GS(";I$;"))=0THENPRINT";Q$;EM$;Q$;
FE$:GOTO1040
1160 FH=INSTR(BI$,Q$):IFFH<1THENPRINT"YOU FORGOT QUOTES.":GOTO1040
1170 IFFH<1THEN1190:ELSE FJ=INSTR(FH+1,BI$,Q$):IF FJ<1THENBI$="":G
OTO1160
1180 FH$=MID$(BI$,FH+1,FJ-FH-1):LN=LN+10:PRINT#1,LN;"IFINSTR(GS(";
I$;"),";Q$;FH$;Q$;")>0THENPRINT";Q$;EM$;Q$;FE$:FH=0:GOTO1040
1190 PRINT"SYNTAX ERROR. CONTAINS should not be set off by quotes.
":GOTO1040
1200 FH=INSTR(BI$,"LENGTH"):LN=LN+10:PRINT#1,LN;"IFLEN(GS(";I$;"))
";MID$(BI$,FH+6);"THENPRINT";Q$;EM$;Q$;FE$:FH=0:GOTO1040
1210 IF INSTR(BI$,"<")<1ANDINSTR(BI$,">")<1ANDINSTR(BI$,"=")<1THEN
1220ELSE1230
1220 PRINT"SYNTAX ERROR. Repeat command, please.":GOTO1040
1230 C=INSTR(BI$,Q$):IFC<1THEN1250
1240 LN=LN+10:PRINT#1,LN;"IFGS(";I$;")";BI$;"THENPRINT";Q$;EM$;Q$;
FE$:FH=0:GOTO1040
1250 LN=LN+10:PRINT#1,LN;"IFVAL(GS(";I$;"))";BI$;"THENPRINT";Q$;EM
S;Q$;FE$:FH=0:GOTO1040
1260 LN=LN+10:IFK$="PI"THENPRINT#1,LN;"GS(";I$;")=MKI$(VAL(GS(";I$
;")))"
1270 IFK$="PH"THENPRINT#1,LN;"GS(";I$;")=FNPH$(VAL(GS(";I$;")))"
1280 IFK$="PI"THENPRINT#1,11900+I;"IFUF<>";I$;"THENGS(";I$;")=MKI$
(VAL(GS(";I$;")))"
1290 IFK$="PH"THENPRINT#1,11900+I;"IFUF<>";I$;"THENGS(";I$;")=FNPH
$(VAL(GS(";I$;")))"
1300 IFK$="PI"THENPRINT#1,28000+RL;"GS(";I$;")=STR$(CVI(F$(";I$;")
))"
1310 IFK$="PH"THENPRINT#1,28000+RL;"GS(";I$;")=STR$(FNUH(F$(";I$;"
)))"
1320 IFK$="PS"THENPRINT#1,LN;"GS(";I$;")=MKS$(VAL(GS(";I$;")))"
1330 IFK$="PS"THENPRINT#1,11900+I;"IFUF<>";I$;"THENGS(";I$;")=MKS$
(VAL(GS(";I$;")))"
1340 IFK$="PS"THENPRINT#1,28000+RL;"GS(";I$;")=STR$(CVS(F$(";I$;")
))"
1350 IFK$="PD"THENPRINT#1,LN;"GS(";I$;")=MKD$(VAL(GS(";I$;")))"
1360 IFK$="PD"THENPRINT#1,11900+I;"IFUF<>";I$;"THENGS(";I$;")=MKD$
(VAL(GS(";I$;")))"
1370 IFK$="PD"THENPRINT#1,28000+RL;"GS(";I$;")=STR$(CVD(F$(";I$;")
))"
1380 IFK$="N"ORK$="C"THENPRINT#1,RL+28000;"GS(";I$;")=F$(";I$;")"
1390 RL=RL+10:LN=LN+10:PRINT#1,LN;"RETURN":NEXT
1400 PRINT#1,"1006 ONFCGOSUB";:FORI=1TOA-1:PRINT#1,MID$(STR$(D%(I)
-10),2,4);",";:NEXT:PRINT#1,MID$(STR$(D%(A)-10),2)
1410 PRINT#1,"1007 IFFC<";A;"THEN1005ELSE";LN+10
1420 PRINT#1,"29010 FORI=1TO";A:DIMH$(50):CLS:PRINT"When the recor
d is displayed, what title should be used for":FORI=1TOA:PRINT"FIE
LD NUMBER:";I;:LINEINPUTH$(I):PRINT#1,22000+I;"DATA";Q$;H$(I);Q$:N
EXT
1430 CLS:PRINT"Do you want to display computed data with your reco
rd (Y/N)?";
1440 DC$=INKEY$:IFDC$=""THEN1440:ELSECV$=DC$:GOSUB20:DC$=CV$:PRINT
DC$:IFINSTR("YN",DC$)<1THEN1430:ELSEIFDC$="N"THEN1510
1450 CLS:PRINT"All right. You may display computed data on as many
 fields as":PRINT"you desire. However, you should remember that, t
o do this, you":PRINT"will have to tell me in some detail just wha
t you want printed"
1460 PRINT"and how it should be printed.":PRINT"All fields are ava
ilable for printing or calculations. Any":PRINT"operation allowed
in BASIC is permitted, including all string,":PRINT"logical, and m
athematical operations."
1470 PRINT"HOWEVER: ALL FIELDS ARE TO BE TREATED AS STRINGS. There
fore,":PRINT"if you want to display the sum of field 7 and field 9
, you must":PRINT"tell me to display VAL(G$(7))+VAL(G$(9)). All fi
elds must be"
1480 PRINT"referenced by means of this G$() array, where the subsc
ript":PRINT"(number inside parentheses) is the number of the field
 with":PRINT"which you are working. I will ask you for a command l
ine, a"
1490 PRINT"display title, and a print format for each computed or
derived":PRINT"field you want to display. NOW DEPRESS ANY KEY TO C
ONTINUE.";
1500 DC$=INKEY$:IFDC$=""THEN1500:ELSECLS:JL=29030:GOSUB2000
1510 IF RS<256THENPRINT#1,"12910 FIELD#1,";RS;"AS DL$:LSET DL$=STR
ING$(";RS;",250):PUT1,RP:LSET KP$=MKI$(-RP):PUT2,KP:GOTO";SL
1520 IFRS=256THENPRINT#1,"12910 FIELD#1,255AS DL$,1 AS DX$:LSET DL
$=STRING$(255,250):LSETDX$=CHR$(250):PUT1,RP:LSET KP$=MKI$(-RP):PU
T2,KP:GOTO";SL
1530 PRINT#1,"10860 GOTO";SL
1540 PRINT#1,"11840 IFUF>";A;"ORUF<0THENPRINT";Q$;"INVALID FIELD."
;Q$;":GOTO11810":PRINT#1,"11850 IFUF=0THEN";SL:PRINT#1,"11860 PRIN
T@960,STRING$(63,32)::PRINT@960,";Q$;Q$;";"
1545 PRINT #1,"11865 ONUFGOSUB";
1550 FORTZ=1TOA-1:PRINT#1,MID$(STR$(D%(TZ)-10),2);",";:NEXT:PRINT#
1,MID$(STR$(D%(A)-10),2):PRINT#1,"11870 IFUF<>";KF;"THEN11900ELSEZ
Z$=STRING$(F%(";KF;"),32):LSETZZ$=G$(";KF;
1560 IFRS<256THENPRINT#1,"):FIELD#1,";RS;"AS DL$:LSET DL$=STRING$(
";RS;",250):PUT1,RP:LSET KP$=MKI$(-RP):PUT2,KP:GOSUB26000:KP=RP"
1570 IFRS=256THENPRINT#1,"):FIELD#1,255AS DL$,1AS DX$:LSET DL$=STR
ING$(255,250):LSETDX$=CHR$(250):PUT1,RP:LSET KP$=MKI$(-RP):PUT2,KP
:GOSUB26000:KP=RP"
1580 PRINT#1,"11999 FORI=1TO";A;":LSET F$(I)=G$(I):NEXT:PUT1,RP:G$
=";Q$;Q$;":GOTO11200":PRINT#1,"27010 CLS:GOSUB";D%(KF)-10;":KF$=ST
RING$(F%(";KF;"),32):LSETKF$=G$(";KF;")"
1590 PRINT#1,"27040 KP=KP+1:IFKP>";MS;"ORKP<2THENKP=1:REM CIRCULAT
E AROUND"
1600 PRINT#1,"27050 GET2,KP:RP=CVI(KP$):IFRP<-1THEN27040:ELSEIFRP=
0THENERROR62:ELSEIFRP=-1THENPRINT";Q$;"RECORD NOT FOUND.";Q$;":FOR
I=1TO2000:NEXT:RUN"
1610 PRINT#1,"27055 GET1,RP:IFLEFT$(F$(";KF;"),LEN(ZZ$))=ZZ$THENRE
TURN:ELSE27040"
1620 '
1630 LN=LN+10:PRINT#1,LN;"ZZ$=STRING$(F%(";KF;"),32):LSETZZ$=G$(";
MID$(STR$(KF),2);"):GOSUB26000:KP=RP"
1640 PRINT#1,"26040 RP=";MID$(STR$(MS),2);"*RP/9999:RP=FIX(RP):RET
URN"
1650 LN=LN+10:PRINT#1,LN;"REM*LOOK FOR RECORD SPACE":LN=LN+10:PRIN
T#1,LN;"GOSUB";LN+10;":GOTO";LN+60:LN=LN+10:PRINT#1,LN;"KP=KP+1:IF
KP>";MS;"ORKP<2THENKP=2":PRINT#1,"11890 KP=KP+1:IFKP>";MS;"ORKP<2T
HENKP=2"
1660 PRINT#1,"11895 GET2,KP:IFCVI(KP$)<>-1THEN11890":PRINT#1,"1189
6 GET2,1:RP=CVI(KP$)+1:LSETKP$=MKI$(RP):PUT2,1:LSETKP$=MKI$(RP):PU
T2,KP"
1670 LN=LN+10:PRINT#1,LN;"GET2,KP:RP=CVI(KP$):IFRP>0THEN";LN-10:LN
=LN+10:PRINT#1,LN;"IFRP<-1THENRETURN":LN=LN+10:PRINT#1,LN;"IFRP=0T
HENERROR62":LN=LN+10:PRINT#1,LN;"RETURN":LN=LN+10:PRINT#1,LN;"FORK
=1TO";A
1680 LN=LN+10:PRINT#1,LN;"LSETF$(K)=G$(K):NEXT"
1690 LN=LN+10:PRINT#1,LN;"IFRP<-1THENPUT1,ABS(RP):LSETKP$=MKI$(ABS
(RP)):PUT2,KP:ELSEPK=KP:GET2,1:KP=1+CVI(KP$):LSETKP$=MKI$(KP):PUT2
,PK:PUT1,KP:PUT2,1"
1700 CLS:PRINT"Now we are almost done. We need to ask the operator
 if he is":PRINT"done with entry or update. You may choose the mes
sage you wish."
1710 PRINT"The operator's input will be limited to a single key de
pression.":PRINT"I would suggest something of the form:"
1720 PRINT"Are you finished with entry? (Y/N)      Enter your choic
e below:":LINEINPUTTM$
1730 PRINT"If the operator is NOT done, what should the reply be?"
;
1740 ND$=INKEY$:IFND$=""THEN1740:ELSECV$=ND$:GOSUB20:ND$=CV$:PRINT
ND$:IFND$<" "THEN1730
1750 PRINT"If the operator IS done, what should the reply be?";
1760 ID$=INKEY$:IFID$=""THEN1760:ELSECV$=ID$:GOSUB20:ID$=CV$:PRINT
ID$:IFID$<" "THEN1750:ELSELN=LN+10:PRINT#1,LN;"PRINT";Q$;TM$;Q$;";
";:LN=LN+10:PRINT#1,LN;"TM$=INKEY$:IFTM$="";Q$;Q$;"THEN";LN;":ELSEPR
INTTM$":BL=LN-10
1770 LN=LN+10:PRINT#1,LN;"REM*IF DONE, RETURN TO MENU ELSE CONTINU
E ENTRY OR RE-DISPLAY CHOICE":LN=LN+10:PRINT#1,LN;"IFTM$=";Q$;ID$;
Q$;"THEN";SL
1780 LN=LN+10:PRINT#1,LN;"IFTM$<>";Q$;ND$;Q$;"THENPRINT";Q$;"PLEAS
E ANSWER ";ND$;" OR ";ID$;Q$;":GOTO";BL:LN=LN+10:PRINT#1,LN;"FORI=
1TO";A;":GS(I)=";Q$;Q$;":NEXT:FC=0:CLS:UF=0:GS=";Q$;Q$;":GOTO1000"
1790 PRINT"Do you want to cut down on remarks in the generated pro
gram?":PRINT"Please answer Y or N: ";
1800 CV$=INKEY$:IFCV$=""THEN1800:ELSEGOSUB20:PRINTCV$
1810 IFCV$="Y"THENOPEN"I",2,"C451MIN/LIB":ELSEIFCV$="N"THENOPEN"I"
,2,"C451/LIB":ELSE1790
1820 CLS:PRINT"WAIT A MOMENT WHILE I FINISH YOUR PROGRAM."
1830 IF EOF(2)THEN 1900
1840 LINEINPUT#2,A$
1850 IFS3%<1ANDVAL(A$)>29999ANDVAL(A$)<30999THEN1830
1860 IFS4%<1ANDVAL(A$)>30998ANDVAL(A$)<31900THEN1830
1870 IFS1%<1ANDVAL(A$)>=40000ANDVAL(A$)<40500THEN1830
1880 IFS2%<1ANDVAL(A$)>=40500ANDVAL(A$)<41000THEN1830
1890 PRINT#1,A$:GOTO1830
1900 CLOSE
1910 CLS:PRINT"Your program has been written and saved on the spec
ified drive.":PRINT"You may now load it and run it, if you wish. R
emember, your":PRINT"data file does NOT exist until you initialize
 it!"
1920 PRINT"Thanks for running the CREATOR!":FORI=1TO5000:NEXT:NEW
1930 IF ERL=260 AND ERR=106 THEN CLOSE 1:RESUME 270
1940 IF ERR=104 THEN PRINT"You entered BASIC without specifying an
y files for disk access!":PRINT"Go back to DOS and re-enter BASIC
correctly!":NEW
1950 IF (ERR=126 OR ERR=38)AND ERL=260 THEN PRINT"FILE ALREADY EXI
STS. DEPRESS C TO WRITE OVER IT.":PRINT"DEPRESS ANY OTHER KEY TO R
E-START."
1960 IF (ERR<>126 AND ERR<>38) OR ERL<>260 THEN 1990
1970 XN$=INKEY$:IF XN$=""THEN1970:ELSEIFXN$="C"THENCLOSE1:RESUME27
0
1980 END
1990 PRINT"ERROR NUMBER";ERR;"AT LINE";ERL:ON ERRORGOTO0
2000 PRINT"REMEMBER, ALL FIELDS ARE STRINGS WITHIN THE ARRAY G$().
"
2010 PRINT"TO PERFORM ARITHMETIC ON A FIELD, IT IS NECESSARY TO TA
KE ITS":PRINT"VALUE FIRST, WITH THE VAL COMMAND. FOR EXAMPLE, TO P
RINT THE":PRINT"SUM OF FIELDS 1,2, AND 3 YOU WOULD TYPE THE COMMAN
D LINE"
2020 PRINT"VAL(GS(1))+VAL(GS(2))+VAL(GS(3))"
2030 PRINT:PRINT"Now, type in a command line which will tell me wh
at operations":PRINT"I should perform. Please don't type more than
 240 characters.":PRINT"HIT THE RETURN WHEN YOU ARE DONE."
```

Listing 1 continued

VS-100 :

Will YOU like your voice synthesizer ?

"The VS-100 has been THE most exciting addition since I purchased the computer. It suddenly made it a very intriguing gizmo to all my friends and suddenly they ALL want to buy a computer in order purchase your VS-100."
--C.M. New York

"I love the VS-100. It has turned out to be everything I expected. The manual especially will make it easy to use in assembly applications." --N.N. US Army

"Even my wife was impressed with it, and she is darned hard to convince that I need another 'add-on' for the computer." --M.H. Kentucky

"Fantastic piece of hardware!!! Great price, great software. You have done it again; come up with something great for the TRS-80 computer. THANK YOU." --J.Y. South Carolina

"It's easy to use and sounds as good as $400 units." -- S.C. Ohio

"As you will notice by the enclosed letter, I am placing an order for two more VS-100's for friends who have heard it and can't wait to obtain one." --R.F. Australia

"Well done -- a nice piece of equipment at a reasonable price ! encourage product reviews with 80-Micro and Computer User magazines."
--A.S. Canada

"I am very surprised and pleased with the performance of the VS-100."
--V.G. Pennsylvania

"Simple installation and excellent software support." --G.R. South Carolina

"Impressive device at excellent price."
--V.R. Illinois

"Demo is great -- particularly 'talking clock'"
--P.L. Virginia

"...oustanding software and ease of use."
--A.O. Maryland

"Interesting, Arresting, worth the price."
--K.M. North Carolina

"I was amazed at the quality and the low, low price. I've been searching for a voice synthesizer for years. They all cost at least $150.00. Good job." --B.D. Illinois

"...love the VS-100 overall"
--R.M. California

"The included software is some of the best I've seen for both the beginner and advanced programmers. Merging to existing applications is a snap. Looking forward to many applications!" --J.M. California

"Your product, staff, and dedication to getting it right are appreciated & applauded. Keep it up!!!" --P.S. Florida

"The price was great and the hardware excellent." --T.L. Oregon

"Speech quality better than $400 models."
--J.A. Florida

"More business from me is very likely! -- The VSEDIT program is terrific. The built in amp is thoughtful." --B.G. California

"Amazing and entertaining" --W.A. Tennesse

"Very nice! Incredidibly simple to use. Good speech quality." --D.P. Wisconsin

"My compliments to the author of the phoneme editor and the machine language driver. Your VS-100 may already qualify as the best-supported voice synthesizer."
--D.D. North Carolina

DEMO HOT LINE (212) 296 0399

FOR DETAIL ON THE VS-100 VOICE SYNTHESIZER:SEE THE RED AND GREEN AD

ALPHA Products

79-04 Jamaica Ave., Woodhaven, NY 11421 (212) 296-5916

17

Toll Free Order Line
800-221-0916
Orders Only. NY & Info call (212) 296-5916.
Hours: 9-5 Eastern Standard Time

Add $ .00 per order for shipping/handling. We accept Visa, Mastercard, Checks, M.O. C.O.D. Add $3.00 extra. N.Y. Residents add sales tax. Overseas, FPO, APO: Add 10% Dealer discounts available.

and write an index file to 9,513 random integers in under 15 minutes.

By the way, the program reads the sort index file either randomly or sequentially. If you want to do binary searches of a data file, feel free to use the index this way, though it's still slower than hashing.

The Sort/Merge utility for the CP/M and PC users is faster than the Radio Shack version.

Two Utility Programs

The first utility program (REKEY/BAS) creates or re-creates a key for any random file on any field (see Program Listing 8). Using it, you can write a Creator program to match the fields for an existing file, use this utility to create the needed key file, then use the generated program to look up, update, or display any record on that key.

A second utility program (CXFER/BAS) lets you convert any existing Creator-type file to the new, more efficient format (see Program Listing 9). I don't include this program on the CP/M or PC versions since it isn't needed.

What Are the Trade-offs?

You need 48K of memory on a Model I or III for the new Creator/Reportor. However, you need a Model III or a non-TRSDOS Model I (CMD"0" again) if you want to use the sort program. By the way, you can use the machine-language sort in Lewis Rosenfelder's *BASIC Faster and Better and Other Mysteries* (IJG Inc.) on a Model I in place of the CMD"0". If you have that book or the disk that accompanies it, you can run my sort/merge on a Model I.

If you see any ads comparing the Creator to someone else's program generator, be sure it's my Creator they're comparing. Another program called the Creator is out, and it's not mine.

Ordering Information

I'll send you a guaranteed disk copy of the Creator II/Reportor, with the utility programs and Sort/Merge for $11. CP/M and IBM PC versions cost $27. This includes all postage and handling for the United States, U.S. possessions with zip codes, and Canada; orders from Mexico should add $20 and foreign orders should add $6 in U.S. funds for overseas air mail. Sorry, I cannot take charge cards. For C.O.D. or purchase orders, add $10 for my running around time.

Tell me exactly what you're ordering and for what machine and DOS. I'll send copies in Model I or Model III formats only; if I know your DOS, I may be able to advise you of any problems.

"I can't understand why something like this wasn't put in the public domain a long time ago."

Note: If you have a Model I and are running a DOS that doesn't recognize CMD"0", please tell me when ordering. I'll see that you get a copy of the sort that doesn't use the CMD"0" function. It will sort about 50 percent as fast as the CMD"0" for sorts of fewer than 100 records, and about 50 records per minute for sorts of more than 100 records. The regular version of Cheapsort can sort about 200 records per minute on my Model III, up to about 1,100 per minute on my Altos CP/M machine, and up to 1,300 per minute on the IBM PC. All sort times include all disk I/O times, and were run on floppy disk systems. Sort times on RAM disk and hard disk systems will be better.

Manuals for the new version sell for $11, including $1 postage and handling anywhere in the United States, Canada, or Mexico. You'd do well to order the manual; if your club wants 20 copies of the program, buy one disk and 20 manuals. (You could buy 20 disks, but why spend the extra dough?) Overseas, add $6 per manual for air mail.

A newsletter for Creator II users will cost $10 for one year (four issues). If you live outside the United States or Canada, add $5 annually. Please make foreign checks through a bank with a correspondent bank in the United States, if possible. Otherwise, send dollars. ■

Order from: Bruce Tonkin, 34069 Hainesville Road, Round Lake, IL 60073.

Listing 1 continued

```
2040 PRINT:PRINT"COMMAND LINE:";:LINE INPUT CV$:IFLEN(CV$)>240THEN
PRINT"TOO LONG! TRY AGAIN!":GOTO2040:ELSEIFLEN(CV$)<1THENRETURN
2050 K=0:JK=0:J=0:FORI=1TOLEN(CV$):IFMID$(CV$,I,1)=QSTHENJ=ABS(J-1
)
2060 IFJTHEN2080
2070 IFMID$(CV$,I,1)=")"THENK=K+1:ELSEIFMID$(CV$,I,1)="("THENJK=JK
+1
2080 NEXT:IFJK<>KTHENPRINT"MISHATCHED PARENTHESES SOMEWHERE. PLEAS
E TRY AGAIN."
2090 IFJTHENPRINT"MISHATCHED QUOTATION MARKS (";QS;") IN YOUR COMM
AND. PLEASE TRY AGAIN."
2100 IFJ>0OR(JK<>K)THEN2040
2110 PRINT"What title should I use for this data:";:LINE INPUT TI$
2120 IFINSTR(TI$,QS)THENPRINT"NO QUOTATION MARKS ALLOWED.":GOTO211
0
2130 JL=JL+10:PRINT#1,JL;"CL=CL+1:GOSUB41000":PRINT#1,JL+5;"PRINTC
HR$(133);";QS;TI$;QS;";CHR$(133);"
2140 PRINT"If this field is numeric, you may use a formatted print
.":PRINT"If it is not, or you do not wish to print it in a formatt
ed":PRINT"way, you need not. DO YOU WISH TO USE A FORMATTED PRINT
(Y/N)"
2150 TI$=INKEY$:IFTI$=""THEN2150:ELSEIFTI$="N"THEN2250
2160 IFTI$<>"Y"THEN2140
2170 LINEINPUT"Number of places to the left of the decimal point="
;NL$:IFVAL(NL$)<0ORVAL(NL$)>16THENPRINT"Illegal (0-16 only).":GOTO
2170
2180 LINEINPUT"Number of places to the right of the decimal point"
;NR$:IFVAL(NR$)<0ORVAL(NL$)+VAL(NR$)>16THENPRINT"Illegal number of
 places or illegal total places (0-16).":GOTO2180
2190 PRINT"Do you want to use commas (Y/N): ";
2200 NC$=INKEY$:IFNC$=""THEN2200:ELSEIFNC$<>"N"ANDNC$<>"Y"THEN2200
2210 PRINTNC$
2220 TI$="USING"+QS+STRING$(VAL(NL$),"#"):IFNC$="Y"THENTI$=TI$+","
2230 IFVAL(NR$)>0THENTI$=TI$+"."
2240 TI$=TI$+STRING$(VAL(NR$),"#")+QS+";":CV$=TI$+CV$
2250 PRINT#1,JL+6;"PRINT";CV$
2260 PRINT"Your commands have been accepted. Ready for the next co
mmand.":GOTO2030
2270 END
```

Program Listing 2. The Reportor.

```
10 'REPORTOR; WRITTEN ON 1/16/80 BY BRUCE W. TONKIN
20 'ADAPTED FOR CP/M AND MICROSOFT BASIC 5.01 ON 5/11/80
30 'FOR 4.51 ON 3/18/81, AND FOR MODEL III ON 2/20/83
40 'THIS IS PUBLIC DOMAIN SOFTWARE AND MAY NOT BE SOLD OR INCORPOR
ATED IN ANY COMMERCIAL SOFTWARE WITHOUT MY EXPRESS PERMISSION
50 CLEAR 0:CLEAR 5000:GOTO 110
60 FX=0:IFLEN(CV$)<1THENRETURN
70 FORII=1TOLEN(CV$):CV%=ASC(MID$(CV$,II,1)):IFCV%=34THENFX=ABS(FX
-1)
80 IFFX=0ANDCV%>90THENCV%=CV%AND95:MID$(CV$,II,1)=CHR$(CV%)
90 NEXT:RETURN
110 CLS:PRINTCHR$(23);"THE REPORTOR: A PROGRAM THAT":PRINT"WRITES
REPORT PROGRAMS.":PRINT"PLACED IN THE PUBLIC DOMAIN":PRINT"BY BRUC
E W. TONKIN":PRINT"COPYRIGHT, 1983, BRUCE W. TONKIN"
120 PRINT"THIS PROGRAM MAY NOT BE":PRINT"DISTRIBUTED FOR PROFIT":P
RINT"WITHOUT WRITTEN PERMISSION":PRINT"OF THE AUTHOR!"
130 PRINT"BRUCE W. TONKIN":PRINT"34069 HAINESVILLE RD.":PRINT"ROUN
D LAKE, IL 60073"
140 FOR I=1TO3000:NEXT
150 CLS
160 PRINT"This is the REPORTOR, a report-writing program. Please c
hoose":PRINT"one of the following:":PRINT:PRINT"Run a previously w
ritten program.....Depress R"
170 PRINT"Write a new report.....Depress
 W"
180 PRINT"Exit this program.....Depress
 X"
190 PRINT:PRINT"Please depress the key corresponding to your choic
e:";
200 A$=INKEY$:IFA$=""THEN200:ELSECV$=A$:GOSUB60:PRINTCV$:A$=CV$
210 ONINSTR("RWX",A$)GOTO230,240,2680
220 PRINT"YOU MUST CHOOSE R, W, OR X. PLEASE TRY AGAIN.":FORI=1TO1
500:NEXT:GOTO160
230 PRINT"What is the complete name of your program: ";:LINEINPUTC
V$:GOSUB60:RUNCV$
240 'begin
```

Listing 2 continued

Learn to Program Like a Professional!

THE COMPLETE BOOK OF RANDOM ACCESS & DATA FILE PROGRAMMING

Written for TRS-80™ I, II, & III - IBM™/PC - APPLE™ M/S - OSBORNE™ - HEATH™ - DEC™ - SUPERBRAIN™ - and all Computers using CP/M with Microsoft BASIC™

The last word on disk random access and file handling techniques, this series is intended for everyone — beginning programmers, businessmen and professionals will learn how to create custom programs to handle inventories, mailing lists, work scheduling, record keeping, and many other tasks, while more experienced programmers will learn advanced, professional programming techniques for faster, more efficient data storage and retrieval.

Although random access file handling is a matter of some complexity, the subject has been treated in a simple and down-to-earth fashion, so that anyone with some small familiarity with programming in Microsoft BASIC will be able to cope with the material. Each stage of learning uses a sample program as a starting point. The programs grow in capability and complexity as the books progress into all of the various aspects of file handling and record manipulation. An extensive effort has been made to keep the material coherent and every program line is explained in detail.

Volume I
BASIC FILE HANDLING

- The writing of a Menu to summarize program functions
- Screen format for data entry
- The creation of a basic record
- The FIELD and LSET routines for buffer preparation
- Disk storage of random access records
- Changing or editing stored records
- The LPRINT capability from disk using three different formats
- Sorting the random file
- Searching by name or key field
- Search in "next" or "prior" fashion
- Purging deleted records
- Using disk file data for calculations
- Future expansion of data fields
- Using flags to prevent program crashes
- Date setting, printer on-line and many other routines to make a program run like a commercially written program

VOLUME I **$29.95**
Option Vol. I Program Disk
TRS-80 Model I/III **$28.50**
TRS-80 Model II **$32.50**

Volume II
ADVANCED FILE HANDLING TECHNIQUES

- Relational database programming
- Comprehensive self-balancing accounting system with printouts
- Hashcoded data file manipulation -- (probably the fastest method of data retrieval). Hashing the input key and recovery method explained
- Span-blocking techniques allow creation of records longer than 256 bytes without wasted space
- Blocking & Deblocking
- Shell-Metzner sort
- In-place screen editing
- Recovery of deleted record space
- Alpha-index record retrieval
- Fast machine language/BASIC sort
- Linked list record structure and sort-merge, deleted record removal and file reorganization
- Multi-key file reorganization and record searching

VOLUME II **$29.95**
Optional Vol. II Program Disk
TRS-80 Model I/II/III **$49.95**

Listing 2 continued

```
250 '
260 PRINT"What is to be the name of your program? ";:LINEINPUTCV$:
GOSUB60:PN$=CV$:IFCV$<"A"THENPRINT"Illegal name!":GOTO260
270 IFLEN(PN$)>8THENPRINT"Name may not be more than 8 characters l
ong!":GOTO260
280 IFINSTR(PN$,":")THENPRINT"Do not include the drive number!":GO
TO260
290 IFINSTR(PN$,".")>0ORINSTR(PN$,"/")>0THENPRINT"Illegal characte
r in file name!":GOTO260
350 PRINT"The drives are numbered, 0-3. Which drive do you want yo
ur":PRINT"program written on:";
360 DR$=INKEY$:IFDR$=""THEN360:ELSEPRINTDR$
370 IFDR$<"0"ORDR$>"3"THENPRINT"Invalid drive!":GOTO350
390 PN$=PN$+"/BAS"+":"+DR$
410 ON ERROR GOTO 2640
420 OPEN"I",1,PN$:ERROR 126
430 OPEN"O",1,PN$:QS=CHR$(34)
440 CLS:PRINT"What is to be the title displayed for your report? T
ype it on":PRINT"the next line or lines."
450 LINEINPUTT$
460 PRINT"What is the name of the file you are going to read for y
our":PRINT"report? Please type in the FULL name on the next line.
Do not"
470 PRINT"omit the / and the file type extension, if present. Plea
se":PRINT"EXCLUDE the drive number, though."
490 LINEINPUTCV$:GOSUB60:F$=CV$
495 IF INSTR(F$,":")>0THENPRINT"Drive number not allowed!":GOTO460
500 PRINT"What is the record length of each record in the file? ";
:LINE INPUTRL$:RL=VAL(RL$)
510 IFRL>256ORRL<1THENPRINT"Illegal record length (1-256, please!)
":GOTO500
600 PRINT#1,"12 DF$=";QS;F$;QS
660 PRINT#1,"30 TI$=";QS;T$;QS
690 IFRL=256THENPRINT#1,"40 OPEN";QS;"R";QS;",1,DF$":ELSEPRINT#1,"
40 OPEN";QS;"R";QS;",1,DF$,";RL
700 CLS:PRINT"Now you must describe the fields in each record of y
our file,":PRINT"from left to right, in order. Input the amount of
 space each":PRINT"field takes up. When you are done, enter 999 fo
r the space."
730 A=1:DIMF%(50),D$(50)
740 PRINT"SPACE REMAINING=";RL-F%(0);"; FIELD NUMBER: ";A;"=";:LIN
EINPUTF1$:F%(A)=VAL(F1$)
750 IF F%(A)=999THEN830ELSEIFF%(A)>RLTHENPRINT"INVALID. MUST BE";R
L;"OR LESS.":GOTO740
760 F%(0)=F%(0)+F%(A):IFF%(0)>RLTHENPRINT"NOT ENOUGH SPACE LEFT FO
R THAT FIELD!":F%(0)=F%(0)-F%(A):GOTO740
780 PRINT"Is this correct (Y/N)? ";
790 JX$=INKEY$:IFJX$=""THEN790:ELSECV$=JX$:GOSUB60:PRINTCV$:JX$=CV
$:IFJX$="N"THENF%(0)=F%(0)-F%(A):GOTO740
800 IFJX$<>"Y"THEN780
810 IFRL-F%(0)<1THENA=A+1:GOTO830
820 A=A+1:GOTO740
830 XF$="GOSUB 30000"
840 A=A-1:PRINT#1,"30805 FORIX=1TO";A
850 FORI=1TOA:PRINT#1,30000+I;"FF(";I;")=";F%(I):NEXT
900 PRINT#1,"50 DIMF$(";A;"),FF(";A;"),P(";A;"),P$(";A;"),C#(50),P
#(";A;"),C(50),C$(50),H$(50)"
910 PRINT#1,"60 ";XF$:XF$="GOSUB 30000"
930 CLS:PRINT"What are your column headings to be? Please enter a
legend":PRINT"for each column you intend to use. Type 999 to end."
940 DIM C%(50):FORJ=1TO50:C%(J)=J:NEXT
950 DIM C$(50)
960 B=1
980 PRINT"Column number";B;:LINEINPUTC$(B):IFC$(B)="999"THENC$(B)=
"":B=B-1:ELSEB=B+1:GOTO980
990 J=0:FORJJ=1TOB-1:IFLEN(C$(C%(JJ)))<LEN(C$(C%(JJ+1)))THENJ=C%(J
J):C%(JJ)=C%(JJ+1):C%(JJ+1)=J:J=J-1
1000 NEXT:IFJTHEN990:ELSELN=70
1010 FORI=1TOB:I$=MID$(STR$(I),2):PRINT#1,LN;"H$(";I$;")=";QS;C$(I
);QS:LN=LN+10:NEXT
1020 M$=STRING$(50,32):GOSUB 2055
1030 DIMT(50)
1040 CLS:PRINT"Now we need to set the tabs for your";B;"columns. I
ndicate":PRINT"the tab setting as a number, from 1 to 255."
1050 FORI=1TOB:PRINT"Tab setting for column number";I;":";:LINEINP
UTF1$:T(I)=VAL(F1$):IFT(I)>255ORT(I)<1THENPRINT"NOT A VALID TAB SE
TTING!":I=I-1
1070 NEXT
1090 PRINT#1,"31070 DATA ";
1100 FORI=1TOB-1:PRINT#1,MID$(STR$(T(I)),2);",";:NEXT
1130 PRINT#1,MID$(STR$(T(B)),2)
1140 LN=LN+10:PRINT#1,LN;"PG=1:'INITIALIZE THE PAGE COUNTER"
1150 PRINT#1,LN+5;"'NOW FOLLOWS THE RECORD RETRIEVAL SECTION"
1160 PRINT"Do you want the pages numbered (Y/N)? ";
1170 CV$=INKEY$:IFCV$=""THEN1170:ELSEGOSUB60:PRINTCV$:PG$=CV$:IFPG
$<>"Y"ANDPG$<>"N"THENPRINT"ILLEGAL!":GOTO1160
1180 LN=LN+10:PRINT#1,LN;"FORI=1TO32767"
1190 PRINT#1,LN+5;"IFSR$<>";QS;"N";QS;"THENINPUT#2,I"
1200 LN=LN+10:KZ=LN
1210 IFRL<>256THENPRINT#1,LN+1;"IFZU$=STRING$(";RL;",0)THEN20000":
ELSEPRINT#1,LN+1;"IFZU$=STRING$(255,0)THEN20000"
1220 LN=LN+10:PRINT#1,LN;"ON ERROR GOTO 20000"
1230 IFPG$="Y"THENPRINT"How many lines per page? ";:LINEINPUTF1$:L
P=VAL(F1$)-1
1240 CLS:J=0:FORI=1TOA:IFF%(I)=1ORF%(I)=2ORF%(I)=4ORF%(I)=8THENJ=1
:I=A
1250 NEXT:IFJ=0THENFORI=1TOA:D$(I)="C":NEXT:GOTO1500
1260 PRINT"You have some fields which might be packed data. Please
 tell":PRINT"me if they are PH (packed half precision), packed int
eger (PI),":PRINT"packed single precision (PS) or packed double pr
ecision (PD)."
1300 FORI=1TOA:IFF%(I)<>1ANDF%(I)<>2ANDF%(I)<>4ANDF%(I)<>8THEND$(I
)="C":GOTO1380
1310 PRINT"Field number";I;":Length is";F%(I);": Data type is: ";:
INPUTCV$:GOSUB60:D$(I)=CV$:L=F%(I)
1330 IFD$(I)<>"N"ANDD$(I)<>"C"ANDD$(I)<>"PH"ANDD$(I)<>"PI"ANDD$(I)
<>"PS"ANDD$(I)<>"PD"THENPRINT"ILLEGAL VARIABLE TYPE!":GOTO1310
1340 IF(CV$="PH"ANDL=1)OR(CV$="PI"ANDL=2)OR(CV$="PS"ANDL=4)OR(CV$=
"PD"ANDL=8)THEN1375
1350 IFCV$="N"ORCV$="C"THEN1375
1370 PRINT"ILLEGAL LENGTH FOR THIS VARIABLE TYPE!":GOTO1310
1375 PRINT"Is this correct (Y/N)? ";
1376 CV$=INKEY$:IFCV$=""THEN1376:ELSEGOSUB60:PRINTCV$
1377 IFCV$="N"THEN1310:ELSEIFCV$<>"Y"THEN1375
1380 NEXT
1500 FORI=1TOA:I$=MID$(STR$(I),2)
1510 LN=LN+10:IFD$(I)="C"ORD$(I)="N"THENPRINT#1,LN;"P$(";I$;")=F$(
";I$;")"
1520 IFD$(I)="PH"THENPRINT#1,LN;"P#(";I$;")=ASC(F$(";I$;"))-128"
1530 IFD$(I)="PI"THENPRINT#1,LN;"P#(";I$;")=CVI(F$(";I$;"))"
1540 IFD$(I)="PS"THENPRINT#1,LN;"P#(";I$;")=CVS(F$(";I$;"))"
1550 IFD$(I)="PD"THENPRINT#1,LN;"P#(";I$;")=CVD(F$(";I$;"))"
1560 NEXT
1570 CLS:PRINT"Now we must describe what goes into the columns of
the report.":PRINT"You should write sentences of the following for
m:"
1580 PRINT"Column name=FIELD(9)"
1590 PRINT"Column name=FIELDS(5)"
1600 PRINT"IF Column.name.1=Column.name.2 THEN Column.name.3=";Q$;
"SAME";Q$
1610 PRINT"V1=Name1/Name2:V2=100*V1:Name3=V2"
1620 PRINT"IF Name1=";Q$;"OVERDUE";Q$;" THEN SKIP"
1630 PRINT"IF Name2<1 THEN SKIP"
1640 PRINT"Please note that you should use the HEADING NAMES as yo
ur":PRINT"variables. Variables beginning with the letter V are wor
k":PRINT"variables. You may issue a number of commands on one line
, if":PRINT"you separate them with a colon (:)"
1650 PRINT"Also, note the quotation marks around string constants,
":PRINT"and the $ sign used for string fields.":PRINT"DEPRESS ANY
KEY FOR MORE INFORMATION: ";
1660 CV$=INKEY$:IFCV$=""THEN1660
1670 PRINT:PRINT"YOU MAY USE THE FOLLOWING VARIABLE NAMES:"
1680 FORI=1TOB:PRINTC$(I).:NEXT:PRINT:PRINT"TYPE 999 TO EXIT."
1690 PRINT"READY FOR YOUR COMMAND: ";
1700 LINE INPUT SX$:IFVAL(SX$)=999THEN2050
1710 FORI=1TOB:J=C%(I)
1720 F1%=INSTR(SX$,C$(J)):IFF1%<1THEN1790
1730 J$=MID$(STR$(J),2):I$=MID$(STR$(I),2):IFMID$(H$,J,1)="S"THENF
1$="CS(":ELSEF1$="C#("
1740 F1$=F1$+J$+")":GOSUB5000:GOTO1720
1790 NEXT
1800 F1%=INSTR(SX$,"FIELD"):IFF1%<1THEN1850
1810 IF F1%=1THENPRINT"FIELDS CANNOT BE RE-DEFINED.":GOTO1690
1815 F2%=INSTR(F1%,SX$,"("):F3%=INSTR(F1%,SX$,"$"):F4%=INSTR(F1%,S
X$,"#"):IF(F4%=0ORF4%>F2%)AND(F3%=0ORF3%>F2%)THENSY$="#":ELSESY$="
"
1820 SX$=LEFT$(SX$,F1%-1)+"P"+SY$+MID$(SX$,F1%+5):GOTO1800
1850 PRINT"I interpret your command to be:":PRINTSX$:PRINT"Is this
 correct (Y/N)? ";
1860 CV$=INKEY$:IFCV$=""THEN1860:ELSEGOSUB60:PRINTCV$:IFINSTR("YN"
,CV$)<1THEN1850
1870 IFCV$="N"THENPRINT"INSTRUCTION DELETED.":GOTO1690
1880 F1%=INSTR(SX$,"SKIP"):IFF1%>1THENSX$=LEFT$(SX$,F1%-1)+"19000"
+MID$(SX$,F1%+4):GOTO1880
1890 LN=LN+10:PRINT#1,LN;SX$:GOTO1670
2050 LN=LN+10:PRINT#1,LN;"FORJ=1TO";B;":T#(J)=T#(J)+C#(J):NEXT"
2052 GOTO 2310
2055 CLS:PRINT"All right. Now we need to define the column print f
ormats.":PRINT"I will ask you to tell me if a column will contain
letters or":PRINT"numbers. If a column contains numbers, but those
 numbers come"
2060 PRINT"from an unpacked field, and are not used as numbers (fo
r ":PRINT"example, phone numbers or ID numbers), then that column
should":PRINT"be formatted as if it were LETTERS. On the other han
d, a "
2070 PRINT"column containing the result of a computation, or a col
umn":PRINT"containing numbers coming from a PACKED numeric field s
hould":PRINT"be formatted as NUMBERS. If you do not tell me the co
rrect"
2080 PRINT"format, your report will contain zeros or blanks."
2090 DIMCF$(B):FORI=1TOB
2100 PRINT"COLUMN NAME: ";C$(I);": NUMBERS OR LETTERS (N/L)? ";
2110 CV$=INKEY$:IFCV$=""THEN2110:ELSEGOSUB60:PRINTCV$:IFINSTR("NL"
,CV$)<1THEN2100
2120 IFCV$<>"L"THEN2200
2130 PRINT"How many characters might be printed in this column";:I
NPUTFX:IFFX<1ORFX>255THENPRINT"Invalid!":GOTO2130
2140 IFFX=1THENCF$(I)="!":ELSECF$(I)="%"+STRING$(FX-2,32)+"%"
2150 GOTO2300
2200 PRINT"How many digits will be printed to the LEFT of the deci
mal";:INPUTFX:IFFX<0ORFX>16THENPRINT"Illegal!":GOTO2200
2210 CF$(I)=STRING$(FX,"#")
2212 IFFX<4THEN2220:ELSEPRINT"Do you want the number printed with
commas (Y/N)? ";:
2214 CV$=INKEY$:IFCV$=""THEN2214:ELSEGOSUB60:PRINTCV$:IFINSTR("YN"
,CV$)<1THEN2212
2216 IFCV$="Y"THENCF$(I)=CF$(I)+","
2220 PRINT"How many digits will be printed to the RIGHT of the dec
imal";:INPUTFX:IFFX<0ORFX>16THENPRINT"Illegal!":GOTO2220
2230 IFFXTHENCF$(I)=CF$(I)+"."+STRING$(FX,"#")
2235 IFCF$(I)=""THENPRINT"ILLEGAL FORMAT!":GOTO2200
2300 IFINSTR(CF$(I),"#")THENMID$(M$,I,1)="M":ELSEMID$(M$,I,1)="S"
2301 NEXT
2305 RETURN
2310 FORI=1TOB:PRINT#1,40000+I;"CF$(";MID$(STR$(I),2);")=";QS;CF$(
I);QS:NEXT
2320 PRINT#1,"56 M$=";QS;M$;QS;":'FIELD TYPES - S = STRING, M =MUL
TIPLE PRECISION"
2325 CLS:PRINT"How many columns wide is the paper or screen the re
port will":PRINT"be printed on";:INPUTFX:IFFX<1ORFX>255THENPRINT"I
llegal!":GOTO2325
2330 LN=LN+10:PRINT#1,LN;"IFL=0ANDPD$<>";QS;"S";QS;"THENLPRINTCHR$
(12);TAB(";(FX-LEN(T$))/2;");TI$";:IFLP<1THENPRINT#1," ":ELSEPRINT
#1,";TAB(";FX-15;");";QS;"PAGE NUMBER";QS;";PG"
2335 LN=LN+10:PRINT#1,LN;"IFL=0ANDPD$<>";QS;"P";QS;"THENPRINTCHR$(
12);TAB(";(FX-LEN(T$))/2;");TI$";:IFLP<1THENPRINT#1," ":ELSEPRINT#
1,";TAB(";FX-15;");";QS;"PAGE NUMBER";QS;";PG"
2340 LN=LN+10:PRINT#1,LN;"IFPD$<>";QS;"P";QS;"ANDL=0THENPRINT:GOSU
B31000"
2350 LN=LN+10:PRINT#1,LN;"IFL=0THENGOSUB31000";"ANDL=0THENLPRINT:GOS
UB31000"
2360 PRINT#1,"31006 FORJ=1TO";B:LN=LN+10:PRINT#1,LN;"FORJ=1TO";B
2370 LN=LN+10:PRINT#1,LN;"READ T:IFPD$=";QS;"P";QS;"THEN";LN+30.
2380 LN=LN+10:PRINT#1,LN;"IFMID$(M$,J,1)=";QS;"S";QS;"THENPRINTTAB
(T);:PRINTUSINGCF$(J);C$(J);"
2390 LN=LN+10:PRINT#1,LN;"IFMID$(M$,J,1)=";QS;"M";QS;"THENPRINTTAB
(T);:PRINTUSINGCF$(J);C#(J);"
2400 LN=LN+10:PRINT#1,LN;"IFPD$=";QS;"S";QS;"THEN";LN+30
2402 LN=LN+10:PRINT#1,LN;"IFMID$(M$,J,1)=";QS;"S";QS;"THENLPRINTTA
B(T);:LPRINTUSINGCF$(J);C$(J);"
2404 LN=LN+10:PRINT#1,LN;"IFMID$(M$,J,1)=";QS;"M";QS;"THENLPRINTTA
B(T);:LPRINTUSINGCF$(J);C#(J);"
2406 LN=LN+10:PRINT#1,LN;"NEXT:RESTORE:IFPD$<>";QS;"S";QS;"THENLPR
INT:ELSEPRINT"
2408 LN=LN+10:PRINT#1,LN;"L=L+1";:IFLPTHENPRINT#1,":IFL>";LP;"THEN
PG=PG+1:L=0":ELSEPRINT#1,":'LINE COUNTER"
2410 LN=19000:PRINT#1,LN;"NEXT"
2412 IFRL<256THENPRINT#1,KZ;"GET1,I:IFZU$=STRING$(";RL;",255)ORZU$
=STRING$(";RL;",250)THEN";LN:ELSEPRINT#1,KZ;"GET1,I:IFZU$=STRING$(
255,250)ORZU$=STRING$(255,255)THEN";LN
2414 IFRL<256THENPRINT#1,"30801 FIELD#1,";RL;"AS ZU$":ELSEPRINT#1,
"30801 FIELD#1,255 AS ZU$"
2416 CLS:PRINT"Have you saved any totals you want printed at the e
nd of your":PRINT"report (Y/N)? ";
2418 CV$=INKEY$:IFCV$=""THEN2418:ELSEGOSUB60:PRINTCV$:IFINSTR("YN"
,CV$)<1THENPRINT"ILLEGAL!":GOTO2418
2420 IFCV$="N"THENPRINT#1,"20000 CLOSE:END":GOTO2630
2430 CLS:PRINT"Then you must tell me which variables to print in w
hich columns":PRINT"of your report. Please be careful here: you CA
NNOT print":PRINT"strings in numeric columns, or vice-versa!"
2440 PRINT"You may use the BASIC operations +, *, /, and - (and ot
```

Listing 2 continued

✓ See List of Advertisers on Page 211

Listing 2 continued

```
hers)":PRINT"if you wish. As a default, just hit the enter key: I
will":PRINT"print the grand total for a column, using the same pri
nt"
2450 PRINT"format as the previous data in that column.":LN=20000
2451 PRINT#1,LN;"IFPD$<>";Q$;"S";Q$;"THENLPRINT":LN=LN+10:PRINT#1,
LN;"IFPD$<>";Q$;"P";Q$;"THENPRINT"
2454 PRINT"If you want to use the totals in your computations, the
 totals":PRINT"are stored in the variable array T#(X), where X is
the column":PRINT"number. Feel free to use these totals in computa
tions."
2455 FORI=1TOB
2460 PRINT"COLUMN NAME: ";C$(I);": DEFAULT IS ";
2470 IF MID$(M$,I,1)="S"THENPRINT"BLANK":ELSEPRINT"T#(";I;")":PRIN
T"COLUMN PRINT FORMAT IS:";CF$(I)
2480 PRINT"What should I print in this column: ";:LINEINPUTCV$
2485 IFCV$<=STRING$(LEN(CV$),32)THENCV$=""
2490 IFCV$=""THENIFMID$(M$,I,1)="S"THENCV$=Q$+Q$
2500 IFCV$=""THENIFMID$(M$,I,1)="N"THENCV$="T#("+MID$(STR$(I),2)+"
)"
2510 LN=LN+10:PRINT#1,LN;"READ T:IFPD$<>";Q$;"S";Q$;"THENLPRINTTAB
(T);:LPRINTUSINGCF$(";MID$(STR$(I),2);");";CV$;";"
2520 LN=LN+10:PRINT#1,LN;"IFPD$<>";Q$;"P";Q$;"THENPRINTTAB(T);:PRI
NTUSINGCF$(";MID$(STR$(I),2);");";CV$;";"
2530 NEXT:LN=LN+10:PRINT#1,LN;"IFPD$<>";Q$;"S";Q$;"THENLPRINTCHR$(
12)":LN=LN+10:PRINT#1,LN;"IFPD$<>";Q$;"P";Q$;"THENPRINT:PRINT";Q$;
"END OF REPORT";Q$
2540 LN=LN+10:PRINT#1,LN;"CLOSE:END"
2630 OPEN"I",2,"R451/LIB"
2631 ON ERROR GOTO 2635
2632 LINE INPUT #2,A$:PRINT#1,A$:GOTO2631
2635 FORI=1TOB:RL=RL+LEN(CF$(I))+LEN(C$(I))+2*F%(I):NEXT:PRINT#1,"
5 CLEAR";2000+RL
2636 CLS:PRINT"Your report program has been written. You may run i
t any time.":PRINT"If you wish, you may load your program, then re
-save it: this":PRINT"will decrease subsequent program load times.
"
2639 CLOSE:END
2640 IFERR=38ANDERL=420THENPRINT"PROGRAM ALREADY EXISTS. DEPRESS C
 TO WRITE OVER IT. ANY OTHER";PRINT"TO QUIT":ELSE2670
2650 Q$=INKEY$:IFQ$=""THEN2650:ELSEIFQ$="C"THENCLOSE1:RESUME430
2660 IFQ$="c"THENCLOSE1:RESUME430:ELSECLOSE:END
2670 IFERR=104THENPRINT"YOU ENTERED BASIC WITHOUT SPECIFYING ANY F
ILES.";PRINT"YOU NEED AT LEAST TWO. GO BACK TO DOS AND ENTER BASIC
 CORRECTLY.":CLOSE:END
2672 IF ERR=106 AND ERL=420THENRESUME430
2675 PRINT"ERROR NUMBER";ERR;"AT LINE";ERL;":PROGRAM ABORTED.":ONE
RRORGOTO0
2680 CLS:CLOSE:END
4999 'REPLACE A STRING WITH ANOTHER
5000 IF F1%=1THENSX$=F1$+MID$(SX$,LEN(C$(J))+1):RETURN
5010 SX$=LEFT$(SX$,F1%-1)+F1$+MID$(SX$,F1%+LEN(C$(J))):RETURN
```

Program Listing 3. R451/LIB. The Reportor's library file.

```
10 PRINT"Should this report be on the screen (S), printer (P), or
both":PRINT"screen and printer (B)";:INPUTPD$:IFINSTR("SsPpBb",PD$
)<1THEN10
11 PD$=CHR$(ASC(PD$)AND95)
15 PRINT"What drive is the data file on (0-3):";:LINEINPUTDR$:IFVA
L(DR$)<0ORVAL(DR$)>3THEN15:ELSEDR$=MID$(STR$(VAL(DR$)),2)
17 DF$=DF$+":"+DR$
18 PRINT"OK, depress any key when the data file is on that drive:"
;
19 A$=INKEY$:IFA$=""THEN19
20 PRINT:PRINT"Are you going to read an index file for a sorted re
port (Y/N)";:INPUTSR$
21 IFINSTR("YyNn",SR$)<1THEN20:ELSESR$=CHR$(ASC(SR$)AND95)
22 IFSR$="N"THEN29
23 PRINT"Please give me the complete name of your index file. For"
:PRINT"example, MYFILE/INX:1      ";:LINEINPUTSR$
29 '
35 IFPD$="S"ORPD$="B"THENCLS
36 IFSR$<>"N"THENOPEN"I",2,SR$
45 DIMCF$(50),T#(50):GOSUB40000:'DIMENSIONING THE MATRICES WE WILL
 USE
65 'FOLLOWING ARE THE COLUMN HEADINGS
19990 'PRINTING TOTALS SECTION
30000 'FIELDING THE FILE
30800 TT=0
30810 FIELD#1,(TT) AS DY$,(FF(IX)) AS F$(IX)
30820 TT=TT+FF(IX):NEXT:RETURN
30990 'BEGIN PRINTING THE COLUMN HEADINGS
31000 IFPD$<>"S"THENLPRINT
31005 IFPD$<>"P"THENPRINT
31010 READ T:OF%=0:IF MID$(M$,J,1)="N" THEN OF%=LEN(CF$(J))-LEN(H$
(J))
31012 IFPD$<>"P"THENPRINTTAB(T+OF%);H$(J);
31015 IFPD$<>"S"THENLPRINTTAB(T+OF%);H$(J);
31020 NEXT:RESTORE
31030 IFPD$<>"P"THENPRINT:PRINT
31040 IFPD$<>"S"THENLPRINT:LPRINT
31050 RETURN
31060 'THESE ARE THE TAB SETTINGS
40000 'COLUMN PRINT FORMATS
40100 RETURN
```

Program Listing 4. The menu program.

```
10 CLS:CLEAR3000:DEFINTA-Z:PRINT
20 PRINTTAB(18);"CREATOR AND REPORTOR MENU":PRINT:PRINT
30 PRINTTAB(15);"You have the following choices:":PRINT
40 PRINT"       1. Write a program using the CREATOR."
50 PRINT"       2. Write a report program using the reportor."
60 PRINT"       3. Run a program or a report."
70 PRINT"       4. Sort a data file."
80 PRINT"       5. Create or re-create a key file."
90 PRINT"       6. Transfer data from a file created using the orig
inal"
100 PRINTTAB(9);"CREATOR to a file useable by the new version."
110 PRINT"       7. Quit, and exit to BASIC.":PRINT
120 PRINTTAB(10);"Please depress the number of your choice: ";
130 AN$=INKEY$:IFAN$=""THEN130
140 AN=VAL(AN$):IFAN<1ORAN>7THENRUN
150 IFAN=1THENCLS:PRINT"LOADING THE CREATOR.":RUN"CREATOR/BAS"
160 IFAN=2THENCLS:PRINT"LOADING THE REPORTOR.":RUN"REPORTOR/BAS"
170 IFAN=4THENCLS:PRINT"LOADING CHEAPSORT.":RUN"CSORT/BAS"
180 IFAN=5THENCLS:PRINT"LOADING REKEY.":RUN"REKEY/BAS"
190 IFAN=6THENCLS:PRINT"LOADING CXFER.":RUN"CXFER/BAS"
200 IFAN=7THENNEW
210 PRINT@960,STRING$(63,32):PRINT@960,"What's the name of the pro
gram to run: ";
220 LINEINPUTF$:CLS:PRINT"LOADING ";F$:RUNF$
```

Program Listing 5. C451/LIB. The Creator's library file.

```
21 DEFFNPH$(X%)=CHR$(X%+128):DEFFNUH(X$)=ASC(X$)-128
100 PRINT
110 PRINT"Enter data.....................................................De
press E"
120 PRINT"Look up a record...............................................De
press L"
130 PRINT"Scan all records...............................................De
press S"
140 PRINT"Update a record................................................De
press U"
150 PRINT"Delete a record................................................De
press D"
160 PRINT"Exit the program...............................................De
press X"
170 PRINT"Initialize the file............................................De
press I"
175 ONERRORGOTO60000
180 GET2,1:PRINT"   CURRENT NUMBER OF RECORDS IN FILE=";CVI(KP$);:
IFCVI(KP$)<1THENPRINT"PLEASE INITIALIZE!":ELSEPRINT
185 PRINT"Please depress the key corresponding to your choice."
186 ONERRORGOTO25000
1000 CLS:REM*BEGIN ENTRY
1005 FC=FC+1
10000 REM*BEGIN FILE LOOK UP
10010 GOSUB27000:'TRY TO FIND THE RECORD
10200 GOSUB28000:'UNPACK THE FIELDS
10800 GOSUB29000:'DISPLAY RECORD IF KEY FIELD MATCHES
11000 REM*BEGIN FILE LOOK UP
11010 GOSUB27000:'TRY TO FIND THE RECORD
11200 GOSUB28000:'UNPACK THE FIELDS
11800 GOSUB29000:'DISPLAY RECORD IF KEY FIELD MATCHES
11810 UF$="":PRINT@960,STRING$(63,32);:PRINT@960,"What field numbe
r do you want to update? ";
11820 UX$=INKEY$:IFUX$>="0"ANDUX$<="9"THENPRINTUX$;:UF$=UF$+UX$:GO
TO11820ELSEIFUX$<>CHR$(13)THEN11820:ELSEUF=VAL(UF$)
11900 REM*BEGIN OUTPUT*
11998 REM*INSERT CHANGED FIELDS AND SEND TO DISK*
12000 REM*BEGIN RECORD DELETE*
12010 GOSUB27000:'TRY TO FIND THE RECORD
12200 GOSUB28000:'UNPACK THE FIELDS
12800 GOSUB29000:'DISPLAY RECORD IF KEY FIELD MATCHES
12900 REM*DELETE CODE WRITTEN INTO ALL FIELDS*
22000 REM*FIELD TITLES FOR DISPLAY*
25000 IF ERL<36000 AND ERL>35000 THEN RESUME 35000:ELSE IF ERR=20
THEN RESUME NEXT
25001 IF ERR=38THENPRINT"YOU DIDN'T INITIALIZE YOUR DATA FILE, DO
SO!":FORI=1TO1000:NEXT:RUN
25010 PRINT"ERROR ENCOUNTERED IN LINE";ERL
25015 IF ERR=10 THEN PRINT"You have input a number too large for t
he field type.":RESUME NEXT
25020 IF ERL<10000 AND ERL>1000 THEN PRINT"You have probably made
an error in edit specifications."
25030 PRINT"ERROR NUMBER IS";ERR:CLOSE:END
25999 REM*HASHING ALGORITHM
26000 FOR ZZ=1TOLEN(ZZ$)
26010 SP=ASC(MID$(ZZ$,ZZ,1)):X#=X#+ZZ*(SP+1/SP)
26020 NEXT
26030 IF X#<1E+17THENX#=X#*X#:GOTO26030
26035 SP=ASC(ZZ$)+ASC(RIGHT$(ZZ$,1)):SP=SP-10*(INT(SP/10)):SP=SP+4
:XS=STR$(X#):RP=VAL(MID$(XS,SP.4)):X#=0
27000 REM*LOOK FOR THE RECORD*
27020 'NOW WE HAVE INPUT THE KEY FIELD
27030 ZZ$=KF$:GOSUB26000:KP=RP:'GO TO HASHING ROUTINE AND GET POSI
TION
27999 REM*UNPACK FIELDS IN RECORD*
28998 RETURN
28999 REM*DISPLAY FOUND RECORD*
29000 CLS:CL=1
29020 READ RS:CX=LEN(G$(I))
29021 IFCX>10THENIFMID$(G$(I),CX-9,10)="          "THENCX=CX-10:GO
TO29021
29022 IFCX<LEN(G$(I))THENG$(I)=LEFT$(G$(I),CX)
29024 LC=POS(0):IFLC<5THENPRINT"#";I;CHR$(133);R$;CHR$(133);G$(I);
:GOTO 29030
29025 IFCL>14THENGOSUB41010:GOTO29024
29026 IF(LC<32 AND LEN(RS)+LEN(G$(I))+37<64)THEN PRINT TAB(32);"#"
;I;CHR$(133);R$;CHR$(133);G$(I);:ELSEPRINT:CL=CL+1:GOTO29024
29030 NEXT:RESTORE:PRINT
29930 IF AN$="S"THENRETURN
29940 PRINT@960,"IS THIS IT? (DEPRESS Y IF SO, ANY OTHER IF NOT)";
29950 AN$=INKEY$:IFAN$=""THEN29950:ELSEPRINT@960,STRING$(60,32);:P
RINT@960,"";:IFAN$<>"Y"THENGOSUB27040:GOSUB28000:GOTO29000
29960 RETURN
29999 REM*NUMERIC FIELD EDIT CHECK SUBROUTINE*
30000 CD=INSTR(CD$,CHR$(32)):IFCD>1THENCD$=LEFT$(CD$,CD-1)+MID$(CD
$,CD+1):GOTO30000:ELSEIFCD=1THENCD$=MID$(CD$,2):GOTO30000
30005 FORZZ=1TOLEN(CD$)
30010 IFMID$(CD$,ZZ,1)<"0"ORMID$(CD$,ZZ,1)>"9"THENIFMID$(CD$,ZZ,1)
<>"."ANDMID$(CD$,ZZ,1)<>"-"THENE=1
30020 NEXT:IF E THEN RETURN
30030 CD=INSTR(CD$,"-"):IFCD>0ANDINSTR(CD+1,CD$,"-")>0THENE=1:RETU
RN:ELSEIF(CD>0ANDCD<>1)THENE=1:RETURN
30040 CD=INSTR(CD$,"."):IFCD>0ANDINSTR(CD+1,CD$,".")>0THENE=1:RETU
RN
30050 RETURN
30999 REM*ALPHA FIELD EDIT CHECK*
31000 FORZZ=1TOLEN(CD$)
31010 IFMID$(CD$,ZZ,1)<="9"ANDMID$(CD$,ZZ,1)>="0"THENE=1
31020 NEXT
31030 RETURN
32000 REM*INITIALIZE*
32010 PRINT"This will erase all previous data, if any.":PRINT"To c
ontinue initialization, depress the C key.";
32030 AN$=INKEY$:IFAN$=""THEN32030:ELSEIFAN$<>"C"THENRUN
32035 CLS:PRINT"This will take a little while. Please be patient."
32060 PUT2,I
32070 NEXT:LSETKP$=MKI$(1):PUT2,1:RUN
34999 REM*BEGIN RECORD SCAN*
35001 INPUT"Numeric or alphabetic scan (N/A):";NS$:NS%=ASC(NS$):NS
%=NS% AND 95:NS$=CHR$(NS%):IFNS$<>"N"ANDNS$<>"A"THEN35001
35002 INPUT"Smallest (numeric or alpha) to display:";SM$
35003 INPUT"Largest (numeric or alpha) to display:";LA$
35004 INPUT"Should I delay after displaying each record (Y/N)?";DY
```

Listing 5 continued

Tired of swapping Disks from Inventory to Accounts Receivable to Accounts Payable etc.? Now, one system does it all.

Introducing

The M.B.S. Business Management System

At last a completely Integrated, Menu driven System for:

- **INVOICING**
  - Opens Customer Files
  - Opens A/R Accounts
  - Updates Inventory
  - Stores Mail List Files
  - Stores Sales Records
  - Computes Sales Tax
- **CUSTOMER FILES**
  - Maintains Order Status
  - Prints Labels
  - Prints Customer Balances
  - Stores Order Amounts
  - Stores Order Payments
  - Prints Statements
- **MAIL LABELS**
  - Stores by Variable File Names
  - Sorts by Zip Code
  - Sorts by Name
- **INVENTORY**
  - Sets Upper and Lower Limits
  - Generates Purchase Orders
  - Lists Inventory by Vendor
- **ACCOUNTS RECEIVABLE**
  - Open A/R Accounts
  - Generates Monthly Statements
  - Interest and Non-Interest Accounts
  - Listing of Accounts Balances
  - Manually Enter Charges and Payments
- **ACCOUNTS PAYABLE**
  - Enter Charges to Accounts
  - Enter Payments to Accounts
  - List Payable Balances
- **CHECK WRITING**
  - Print or Record Checks
  - Maintains Bank Balance
  - Credit Accounts Payable
  - Stores Expense Totals
- **FINANCIAL STATEMENTS**
  - Prints Sales Reports
  - Prints Operating Statements
  - Prints Receipts Reports
  - Modifies Expense and Sales Totals

Yes, now there is a complete business system for the small business man. With our Business Management System, you can increase sales with our mail label function. Complete your Schedule C in as little as 15 minutes. Know what your business is doing, and maintain other important business functions.

This easy to use system comes complete with instruction manual, and diskette on Dosplus mini TDOS operating system with extended Basic, and one year support. All programs are in Basic, and require two disk drives and 48K RAM. If you have waited for the right business software for your business, your wait is over. Our current users love our system and you will too.

Special Introductory Price $249.95

GAMEMASTER

Do you get complaints from your spouse, family and friends that you isolate yourself from them while "playing" with your computer? Do you wish there was a game that could get everyone involved? Do you enjoy creative expressions as well as playing games? If so, we have a unique Program/Game wich could be the answer. It's called GAMEMASTER.

This program lets you easily develop and play your own adventure games. It does not create a game which you play against the computer, but creates a game which you the game creator, direct the game play while your computer maintains the game conditions and resolves combat.

GAMEMASTER package includes routines for creating player characters, creatures, items (weapons, armour, implements, or treasures), and environment (time, location etc.). Comes on diskette with sample scenerio. Requires one disk drive and 48k ram.

Price $39.95

Specify Model I, Model III, Model IV, or LNW
Dealer Inquiries Invited

PRINTERS		PERIPHERALS		L.N.W.		SOFTWARE	
Prowriter	$ 379	Hayes Smartmodem	$ 219	LNW80II	$1495	DOSPLUS 3.5	$ 119
Prowriter	$ 669	Hayes 300/1200	$ 539	LNDoubler W/DOS	$ 199	DOSPLUS IV	$ 119
Starwriter F10	$1229	Holmes VID 80	$ 259	System Expansion	s 339	MTERM	$ 69
Okidata 82A	Call	Holmes CPM 2.2	$ 109	**MONITORS**		TRSDOS 6.0 Plus Enhance	$ 45
Okidata 83A	Call	64 K Ram	$ 114	Amdek 300G	$ 149	Super Utility Plus	$ 69
Okidata 92	Call	Holmes DX3DC	$ 157	Amdek 300A	$ 159	Newscript 7.1	$ 99
Okidata 93	Call	Holmes DX4DC	$ 157	Amdek Color I	$ 339	Newscript w/labels	$ 109
Silver Reed	$499	Holmes Sprinter I	$ 89	Amdek Color II	$ 469	File Converter	$ 20
Mannesman Tally	Call	Holmes Sprinter III	$ 89	Taxan Amber	$ 149	Trashman	$ 35
NEC	Call	Holmes Sprinter KX	$ 129	Taxan Green	$ 139	Faster	$ 27
Transtar	Call	Disk Drives	Call	Taxan RGB I	$ 319	Tallymaster	$ 69
		Diskettes	Call	Taxan RGB III	$ 599	Electric Webster	$ 129

Microcomputer Business Systems
14030 South Springfield Road
Brandywine, Maryland 20613
1-800-638-1857
in Maryland 1 (301) 372-8555 — Washington, D.C. Local Call ↙47

↙ See List of Advertisers on Page 211

Listing 5 continued

```
$:NS%=ASC(DY$):NS%=NS% AND 95:DY$=CHR$(NS%):IFDY$<>"N"ANDDY$<>"Y"T
HEN35004
35050 GOSUB28000:'UNPACK THE RECORD
35060 IFNS$="A"AND(G$(NS)<SM$ORG$(NS)>LA$)THEN35990
35070 IFNS$="N"AND(VAL(G$(NS))<VAL(SM$)ORVAL(G$(NS))>VAL(LA$))THEN
35990
35960 GOSUB29000:'DISPLAY RECORD
35965 IFDY$="Y"THEN35970ELSE35990
35970 FORJ=1TO2000:NEXT:'WAIT BEFORE DISPLAYING NEXT RECORD
36000 REM*FIELD LENGTHS AND FIELDING FILE*
36820 FIELD#1,CD%AS DD$,F%(ZZ) AS F$(ZZ):CD%=CD%+F%(ZZ):NEXT:RETUR
N
38000 CLOSE:NEW
40000 G1=INSTR(G$(UF),"MORE"):IFG1>0THENG%=CINT(VAL(G$(UF))+VAL(G$
)):G$(UF)=MID$(STR$(G%),FIX(2+SGN(G%)/2)):RETURN
40010 G1=INSTR(G$(UF),"LESS"):IFG1>0THENG!=CSNG(-VAL(G$(UF))+VAL(G
$)):G$(UF)=MID$(STR$(G!),FIX(2+SGN(G!)/2)):RETURN
40020 G1=INSTR(G$(UF),"+"):IFG1>1THENG%=CINT(VAL(G$(UF))+VAL(G$)):
G$(UF)=MID$(STR$(G%),FIX(2+SGN(G%)/2)):RETURN
40030 G1=INSTR(G$(UF),"-"):IFG1>1THENG!=CSNG(-VAL(G$(UF))+VAL(G$))
:G$(UF)=MID$(STR$(G!),FIX(2+SGN(G!)/2)):RETURN
40040 G1=INSTR(G$(UF),"*"):IFG1>0THENG%=CINT(VAL(G$(UF))*VAL(G$)):
G$(UF)=MID$(STR$(G%),FIX(2+SGN(G%)/2)):RETURN
40050 G1=INSTR(G$(UF),"/"):IFG1>0THENG%=CINT(VAL(G$)/VAL(G$(UF))):
G$(UF)=MID$(STR$(G%),FIX(2+SGN(G%)/2)):RETURN
40100 RETURN
40500 G3=0:G2=INSTR(G$(UF),"RO"):IFG2THENG3=VAL(MID$(G$(UF),G2+2))
40510 G1=INSTR(G$(UF),"LESS"):IFG1>0THENG#=CDBL(-VAL(G$(UF))+VAL(G
$)):G$(UF)=MID$(STR$(G#),FIX(2+SGN(G#)/2)):GOTO40570
40520 G1=INSTR(G$(UF),"+"):IFG1>1THENG#=CDBL(VAL(G$(UF))+VAL(G$)):
G$(UF)=MID$(STR$(G#),FIX(2+SGN(G#)/2)):GOTO40570
40530 G1=INSTR(G$(UF),"-"):IFG1>1THENG#=CDBL(-VAL(G$(UF))+VAL(G$))
:G$(UF)=MID$(STR$(G#),FIX(2+SGN(G#)/2)):GOTO40570
40540 G1=INSTR(G$(UF),"*"):IFG1>0THENG#=CDBL(VAL(G$(UF))*VAL(G$)):
G$(UF)=MID$(STR$(G#),FIX(2+SGN(G#)/2)):GOTO40570
40550 G1=INSTR(G$(UF),"/"):IFG1>0THENG#=CDBL(VAL(G$)/VAL(G$(UF))):
G$(UF)=MID$(STR$(G#),FIX(2+SGN(G#)/2)):GOTO40570
40560 G1=INSTR(G$(UF),"MORE"):IFG1>0THENG#=CDBL(VAL(G$(UF))+VAL(G$
)):G$(UF)=MID$(STR$(G#),FIX(2+SGN(G#)/2))
40570 IFG2<1THENRETURN
40580 G#=CDBL(VAL(G$(UF)))
40590 IFG3THENFORG2=1TOG3:G#=G#*10:NEXT:G#=FIX(G#+.500001#*SGN(G#)
)
40600 IFG3THENFORG2=1TOG3:G#=G#/10:NEXT:ELSEG#=FIX(G#+.500001#*SGN
(G#))
40610 G$(UF)=MID$(STR$(G#),FIX(2+SGN(G#)/2)):RETURN
41000 IFCL<15THENRETURN
41010 PRINT@960,"THERE ARE MORE FIELDS! HIT ANY KEY TO SEE THE RES
T:";:PRINT@960,STRING$(60,32);:Y$=INKEY$:IFY$=""THEN41010:ELSECLS:
CL=0:RETURN
60000 LSETKP$=MKI$(0):RESUMENEXT
```

Program Listing 6. C451MIN/LIB. Creator inventory/update program.

```
21 DEFFNPH$(X%)=CHR$(X%+128):DEFFNUH(X$)=ASC(X$)-128
100 PRINT
110 PRINT"Enter data.....................................De
press E"
120 PRINT"Look up a record...............................De
press L"
130 PRINT"Scan all records...............................De
press S"
140 PRINT"Update a record................................De
press U"
150 PRINT"Delete a record................................De
press D"
160 PRINT"Exit the program...............................De
press X"
170 PRINT"Initialize the file............................De
press I"
175 ONERRORGOTO60000
180 GET2,1:PRINT"   CURRENT NUMBER OF RECORDS IN FILE=";CVI(KP$);:
IFCVI(KP$)<1THENPRINT"PLEASE INITIALIZE!":ELSEPRINT
185 PRINT"Please depress the key corresponding to your choice."
186 ONERRORGOTO25000
1000 CLS
1005 FC=FC+1
10000 '
10010 GOSUB27000
10200 GOSUB28000
10800 GOSUB29000
11000 '
11010 GOSUB27000
11200 GOSUB28000
11800 GOSUB29000
11810 UF$="":PRINT@960,STRING$(63,32);:PRINT@960,"What field numbe
r do you want to update? ";
11820 UX$=INKEY$:IFUX$>="0"ANDUX$<="9"THENPRINTUX$;:UF$=UF$+UX$:GO
TO11820ELSEIFUX$<>CHR$(13)THEN11820:ELSEUF=VAL(UF$)
11900 '
12000 '
12010 GOSUB27000
12200 GOSUB28000
12800 GOSUB29000
12900 '
22000 '
25000 IF ERL<36000 AND ERL>35000THENRESUME35000:ELSE IF ERR=20 THE
N RESUME NEXT
25001 IF ERR=38THENPRINT"YOU DIDN'T INITIALIZE YOUR DATA FILE. DO
SO!":FORI=1TO1000:NEXT:RUN
25010 PRINT"ERROR ENCOUNTERED IN LINE";ERL
25015 IF ERR=10 THEN PRINT"You have input a number too large for t
he field type.":RESUME NEXT
25020 IF ERL<10000 AND ERL>1000 THEN PRINT"You have probably made
an error in edit specifications."
25030 PRINT"ERROR NUMBER IS";ERR:CLOSE:END
26000 FOR ZZ=1TOLEN(ZZ$)
26010 SP=ASC(MID$(ZZ$,ZZ,1)):X#=X#+ZZ*(SP+1/SP)
26020 NEXT
26030 IF X#<1E+17THENX#=X#*X#:GOTO26030
26035 SP=ASC(ZZ$)+ASC(RIGHT$(ZZ$,1)):SP=SP-10*(INT(SP/10)):SP=SP+4
:X$=STR$(X#):RP=VAL(MID$(X$,SP,4)):X#=0
27000 '
27020 '
27030 ZZ$=KF$:GOSUB26000:KP=RP
28998 RETURN
29000 CLS:CL=1
29020 READ R$:CX=LEN(G$(I))
29021 IFCX>10THENIFMID$(G$(I),CX-9,10)="          "THENCX=CX-10:GO
TO29021
29022 IFCX<LEN(G$(I))THENG$(I)=LEFT$(G$(I),CX)
29024 LC=POS(0):IFLC<5THENPRINT"#";I;CHR$(133);R$;CHR$(133);G$(I);
:GOTO 29030
29025 IFCL>14THENGOSUB41010:GOTO29024
29026 IF(LC<32 AND LEN(R$)+LEN(G$(I))+37<64)THEN PRINT TAB(32);"#"
;I;CHR$(133);R$;CHR$(133);G$(I);:ELSEPRINT:CL=CL+1:GOTO29024
29030 NEXT:RESTORE:PRINT
29930 IF AN$="S"THENRETURN
29940 PRINT@960,"IS THIS IT? (DEPRESS Y IF SO, ANY OTHER IF NOT)";
29950 AN$=INKEY$:IFAN$=""THEN29950:ELSEPRINT@960,STRING$(60,32);:P
RINT@960,"";:IFAN$<>"Y"THENGOSUB27040:GOSUB28000:GOTO29000
29960 RETURN
30000 CD=INSTR(CD$,CHR$(32)):IFCD>1THENCD$=LEFT$(CD$,CD-1)+MID$(CD
$,CD+1):GOTO30000:ELSEIFCD=1THENCD$=MID$(CD$,2):GOTO30000
30005 FORZZ=1TOLEN(CD$)
30010 IFMID$(CD$,ZZ,1)<"0"ORMID$(CD$,ZZ,1)>"9"THENIFMID$(CD$,ZZ,1)
<>"."ANDMID$(CD$,ZZ,1)<>"-"THENE=1
30020 NEXT:IF E THEN RETURN
30030 CD=INSTR(CD$,"-"):IFCD>0ANDINSTR(CD+1,CD$,"-")>0THENE=1:RETU
RN:ELSEIF(CD>0ANDCD<>1)THENE=1:RETURN
30040 CD=INSTR(CD$,"."):IFCD>0ANDINSTR(CD+1,CD$,".")>0THENE=1:RETU
RN
30050 RETURN
31000 FORZZ=1TOLEN(CD$)
31010 IFMID$(CD$,ZZ,1)<="9"ANDMID$(CD$,ZZ,1)>="0"THENE=1
31020 NEXT
31030 RETURN
32000 PRINT"This will erase all previous data, if any.":PRINT"To c
ontinue initialization, depress the C key.";
32030 AN$=INKEY$:IFAN$=""THEN32030:ELSEIFAN$<>"C"THENRUN
32035 CLS:PRINT"This will take a little while. Please be patient."
32060 PUT2,I
32070 NEXT:LSETKP$=MKI$(1):PUT2,1:RUN
35001 INPUT"Numeric or alphabetic scan (N/A):";NS$:NS%=ASC(NS$):NS
%=NS% AND 95:NS$=CHR$(NS%):IFNS$<>"N"ANDNS$<>"A"THEN35001
35002 INPUT"Smallest (numeric or alpha) to display:";SM$
35003 INPUT"Largest (numeric or alpha) to display:";LA$
35004 INPUT"Should I delay after displaying each record (Y/N)?";DY
$:NS%=ASC(DY$):NS%=NS% AND 95:DY$=CHR$(NS%):IFDY$<>"N"ANDDY$<>"Y"T
HEN35004
35050 GOSUB28000
35060 IFNS$="A"AND(G$(NS)<SM$ORG$(NS)>LA$)THEN35990
35070 IFNS$="N"AND(VAL(G$(NS))<VAL(SM$)ORVAL(G$(NS))>VAL(LA$))THEN
35990
35960 GOSUB29000
35965 IFDY$="Y"THEN35970ELSE35990
35970 FORJ=1TO2000:NEXT
36000 '
36820 FIELD#1,CD%AS DD$,F%(ZZ) AS F$(ZZ):CD%=CD%+F%(ZZ):NEXT:RETUR
N
38000 CLOSE:NEW
40000 G1=INSTR(G$(UF),"MORE"):IFG1>0THENG%=CINT(VAL(G$(UF))+VAL(G$
)):G$(UF)=MID$(STR$(G%),FIX(2+SGN(G%)/2)):RETURN
40010 G1=INSTR(G$(UF),"LESS"):IFG1>0THENG!=CSNG(-VAL(G$(UF))+VAL(G
$)):G$(UF)=MID$(STR$(G!),FIX(2+SGN(G!)/2)):RETURN
40020 G1=INSTR(G$(UF),"+"):IFG1>1THENG%=CINT(VAL(G$(UF))+VAL(G$)):
G$(UF)=MID$(STR$(G%),FIX(2+SGN(G%)/2)):RETURN
40030 G1=INSTR(G$(UF),"-"):IFG1>1THENG!=CSNG(-VAL(G$(UF))+VAL(G$))
:G$(UF)=MID$(STR$(G!),FIX(2+SGN(G!)/2)):RETURN
40040 G1=INSTR(G$(UF),"*"):IFG1>0THENG%=CINT(VAL(G$(UF))*VAL(G$)):
G$(UF)=MID$(STR$(G%),FIX(2+SGN(G%)/2)):RETURN
40050 G1=INSTR(G$(UF),"/"):IFG1>0THENG%=CINT(VAL(G$)/VAL(G$(UF))):
G$(UF)=MID$(STR$(G%),FIX(2+SGN(G%)/2)):RETURN
40100 RETURN
40500 G3=0:G2=INSTR(G$(UF),"RO"):IFG2THENG3=VAL(MID$(G$(UF),G2+2))
40510 G1=INSTR(G$(UF),"LESS"):IFG1>0THENG#=CDBL(-VAL(G$(UF))+VAL(G
$)):G$(UF)=MID$(STR$(G#),FIX(2+SGN(G#)/2)):GOTO40570
40520 G1=INSTR(G$(UF),"+"):IFG1>1THENG#=CDBL(VAL(G$(UF))+VAL(G$)):
G$(UF)=MID$(STR$(G#),FIX(2+SGN(G#)/2)):GOTO40570
40530 G1=INSTR(G$(UF),"-"):IFG1>1THENG#=CDBL(-VAL(G$(UF))+VAL(G$))
:G$(UF)=MID$(STR$(G#),FIX(2+SGN(G#)/2)):GOTO40570
40540 G1=INSTR(G$(UF),"*"):IFG1>0THENG#=CDBL(VAL(G$(UF))*VAL(G$)):
G$(UF)=MID$(STR$(G#),FIX(2+SGN(G#)/2)):GOTO40570
40550 G1=INSTR(G$(UF),"/"):IFG1>0THENG#=CDBL(VAL(G$)/VAL(G$(UF))):
G$(UF)=MID$(STR$(G#),FIX(2+SGN(G#)/2)):GOTO40570
40560 G1=INSTR(G$(UF),"MORE"):IFG1>0THENG#=CDBL(VAL(G$(UF))+VAL(G$
)):G$(UF)=MID$(STR$(G#),FIX(2+SGN(G#)/2))
40570 IFG2<1THENRETURN
40580 G#=CDBL(VAL(G$(UF)))
40590 IFG3THENFORG2=1TOG3:G#=G#*10:NEXT:G#=FIX(G#+.500001#*SGN(G#)
)
40600 IFG3THENFORG2=1TOG3:G#=G#/10:NEXT:ELSEG#=FIX(G#+.500001#*SGN
(G#))
40610 G$(UF)=MID$(STR$(G#),FIX(2+SGN(G#)/2)):RETURN
41000 IFCL<15THENRETURN
41010 PRINT@960,"THERE ARE MORE FIELDS! HIT ANY KEY TO SEE THE RES
T:";:PRINT@960,STRING$(60,32);:Y$=INKEY$:IFY$=""THEN41010:ELSECLS:
CL=0:RETURN
60000 LSETKP$=MKI$(0):RESUMENEXT
```

Program Listing 7. Sort/merge utility.

```
10 CLEAR 0:FRI=FRE(0):CLEAR FRI-8000:DIMX%(8)
20 GOSUB30:GOTO50
30 DEFFNCI$(XS)=RIGHT$(MKI$(-CVI(X$)),1)+LEFT$(X$,1)
40 RETURN
50 GOTO140
60 FORX1=1TOLEN(X$):X2=ASC(MID$(X$,X1,1)):IFX2>95THENX2=X2AND95:MI
D$(X$,X1,1)=CHR$(X2)
70 NEXT:RETURN
80 XX%=LEN(X$)/2:Z%=LEN(X$)+1:IFASC(MID$(X$,Z%-1,1))=0THENX$=CHR$(
129)+STRING$(Z%-2,0):RETURN
90 FORX%=1TOXX%:XX$=MID$(X$,X%,1):MID$(X$,X%,1)=MID$(X$,Z%-X%,1):M
ID$(X$,Z%-X%,1)=XX$:NEXT:FORX%=0TO1:X%(X%)=ASC(MID$(X$,X%+1,1)):NE
XT:IFX%(1)>127THENXY%=1:ELSEXY%=0
100 XZ%=X%(0)AND1:X%(1)=(X%(1)AND127)+128*XZ%:X%(0)=(X%(0)/2)OR128
110 IFXY%THENFORX%=2TOZ%-2:X%(X%)=ASC(MID$(X$,X%+1,1)):NEXT:FORX%=
0TOZ%-2:X%(X%)=255ANDNOTX%(X%)AND255:NEXT:FORX%=2TOZ%-2:MID$(X$,X%
+1,1)=CHR$(X%(X%)):NEXT
120 FORX%=0TO1:MID$(X$,X%+1,1)=CHR$(X%(X%)):NEXT
130 RETURN
140 DIMKS%(50),KE%(52),TY%(50),PT%(200,1),RS(50)
150 AD$=STRING$(50,0)
160 PRINT"CSORT, A RANDOM FILE SORT/MERGE."
170 PRINT"COPYRIGHT (C). 1983, BY TNT SOFTWARE, INC."
180 PRINT
190 PRINT"NAME OF FILE TO BE SORTED: ";:LINE INPUTSOS:IFSOS<>""THE
N190
200 PRINT"SORT FILE RECORD LENGTH=";:INPUTRL%:IFRL%<10ORRL%>256THEN
200
```

Listing 7 continued

PRICES AND SERVICE TOO GOOD TO PASS UP!

PRODUCTS FOR THE TRS-80®

SOFTWARE

NEWDOS80/Vers. 2	$124.95
COMREF for Newdos80	13.95
LDOS 5.1	109.95
DOSPLUS 3.5	119.95
DOSPLUS IV	119.95
6.0 Plus	39.50
DOSPLUS II	195.00
Micro Clinic - Mod 1	24.50
Micro Clinic - Mod 3	28.50
Super Utility + 3	59.95
Trashman	34.50
Faster	24.95
R.P.M. - Disk Timer	22.50
Basic Editor by C.A.U.	24.75
M-ZAL Release Two by C.A.U.	75.00
LDOS Utility Disk #1	42.50
The Toolbox for LDOS	59.95
EDAS IV	84.95
The BASIC Answer	59.95
Dot Writer 3.0	66.50
Dot Writer 3.0 w/letterset	84.50
Dot Writer Font Disks	23.50
GEAP	38.50
Maxi Manager	119.50
Maxi Mail (Mod 3 Only)	64.95
Maxi Stat	164.95
Maxi CRAS	79.95
Powermail Plus	129.00
Tallymaster	62.95
Microterm Mod 1, 3, or 4	64.95
Omniterm Mod 1/3	79.95
Omniterm Mod II	139.95
LNW Doubler 5/8 w Dos +	194.95
LNW Doubler 5/8 w/o Dos	159.95
LNW System Expansion II	329.95
SBSG Accounting Modules	
Model 1 or 3	175.00
Model 2	250.00

SPECIAL — Verbatim disks

5" SS/DD	**$23.75**
5" DS/DD	**$35.75**

THE HOME ACCOUNTANT

The #1 best selling program for home and small business accounting is now available for TRS-80 Model 3. Handles up to 99 accounts, five checkbooks, multiple income accounts, and can split transactions to any number of accounts. Prints net worth statements, income statements, as well as custom designed reports. Displays a bar chart and trend for any selected category. It is easy to use and yet provides all the power you'll need. (By Continental Software, requires Model 3, 2 drives, 48K.)

SPECIAL — The Home Accountant has been selected for the Tandy line, so we are closing out our stock. While our supply lasts you can save with our super low close-our price — **$59.95**

WORD PROCESSING

Lazy Writer 3.4	139.95
LeScript	109.95
Zorloff II	58.95

NewScript™

Newscript 7.1	99.95
Newscript 7.1 plus Labels	114.95
Newscript DW Proportional	39.95
Newscript File Converter	19.95

Electric Webster + Corrections	119.50
Elec. Webster Grammar	39.95
Elec. Webster Hyphenation	39.95

WRITE FOR OUR FREE CATALOG

MODEMS

U.S. ROBOTICS INC.™

PASSWORD

Password puts your TRS-80 into telephone communications with 300 or 1200 baud speed, programmable auto dial and auto answer capability, auto mode and auto speed select, audio phone line monitor, and other advanced features. Simple one button operation, compact, and dependable. **$349.95**

PHONE LINK	**$165.95**
MICRO LINK 300	**$199.95**
AUTO LINK 300	**$229.95**
MICRO LINK	**$349.95**
AUTO LINK 1200	**$385.95**
AUTO LINK 212A	**$419.95**
AUTO DIAL 212A	**$449.95**
PASSWORD	**$349.95**
COURIER	**$389.95**

—SAVE AN ADDITIONAL $35.00—
When you purchase your US Robotics modem from AMI we will give you an Omniterm Mod 1/3 at $35 off the retail price— **$65.00**

OMNITERM

The best "smart" terminal package available for the Model I/III. It's menu driven and includes a text editor, four conversion utilities, and setting files to access all of the popular data bases such as CompuServ, the Source, Dow Jones, etc. User defined keys and automatic reformatting of data to the size of your screen are only two of many extra features.

$79.95

LNW80 II COMPUTER

System complete with monitor and 2 disk drives. Includes DOSPLUS 3.5 and CP/M 2.2. It's ready to run!

$1,995.00

- Free Shipping on Orders over $100
- Friendly, Honest, Reliable Service
- Toll Free Order Line
- Free Use of Credit Cards
- 24-Hour Shipping for Items in Stock
- Large Selection-Call for Items Not Listed

We accept Visa, Master Card, check, cash, money orders, and COD. In the 48 continental States add $2.50 for UPS standard shipping, we'll pay shipping if your order is over $100. Alaska and Hawaii orders are charged actual shipping charges. COD orders are charged an extra $2.00 and require cash or certified check on delivery. POs accepted upon approval.

When ordering by mail, include your telephone number, credit card information, computer model, memory size, and number of drives. Colorado residents add appropriate sales tax (6½% in Denver).

Prices are subject to change without notice.

IN COLO CALL 861-9250

ORDER NOW TOLL FREE **1-800-468-4474**

AMI applied microsystems, inc.

612 Washington, Denver, CO 80203 ⌐390

LDOS is a TM of Logical Systems, Inc. **M-ZAL** is a TM of CAU, Inc. **TRS-80** is a TM of Tandy Corp. **NEWSCRIPT** is a TM of Prosoft

Listing 7 continued

```
210 PRINT"OUTPUT INDEX FILE NAME IS: ";:LINE INPUTOF$:IFOF$<>""THE
N210
220 PRINT"NOW I NEED TO KNOW THE FIELD TYPE, AND WHETHER THE SORT
WILL"
230 PRINT"BE ASCENDING OR DESCENDING ORDER, FOR EACH KEY."
240 I%=I%+1
250 PRINT:PRINT"INPUT THE NUMBER 0 TO END, OTHERWISE"
260 PRINT"INPUT THE STARTING POSITION FOR KEY #";I%;":";:INPUT KS%
(I%)
270 IFKS%(I%)=0THEN460
280 PRINT"INPUT THE LENGTH OF THIS KEY:";:INPUTKE%(I%)
290 IFKE%(I%)<=0THENPRINT"INVALID!":GOTO280
300 PRINT"FIELD TYPE:"
310 PRINT"I=PACKED INTEGER,"
320 PRINT"C=PACKED HALF-PRECISION, OR ANY CHARACTER FIELD,"
330 PRINT"N=NUMERIC UNPACKED."
340 PRINT"O=OTHER FIELD TYPES, FLOATING POINT OR PACKED."
350 PRINT:PRINT"FIELD TYPE FOR THIS FIELD IS: ";:LINEINPUTTY$
360 X$=TY$:GOSUB60:TY$=X$
370 TY%=INSTR("IONC",TY$):IFTY%<1THENPRINT"INVALID!":GOTO300
380 TY%(I%)=TY%
390 IFTY%(I%)<>3THENTT%=TT%+KE%(I%):ELSETT%=TT%+8
400 PRINT"ASCENDING OR DESCENDING ORDER ON THIS FIELD (A/D): ";:LI
NEINPUTAO$
410 X$=AO$:GOSUB60:AO$=X$
420 AD%=INSTR("AD",AO$):IFAD%<1THEN400
430 MID$(AD$,I%,1)=AO$:PRINT50-I%;"SORT KEYS, AND"
440 PRINT253-TT%;"BYTES OF SORT KEY SPACE REMAIN."
450 GOTO240
460 PRINT"DRIVE TO USE FOR WORK FILES (0-3): ";:LINE INPUT DR$
470 IFDR$<"0"ORDR$>"3"ORLEN(DR$)<>1THENPRINT"ILLEGAL!":GOTO460
480 DR$=":"+DR$
490 OPEN"R",1,SO$,RL%:J%=0:FS%=RL%AND255:IFRL%>255THENFS%=255
500 FIELD#1,FS% AS R$
510 FORK%=1TOI%-1:FIELD#1,(KS%(K%)-1) AS D$,KE%(K%) AS R$(K%):NEXT
520 OPEN"O",2,OF$
530 OPEN"R",3,"ZZZ/TMP"+DR$,TT%+2
540 FRI=20000/(TT%+10)
550 DIM D$(FRI)
555 D$=STRING$(TT%+2,32)
560 PRINT"PRINT LEVEL TO USE FOR INFORMATIONAL MESSAGES (1-4); LEV
EL 1"
570 PRINT"IS MOST RESTRICTED, LEVEL 4 MOST COMPLETE:";:LINEINPUTLE
V$
590 LEV%=VAL(LEV$):IFLEV%<1ORLEV%>4THENPRINT"ILLEGAL!":GOTO560
595 ONERRORGOTO710
600 FR%=CINT(FRI):IFLEV%=4THENPRINT"READING THROUGH RECORD NUMBER:
"

610 IF LEV%=3THENPRINT"SORTING. . ."
620 J%=J%+1:GET1,J%:IFLE%=4ANDJ%AND1THENPRINTJ%:PRINTCHR$(27);
630 IFR$=STRING$(FS%,0)THEN710:ELSEIFR$=STRING$(FS%,250)ORR$=STRIN
G$(FS%,255)THEN620
650 CO%=CO%+1:VC%=VC%+1:FORK%=1TOI%-1:Y%=VARPTR(D$):POKEY%,KE%(K%)
:LSETD$=R$(K%):CP%=CP%+KE%(K%)
660 IFTY%(K%)=1THENLSETD$=FNCI$(D$):ELSEIFTY%(K%)=2THENX$=D$:GOSUB
80:LSETD$=X$:ELSEIFTY%(K%)=3THENLSETD$=MKD$(VAL(D$)):X$=D$:GOSUB80
:LSETD$=X$
670 IFMID$(AD$,K%,1)="D"THENM%=LEN(D$):FORL%=1TOM%:F%=ASC(MID$(D$,
L%,1)):F%=255ANDNOTF%AND255:MID$(D$,L%,1)=CHR$(F%):NEXT
680 IFLEN(D$(CO%))=TT%+2THENLSETD$(CO%)=LEFT$(D$(CO%),CP%-LEN(D$))
+D$:ELSED$(CO%)=D$(CO%)+D$
685 NEXT:CP%=0:IFLEN(D$(CO%))=TT%+2THENLSETD$(CO%)=LEFT$(D$(CO%),T
T%)+MKI$(J%):ELSED$(CO%)=D$(CO%)+MKI$(J%)
690 IFCO%>=FR%THENGOSUB760
700 GOTO620
710 IFMR%THEN820
720 CMD"O",CO%,D$(1)
730 PRINT:IFLEV%>1THENPRINTCO%;"RECORDS SORTED. NOW WRITING INDEX
FILE."
740 FORK%=1TOCO%:J%=CVI(RIGHT$(D$(K%),2)):PRINT#2,STRING$(6-LEN(ST
R$(J%)),32);J%:NEXT
750 GOTO1140
760 IFLEV%>2THENPRINT"WRITING TO TEMPORARY MERGE FILE."
770 CMD"O",CO%,D$(1)
775 FIELD#3,TT%+2 AS D$
780 FORK%=1TOCO%:LSETD$=D$(K%):OP%=OP%+1:PUT3,OP%:NEXT
790 CO%=0:MR%=1
800 IFLEV%>2THENPRINT"RESUMING SORT."
810 RETURN
820 IFCO%<=1999ANDCO%>0THENGOSUB760
830 FIELD#3,TT%+2 AS D$:LSET D$=STRING$(255,0):PUT3,OP%+1
840 CLOSE3
850 OPEN"R",3,"ZZZ/TMP"+DR$,TT%+2
860 FIELD#3,(TT%+2) AS T$
870 PRINT:CT%=0:IFLEV%>2THENPRINT"MERGING. . ."
880 IFLEV%=4THENPRINT"APPROXIMATE NUMBER MERGED:"
890 BL$=STRING$(TT%+2,0)
900 J%=(VC%/FR%)+1
910 FORK%=1TOJ%:PT%(K%,1)=K%*FR%:NEXT
920 LSETD$(J%+1)=BL$:PT%(J%+1,1)=VC%+1
930 FORK%=1TOJ%:PT%(K%,0)=1+FR%*(K%-1):NEXT:PT%(J%+1,0)=VC%+1
940 FORBM%=1TOJ%+1:KE%(BM%)=0:NEXT
950 K%=FR%/J%:FORM%=1TOJ%:FORI%=1TOK%:GET3,PT%(M%,0):LSETD$((M%-1)
*K%+I%)=T$:PT%(M%,0)=PT%(M%,0)+1:IFT$=BL$THENI%=K%:M%=J%
960 NEXT:NEXT
970 FORI%=1TOJ%:KE%(I%)=K%*(I%-1)+1:NEXT
980 TP%=1:BM%=0
990 IFTP%>J%THEN1020:ELSEIFD$(KE%(TP%))=BL$THENTP%=TP%+1:GOTO990
1000 IFBM%=0THENBM%=TP%:TP%=TP%+1:GOTO990:ELSEIFD$(KE%(TP%))<D$(KE
%(BM%))THENBM%=TP%
1010 TP%=TP%+1:GOTO990
1020 IFBM%=0THEN1140
1030 F%=CVI(RIGHT$(D$(KE%(BM%)),2)):PRINT#2,STRING$(6-LEN(STR$(F%)
),32);F%
1040 KE%(BM%)=KE%(BM%)+1
1050 CT%=CT%+1:IFLEV%=4ANDCT%AND1THENPRINTCT%:PRINTCHR$(27);
1060 IFKE%(BM%)>BM%*K%THENGOSUB1080
1070 GOTO980
1080 F%=(BM%-1)*K%+1:M%=BM%*K%
1090 FORI%=F%TOM%
1100 IFPT%(BM%,0)>PT%(BM%,1)THEND$(I%)=BL$:I%=M%:ELSEGET3,PT%(BM%,
0):LSETD$(I%)=T$:PT%(BM%,0)=PT%(BM%,0)+1:IFBL$=T$THENI%=M%
1110 NEXT
1120 KE%(BM%)=F%
1130 RETURN
1140 CLOSE:PRINT:PRINTVC%;"RECORDS SORTED AND MERGED."
1150 PRINT"SORT COMPLETE.":KILL"ZZZ/TMP"+DR$:CLEAR0:END
```

Program Listing 8. REKEY/BAS. Rekey utility.

```
10 CLEAR 3000
20 CLS:PRINT"This is the RE-KEY utility program for CREATOR data f
iles.":PRINT:PRINT
30 PRINT"This program will allow you to do the following:"
40 PRINT"1. Repair a damaged key file."
50 PRINT"2. Create a new key file for a field not previously a key
."
60 PRINT"3. Create added keys for records you may have appended to
 an";PRINT"    existing data file."
70 PRINT:PRINT"You must be able to tell me a few things about your
 data file."
80 PRINT"For example, the record length, the beginning position an
d";PRINT"length of the key field, and the maximum number of record
s";PRINT"allowed in the file. Of course, you must know the name of
 the"
90 PRINT"data file, as well. DEPRESS ANY KEY TO CONTINUE: ";
100 AN$=INKEY$:IF AN$=""THEN100
110 CLS:PRINT"You may not use a packed field as a key field. Recor
d length":PRINT"can be up to 256, key length up to 255.":PRINT"Whe
n I ask for file names, please give the complete file name,":PRINT
"including the / type and drive!"
120 PRINT:PRINT"Now give me the complete DATA file name: ";
130 LINEINPUTDF$:IF DF$<"A"ORLEN(DF$)>14THENPRINT"Illegal name!":G
OTO120
140 PRINT"Now tell me the maximum number of records allowed in thi
s":PRINT"file, per data disk: ";
150 LINE INPUT MR$:IFVAL(MR$)<1ORVAL(MR$)>32767THENPRINT"Illegal n
umber of records!":GOTO140
160 PRINT"What is the record length of each record in the data fil
e:";:LINE INPUT RL$
170 IFVAL(RL$)>256ORVAL(RL$)<1THENPRINT"Illegal record length!":GO
TO160
180 PRINT"What is the name of the proposed KEY file? ";:LINEINPUTK
F$
190 IFKF$<"A"ORLEN(KF$)>14THENPRINT"Illegal key file name!":GOTO18
0
200 PRINT"What is the beginning position of the key field in the d
ata":PRINT"record? THIS IS NOT USUALLY THE FIELD NUMBER! ";
210 LINEINPUTST$:IFVAL(ST$)<1ORVAL(ST$)>VAL(RL$)THENPRINT"Illegal
 starting position: should be from 1 to";VAL(RL$):GOTO200
220 PRINT"What is the length of the key field: ";:LINE INPUTKL$:IF
 VAL(KL$)<1ORVAL(KL$)>255ORVAL(KL$)>VAL(RL$)THENPRINT"Illegal key
length!":GOTO220
230 IFVAL(RL$)<256THENOPEN"R",1,DF$,VAL(RL$):ELSE OPEN"R",1,DF$
240 OPEN"R",2,KF$,2
250 FIELD#1,VAL(ST$)-1 AS D$,VAL(KL$) AS K$
260 RL%=VAL(RL$):IFRL%>255THENRL%=255
270 FIELD #1,RL% AS B$
280 MR%=VAL(MR$)
290 DIM R%(MR%):FORI%=1TOMR%:R%(I%)=-1:NEXT:I%=1:KL%=VAL(KL$)
300 LSET B$=STRING$(RL%,250):PUT1,1:PRINT"RE-KEYING THROUGH RECORD
 NUMBER ";
305 ONERRORGOTO420
310 I%=I%+1:GET1,I%:F%=KL%:IFB$=STRING$(RL%,0)THEN420
320 PRINT I%;
340 IF K$<CHR$(250)THEN390
350 RP=RND(MR%)
360 RP=RP+1:IFRP>MR%ORRP<2THENRP=2
370 IF R%(RP)<>-1THEN360
380 R%(RP)=-I%:GOTO310
390 ZZ$=LEFT$(K$,F%):GOSUB460
400 RP=RP+1:IFRP>MR%ORRP<2THENRP=2
410 IFR%(RP)=-1THENR%(RP)=I%:GOTO310:ELSE400
420 CLOSE 1:PRINT:PRINT"WRITING KEY POINTERS NOW."
430 FIELD #2,2 AS KP$
440 R%(1)=I%-1:FORI%=1TOMR%:LSETKP$=MKI$(R%(I%)):PUT2,I%:NEXT
450 PRINT:PRINT"RE-KEY OPERATION IS COMPLETE.":END
460 FOR ZZ=1TOLEN(ZZ$)
470 SP=ASC(MID$(ZZ$,ZZ,1)):X#=X#+ZZ*(SP+1/SP)
480 NEXT
490 IF X#<1E+17THENX#=X#*X#:GOTO490
500 SP=ASC(ZZ$)+ASC(RIGHT$(ZZ$,1)):SP=SP-10*(INT(SP/10)):SP=SP+4:X
$=STR$(X#):RP=VAL(MID$(XS,SP,4)):X#=0
510 RP=NR%*RP/9999:RP=FIX(RP):RETURN
```

Program Listing 9. CXFER/BAS. Creator transfer utility.

```
10 CLS:CLEAR 0:CLEAR 3000
20 PRINT"This is CXFER/BAS, a utility program which will read your
 old"
30 PRINT"data files, created under the original CREATOR, and allow
 you"
40 PRINT"to squeeze out all deleted and unallocated records, and t
o"
50 PRINT"reformat the remaining records so they will span sectors.
"
60 PRINT:PRINT"You should NOT use this utility program on data fil
es created":PRINT"with the new CREATOR; IF YOU DO, YOU WILL MANGLE
 YOUR FILES!":PRINT:PRINT
70 LINEINPUT"Please type your old file name: ";F$
80 LINEINPUT"Now tell me your proposed new file name: ";N$
90 LINEINPUT"What was the old record length (1-255): ";OL$
100 IF VAL(OL$)>255 OR VAL(OL$)<1 THEN PRINT"INVALID LENGTH!":GOTO
90
110 LINEINPUT"What is the proposed record length: ";NL$
120 IF VAL(NL$)>256 OR VAL(NL$)<1 THEN PRINT"INVALID LENGTH!":GOTO
110
130 IF VAL(OL$)<>VAL(NL$) THEN PRINT"WARNING! NEW LENGTH DOES NOT
MATCH OLD. YOU MAY HAVE DATA":PRINT"INCOMPATIBILITIES UNLESS YOUR
NEW PROGRAM WAS DESIGNED TO USE":PRINT"THE DIFFERENT LENGTH."
140 OPEN"R",1,F$
150 OPEN"R",2,N$,VAL(NL$)
160 FIELD #2,1 AS D$,(VAL(NL$)-1) AS E$
170 OL=VAL(OL$):L=0:K=0:I=0:J=256/OL
175 ONERRORGOTO300
180 J=INT(J):K=K+1:I=I+1:FIELD#1,((K-1)*OL) AS A$,(OL) AS B$:IFK>J
-1THENK=0
190 GET1,1+INT((I-1)/J):IFB$=STRING$(OL,0)THEN300
200 IFB$=STRING$(OL,250)ORB$=STRING$(OL,255)THEN180
205 IFL=0THENL=1:LSETD$=CHR$(250):LSETE$=STRING$(OL,250):PUT2,1
210 LSET D$=LEFT$(B$,1):LSET E$=MID$(B$,2):L=L+1:PUT2,L:GOTO180
300 IFL=0THENPRINT"ERROR: EMPTY DATA FILE; NO RECORDS TRANSFERRED.
":ELSEPRINTL;"RECORDS TRANSFERRED."
310 IF1=1ANDL=0THENPRINT"ORIGINAL DATA FILE DID NOT EXIST AS NAMED
."
320 PRINTI;"SOURCE RECORDS READ.":PRINTI-L;"BLANK OR DELETED RECOR
DS REMOVED.":CLOSE
330 PRINT256*INT(I/J+.995)-VAL(NL$)*L;"BYTES OF DISK SPACE SAVED."
340 PRINT"RECORD TRANSFER IS COMPLETE.":END
```

✓ See List of Advertisers on Page 211

Grand Opining

by David C. Andresen

If you ever wanted to use your TRS-80 Model I or III to tabulate opinions for your favorite political candidate or election results for a local service club, then MicroTab (see the Program Listing), a general-purpose cross-tabulation program, will interest you. MicroTab gives you low-cost, accurate, easy-to-understand tables. With it you can run many different tabulation projects without special programming.

From political surveys to preference tests, MicroTab tabulates the results of opinion polls.

Back to Basics

To produce a finished table, you need a set of completed questionnaires (see Fig. 1) and a drawing of the way you want your table to look. Figure 2 illustrates how a skeleton table might look for a political preference poll.

Let's say you interviewed 20 prospective voters to see whether they are leaning toward candidate Smith or candidate Jones. You now want to make a table that divides the results according to the respondents' sex.

To do this by hand, you'd have to draw a table on a piece of paper, then go through each questionnaire and make a check mark in the appropriate intersection of the rows and columns in the table. After examining all the questionnaires, you would then total each column and enter the result in the total row at the bottom. Then you could calculate percentages based on each column total. This procedure is fine for one or two tables, but suppose you want to make 10 different tables. You can imagine the tedium involved.

Public Opinion Survey
Candidate Preferences

Hello, I'm taking a public opinion survey about political candidates and I would like to include your opinions. My first question is . . .

1. If the election for mayor were held tomorrow, who would you most likely vote for? Would it be Smith or Jones? (interviewer: circle number by answer)

Smith 1
Jones 2
(Don't Know) 3
(Refused) 4

2. Now I'm going to read several income categories. Would you tell me when I get to the category that best describes your household's annual income? (interviewer: read categories. Circle number for answer)

Less than $10,000 1
$10,000 to $14,999 2
$15,000 to $19,999 3
$20,000 to $24,999 4
$25,000 to $29,999 5
$30,000 or more 6
(Refused) 7

3. Sex of respondent:

Female 1
Male 2

Figure 1. Sample questionnaire.

Enter MicroTab

MicroTab automatically goes through the procedure outlined above. You specify what you want the table to look like, enter the questionnaire data, and let the program count the responses and calculate the percentages. In addition to column percentages, MicroTab computes row percentages.

It also gives you several printing options (see Fig. 3): raw counts (frequencies), column percentages, row percentages, no data (useful for printing labels alone), or no print (used for

The Key Box

Models I and III
16K RAM
Cassette or Disk Basic
132-column printer

dummy rows where you calculate results but don't want to show the data or label).

The number of questionnaires MicroTab processes depends on the length of the questionnaire and the amount of memory in your computer. Experimentation is the best bet for determining the capacity of your system for a particular tabulation project.

The program is designed to accommodate a 132-column printer. It doesn't display tables on the screen because of its small size. If you need, you can change the LPRINTs to Prints or divert the printer to the screen with POKEs. However, the program can display only very small tables on the screen. POKE 16422,88: POKE 16423,4 diverts printer data to the screen on the Model I. POKE 16422,141: POKE 16423,5 restores normal printer operation.

Using MicroTab

The program consists of two components. The first (lines 10–2350) takes care of calculations and printing. You normally wouldn't change this part of the program unless you need custom modifications.

The second component consists of data statements that the user adds to the program. They contain the specifications for the table and all the questionnaire data. To change a table or run a new one, change the data statements and their specifications, but leave the questionnaire data alone. If you have a Merge utility, you can set up specifications for several tables in files (disk or tape), then merge them into the program as needed.

The specification part of the program has several sections: Lines 3000–3999 are the table column specifications; lines 4000–4999 are the table row specifications; lines 5000–5999 are the table title section; lines 6000–6999 are the column titles section; lines 7000–7999 are the row titles section; and lines 8000–8999 are the row print options. Line 9040 contains the number of questionnaires and line 9090 contains the number of questions per questionnaire. Line 9140 contains the row to base column percentages on and line 9190 contains the column to base row percentages on.

The table column specifications section contains the specifications for each of the columns that appear in the table. For example, in Fig. 2 the first column is Women, the second is Men, and so on. The specifications themselves consist of regular statements in Basic. They must follow a certain format, however. First, each statement identifies the table column (COL = n).

Program Listing. MicroTab.

```
10 REM * COPYRIGHT 1981 BY DAVID C. ANDRESEN *
20 REM * 8429 EDGEWATER DR., TACOMA, WA 98499 *
30 REM ********************************************************
40 REM                          SHOW TITLES
50 REM ********************************************************
60       CLS
70       PRINT TAB(26) "MICRO-TAB"
80       PRINT TAB(17) "A CROSS TABULATION PROGRAM"
90       PRINT TAB(21) "FOR MARKET RESEARCH"
100 REM *******************************************************
110 REM                            MENU
120 REM *******************************************************
130      PRINT
140      PRINT "SELECT DESIRED OPERATION:"
150      PRINT
160      PRINT TAB(5) "1) RUN TABLE                  7) EDIT ROW P
RINT SPECS"
170      PRINT TAB(5) "2) EDIT COL SPECS             8) EDIT NO QS
TRES"
180      PRINT TAB(5) "3) EDIT ROW SPECS             9) EDIT NO. Q
STNS/QSTRE"
190      PRINT TAB(5) "4) EDIT TABLE TITLES         10) EDIT COL %
 BASE"
200      PRINT TAB(5) "5) EDIT COL TITLES           11) EDIT ROW %
 BASE"
210      PRINT TAB(5) "6) EDIT ROW TITLES           12) EDIT QSTRE
 DATA"
220      PRINT:INPUT "YOUR CHOICE";A
230      IF A<1 OR A>12 THEN 60
240      IF A=1 THEN 350
250      CLS
260      IF A=2 THEN LIST 3000-3050 ELSE IF A=3 THEN LIST 4000-40
50
270      IF A=4 THEN LIST 5000-5060 ELSE IF A=5 THEN LIST 6000-60
90
280      IF A=6 THEN LIST 7000-7090 ELSE IF A=7 THEN LIST 8000-81
00
290      IF A=8 THEN LIST 9000-9040 ELSE IF A=9 THEN LIST 9050-90
90
300      IF A=10 THEN LIST 9100-9140 ELSE IF A=11 THEN LIST 9150-
9190
310      LIST 9900-9905
320 REM *******************************************************
330 REM                    INITIALIZE VARIABLES
340 REM *******************************************************
350     CLEAR 4000
360     DEFINT A-Z
370     FF$="########":PF$="####.##"
380     DIM TT$(20), CT$(10),RT$(50)
390 REM *******************************************************
400 REM                    INITIALIZE PRINTER (MX-80) TO
410 REM                         TO 132-CHARACTER MODE
420 REM *******************************************************
430      LPRINT CHR$(15);
440 REM *******************************************************
450 REM                    READ TABLE TITLES
460 REM *******************************************************
470      CLS: PRINT "READING TABLE TITLES"
480      NT=0
490      READ W$
500      IF W$="END TABLE TITLES" THEN GOTO 560
510      NT=NT+1
520      IF NT>20 THEN 2250
530      TT$(NT)=LEFT$(W$,110)
540      GOTO 490
550 REM *******************************************************
560 REM                    READ COLUMN TITLES
570 REM *******************************************************
580      PRINT "READING COLUMN TITLES"
590      NC=0
600      READ W$
610      IF W$="END COLUMN TITLES" THEN 670
620      NC=NC+1
630      IF NC>10 THEN 2260
640      CT$(NC)=LEFT$(W$,40)
650      GOTO 600
660 REM *******************************************************
670 REM                         READ ROW TITLES
```

Listing continued

Table 1.
Public Opinion Poll—Mayor Candidate Preferences

	Women	Men	Total Responses
Smith			
Jones			
Total Responses			

Figure 2. Skeleton table before tabulation.

FREQ—Print the frequency or tally
COL %—Print the column percentage
ROW %—Print the row percentage
NO DATA—Print the row title but not the data in that row
NO PRINT—Don't print anything for this row

Figure 3. Printing options.

Table 1.
Public Opinion Poll—Mayor Candidate Preferences

	Women	Men	Total Responses	
Smith	7	3	10	← Frequency
	63.64	33.33	50.00	← Col. %
Jones	4	6	10	
	36.36	66.67	50.00	
Total Responses	11	9	20	
	100.00	100.00	100.00	

Figure 4. Finished table.

Listing continued

```
680 REM ******************************************************
690     PRINT "READING ROW TITLES"
700     NR=0
710     READ W$
720     IF W$="END ROW TITLES" THEN 800
730     NR=NR+1
740     IF NR>50 THEN 2270
750     RT$(NR)=LEFT$(W$,20)
760     GOTO 710
770 REM ******************************************************
780 REM                READ ROW PRINT SPECIFICATIONS
790 REM ******************************************************
800     DIM PS(NR,8)
810     PRINT "READING ROW PRINT SPECS."
820     W=1
830     FOR I=1 TO NR
840         READ W$
850         PRINT @ 218,"ROW: ";I;"  ";W$+"      "
860         IF W$="FREQ" THEN W1=1
870         IF W$="ROW %" THEN W1=2
880         IF W$="COL %" THEN W1=3
890         IF W$="NO PRINT" THEN W1=4
900         IF W$="NO DATA" THEN W1=5
910         IF W$="END" THEN PS(I,0)=W-1:W=1:GOTO 960
920         PS(I,W)=W1
930         W=W+1
940         IF W>3 THEN 2280
950         GOTO 840
960     NEXT
970     READ W$
980     IF W$<>"END PRINT SPECS" THEN 2290
990 REM ******************************************************
```

Listing continued

Next, the program determines whether the answer to a particular question in the questionnaire qualifies to go in that column. The format is IF Q(x) = y THEN GOSUB 4060, where x is the question number and y is the value to be tested for. If the test is successful, then the program executes the GOSUB, which goes to the row tests. You can use any relational operator permitted by Basic in place of the equal sign.

An example will illustrate the procedure. Let's say you want the first column in the table to be for women. Further, let's suppose that the third question in the questionnaire gives the sex of the respondent, with a one meaning a woman and a two meaning a man. The specification, then, would be written this way: 3060 COL = 1:IF Q(3) = 1 THEN GOSUB 4060.

Write all column specifications in a similar manner. Just remember that you have to give the column number, the question number, and question value.

Row specifications are similar to column specifications. They tell the program which row you are working with and what qualifies to go in that row. Their format is ROW = n: IF Q(x) = y THEN GOSUB 2200, where n is the number of the row, x is the question number, and y is the value the program tests. If the test is met, the program goes to the section where it tallies answers (lines 2200–2210).

For example, the first row in the table is for Smith, and the first question in the questionnaire asks which candidate respondents favor, with a one meaning Smith and a two signifying Jones (see Fig. 1). The specification for this row is 4060 ROW = 1: IF Q(1) = 1 THEN GOSUB 2200. Write all row specifications in a similar way. The program operates by testing each column. If the column test is met, it then tests all the rows, putting a tally mark on its imaginary tally sheet in each row/column intersection where that test is successful.

The table titles section lets you put titles on your tables. Such titles might be "Table 1" or "Public Opinion Poll." Simply enter each title as a data statement, with one title per statement. The maximum length is 110 characters—the program truncates anything longer. The maximum number of titles is 20.

Column titles are the labels of the columns in the table. Enter each title as a data statement (only one title per statement). Be sure the number of

THE COMPUTER TANDY SHOULD HAVE BUILT.

Software-selectable screen formats let you run either 24 x 80 CP/M or 16 x 32 and 16 x 64 LDOS programs.

Monitor comes in your choice of green or amber phosphor display.

With our built-in disk interfaces, you can simply plug in any combination of peripherals.

Standard Centronics-type parallel port accommodates a wide variety of printers.

Our 5MHz Z-80B processor runs your programs 25% faster than TRS-80 Model 4.

Rugged construction includes heavy duty case and full-stroke sculptured keyboard.

Available storage includes Lobo's 5-1/4" floppy, 8" floppy and Winchester hard disks.

All Lobo hardware products carry a limited one-year parts and labor warranty.

AT THE PRICE TANDY SHOULD HAVE CHARGED.

The bottom line is this. For far less than the price of a TRS-80* Mod 4, you can own a lot more computer.

A computer that's 25% faster.

That supports both the CP/M* Plus and LDOS™ operating systems.

A computer that features a price tag of only $945 for the MAX-80* processor. And that expands with your needs without breaking your budget.

A FLEXIBLE SYSTEM AT A FLEXIBLE PRICE

When you put it all together—an entire system, including a dual 5-1/4" floppy drive subsystem (320 Kb of disk storage), monitor, CP/M and LDOS—totals up to just $1599.

Totally remarkable.

But that's not all. Because this basic MAX-80 system is incredibly flexible. Its dual operating systems run far more software than any other computer. (Including our specially-discounted Perfect-Calc™ and Perfect-Writer™ software packages.)

And since our MAX-80 features two RS-232C serial ports, you can easily add both printer and modem without switching back and forth.

SUPPORTS A WIDE VARIETY OF PERIPHERALS

What's more, Lobo offers almost any peripheral you might want to add to your MAX-80. At very special prices.

Like $995 for a 5Mb Winchester hard disk, the ultimate in fast, accurate data storage.

Or a whole line of affordable dot matrix and letter-quality printers. Or an 8" floppy drive. Or even a MAX-80 local area network.

EVERY PIECE OF HARDWARE BACKED BY OUR UNBEATABLE WARRANTY

Because every part of the MAX-80 system is so rugged and reliable, we go beyond industry-standard 90 day warranties. We back every piece of Lobo hardware with a full year warranty.

So now, you can run virtually all your TRS-80 programs faster and cheaper.

Just give us your Visa or Master Card number. We'll rush you the MAX-80. Try it out. Then, if for any reason you change your mind, return it within 30 days. We'll return all your money.

So call us. Because owning a MAX-80 costs you $945. But finding out about it costs you nothing.

1-800-235-1245

(1-800-322-6103 in California)

LOBO SYSTEMS, INC.

358 South Fairview Avenue, Goleta, California 93117

535

See List of Advertisers on Page 211

Listing continued

```
1000 REM      READ NO. QUESTIONNAIRES, NO. QUESTIONS,
1010 REM        BASES FOR CALCULATING PERCENTAGES
1020 REM ************************************************
1030     PRINT "READING NO. OF QUESTIONNAIRES:  ";
1040     READ NQ
1050     PRINT NQ
1060     PRINT "READING NO. OF ENTRIES/QUESTIONNAIRE:  ";
1070     READ NE
1080     PRINT NE
1090     PRINT "COL. & BASE:  ROW NO.";
1100     READ CB
1110     PRINT CB
1120     IF CB<1 OR CB>NR THEN 2330
1130     PRINT "ROW & BASE:  COLUMN NO.";
1140     READ RB
1150     PRINT RB
1160     IF RB<1 OR RB>NC THEN 2340
1170     DIM Q(NE+1), TB(NR,NC)
1180 REM ************************************************
1190 REM              READ QUESTIONNAIRE
1200 REM ************************************************
1210    PRINT "PROCESSING QUESTIONNAIRE NO."
1220    ON ERROR GOTO 2300
1230    FOR I=1 TO NQ
1240          PRINT @ 540,I
1250            FOR J=1 TO NE+1
1260                  READ Q(J)
1270            NEXT J
1280          IF Q(NE+1)<>-1 THEN 2320
1290          GOSUB 3060
     'DO COLUMN SPECIFICATION CHECKS
1300     NEXT I
1310 REM ************************************************
1320 REM           PREPARE TO PRINT TABLE
1330 REM ************************************************
1340     LPRINT CHR$(7);      'SOUND BUZZER ON MX-80 PRINTER
1350     PRINT:PRINT "PROCESSING COMPLETE."
1360     LINEINPUT "PRESS ENTER TO PRINT TABLE.";A$
1370 REM ************************************************
1380 REM                 PRINT TABLE
1390 REM ************************************************
1400     FOR I=1 TO 10:LPRINT " ":NEXT   'SPACE DOWN 10 LINES
1410     IF NT=0 THEN 1450     'IF NO TITLES GOTO 1450
1420        FOR I=1 TO NT
1430            LPRINT TT$(I)        'PRINT TITLES
1440        NEXT
1450     LPRINT" ":LPRINT " ":LPRINT " "     'SKIP DOWN 3 LINES
1460 REM ************************************************
1470 REM           STACK COLUMN TITLES
1480 REM ************************************************
1490     IF NC<=8 THEN ST=10 ELSE ST=8
1500     FOR I=1 TO NC
1510        AF=0
1520        IF LEN(CT$(I))>40 THEN CT$(I)=LEFT$(CT$(I),40)
1530        IF LEN(CT$(I))/ST - INT(LEN(CT$(I))/ST)=0 THEN 1560
1540        AF=ST*(1-(LEN(CT$(I))/ST-INT(LEN(CT$(I))/ST)))
1550        AF=INT(AF+.5)
1560        CT$(I)=CT$(I)+STRING$(AF," ")
1570        BF=40-LEN(CT$(I))
1580        CT$(I)=STRING$(BF," ") + CT$(I)
1590     NEXT
1600 REM ************************************************
1610 REM              PRINT STACKED TITLES
1620 REM ************************************************
1630     FOR I=40 TO 1 STEP -ST
1640        LPRINT TAB(20);
1650        FOR J=1 TO NC
1660              LPRINT STRING$(3," ");MID$(CT$(J),41-I,ST);
1670        NEXT
1680        LPRINT " "
1690     NEXT
1700     LPRINT TAB(20);
1710 REM ************************************************
1720 REM              UNDERLINE COLUMN TITLES
1730 REM ************************************************
1740     FOR I=1 TO NC
1750        LPRINT "   ";STRING$(ST,"-");
1760     NEXT I
1770     LPRINT " ":LPRINT " "   'SPACE DOWN 2 LINES
1780 REM ************************************************
1790 REM           PRINT ROW TITLES AND DATA
1800 REM ************************************************
1810     FOR I=1 TO NR
1820       IF PS(I,1)=4 THEN 1940   'NO PRINT OPTION
1830       LPRINT RT$(I);     'PRINT ROW TITLE
1840       IF PS(I,1)=5 THEN LPRINT " ":GOTO 1930   'NO DATA OPT
ION
```

Listing continued

titles matches the number of column specifications or an error results. The maximum title length is 40 characters.

The program automatically wraps around and stacks column titles as it needs to. This may lead to some strange-looking word breaks. Prevent that by inserting spaces in the title—experiment to see what I mean. The maximum number of columns is 10. This assumes a 132-column printer. Use fewer columns if your printer won't go that wide.

The row titles section operates like the column titles section. Enter row titles as data statements. The maximum row title length is 20 characters, and the maximum number of rows is 50.

The row print specifications let you determine the data printed in each row of the table. Figure 3 lists the options.

Enter the specifications as data statements in row order. Make sure the number of print specifications matches the number of row titles. Be sure to put a comma between each option and terminate the print options for each row with the word END.

As long as there is no conflict, each row specification may contain one or more options in any order. For example, the statement FREQ,COL %,ROW % prints the tally, column percentage, and row percentage in that order. You can use a maximum of three options at one time per row.

Enter the number of questionnaires you want to tabulate in line 9040. Then put the number of questions per questionnaire in line 9090. MicroTab uses this information to perform some gross checking on the questionnaire data.

Type the number of the row you want to use as a base for column percentages in line 9140. Do the same for row percentages in line 9190. Enter these two numbers even if you don't plan to use column or row percentages.

If you want to alter a specification, simply edit the appropriate line number. If you want to make a new table, then alter only those specifications that need to be changed. For example, if you use the same set of table columns (header) for many tables, then change the row specifications.

Entering the Data

Lines 10000 onward contain the data for each questionnaire in data statements. The program is set up so that all questionnaire data must be in-

Continued on p. 93

✓ See List of Advertisers on Page 211

DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES

PRICE BREAKTHROUGH

Super Sale on New Disk Drives

Starting at **$169.00!**

Tandon — Siemens — Remex — MPI — Teac — Shugart — Tabor

40 or 80 Tracks — Single or Dual Head — New 3½" Drivette™
Our Disk Drives Are Capable Of Single **And** Dual Density Operation

The **NEWEST** Technology Capable Of Operating On Most Popular Computers

Drive a Hard Bargain!!™ — 5 M.B.-20 M.B. Complete Systems . . . . . from $999.95

Diskette Breakthrough — 10 Pack in Library Case — ~~$18.95~~ NEW LOW PRICE

Since We Are Always Finding Ways To Save You Money, Please CALL For Our Most Current Pricing.

TOLL FREE ORDERING 1-800-343-8841	GENERAL AND TECHNICAL 1-617-872-9090
Model I/III/IV Drives (0 1 2 3)	starting at **$169.00**
Color Computer Drive (0 1 2 3)	**$ Call Toll Free**
Apple/Franklin Compatible Add-On Drives with Case & Cable Now 40 Tracks	~~$219.95~~ **$199.95** NEW LOW PRICE
Apple/Franklin Compatible Drives with Controller Now 40 Tracks	~~$299.95~~ **$259.95** NEW LOW PRICE
Model I/III/IV Memory Upgrade	**Call Toll Free**
Printers — Daisywheel/Dot Matrix	**$ Call Toll Free**
Diskettes in Library Cases	~~$18.95~~ **New Low Price**
Cases and Power Supplies — (Single-Dual-1/2 Height)	starting at $44.95
Printer **Buffers** 8K to 512K	starting at $143.95
Percom Double Density Controller (Model I)	**$ Call Toll Free**
Holmes Model I/III Speed-up Mod-VID/80	starting at $90.00
Color Computer Printer Interfaces	starting at $29.95
Cables — Printer/Disk Drive	starting at $23.00
DOSPLUS	**$ Special Prices**
Repair Services Now Offered — FAST Turn-a-Round	**$ Call Toll Free**

Warranty on Disk Drives — 6 Months — Extended Warranty $ Call Toll Free

SOFTWARE SUPPORT, INC.

327

One Edgell Road, Framingham, MA 01701 (617) 872-9090

Hours: Mon. thru Fri. 9:30 am to 5:30 (E.S.T.) Sat. 10 am to 4:30 pm

DEALER INQUIRIES INVITED.

TERMS:
M.C./Visa/Amex and personal checks accepted at no extra charge.
C.O.D., please add $3.00.
Shipping: Please call for amount.
Not responsible for typographical errors.

CANADA
MICRO R.G.S. INC.
751, CARRE VICTORIA, SUITE 403
MONTREAL, QUEBEC, CANADA, H2Y 2J3
Regular **Tel. (514) 845-1534**
Canadian Toll Free 800-361-5155

Service! Service!

All in stock products are shipped within 24 hours of order. Repair/Warranty service is performed within 24 hours of receipt unless otherwise noted. We accept C.O.D., foreign and APO orders. Schools and D&B corporate P.O.s accepted.

DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES

CLONE I

CLONE III

Dear Customer!

If you care about TIME, you need Clone. Why did you buy your computer? If you are like us, you need your computer to save you time. You follow proper programming procedures and backup everything twice. THIS TAKES TIME! If you use more than one disk operating system, IT TAKES TIME. At the end of our programming day we have at least 8 disks to backup twice. With the disk backup utilities that came with our disk operating systems, formatting and verifying that many disks could take 64 minutes, not to mention the amount of time that it took to initialize and answer the opening inquiries for the various disk operating systems we use. Now we use the Clone duplication system and the entire process takes less than 23 minutes. Clone is so reliable at verifying that we never worry about having a bad duplication. We are sure you know that a disk which has not been properly verified might cause problems you would not detect for months. Clone is so advanced that passwords, densities or different disk operating systems don't affect its efficient operation. Clone is so flexible that we are also able to duplicate Atari 400, Atari 800, TRS-80 Color Computer as well as TRS-80 Model I, III, or IV disks. Clone is so sophisticated that if it encounters a damaged disk and is unable to read it, you can ask it to keep trying, take it's best guess, or give up. Most backup utilities just give up. Clone's error messages will explain exactly what the trouble is. Finally, Clone IS FAST! It takes just 1 minute 25 seconds to format, duplicate, and verify a disk that used to take us 4 minutes to complete.

We are sure that everyone who owns a TRS-80 Model I, III, or IV would benefit from owning the Clone duplication system. Clone will become an indispensable part of your programing library. Unlike copying utilities, Clone will have a lasting usefulness which is not dependent on any other program's availability. That is why we at Gibberman Enterprises are proud to offer you Clone I for TRS-80 Model I or Clone III for TRS-80 Model III or IV.

Sincerely,

James Schoengarth

James Schoengarth
Marketing Director
Gibberman Enterprises

HARDWARE REQUIREMENTS

CLONE I OR CLONE III
TRS-80 MODEL III, IV OR
TRS-80 MODEL I WITH LN DOUBLER OR PERCOM DOUBLER ONLY
2 DISK DRIVES OR MORE
32K RAM OR MORE

NOTE: A MODEL I WITH SINGLE DENSITY MAY ALSO BE USED, HOWEVER YOU WILL ONLY BE ABLE TO COPY STANDARD MODEL I SINGLE DENSITY DISKS!

ORDER INFORMATION

$79.97

CLONE I for TRS-80 MODEL I
CLONE III for TRS-80 MODEL III or IV
For VISA or MASTER CARD Orders
CALL NOW - TOLL FREE (800) 824-7888
Ask for operator # 797
Hawaii & Alaska (800) 824-7919
For more information call (213) 367-0887

Add $3.03 Postage & Handling
California Residents add sales tax

-523

GIBBERMAN ENTERPRISES

Clone I and Clone III are available exclusively through Gibberman Enterprises, authorized dealer for Pegasus Research.

PEGASUS RESEARCH

Gibberman Enterprises - 13000 San Fernando Rd - No. 5 - Sylmar, CA 91342

Continued from p. 89

teger numbers in the range of −32768 to 32767, with the exception of a minus one, which is used as an end-of-questionnaire marker. This means that any question may have somewhat more than 65,000 possible answer categories. While no one would use that many, it illustrates the program's flexibility.

Enter a number for each question even if it would be a blank. This is best handled by assigning a certain number to indicate a blank. As with any data statement, separate each number with a comma. You can spread a questionnaire over any number of data statements. Of course, the longer the questionnaire, the fewer you can get into the computer.

Signify the end of a questionnaire with a negative one. The program uses such a marker to ensure that a data element is not dropped somewhere.

"You can spread a questionnaire over any number of data statements."

Since the data is the last part of the MicroTab program, you don't need to worry about running into other parts of the program as you add data. I usually start the numbers with 10001 and increase it in increments of one. If you enter the questionnaires in their numerical order, it's easy to cross-check data.

Running the Program

After you enter all your specifications and data, type RUN. The program displays a menu that allows you to edit the specifications or process the data into a table. If you choose an editing function, you have to type RUN again when you finish. Be sure your printer is ready when you select the Run Table option.

Several error traps in the program catch mistakes in the specifications or data. They trap most but not all errors. Figure 4 shows an example of a finished table.

Program Notes

Line 1260 reads a questionnaire. Control then passes via a GOSUB to

Listing continued

```
1850       FOR J=1 TO PS(I,0)     'NO. OF DATA TYPES TO PRINT
1860       FOR K=1 TO NC
1870              TB=23+13*(K-1)+ST-LEN(FF$)  'CALCULATE TAB PO
SITION
1880              LPRINT STRING$(TB-PEEK(16539),32);   'TAB OVE
R
1890              ON PS(I,J) GOSUB 2110,2130,2150
1900       NEXT
1910       LPRINT " "
1920       NEXT
1930       LPRINT " "
1940     NEXT
1950 REM ******************************************************
1960 REM                MENU AFTER TABLE PRINTED
1970 REM ******************************************************
1980     CLS
1990     PRINT:PRINT:PRINT "OPTIONS:":PRINT
2000     PRINT TAB(5) "1) PRINT TABLE AGAIN"
2010     PRINT TAB(5) "2) START PROGRAM OVER"
2020     PRINT TAB(5) "3) END PROGRAM"
2030     PRINT:INPUT "YOUR CHOICE";A
2040     IF A<1 OR A>3 THEN 1980
2050     ON A GOTO 1360,2060,2070
2060     RUN
2070     END
2080 REM ******************************************************
2090 REM                DATA TYPES TO PRINT
2100 REM ******************************************************
2110     LPRINT USING FF$;TB(I,K);
       'FREQUENCY
2120     RETURN
2130     IF TB(I,RB)=0 THEN LPRINT USING PF$;0;ELSE LPRINT USING
 PF$;TB(I,K)/TB(I,RB)*100;     'ROW %
2140     RETURN
2150     IF TB(CB,K)=0 THEN LPRINT USING PF$;0;ELSE LPRINT USING
 PF$;TB(I,K)/TB(CB,K)*100;    'COL %
2160     RETURN
2170 REM ******************************************************
2180 REM                TALLY TABLE FREQUENCIES
2190 REM ******************************************************
2200    TB(ROW,COL)=TB(ROW,COL)+1      'TOTAL CELL FREQUENCY
2210    RETURN
2220 REM ******************************************************
2230 REM                ERROR MESSAGES
2240 REM ******************************************************
2250     PRINT "NO 'END' STATEMENT OR TOO MANY TITLES.  MAX = 20
.":GOTO 2350
2260     PRINT "NO 'END' STATEMENT OR TOO MANY COLUMNS.  MAX = 1
0.":GOTO 2350
2270     PRINT "NO 'END' STATEMENT OR TOO MANY ROWS.  MAX = 50."
:GOTO 2350
2280     PRINT "TOO MANY PRINT SPECS. FOR ROW";I:GOTO 2350
2290     PRINT "ERROR IN ROW PRINT SPECS":GOTO 2350
2300     IF ERR/2+1=4 THEN PRINT "NOT ENOUGH QUESTIONNAIRES HAVE
 BEEN ENTERED.":RESUME 2350
2310     PRINT "ERROR IN QUESTIONNAIRE DATA":RESUME 2350
2320     PRINT "ERROR IN RECORD NO.";I:GOTO 2350
2330     PRINT "INVALID ROW NO.":GOTO 2350
2340     PRINT "INVALID COLUMN NO."
2350     LPRINT CHR$(7):STOP   'SOUND BUZZER THEN STOP
2900 REM ******************************************************
2910 REM
2920 REM          USER SPECIFIES THE KIND OF TABLE HE
2930 REM          WANTS IN THE NEXT SECTION.
2940 REM
2950 REM ******************************************************
2990 REM
3000 REM ******************************************************
3010 REM              TABLE COLUMN SPECIFICATIONS
3020 REM
3030 REM       PUT SPECIFICATIONS FOR TABLE COLUMNS HERE
3040 REM                  10 COLUMNS MAXIMUM
3050 REM ******************************************************
3060 COL=1:IF Q(3)=1 THEN GOSUB 4060  'COL. 1 SPEC -- FEMALE
3070 COL=2:IF Q(3)=2 THEN GOSUB 4060
 'COL. 2 SPEC -- MALE
3080 COL=3:GOSUB 4060  'COL. 3 SPEC -- TOTAL RESPONSES
3090 REM    ROOM FOR COL 4 SPEC
3100 REM    ROOM FOR COL 5 SPEC
3110 REM    ROOM FOR COL 6 SPEC
3120 REM    ROOM FOR COL 7 SPEC
3130 REM    ROOM FOR COL 8 SPEC
3140 REM    ROOM FOR COL 9 SPEC
3150 REM    ROOM FOR COL 10 SPEC
3999 RETURN
4000 REM ******************************************************
4010 REM              TABLE ROW SPECIFICATIONS
```

Listing continued

Listing continued

```
4020 REM
4030 REM        PUT SPECIFICATIONS FOR TABLE ROWS HERE
4040 REM                    50 ROWS MAXIMUM
4050 REM*****************************************************
4060 ROW=1:IF Q(1)=1 THEN GOSUB 2200  'ROW 1 SPEC -- SMITH
4070 ROW=2:IF Q(1)=2 THEN GOSUB 2200  'ROW 2 SPEC -- JONES
4080 ROW=3:IF Q(1)>0 AND Q(1)<3 THEN GOSUB 2200  'ROW 3 SPEC --
TOTAL RESPONSES
4090 REM     ROW 4 SPEC
4100 REM     ROW 5 SPEC
4110 REM     ROW 6 SPEC
4120 REM     ROW 7 SPEC
4130 REM     ROW 8 SPEC
4140 REM     ROW 9 SPEC
4150 REM     ROW 10 SPEC
4160 REM     CONTINUE FOR UP TO 50 ROWS
4999 RETURN
5000 REM *****************************************************
5010 REM               TABLE TITLES SECTION
5020 REM
5030 REM        ENTER TABLE TITLES HERE AS DATA STATEMENTS
5040 REM             MAXIMUM LENGTH = 110 CHARS.
5050 REM             MAXIMUM NO. = 20
5060 REM *****************************************************
5070 DATA TABLE 1.:REM     TITLE LINE 1
5080 DATA PUBLIC OPINION POLL -- MAYOR CANDIDATE PREFERENCES:REM
    TITLE LINE 2
5090 REM     TITLE LINE 3
5100 REM     TITLE LINE 4
5110 REM     TITLE LINE 5
5120 REM     TITLE LINE 6
5130 REM     TITLE LINE 7
5140 REM     CONTINUE THIS WAY FOR UP TO 20 TITLE LINES
5999 DATA END TABLE TITLES
6000 REM *****************************************************
6010 REM               COLUMN TITLES SECTION
6020 REM
6030 REM       ENTER COLUMN TITLES HERE AS DATA STATEMENTS
6040 REM             MAXIMUM LENGTH = 40 CHARS.
6050 REM             MAXIMUM NUMBER = 10
6060 REM
6070 REM     NUMBER MUST MATCH NO. OF COLUMN SPECIFICATIONS
6080 REM     GIVEN IN LINES 3000-3999.
6090 REM *****************************************************
6100 DATA WOMEN:REM        COL. 1 TITLE
6110 DATA MEN:REM          COL. 2 TITLE
6120 DATA TOTAL     RESPONSES:REM      COL. 3 TITLE
6130 REM     COL 5 TITLE
6140 REM     COL 6 TITLE
6150 REM     COL 7 TITLE
6160 REM     COL 8 TITLE
6170 REM     COL 9 TITLE
6180 REM     COL 10 TITLE
6999 DATA END COLUMN TITLES
7000 REM *****************************************************
7010 REM                ROW TITLES SECTION
7020 REM
7030 REM        ENTER ROW TITLES HERE AS DATA STATEMENTS
7040 REM             MAXIMUM LENGTH = 20 CHARS.
7050 REM             MAXIMUM NO = 50
7060 REM
7070 REM        NUMBER OF ROW TITLES MUST MATCH NUMBER OF
7080 REM        ROW SPECIFICATIONS IN LINES 4000-4999
7090 REM *****************************************************
7100 DATA SMITH:REM        ROW 1 TITLE
7110 DATA JONES:REM        ROW 2 TITLE
7120 DATA TOTAL RESPONSES:REM    ROW 3 TITLE
7130 REM     ROW 4 TITLE
7140 REM     ROW 5 TITLE
7150 REM     ROW 6 TITLE
7160 REM     ROW 7 TITLE
7170 REM     CONTINUE THIS WAY FOR TOTAL NO. OF ROWS
7999 DATA END ROW TITLES
8000 REM *****************************************************
8010 REM               ROW PRINT SPECIFICATIONS
8020 REM
8030 REM     THESE SPECS TELL WHAT DATA TO PRINT IN EACH ROW.
8040 REM     ENTER SPECS AS DATA STATEMENTS -- ONE FOR EACH
8050 REM     ROW TO PRINT.
8060 REM     OPTIONS:  FREQ
8070 REM               COL %
8080 REM               ROW %
8090 REM     END EACH SET OF SPECS WITH THE WORD 'END'
8100 REM *****************************************************
8110 DATA FREQ,COL %,END:REM      ROW 1 PRINT SPEC
8120 DATA FREQ,COL %,END:REM      ROW 2 PRINT SPEC
8130 DATA FREQ,COL %,END:REM      ROW 3 PRINT SPEC
8140 REM     ROW 4 PRINT SPEC
```

Listing continued

line 3060 where the program checks the questionnaire against the column specifications one at a time. If it passes the checks, the program goes to line 4060 via a GOSUB. Here the program checks all the row specifications. It makes a tally in array TB(ROW,COL) for each specification that it passes (line 2200).

At the end of processing, the array contains a count of all the questionnaires that fit into each intersection of a row and column. The percentages are computed during printing (lines 2110–2160) and are not kept permanently by the program.

Array PS(n,n) keeps a code for each type of data that is to be printed for each row (FREQ, COL %, and so on). MicroTab processes the code in line 1850 to see how many different items it needs to print and again in line 1890 to determine which one to print at the moment. The actual printing is called from that line.

"Add the capability of calculating means, standard deviations, and standard errors—useful enhancements for marketing researchers."

Modifications

Here are some ideas for modifications.

- Eliminate the data statements and use an editor to create disk or tape files instead. The row and column specifications, which are written in Basic, could be treated as a Basic program and merged with MicroTab when needed.
- Add the capability of calculating means, standard deviations, and standard errors. These are useful enhancements for marketing researchers.
- Save finished tables on disk or tape for future retrieval.
- Allow data to be entered interactively for instant tabulations. ■

David Andresen is a marketing research analyst whose hobbies include ham radio and photography. He can be reached at 8103 104th St., Tacoma, WA 98498.

Listing continued

```
8150 REM     ROW 5 PRINT SPEC
8160 REM     ROW 6 PRINT SPEC
8170 REM     CONTINUE FOR TOTAL NO. OF ROWS
8999 DATA END PRINT SPECS
9000 REM ******************************************************
9010 REM              ENTER NO. OF QUESTIONNAIRES HERE
9020 REM              AS A DATA STATEMENT.
9030 REM ******************************************************
9040 DATA 20:REM       I.E., 20 QUESTIONNAIRES
9050 REM ******************************************************
9060 REM           ENTER NO. OF QUESTIONS PER QUESTIONNAIRE
9070 REM           AS A DATA STATEMENT
9080 REM ******************************************************
9090 DATA 3:REM     I.E., 3 QUESTIONS PER QUESTIONNAIRE
9100 REM ******************************************************
9110 REM        ENTER ROW NO. TO USE FOR CALCULATING
9120 REM        COLUMN PERCENTAGES.
9130 REM ******************************************************
9140 DATA 3:REM     I.E., USE ROW 3
9150 REM ******************************************************
9160 REM        ENTER COLUMN NO. TO USE FOR CALCULATING
9170 REM        ROW PERCENTAGES
9180 REM ******************************************************
9190 DATA 3:REM     I.E., USE COLUMN 3
9900 REM ******************************************************
9901 REM        ENTER QUESTIONNAIRE DATA HERE AS
9902 REM        DATA STATEMENTS.
9903 REM
9904 REM        END EACH QUESTIONNAIRE WITH A -1 MARKER
9905 REM ******************************************************
10001 DATA 1,3,1,-1
10002 DATA 2,1,1,-1
10003 DATA 2,4,2,-1
10004 DATA 1,5,1,-1
10005 DATA 1,3,2,-1
10006 DATA 2,1,2,-1
10007 DATA 2,4,2,-1
10008 DATA 2,2,1,-1
10009 DATA 1,6,2,-1
10010 DATA 2,2,1,-1
10011 DATA 1,4,1,-1
10012 DATA 2,5,2,-1
10013 DATA 1,3,1,-1
10014 DATA 1,1,2,-1
10015 DATA 1,4,1,-1
10016 DATA 2,3,2,-1
10017 DATA 2,6,1,-1
10018 DATA 2,5,2,-1
10019 DATA 1,2,1,-1
10020 DATA 1,4,1,-1
```

Model II/12/16 Conversion

CONVERSION BY

Richard Faber
48 Chinian Path
Newton Centre MA 02159

DELETE THE FOLLOWING LINES:

```
8274
```

ADD THE FOLLOWING LINES:

```
25 REM   Modified for the Model II by Richard L. Faber, 2/83
4055 'Note: var. name RW (Mod II vers.) replaces ROW in Mod III version,
     'since ROW is a reserved word for the Model II.
```

EDIT THE FOLLOWING LINES:

```
70      PRINT TAB(32) "MICRO-TAB"
80      PRINT TAB(23) "A CROSS TABULATION PROGRAM"
90      PRINT TAB(27) "FOR MARKET RESEARCH"
160     PRINT TAB(11) "1) RUN TABLE                    7) EDIT ROW PRINT SPECS"
170     PRINT TAB(11) "2) EDIT COL SPECS               8) EDIT NO QSTRES"
180     PRINT TAB(11) "3) EDIT ROW SPECS               9) EDIT NO. QSTNS/QSTRE"
190     PRINT TAB(11) "4) EDIT TABLE TITLES           10) EDIT COL % BASE"
200     PRINT TAB(11) "5) EDIT COL TITLES             11) EDIT ROW % BASE"
210     PRINT TAB(11) "6) EDIT ROW TITLES             12) EDIT QSTRE DATA"
260     IF A=2 THEN LIST 3000-3999 ELSE IF A=3 THEN LIST 4000-4999
270     IF A=4 THEN LIST 5000-5999 ELSE IF A=5 THEN LIST 6000-6999
280     IF A=6 THEN LIST 7000-7999 ELSE IF A=7 THEN LIST 8000-8999
850         PRINT @ 266,"ROW: ";I;"  ";WS+"     "
1240            PRINT @ 668,1
1570        BF=40-LEN(CT$(I))
1880                LPRINT TAB(TB);     'TAB OVER
2200    TB(RW,COL)=TB(RW,COL)+1      'TOTAL CELL FREQUENCY
2300     IF ERR=4 THEN PRINT "NOT ENOUGH QUESTIONNAIRES HAVE BEEN ENTERED.":RESU
ME 2350
2350  LPRINT CHR$(7):STOP     'SOUND BUZZER THEN STOP
4060 RW=1:IF Q(1)=1 THEN GOSUB 2200  'ROW 1 SPEC -- SMITH
4070 RW=2:IF Q(1)=2 THEN GOSUB 2200  'ROW 2 SPEC -- JONES
4080 RW=3:IF Q(1)>0 AND Q(1)<3 THEN GOSUB 2200  'ROW 3 SPEC -- TOTAL RESPONSES
```


See List of Advertisers on Page 211

LOAD 80

Letter Perfect

by J.C. Sprott

Eliminate spelling mistakes with Word Checker, a program that compiles, maintains, and updates a dictionary of thousands of words.

The English language contains approximately half a million words, yet only 60,000 of them are commonly used. In most applications, you might use only a few thousand words, and for these applications a spelling checker program is useful.

The Model I/III dictionary program I wrote, Word Checker, develops a personalized dictionary that includes a self-updating feature: it stores the words you use frequently and purges the ones you use rarely. Word Checker runs on your word processing and Basic programs, and other specialized applications, checking for spelling errors and improving itself with every use.

By retaining only the words you use most often, you can reduce your dictionary to a few thousand words that fit into RAM and save disk input/output time. A 48K machine can hold almost 10,000 five-letter words.

Although I designed Word Checker for word processing, you can also use it to debug Basic programs or to check any type of ASCII data stored on disk.

Running the Program

The program displays your text on the screen one line at a time. When it encounters a word that isn't in the dictionary, the program stops and displays a ? prompt. (It also beeps on machines with sound capability.) Press the enter key to add the word to the dictionary, or press the space bar to bypass an infrequently used word.

The program also stops if it finds a misspelled word. If you have a line printer connected to your system, you can press the space bar at an incorrectly spelled word and the program will print the line that includes the misspelling. Otherwise you must make note of the incorrect word, since the program makes no provision for editing on the spot.

When Word Checker has reviewed the entire file, it prints the word count, updates the dictionary on disk, and returns to the Basic READY prompt.

Program Operation

On boot-up Word Checker tests available memory and clears all but 4,500 bytes for string use in line 40. It then reads a short machine-language routine that produces an audible beep at the cassette port into the U$ in lines 50–90.

The program reads the current dictionary (file name: WORDS/TXT) into memory in lines 100–120. It stores the dictionary words in 256 strings with a maximum length of 62 characters. The dictionary has a capacity of about 2,645 five-letter words.

In line 140, Word Checker asks you to enter the file name of the disk file you want to check and then reads into memory 128 lines at a time to avoid overloading.

Word Storage

Since a sequential search for the words in the dictionary would be far too slow, I developed a hash code instead. It locates which of the 256 lines of dictionary text contains a desired word by generating a hash code, H, in lines 430–440. The hash code then searches that line for the word, using the INSTR function in line 450.

When Word Checker adds a word, it moves that word to the beginning of its line, pushing other words in the line back (lines 510–540). If you enter a new word, the program places it at the beginning of its line and moves the other words back (line 490). When the resulting line exceeds 62 characters, the program truncates the last word, erasing it from the list. When the dictionary is full, the program continuously puts new words at the beginning of

The Key Box

Models I and III
32K RAM
Disk Basic
Printer Optional

the list and purges words you haven't recently used.

In order to reduce the size of the dictionary, I removed control and graphics characters in line 270. I then replaced all lowercase characters with their uppercase equivalents in line 280. You can determine the end of a word by adding a space, a punctuation mark, or a number. For example, the program stores a word like "can't" as "can" in the dictionary. It ignores words of a single character such as "a" and "I" (line 330).

Finally, I assume that words that end with an "s" are plural, and the program removes the "s" in line 340. Unfortunately, words like "class" appear in the dictionary as "clas." If you use the program to check its own dictionary, you will get occasional mismatches. Running the program on its dictionary also inverts the words, so you should run it twice when you do this check.

Suggestions

In word processor applications, store the text in a disk file and run Word Checker on it before you get a printout. To debug a Basic program, save it in ASCII format and run Word Checker on it. In both cases, Word Checker returns you to the original program to make corrections.

The limited length of the dictionary is less of a problem when you keep a separate file of words for each application. I have one dictionary file for personal correspondence, another for business correspondence, and a third for checking Basic programs. Each file contains its own specialized vocabulary.

Word Checker runs at about the speed you can proofread. My Model III has a 4 MHz speed-up modification, but Southern Software's ACCEL3/4 compiles the program so it runs much faster.

You can minimize delays by using the MID$ function on the left of the equals sign whenever possible, but the program still pauses occasionally while it rearranges string space. Prosoft's Trashman program eliminates this problem.

The first time you run the program, it tries to read your yet-to-be-built text file into memory. Ignore the error message and type GOTO 130 to continue. The program then runs without difficulty.

I run everything I write through Word Checker before making a printout, and find it extremely effective for catching misspelled words. ■

You can reach J.C. Sprott at 5002 Sheboygan #207, Madison, WI 53705. A cassette version is available from the author for $19.95; disk version is $24.95.

```
10 CLS: PRINT TAB(26)"WORD CHECKER": PRINT
20 PRINT TAB(22)"by Prof. J. C. Sprott"
30 PRINT TAB(13)"5002 Sheboygan #207, Madison, WI 53705": PRINT
40 CLEAR 0: CLEAR MEM-4500: DEFINT A-Y: DIM A,K,C,H,L1,J,B$,LB,C$,
JH,IN,WC,L,I,IM,M,CP,U$,X,Z,NF$,E,B$(128),A$(256)
50 FOR I=1 TO 20: READ J: U$=U$+CHR$(J): NEXT
60 DATA 205,127,10,77,62,1,243,6,99,238,3,211,255,16,254,13,32
70 DATA 245,251,201
80 X=FRE(U$): X=VARPTR(U$): Z=PEEK(X+1)+256.0*PEEK(X+2)
90 IF Z>32767 THEN DEFUSR=Z-65536 ELSE DEFUSR=Z
100 PRINT TAB(17)"Loading dictionary into memory"
110 OPEN"I",1,"WORDS/TXT"
120 FOR J=1 TO 256: INPUT#1,A$(J): NEXT: CLOSE
130 CLS: IF PEEK(293)=73 THEN CMD"D:0": PRINT
140 INPUT"FILE NAME";NF$
150 I=1: OPEN"I",1, NF$
160 IM=I-128*INT((I-1)/128): LINE INPUT#1,B$(IM)
170 IF I<=128*M THEN 200
180 IF EOF(1) THEN CLOSE: E=0: GOTO 210
190 E=1: IF IM=128 THEN CLOSE: GOTO 210
200 I=I+1: GOTO 160
210 B$=STRING$(255,32): PRINT CHR$(15);: IF M=0 THEN CLS
220 IF M=0 THEN CLS: PRINT"--> Press <ENTER> to enter word in dict
ionary.": PRINT"--> Press <SPACE BAR> to bypass word.": PRINT
230 FOR J=1 TO IM: CP=256.0*PEEK(16417)+PEEK(16416)-15360
240 PRINT@60, STR$(J+128*M);: PRINT@CP,"";
250 LB=LEN(B$(J)): IF LB=0 THEN 370
260 C=0: L1=1: FOR K=1 TO LB: A=ASC(MID$(B$(J),K,1))
270 IF A>127 OR A<32 THEN A=32
280 PRINT CHR$(A);: IF A>90 THEN A=A-32
290 MID$(B$,K,1)=CHR$(A)
300 IF A<65 THEN IF K>L1+1 THEN C=K-L1: C$=MID$(B$,L1,C)
310 IF A>64 AND A<97 AND K=LB THEN C=K-L1+1: C$=MID$(B$,L1,C)
320 IF A<65 OR K=LB THEN L1=K+1
330 IF C<2 THEN 360
340 IF RIGHT$(C$,1)="S" THEN C$=LEFT$(C$,C-1): C=C-1: IF C<2 THEN
360
350 WC=WC+1: GOSUB 430: C=0
360 NEXT
370 PRINT: NEXT: IF E THEN M=M+1: GOTO 150
380 PRINT: PRINT,"Word count =";WC
390 C$="": PRINT,"Updating dictionary on disk"
400 OPEN"O",1,"WORDS/TXT"
410 FOR JH=1 TO 256: PRINT#1,A$(JH): NEXT: CLOSE
420 CLEAR 50: END
430 H=0: FOR JH=1 TO C: H=H+ASC(MID$(C$,JH)): NEXT
440 H=H-256*INT(H/256)+1
450 L=LEN(A$(H)): IN=INSTR(A$(H),C$): IF IN THEN 510
460 PRINT"<?>";: X=USR(200)
470 IF PEEK(14400)=128 THEN IF (PEEK(14312) AND 240)=48 THEN LPRIN
T USING"######";J+128*M;: LPRINT": ";LEFT$(B$(J),L1-1): GOTO 500 EL
SE 500
480 IF PEEK(14400)<>1 THEN 470
490 IF L+C<62 THEN A$(H)=C$+" "+A$(H) ELSE MID$(A$(H),C+2,61-C)=MI
D$(A$(H),1,61-C): GOSUB 540
500 FOR JH=1 TO 3: PRINT CHR$(8);: NEXT: RETURN
510 IF IN<2 OR IN+C>L THEN RETURN
520 IF MID$(A$(H),IN+C,1)<>CHR$(32) THEN RETURN
530 MID$(A$(H),C+2,IN-1)=MID$(A$(H),1,IN-1)
540 MID$(A$(H),1,C)=C$: MID$(A$(H),C+1,1)=CHR$(32): RETURN
```

Program Listing. Listing for Word Checker.


LOAD 80
TO YOUR
RESCUE

Yes! I want LOAD 80 to rescue me!

February 1984

☐ YES, I want this month's **LOAD 80** on disk for $21.47 each. Includes postage and handling (foreign air mail please add $.45 per item)

☐ YES, I want to save money. I'll subscribe to **LOAD 80** on disk beginning with this month's issue. . . $199.97 for 12 issues.

☐ YES, I want a cassette of this month's **LOAD 80** for $11.47 each. Includes postage and handling (foreign air mail please add $.45 per item)

☐ YES, I want to save money. I'll subscribe to LOAD 80 on cassette beginning with this month's issue. . . $99.97 for 12 issues.

Foreign Airmail please add $25.00 per subscription. U.S. Funds drawn on U.S. bank only)

☐ VISA ☐ AMEX ☐ MC ☐ CHECK/MONEY ORDER

NAME _____

ADDRESS _____

CITY _____ STATE _____ ZIP _____

SIGNATURE _____

CARD# _____ EXP. DATE _____

LOAD 80 •80 Pine St.•Peterborough, NH 03458

2-84

YES, I want Color LOAD 80 1983

☐ 1st Color volume (January, February and March 1983) @ $11.47 each

☐ 2nd Color volume (April, May and June 1983) @ $11.47 each

☐ 3rd Color volume (August, September and October 1983) @ $11.47 each

☐ The Best of '82 (January–December) at $16.47 each

Price includes postage and handling
Foreign air mail, please add $.45 per item
U.S. Funds drawn on U.S. bank only

COLOR

PROGRAMS FROM *80 MICRO*

LOAD 80

☐ MC ☐ VISA ☐ AMEX

☐ CHECK / MONEY ORDER

NAME _____

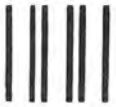
ADDRESS _____

CITY _____ STATE _____ ZIP _____

SIGNATURE _____

CARD# _____ EXP. DATE _____

COLOR LOAD 80 •80 Pine St.•Peterborough, NH 03458 QC-2-84



NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 73 PETERBOROUGH, NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE

WAYNE GREEN INC.
LOAD 80®
ATTN: Jill Jackson
80 PINE STREET
PETERBOROUGH, NH 03458

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 73 PETERBOROUGH, NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE

WAYNE GREEN INC.
LOAD 80®
ATTN: Cecile Giguere
80 PINE STREET
PETERBOROUGH, NH 03458

LET LOAD 80

RESCUE YOU

In the dawn of the computer age programmers needed the patience of Job to manually keyboard all the major program listings from a single issue of 80 MICRO.

Then LOAD 80 burst onto the scene and exploded that practice. Today thousands of TRS-80* owners use LOAD 80 cassettes and disks every month. These time-conscious computerists spend more time enjoying all the benefits of those 80 MICRO programs.

LOAD 80 programs come directly from 80 MICRO giving you complete and detailed documentation right there in the magazine. And now, the new disk transfer system allows Model III users to run LOAD 80 disks without conversions. Save close to 30% on the single issue price by subscribing to LOAD 80 today (choose either monthly cassettes or disks).

Due to the tremendous popularity of color computing we are replacing COLOR LOAD 80 with INSTANT COCO. INSTANT COCO is the color loader backed by Wayne Green's new and exciting color computing magazine, HOT COCO. See the latest issue of HOT COCO for full details. We will still be offering from COLOR LOAD 80, "The Best of '82," and three volumes of color programs from 1983 issues of 80 MICRO.

Start enjoying all the benefits of the programs in 80 MICRO without all the programming hassels. Fill out the attached order form and send it to LOAD 80, 80 Pine Street, Peterborough, NH 03458. Or call toll free and use your Master Card, Visa, or American Express.

****ATTENTION MODEL III USERS****
New, Improved Operating System!
—Ready To Run!!

1-800-258-5473

In NH and outside USA, phone: 603-924-9471

NEW DISK TRANSER SYSTEM ALLOWS **LOAD 80** *TO NOW BE RUN ON MODEL III SINGLE DRIVE UNITS WITH NO CONVERSIONS NECESSARY!!*

LOAD 80 *is simply the listing from* **80 MICRO**. *Use the KEYBOX accompanying each article as your guide to system configurations.* **LOAD 80** *runs on the *TRS-80 Model I and Model III computers only.* **COLOR LOAD 80** *runs on the *TRS-80 Color Computer only. *TRS-80 and Color Computer are trademarks of Radio Shack, a division of Tandy Corp.*

Yes! Send me the February **LOAD** 80. 2-84

☐ Disk $21.47 ☐ Cassette $11.47
Price includes postage and handling.
Foreign air mail please add $.45 per item for postage and handling.
Please enter my subscription for one year, beginning with this month's issue.
☐ Disk $199.97 ☐ Cassette $99.97
Foreign air mail please add $25 per subscription for postage and handling.
U.S. Funds drawn on U.S. Bank only

☐Check/MO ☐MC ☐Visa ☐AE

Card#_____ Exp. date _____

Signature_____

Name _____

Address _____

City _____ State_____ Zip_____

LOAD 80•80 PINE ST.•PETERBOROUGH, NH 03458

Yes! Send me **COLOR LOAD 80** QC-2-84

☐ 1st Color Volume (Jan, Feb, March) @ $11.47
☐ 2nd Color Volume (Apr, May, June) @ $11.47
☐ 3rd Color Volume (August, September, October) @ $11.47
☐ "Best of '82" at $16.47
Price includes postage and handling.
Foreign air mail please add $.45 per item for postage and handling.
U.S. Funds drawn on U.S. Bank only

☐Check/MO ☐MC ☐Visa ☐AE

Card#_____ Exp. date_____

Signature_____

Name _____

Address _____

City _____ State_____ Zip_____

COLOR LOAD 80•80 PINE ST.•PETERBOROUGH, NH 03458

So Much to C

by John B. Harrell III

If you're in the market for software development tools, take a look at this C language compiler from Misosys. LC is a real winner.

LC is a compiler for the C programming language. It's also one of the most advanced and superior products I've had the pleasure of reviewing. If you're writing software in the C language, I strongly recommend you add this compiler to your library. If you're a casual user and want the increased speed of a compiled language, LC is the compiler for you.

LC (otherwise known as Elsie) is a subset of the C programming language. C provides the power and organization of other structured languages like Pascal, but at a lower level. C is considered a medium-level language that allows easy access to the computer's hardware features from within the program.

The Package

The LC package comes in a sturdy binder with the reference manuals for LC and EDAS-IV and the disks for the two systems. The documentation is high quality. Both disks are single-density, 35-track, LDOS-compatible media for the Model I or III.

LC was designed for compatibility with other C compilers and operating systems. In order to make this possible, Misosys incorporated some features of the UNIX operating system into LC.

For example, standard input, output, and console error devices are defined for use with LC. In addition, input/output redirection, full device independence, command line arguments, and dynamic memory allocation are all part of LC's features.

The file redirection that LC allows applies to the standard devices: input, output, and error. The LC program uses these files without any prior declaration, i.e., the system opens these files for you. File redirection lets you assign a file or system device to these C devices at the beginning of program execution.

For example, the short code segment below copies a file from "stdin" to "stdout":

```
main( ) /*COPY FILE stdin TO stdout          */
{
    int c;                    /* declare byte holder */
    while ( (c = getchar( ) ) != -1 /*EOF*/ )
        putchar(c);           /*write byte to stdout */
}
```

If you compile and save this file in the file COPYF/CMD, the command *COPYF<infile>outfile* copies the data from infile to outfile until the program detects the input file's end-of-file marker.

LC allows data in integers (signed or unsigned 16-bit numbers) or characters (8-bit bytes). It doesn't allow any of the other C data types such as floating-point (single- or double-precision). Similarly, the program can define storage classes for the data as *static* (declared, fixed memory space), *auto* (the default storage class, allocated on the stack), *register* (not implemented, regarded as auto), and *extern* (lets variables in one module reference another module).

As an example of LC's programming ease, look at the Program Listing. In lines 5–21, the salient point is the processing of the command line arguments through the argument counter "argc" and the pointer structure "argv". Consider the identifier argv as an array of pointers to character strings, each of which makes up one element of the command line.

The function FOPEN opens the file pointed to by the first parameter. In lines 18 and 20, the expression *++argv increments the pointer to the next element in the command line and uses the resultant address as a pointer to the name of the file. FOPEN returns a pointer to the file parameter block used by LC if the operation is successful.

Actual file reading occurs in line 26. Note that the next byte assignment occurs within the evaluation of the logical

LC and EDAS

★★★★★

Misosys
P.O. Box 4848
Alexandria, VA 22303
Models I and III, 48K
Two disk drives, LDOS 5.1
$150

Easy to use?	★★★★★
Good docs?	★★★★★
Bug-free?	★★★★★
Does the job?	★★★★★

OURS DOES ONE THING RADIO SHACK'S DOESN'T;

SAVES YOU $500.

The Apparat hard disk subsystems for the TRS-80 Model I & III computers give you the high performance, complete compatibility and top value you demand.

These systems use the most advanced 5¼" Winchester technology with transfer rates up to 5 Mbits/sec. and an average access time of 85 msec. And you can add a second drive in the future if your needs grow.

The Apparat hard disks offer complete compatibility with the model III, just plug it in and you're ready to work. They are fully supported by NEWDOS/80 2.5 and RS LDOS 5.1.3.

Top performance and compatibility doesn't have to mean a high price. The disks are available in four configurations; 5 Mb at $1,489, 10 Mb at $1,735, 15 Mb at $1,945 and the 26 Mb at $2,499. These prices offer you top value in a hard disk sub-system. Prices include all required components and software.

For more information write Apparat, Inc., 4401 S. Tamarac Parkway, Denver, CO 80237, 303/741-1778. Dealer inquiries invited. ↵40

Radio Shack and TRS-80 are registered trademarks of Tandy Corp.

Apparat, Inc.

↵ See List of Advertisers on Page 211

expression used to control the While statement.

One of the most powerful features of C is this ability to construct complex expressions performing a variety of operations within a single statement. This is also one of its most confusing aspects.

The conditional expression operator in C is:

```
logical expression ? expression-1 : expression-2
```

This allows the conditional evaluation of an expression dependent on the result of the logical expression. If the logical expression is true, the expression value assumes the value of the first expression and, if false, assumes the value of the second expression.

This is a powerful alternative to the If statement. Note its use in line 29 and 30 to convert the left and right nibbles of the byte to an ASCII character.

Using the increment/decrement (++/--) operators shortens program structure significantly. You can use the increment/decrement operators with any lvalue (an identifier used as the left part of an assignment statement) anywhere in an expression.

The operator can be a prefix or postfix, depending on the desired result. Prefix use performs the operation prior to use in the expression, and postfix use performs the operation after the current value has been used in the expression. The prefix form of ++ in line 34 increments the value of the line counter prior to comparing it to MAXONLINE.

C also provides a compiler preprocessor language consisting of Include, Define, Option, ASM, and ENDASM directives. The use of the Include statement is obvious, but the use of Define is not.

Line 3 in the Listing defines MAXONLINE as equivalent to 20. Any subsequent use of MAXONLINE causes the immediate substitution of the value 20. LC, like C, is case-dependent so the uppercase and lowercase letters have different meanings.

The Option statement tells the compiler/assembler to define certain symbols. These symbols invoke the desired options. For example, the listing uses the Option statement to turn off the ability to perform input/output (I/O) redirection when you load BINHEX. Using the Option statement, you can select other options that affect the assembly phase of translating the program, such as selecting the libraries to scan for undefined references.

"LC is unlike most other language translators. The LC compiler produces Assembly-language source text that you must assemble to produce executable code."

The ASM and ENDASM macro statements let you pass Assembly-language text to the compiler output file without further translation. This lets you write particularly critical portions of the source code in Assembly language. Use of this feature generally destroys the portability of your source code and you should limit it as much as possible.

What LC Does

LC is unlike most other language translators. The LC compiler produces Assembly-language source text that you must assemble to produce executable code.

In order to maintain a relatively small Assembly-language source file, LC uses the many powerful features of EDAS. The entire stack manipulation and data allocation routines are written as macros. This lets you specify a detailed sequence of Assembly-language code with a one-line instruction. Then the assembler expands this macro instruction to produce the necessary code. The macro library is maintained on disk, and EDAS uses the Get command to access the library file at assembly time.

Another innovation in the LC system is Misosys's partitioned data set structure to maintain the source code for the other system libraries LC uses. With EDAS's Search option, the assembler can search partitioned data sets (PDS) in an attempt to resolve undefined references.

EDAS searches through all entries in any order and finds only those entries necessary to complete the assembly. The only requirement is that the PDS member should have the same name as its entry point.

The package includes three function libraries. The program always searches the standard library in any compilation. It doesn't search the installation library and the floating-point library unless requested by the Option statement.

The LC subset of C is defined as strictly an integer subset. However, the floating-point library allows access to the ROM floating-point routines and maintains the floating-point numbers as a character array.

Using this library, you then code algorithms using floating-point arithmetic in Assembly-language calls to the ROM chip. LC provides this through the library function calls, and you can easily calculate complex functions.

EDAS

EDAS is an extremely powerful editor/assembler. The only feature missing is the ability to produce relocatable binary code for use with a linkage editor.

An example of EDAS's versatility is a feature to let you use it as the text editor for LC source programs. EDAS is easy

Test No.	Benchmark Test	LC	Fortran	Pascal-80	TRS-80 Pascal	Tiny Pascal Z80 Code	ZBasic	Disk Basic
1	For/Do Loop	0.260 ms	0.071 ms	0.503 ms	0.710 ms	0.444 ms	0.112 ms	1.872 ms
2	IX = 1	0.019 ms	0.019 ms	0.460 ms	0.246 ms	0.107 ms	0.018 ms	3.192 ms
3	IX = IX + 1	0.045 ms	0.039 ms	0.956 ms	0.486 ms	0.162 ms	0.022 ms	4.723 ms
4	IARRAY(I) = 0	0.131 ms	0.066 ms	2.743 ms	1.548 ms	0.213 ms	0.083 ms	11.142 ms
5	J = K*L	0.618 ms	0.582 ms	1.937 ms	1.274 ms	0.517 ms	0.227 ms	6.041 ms
6	POKE White Out	0.30 sec	0.122 sec	1.24 sec	3.34 sec	0.70 sec	0.116 sec	7.24 sec
7	Set White Out	6.78 sec	n/a	9.47 sec	17.25 sec	6.97 sec	1.85 sec	44.53 sec
8	Sieve Prime No.	13.1 sec	3.94 sec	141.00 sec	50.81 sec	24.03 sec	7.03 sec	501.00 sec
9	Shuffle Cards	4.9 sec	3.52 sec	34.00 sec	19.60 sec	6.82 sec	2.23 sec	149.40 sec

Figure 1. LC execution benchmark data.

1984 OFFICE AUTOMATION CONFERENCE®

A's the place.

LOS ANGELES CONVENTION CENTER • CALIFORNIA
FEBRUARY 20-22, 1984

SPONSORED BY AMERICAN FEDERATION OF INFORMATION PROCESSING SOCIETIES, INC.

OFFICE AUTOMATION & YOU

The 1984 Office Automation Conference, "Office Automation & You," will highlight the most recent advances in OA, as they affect your role as a professional, manager, or user. This is YOUR conference.

OAC '84, the leading annual conference in the office automation industry, offers you the opportunity to:

* Learn more about OA; with over 45 Program Sessions, 8 in-depth Professional Development Seminars, and 6 Industry Workshops.
* Hear authoritative speakers, including Keynote—David T. Kearns, President & CEO of Xerox Corporation.
* See more than 150 major companies exhibiting their products & services.
* View numerous new products and developments introduced at OAC.
* Evaluate and compare specific products and services.

For additional information, complete the coupon below and send to: OAC '84, AFIPS, 1899 Preston White Drive, Reston, VA 22091.

Or, REGISTER TODAY via our toll-free number: 800-OAC-1984, using American Express, MasterCard, or Visa.

111

For more information, mail this coupon today

Name: _____

Title: _____

Company: _____

Address: _____

City: _____ State: _____ Zip: _____

Mail to:
OAC '84
AFIPS
1899 Preston White Dr.
Reston, VA 22091

5 9

See List of Advertisers on Page 211

to use and perfect for creating LC source programs.

EDAS maintains compatibility with EDTASM commands. All the former commands are present and work the same way.

The most important addition to EDAS is the conditional assembly and macro assembly features. These alone give you the ability to construct exceptionally powerful instructions to the assembler and to compress the amount of code you must write. This is one of the advantages of coupling LC to this assembler—LC makes full use of this ability.

Other important features of EDAS are the Get and Search assembler directives. The Get directive assembles source code directly from the file specified. This is equivalent to an Include function.

The Search directive invokes an automatic search of a PDS source code library directory. EDAS searches the library directory, matching entries with the undefined labels in the symbol table. For each such label matched, the corresponding source is assembled into the program. Routines in a PDS can reference routines located in any other PDS without regard to the order in which they are referenced.

LC Performance

To top off LC's power and capabilities as a compiler, the execution performance of the compiled code is superb. The performance criteria that I used to measure LC was very simple. I tried some of the standard benchmarks designed to test program execution, and then I used LC to write just about everything I could think of. Figures 1 and 2 illustrate the results.

Figure 1 represents the data collected from the timing tests using the various languages. I compared LC to each of the other language processors so that I could see how it performed. The times in Fig. 1 indicate that LC is slightly slower than Fortran and ZBasic and faster than any Pascal compiler.

Since LC produces absolute machine code like Fortran and ZBasic, why is there a difference in execution speed? Fortran and ZBasic both produce code with static variable references. Neither

```
1:  # include stdio/csh
2:  # option REDIRECT OFF
3:  # define MAXONLINE 20
4:
5:  main(argc,argv) /* Convert binary file to hex chars          */
6:  int argc;       /* Command format:                          */
7:  int *argv;      /*   BINHEX <binary_file> <hex_file>        */
8:  {
9:      int line_ctr, next_byte, ch1, ch2;
10:     char check_sum;
11:     FILE *bin_file,*hex_file, *fopen( );
12:
13:     if (argc != 3) {
14:         puts("** Bad Parameters\n");
15:         puts("    Proper format: BINHEX <bin_file> <hex_file>\n");
16:         exit(1);
17:     }
18:     if ( (bin_file = fopen(*++argv,"r")) == NULL )
19:         exit(1);
20:     if ( (hex_file = fopen(*++argv,"w")) == NULL )
21:         exit(1);
22:     check_sum = 0;
23:     line_ctr = 1;
24:     puts("\x1C\x1F\x0F");       /* clear the screen */
25:     puts("Converting binary input file to hexadecimal\n\n");
26:     while ( (next_byte = getc(bin_file)) != EOF ) {
27:         ch1 = next_byte & 0x0F;
28:         ch2 = (next_byte & 0xF0) >> 4;
29:         ch1 = (ch1 > 9) ? ch1 + 'A' - 10 : ch1 + '0';
30:         ch2 = (ch2 > 9) ? ch2 + 'A' - 10 : ch2 + '0';
31:         if ( fprintf(hex_file,"%c%c",ch2,ch1) == EOF )
32:             output_error( );
33:         printf("%c%c",ch2,ch1);
34:         if ( ++line_ctr > MAXONLINE ) {
35:             putchar('\n');
36:             putc('\n',hex_file);
37:             line_ctr = 1;
38:         }
39:         else
40:             putchar(' ');
41:         check_sum += next_byte;
42:     }
43:     if ( line_ctr != 1 )
44:         putc('\n',hex_file);
45:     printf("\n\nThe binary file checksum was <%x>\n",check_sum);
46: }
47:
48: output_error( )
49: {
50:     puts("\n** Output file error\n");
51:     exit(1);
52: }
```

Program Listing. Binary to hexadecimal character file converter.

↙ See List of Advertisers on Page 211

of these two languages has dynamic stack manipulation as in LC. The difference in the timing is due to the stack manipulation overhead inherent in any dynamic, re-entrant, recursive language.

Tests 1–5 are simple timing loops designed to test various arithmetic facilities in each of the languages. After calculating the time for the first test, I used this time as a basis for those for each of the remaining statements. Tests 6–9 are more complete programs designed to test other aspects, such as subroutine calls and looping controls.

Speed is not everything. You must also consider overall program size. For example, Fortran is remarkably quick in all respects, but the executable program size is prohibitively large.

When using languages other than Basic, such as compiled languages, it is generally necessary to use some medium to store the resultant intermediate code. LC produces an ASM file as its intermediate work space. Due to the use of the EDAS assembler format, the disk space used is minimal.

The extensive use of macro definitions results in a minimal amount of Assembly-language source code. LC provides the source code in the assembler file as a method of marginally commenting the resultant compiled file. You can further minimize this with a command line option to suppress writing the LC source code file to the ASM file as comments.

"Numerous interest groups are involved in using, and furthering development of, C."

Figure 2 is a comparison of the file sizes generated in the course of producing an executable CMD file under each of the benchmark languages. LC consistently produces far smaller files than either the Fortran or Pascal compilers.

Also, Fortran requires a separate source file for the result of each Edit operation—you cannot overwrite source files. TRS-80 Pascal requires generation of a separate OPT file for each optimization run and a COD file for each CODEGEN (convert to Z80 code) run. All the output files from TRS-80 Pascal are written using ASCII characters, taking roughly twice the space of a binary file.

	Sieve Prime Numbers			*Card Shuffling*		
Language	**Source**	**CMD**	**Other**	**Source**	**CMD**	**Other**
LC	692	3,147	2,870	839	3,332	3,373
Fortran	651	14,903	456	913	6,953	711
Pascal-80	672	12,544	512	970	12,544	512
TRS-80 Pascal	891	13,025	1,054	1,167	13,061	1,273
Tiny Pascal	719	2,000	n/a	874	2,175	n/a
Basic	313	n/a	n/a	408	n/a	n/a
ZBasic	313	2,304	n/a	408	2,304	n/a

Figure 2. LC file size comparisons.

Conclusion

LC is not a limited subset of the C language. It provides the tools for building extremely powerful software.

Anyone can develop this software for resale. The LC license agreement from Misosys states that programs distributed commercially must document the use of portions of the LC libraries and the CMD file must contain the imbedded comment with the copyright notice (this is done automatically by LC/EDAS).

The current version of LC is not the end of development. Misosys intends to provide a full implementation of C in the future. As always, Misosys is very good with consumer support and plans for an upgrade policy.

Two implementation features do require improvement. First, variable declaration does not allow initialization of the values within the declaration statement. This one feature can provide a significant reduction in source code for those variables requiring one-time initialization.

Secondly, the Define macro statement does not function as described by Brian W. Kernighan and Dennis M. Ritchie in *The C Programming Language*. LC provides for a direct substitution of values used in the Define statement. Kernighan and Ritchie describe the Define compiler director as an actual macro definition allowing parameter substitution.

Numerous interest groups are involved in using and furthering development of C. LC is not alone—a special interest user's group with a library of public domain software exists, and Logical Systems has an interest section for LC under the LDOS SIG on CompuServe.

Also, Misosys has recently released LC and EDAS for the native Model 4 mode at the same price. Pro-LC and Pro-Create are among the many packages that Misosys has revised to operate in the Model 4 mode.

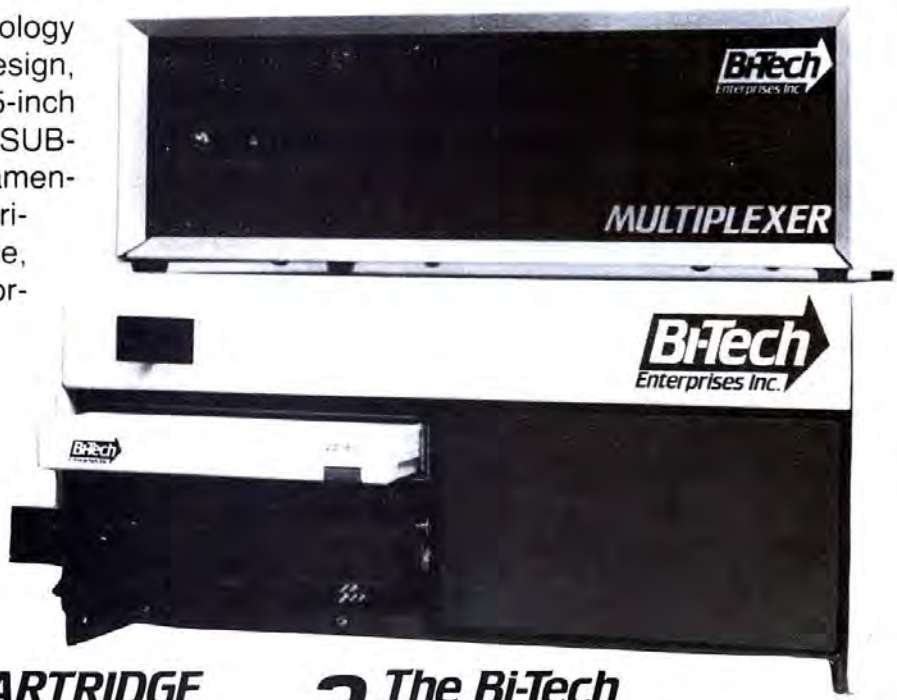
Pro-LC currently lacks the floating point library. Implementing this for the Model 4 involves an extensive development effort to write all the FP routines contained in the Model I/III ROM. This FP library is under development. ■

You can contact John Harrell at 1519A Carswell Circle, Bolling Air Force Base, Washington, DC 20336.

Bi-Tech's 3 ways to make your computer work harder

1 The Bi-Tech HARD DISK SUB SYSTEM

Combining proven technology with innovative Bi-Tech design, the highly reliable 5.25-inch HARD DISK DRIVE SUB-SYSTEM fulfills the fundamental requirements of a peripheral storage device, which provides mass storage, high speed data transfer, and removable backup capability.



2 The Bi-Tech REMOVABLE CARTRIDGE HARD DISK DRIVE

The Bi-Tech REMOVABLE CARTRIDGE used with the REMOVABLE HARD DISK DRIVE is small, lightweight, and portable. Yet, it provides a total storage capacity of 5.0 Mbytes. Pre-recorded embedded servo information provides for the most reliable cartridge interchangeability and data integrity in the industry.

3 The Bi-Tech MULTIPLEXER

Joining the Bi-Tech MULTIPLEXER with your hard disk subsystem enables you to share hard disk storage with many computers. All users can share programs and data present on the hard disk, thus affording a true data base to all of its users. Each computer can run completely independent of the other computers.

BT ENTERPRISES

B.T. Enterprises Dept. 1 B
10 Carlough Road
Bohemia, N.Y. 11716-2996
516 567-8155 (voice)
516 588-5836 (modem)
Orders Only 800-645-1165 ⌐300

Dealer Inquires Welcome
Prices subject to change
N.Y.S. Residents add tax
American Express, Carte Blanche, Diners Club, MasterCard & Visa
B.T. Enterprises is a division of Bi-Tech Enterprises Inc.

⌐ See List of Advertisers on Page 211

Assembly Language Made Simple—Part III

by Hardin Brothers

In this final installment of the Assembly-language series, you'll learn how to break down programs into easily managed parts.

It's time to boot up your assembler, get out the manual, and prepare for your third and final dose of Assembly-language programming (see Parts I and II of "Assembly Language Made Simple," December 1983, p. 74 and January 1983, p. 128). This month, you'll learn another handful of Z80 instructions and a few new opcodes; more important, you'll learn to put a longer program together without getting lost in a maze of instructions.

I include only one program listing this month. While it's long, it's easy to understand. This is characteristic of Assembly language; source code (the code you type in) is longer than object code (the code your editor/assembler assembles).

For instance, the object code of this month's program is only 651 bytes long (equivalent to a 25-line Basic program). But the source code is about 10K bytes long. That's a ratio of 15.7 bytes of source code to 1 byte of object code.

Fully commented source code is long because each line assembles to between 1 and 4 bytes of object code. Lengthy source code is a necessary part of programming in Assembly language. The length of your source code determines the upper limit of your program size more than the length of your object code.

This month's program is a relatively simple game, CRAM 2, based on my Basic game published in *80 Micro* (August 1982, p. 234).

As a first step toward understanding the program and how it works, enter, assemble, and run the program before continuing. As you type in the Program Listing, try to figure out what each instruction does. Once you see how the game works, you'll have an easier time understanding the explanations.

If you use a tape-based assembler on a 16K machine and run out of room, you can omit some of the comments, but try to keep the line numbers the same for reference to the discussion that follows.

Game Overview

For those who wish to ignore my advice to try the game first, here is a short description of what it does. First, the program displays a title screen with flashing graphics. When you press the enter key, the program prompts you to select a difficulty level between zero and 9. The program then clears the screen, frames it, and generates a block cursor in the center.

Use the arrow keys to move the cursor. As it moves around the screen, the cursor leaves a trail of graphics blocks behind it. When the cursor runs into its trail or a wall, the game ends.

If you take your fingers off the keys, the cursor continues in the same direction as your last move. Your score, shown at the top of the screen, is the number of spaces you have filled on the screen.

Planning a Long Program

The first step in planning a lengthy Assembly-language program is to break the task down into small, easily managed chunks. Break each of those chunks into smaller pieces until you end up with a number of easily programmed routines.

The first 700 lines of the Program Listing are almost entirely Call instructions: a good indication of top-down programming.

Lines 300–700 control the logic and flow of the game; the remainder of the program implements the individual routines needed to support that logic.

Top-down programming has four advantages. First, you start by considering only the overall program operation, without concern for individual routines. Second, using your assembler with the No Output and Wait on Errors switches (A/NO/WE or A,NO,WE), always shows you which block you need to write next (it's the first one that shows up as an Undefined Label error during trial assembly).

Third, you can test what you have written at any time by temporarily assembling all unwritten subroutines as simple return (RET) instructions and testing the results of the rest of the program.

The Key Box

Models I and III
16K RAM
Assembly Language
Editor/Assembler

Finally, you need many of the same routines in several different programs. You can solve a programming problem once, and then call the same subroutine over and over, either by chaining it to the end of each program from disk or tape, or by copying it from listings of programs you've already written.

Whatever programming strategy you adopt, it's important to break a program into small sections. Try to limit all routines to a single screen of source code, so you can study the logic of each without scrolling back and forth.

If you simply plunge into the program and try to write everything in a straight line, you may soon find yourself lost in a maze of inconsistent logic and complex instructions. You might know what you are doing now, but if your programming is interrupted for a day or two before you finish, you may never find your way back to the same mental set. Such straight-line programming also makes your program more difficult to debug.

The Program Listing

The first part of the Program Listing is a short section of equates (lines 210 to 250). Video is the address of the video display, and should be familiar from the first two articles in this series. CLS, or 01C9 hexadecimal (hex), is the address of the ROM routine that clears the screen, sets the video processor to 64-character mode, and turns off the cassette relay.

It's a good idea to call 01C9 hex near the beginning of all Assembly-language programs to set the computer into a known state; then you can manipulate the screen and other input/output (I/O) as you wish.

Print, or 0033 hex, is a ROM routine that prints a single character to the current I/O device, usually the screen. It automatically updates the cursor position, and processes control codes as well as normal ASCII and graphics characters.

The fourth equate is a ROM Delay routine (0060 hex) that uses the current value in the BC register as a counter for a simple Delay routine. You could easily write such a Delay routine yourself. For example:

```
LOOP   DEC BC
       LD A,B
       OR C
       JR NZ,LOOP
       RET
```

This routine is the code at 0060 hex in Model I, Level II ROMs. But the Model

Program Listing. Listing for CRAM 2 game program.

```
                00100 ;****************************************
                00110 ;*                                      *
                00120 ;*     Graphics Demonstration Game      *
                00130 ;*                                      *
                00140 ;*           CRAM Version 2             *
                00150 ;*                                      *
                00160 ;*      Written by Hardin Brothers      *
                00170 ;*                                      *
                00180 ;****************************************
                00190 ;
                00200 ;EQU List:
3C00            00210 VIDEO   EQU     3C00H          ;TOP OF SCREEN
01C9            00220 CLS     EQU     01C9H          ;ROM CLEAR-SCREEN ROUTINE
0033            00230 PRINT   EQU     0033H          ;ROM PRINT ROUTINE
0060            00240 DELAY   EQU     0060H          ;ROM DELAY ROUTINE
0049            00250 KEYIN   EQU     0049H          ;ROM KEY INPUT ROUTINE
                00260 ;
7000            00270         ORG     7000H
7000 210070     00280 START   LD      HL,START       ;GET START ADDRESS
7003 F9         00290         LD      SP,HL          ;AND SET STACK BELOW CODE
                00300 ;
7004 CD1270     00310 GAME    CALL    TITLE          ;TITLE ROUTINE
7007 CD3670     00320         CALL    SETUP          ;SETUP GAME BOARD
700A CD4E70     00330         CALL    PLAY           ;PLAY GAME
700D CD6670     00340         CALL    END            ;END OF GAME ROUTINE
7010 18F2       00350         JR      GAME           ;LOOP BACK
                00360 ;
                00370 ;
7012 CDC901     00380 TITLE   CALL    CLS            ;ROM CLEAR-SCREEN ROUTINE
7015 CD7670     00390         CALL    GRPHCL         ;CLEAR SCREEN W/ GRAPHICS
7018 CD8470     00400         CALL    PTITLE         ;PRINT TITLE
701B CD8F71     00410         CALL    SHWSCR         ;AND PREVIOUS SCORE
701E CD9170     00420 TITL10  CALL    FLASH          ;FLASH SCREEN
7021 CDA770     00430         CALL    TONE           ;PLAY TONE
7024 010018     00440         LD      BC,1800H       ;DELAY VALUE
7027 CD6000     00450         CALL    DELAY          ;WAIT A BIT
702A CDDC70     00460         CALL    GETENT         ;CHECK FOR <ENTER> KEY
702D 28EF       00470         JR      Z,TITL10       ;LOOP IF NO <ENTER>
702F CDE270     00480         CALL    NOKEY          ;ELSE WAIT UNTIL KEY UP
7032 CDEF70     00490         CALL    GETDIF         ;GET DIFFICULTY LEVEL
7035 C9         00500         RET                    ;RETURN TO MAIN DRIVER
                00510 ;
7036 CD7670     00520 SETUP   CALL    GRPHCL         ;CLEAR W/ GRAPHICS SPACE
7039 CD1071     00530         CALL    FRAME          ;FRAME SCREEN
703C CD2D71     00540         CALL    SETCRS         ;SET CURSOR & COUNTER
703F CD8171     00550         CALL    DISPLY         ;DISPLAY STARTING POSN
7042 CD3B71     00560         CALL    GAMMSG         ;SHOW GAME MESSAGE
7045 CD4871     00570 SET10   CALL    GETKEY         ;GET KEYSTROKE
7048 28FB       00580         JR      Z,SET10        ;LOOP UNTIL KEYSTROKE
704A CD1071     00590         CALL    FRAME          ;COMPLETE FRAME AGAIN
704D C9         00600         RET                    ;RETURN TO MAIN DRIVER
                00610 ;
704E CD4D71     00620 PLAY    CALL    GETARR         ;GET ARROWS PRESSED
7051 CD5771     00630         CALL    FPOSN          ;FIND NEW POSITION
7054 CD7A71     00640         CALL    CHKEND         ;CHECK IF GAME OVER
7057 C0         00650         RET     NZ             ;RETURN TO DRIVE IF IT IS
7058 CD8171     00660         CALL    DISPLY         ;DISPLAY NEW POSITION
705B CD8771     00670         CALL    SCORE          ;UPDATE SCORE
705E CD8F71     00680         CALL    SHWSCR         ;DISPLAY SCORE
7061 CDCB71     00690         CALL    PAUSE          ;PAUSE BETWEEN MOVES
7064 18E8       00700         JR      PLAY           ;LOOP BACK & START OVER
                00710 ;
7066 060A       00720 END     LD      B,0AH          ;COUNTER FOR FLASHES
7068 C5         00730 END10   PUSH    BC             ;SAVE COUNTER
7069 CD9170     00740         CALL    FLASH          ;FLASH SCREEN
706C 010018     00750         LD      BC,1800H       ;DELAY VALUE
706F CD6000     00760         CALL    DELAY          ;WAIT A BIT
7072 C1         00770         POP     BC             ;GET END COUNTER
7073 10F3       00780         DJNZ    END10          ;REPEAT 10 TIMES
7075 C9         00790         RET                    ;RETURN TO DRIVER
                00800 ;
7076 21003C     00810 GRPHCL  LD      HL,VIDEO       ;HL==>BEGINNING OF SCREEN
7079 11013C     00820         LD      DE,VIDEO+1     ;DE==> NEXT SPACE
707C 01FF03     00830         LD      BC,3FFH        ;BC=# OF SCREEN POSN'S -1
707F 3680       00840         LD      (HL),80H       ;SET 1ST POSITION
7081 EDB0       00850         LDIR                   ;SET ALL THE REST
7083 C9         00860         RET                    ;RETURN TO CALLER
                00870 ;
7084 21FF3C     00880 PTITLE  LD      HL,VIDEO+255   ;HL==>PRINT POSITION
7087 224020     00890         LD      (4020H),HL     ;SET CURSOR POSITION
708A 217272     00900         LD      HL,TMSG        ;HL==>TITLE MESSAGE
708D CDDD71     00910         CALL    PRMSG          ;PRINT THE MESSAGE
7090 C9         00920         RET                    ;THEN RETURN
                00930 ;
7091 21003C     00940 FLASH   LD      HL,VIDEO       ;HL==>BEG. OF SCREEN
7094 7E         00950 FL10    LD      A,(HL)         ;GET CHAR. FROM SCREEN
7095 17         00960         RLA                    ;BIT 7 TO CARRY FLAG
7096 3008       00970         JR      NC,FL20        ;GO IF NO CARRY
7098 17         00980         RLA                    ;BIT 6 TO CARRY FLAG
7099 3805       00990         JR      C,FL20         ;GO IF CARRY
709B 2F         01000         CPL                    ;COMPLEMENT BITS 0 - 5
709C 1F         01010         RRA                    ;RESTORE BIT 6
709D 37         01020         SCF                    ;SET CARRY FLAG
709E 1F         01030         RRA                    ;RESTORE BIT 7
709F 77         01040         LD      (HL),A         ;PUT CHARACTER ON SCREEN
70A0 23         01050 FL20    INC     HL             ;HL==> NEXT CHARACTER
70A1 7C         01060         LD      A,H            ;GET MSB
70A2 FE40       01070         CP      40H            ;TEST FOR END
70A4 C8         01080         RET     Z              ;RETURN IF DONE
70A5 18ED       01090         JR      FL10           ;ELSE DO NEXT CHARACTER
                01100 ;
70A7 DD2AF071   01110 TONE    LD      IX,(TONPTR)    ;IX==>NEXT TONE VALUE
70AB DD7E00     01120         LD      A,(IX)         ;GET VALUE
```

Listing continued

Listing continued

```
70AE B7        01130          OR      A              ;SET FLAGS
70AF 2004      01140          JR      NZ.TONE10      ;GO IF NOT ZERO
70B1 DD21F871  01150          LD      IX,TONTOP      ;ELSE IX==>TOP OF LIST
70B5 CDC170    01160 TONE10   CALL    MAKTON         ;MAKE THE TONE
70B8 DD23      01170          INC     IX             ;BUMP POINTER
70BA DD23      01180          INC     IX             ;TWICE
70BC DD22F071  01190          LD      (TONPTR),IX    ;AND SAVE IT
70C0 C9        01200          RET                    ;RETURN TO CALLER
               01210 ;
70C1 F3        01220 MAKTON   DI                     ;TURN OFF INTERRUPTS
70C2 DD4E00    01230          LD      C,(IX)         ;GET DURATION IN C
70C5 DD4601    01240 MAK10    LD      B,(IX+1)       ;GET FREQ. VALUE
70C8 3E01      01250          LD      A,1            ;OUTPUT VALUE
70CA D3FF      01260          OUT     (0FFH),A       ;SEND TO CASS. PORT
70CC 10FE      01270 MAK20    DJNZ    MAK20          ;DELAY FOR 1/2 CYCLE
70CE DD4601    01280          LD      B,(IX+1)       ;GET FREQ. VALUE AGAIN
70D1 3E02      01290          LD      A,2            ;OUTPUT VALUE
70D3 D3FF      01300          OUT     (0FFH),A       ;SEND TO CASS. PORT
70D5 10FE      01310 MAK30    DJNZ    MAK30          ;DELAY FOR 1/2 CYCLE
70D7 0D        01320          DEC     C              ;DROP DURATION COUNTER
70D8 20EB      01330          JR      NZ.MAK10       ;LOOP UNTIL C=0
70DA FB        01340          EI                     ;TURN INTERRUPTS ON
70DB C9        01350          RET                    ;AND RETURN
               01360 ;
70DC 3A4038    01370 GETENT   LD      A,(3840H)      ;GET <ENTER> ROW
70DF E601      01380          AND     1              ;MASK ALL BUT <ENTER> KEY
70E1 C9        01390          RET                    ;THEN RETURN
               01400 ;
70E2 C5        01410 NOKEY    PUSH    BC             ;SAVE BC ON STACK
70E3 06FF      01420          LD      B,0FFH         ;B = 255
70E5 3AFF3B    01430 NOKY10   LD      A,(3BFFH)      ;CHECK KEYBOARD
70E8 B7        01440          OR      A              ;SET FLAGS
70E9 20FA      01450          JR      NZ.NOKY10      ;LOOP UNTIL NO KEY
70EB 10F8      01460          DJNZ    NOKY10         ;DO 255 TIMES
70ED C1        01470          POP     BC             ;RECOVER BC VALUE
70EE C9        01480          RET                    ;RETURN TO CALLER
               01490 ;
70EF CDC901    01500 GETDIF   CALL    CLS            ;ROM CLEAR SCREEN ROUTINE
70F2 212C72    01510          LD      HL,DIFMSG      ;HL==>DIFFICULTY MESSAGE
70F5 CDDD71    01520          CALL    PRMSG          ;PRINT THE MESSAGE
70F8 CD4900    01530 GD10     CALL    KEYIN          ;GET KEY INPUT
70FB FE30      01540          CP      '0'            ;CHECK LOWER LIMIT
70FD 38F9      01550          JR      C,GD10         ;LOOP IF TOO LOW
70FF FE3A      01560          CP      '9'+1          ;CHECK TOP LIMIT
7101 30F5      01570          JR      NC,GD10        ;LOOP IF TOO HIGH
7103 D62F      01580          SUB     2FH            ;PUT IN 1 - 10 RANGE
```

Listing continued

III uses a slightly different code to give approximately the same time delay for an equal value in BC. By using the ROM routine, you can ensure that your program will run at approximately the same speed on both machines.

KEYIN, or 0049 hex, is a ROM routine that continually strobes the keyboard waiting for a keystroke, translates that keystroke into its equivalent ASCII code, and returns it in the A register. KEYIN is a handy, powerful routine that gets you around the problem of how to read and decode the TRS-80 keyboard.

The ORG pseudo-op on line 270 should be familiar. The address given, 7000 hex, lets this program reside comfortably in 16K, 32K, or 48K memory with either tape or disk operating systems.

The next two instructions, however, need some explanation. Unlike Basic routines, any time you write a machine-language program, you need to be concerned about the location of the stack. You don't want the stack to save values on top of your program and wipe out its instructions. Since the stack builds down in memory, it's wise to place it just below your program, where it can do no harm. However, you must then make sure that the program cannot return directly to DOS or Basic unless you press the reset key. Reset lets the system reestablish the stack in the proper location after program execution.

Line 280 loads the HL register with the program's starting address, and line 290 sets the top of stack to this address. The first value saved on the stack is immediately below the program in memory locations 6FFE hex and 6FFF hex. Too often, novice programmers forget to set the stack and then wonder why their programs work for a while and then bomb out.

Software Drivers

The five instructions starting on line 310 (Title, Setup, Play, End, and Start Over) define the entire logical control of the program. First, the program calls a Title routine and sets up the game board. Then the game runs, the program calls a routine for the end of the game, and everything starts over. This is the whole program.

Notice that as long as each of the four subroutines returns properly, there's no way out of this loop. A loop like this is often called a program driver, because it drives and controls the logic of the program. This driver is very simple in structure; but once you understand the logic

of the major program drivers, you will have solved most of your programming problems.

The driver has no decision paths and no conditional calls or jumps. It does contain one subtlety, however: It repeats the Title section for each game so you can select a new difficulty level.

The Title Routine

The remainder of the program is a collection of subroutines, many of which call other subroutines. The main driver calls the first four subroutines: Title, Setup, Play, and End. These are still high enough in the program to contain little actual code; they usually call the sub-subroutines they need. Each is, by itself, a program driver.

The Title routine, beginning on line 380, calls the ROM CLS routine to reset the video to 64-character mode, and then calls a second Clear Screen routine, GRPHCL, to set the entire screen with graphics spaces, CHR$(128). Setting the screen with graphics spaces lets the Flash routine (described later) operate.

Finally, Title calls PTITLE, the routine that prints the title message on the screen. Next, Title calls SHWSCR, a routine that displays the score from the previous game at the top of the screen.

SHWSCR was a late addition to the Title routine, made when I decided to take out the section of the End routine that looked for a keystroke before returning to the title. This change shows how easy it is to modify program logic using top-down programming techniques: I only had to move a Call subroutine instead of an entire section of code.

The Title routine includes an inner loop, TITL10, starting at line 420. This routine flashes the screen and plays a tone (you'll see the effect when you run the program). Next the program loads the BC register pair with 1800 hex and calls the delay program for a pause of one-third of a second. Machine-language programs often operate too fast, so add pauses, especially to Display routines, to make them look right.

Line 460 calls GETENT, a subroutine that looks to see if you have pressed the enter key and sets the Z flag. If you haven't pressed the enter key, the program loops back to TITL10 to start the sequence over. If you have pressed the enter key, the call to NOKEY waits until you have released all of the keys and then calls GETDIF, a routine that asks the player to set a difficulty level. Finally, in line 500, control returns to the

Listing continued

```
7105 CB27     01590          SLA     A               ;MULTIPLY BY 2
7107 CB27     01600          SLA     A               ;VALUE IN 4 - 40 RANGE
7109 32E671   01610          LD      (DIFBUF),A      ;SAVE VALUE
710C CDE270   01620          CALL    NOKEY           ;WAIT FOR KEY RELEASE
710F C9       01630          RET                     ;AND RETURN
              01640 ;
7110 21FF3B   01650 FRAME    LD      HL,VIDEO-1      ;HL==>SCREEN TOP -1
7113 0641     01660          LD      B,41H           ;TOP POSITIONS + 1
7115 3EBF     01670          LD      A,0BFH          ;FULL GRAPHICS BLOCK
7117 23       01680 FR10     INC     HL              ;POINT TO NEXT POSN
7118 77       01690          LD      (HL),A          ;SET GRAPHICS BLOCK
7119 10FC     01700          DJNZ    FR10            ;REPEAT FOR TOP LINE + 1
711B 060E     01710          LD      B,0EH           ;B = # OF LINES
711D 113F00   01720          LD      DE,3FH          ;OFFSET FOR EACH LINE
7120 19       01730 FR20     ADD     HL,DE           ;SKIP MIDDLE OF SCREEN
7121 77       01740          LD      (HL),A          ;SET BLOCK
7122 23       01750          INC     HL              ;HL==> BEG. OF NEXT LINE
7123 77       01760          LD      (HL),A          ;SET BLOCK
7124 10FA     01770          DJNZ    FR20            ;REPEAT UNTIL SIDES SET
7126 063F     01780          LD      B,3FH           ;SPACES ON BOTTOM LINE -1
7128 23       01790 FR30     INC     HL              ;POINT TO NEXT SPACE
7129 77       01800          LD      (HL),A          ;SET BLOCK
712A 10FC     01810          DJNZ    FR30            ;REPEAT FOR BOTTOM
712C C9       01820          RET                     ;AND RETURN
              01830 ;
712D 21DF3D   01840 SETCRS   LD      HL,VIDEO+479    ;HL==> MIDDLE OF SCREEN
7130 22E771   01850          LD      (CRSPOS),HL     ;SET CURSOR POSITION
7133 AF       01860          XOR     A               ;A = 0
7134 32E971   01870          LD      (SCRBUF),A      ;ZERO BOTH BYTES OF
7137 32EA71   01880          LD      (SCRBUF+1),A    ;  CURRENT SCORE
713A C9       01890          RET                     ;AND RETURN
              01900 ;
713B 21D53F   01910 GAMMSG   LD      HL,3FD5H        ;PRINT POSITION
713E 222040   01920          LD      (4020H),HL      ;SET CURSOR
7141 217472   01930          LD      HL,GMSG         ;HL==>GAME MESSAGE
7144 CDDD71   01940          CALL    PRMSG           ;PRINT IT
7147 C9       01950          RET                     ;AND RETURN
              01960 ;
7148 3AFF3B   01970 GETKEY   LD      A,(3BFFH)       ;CHECK ALL KEYS
714B B7       01980          OR      A               ;SET FLAGS
714C C9       01990          RET                     ;AND RETURN
              02000 ;
714D 3A4038   02010 GETARR   LD      A,(3840H)       ;GET ARROW ROW
7150 E678     02020          AND     78H             ;MASK ALL BUT ARROWS
7152 C8       02030          RET     Z               ;RETURN IF NO ARROWS
7153 32EB71   02040          LD      (SAVARR),A      ;ELSE SAVE VALUE
7156 C9       02050          RET                     ;AND THEN RETURN
              02060 ;
```

Listing continued

Listing continued

```
7157 2AE771    02070 FPOSN   LD      HL,(CRSPOS)     ;HL=CURRENT SCREEN POSN
715A 114000    02080         LD      DE,40H          ;OFFSET BETWEEN LINES
715D 3AEB71    02090         LD      A,(SAVARR)      ;GET ARROW DIRECTION
7160 B7        02100         OR      A               ;RESET CARRY FLAG
7161 CB5F      02110         BIT     3,A             ;CHECK FOR UP-ARROW
7163 2802      02120         JR      Z,FPO10         ;GO IF NOT
7165 ED52      02130         SBC     HL,DE           ;ELSE MOVE UP ONE LINE
7167 CB67      02140 FPO10   BIT     4,A             ;CHECK FOR DOWN-ARROW
7169 2801      02150         JR      Z,FPO20         ;GO IF NOT
716B 19        02160         ADD     HL,DE           ;ELSE MOVE DOWN A LINE
716C CB6F      02170 FPO20   BIT     5,A             ;CHECK FOR LEFT-ARROW
716E 2801      02180         JR      Z,FPO30         ;GO IF NOT
7170 2B        02190         DEC     HL              ;MOVE LEFT ONE SPACE
7171 CB77      02200 FPO30   BIT     6,A             ;CHECK FOR RIGHT-ARROW
7173 2801      02210         JR      Z,FPO40         ;GO IF NOT
7175 23        02220         INC     HL              ;MOVE RIGHT ONE SPACE
7176 22E771    02230 FPO40   LD      (CRSPOS),HL     ;SAVE NEW POSTION
7179 C9        02240         RET                     ;AND RETURN
               02250 ;
717A 2AE771    02260 CHKEND  LD      HL,(CRSPOS)     ;HL==>NEXT SCREEN POSN
717D 7E        02270         LD      A,(HL)          ;GET CHARACTER THERE
717E FE80      02280         CP      80H             ;IS IT GRAPHICS SPACE?
7180 C9        02290         RET                     ;RETURN WITH FLAG SET
               02300 ;
7181 2AE771    02310 DISPLY  LD      HL,(CRSPOS)     ;HL==>NEXT SCREEN POSN
7184 36BF      02320         LD      (HL),0BFH       ;SET WITH GRAPH. BLOCK
7186 C9        02330         RET                     ;AND RETURN
               02340 ;
7187 2AE971    02350 SCORE   LD      HL,(SCRBUF)     ;GET CURRENT SCORE
718A 23        02360         INC     HL              ;ADD ONE
718B 22E971    02370         LD      (SCRBUF),HL     ;SAVE NEW SCORE
718E C9        02380         RET                     ;AND RETURN
               02390 ;
718F CD9F71    02400 SHWSCR  CALL    XLATE           ;TRANSLATE BIN => ASCII
7192 211E3C    02410         LD      HL,VIDEO+30     ;GET DISPLAY POS'N
7195 222040    02420         LD      (4020H),HL      ;SET CURSOR
7198 21EC71    02430         LD      HL,ASCORE       ;HL==>ASCII SCORE
719B CDDD71    02440         CALL    PRMSG           ;PRINT IT
719E C9        02450         RET                     ;AND RETURN
               02460 ;
719F 2AE971    02470 XLATE   LD      HL,(SCRBUF)     ;GET BIN. SCORE FROM BUF.
71A2 FD21F271  02480         LD      IY,PWRTAB       ;IY==>POWERS OF 10
71A6 DD21EC71  02490         LD      IX,ASCORE       ;IX==>ASCII SCORE BUFFER
71AA FD5E00    02500 XLA10   LD      E,(IY+0)        ;GET LSB OF POWER
71AD FD5601    02510         LD      D,(IY+1)        ;GET MSB OF POWER
71B0 AF        02520         XOR     A               ;A = 0
71B1 B7        02530 XLA20   OR      A               ;RESET CARRY FLAG
```

Listing continued

main program driver.

Notice that the program makes only one decision in the Title section: whether or not to loop back to TITL10. The logic in all of the program drivers is straightforward and simple to understand.

The Setup Routine

The program's second subroutine, Setup, is another sub-driver that makes only one decision. In line 520, Setup calls the GRPHCL routine and fills the screen with graphics spaces (used with Flash when the game is over).

Then it calls the Frame routine to set a border around the screen (a technique you learned in my January installment). The routine then calls SETCRS to establish the starting position of the game cursor and reset the score counter to zero.

The DISPLY subroutine displays the game cursor in the starting position, and the GAMMSG subroutine prints the game instruction at the bottom of the screen.

Once set up, the game waits for the player to begin. The SET10 loop uses the GETKEY routine to see if you have pressed any key. If you haven't, the routine sets the Z flag and performs

continual calls to GETKEY. Push any key to begin the game. In line 590, the Frame routine redraws the screen border and erases the game instructions. Finally, control returns to the main driver for the Play routine.

The Play Routine

So far the program has been quite simple. I have displayed a title screen, selected a difficulty level, and set up the game board.

The Play sub-driver in line 620 is no more complex than Title and Setup. The arrow keys check the GETARR routine to see if the player wants to change directions. Then FPOSN (find position) finds the next position for the game cursor on the board.

CHKEND then calls to see if the next cursor position will be a collision and an end to the game. If no collision occurs, the routine sets the Z flag and the game continues; otherwise, control returns to the main driver with the conditional RET NZ instruction (line 650).

Assuming the game isn't over, DISPLY displays the game cursor at the new position and Score updates the player's score (displayed at the top of the screen by SCRSHW). Then the

Listing continued

```
71B2 ED52     02540          SBC     HL,DE          ;SUBTRACT CURRENT POWER
71B4 3803     02550          JR      C,XLA30        ;GO IF CARRY
71B6 3C       02560          INC     A              ;ELSE INCR. COUNTER
71B7 18F8     02570          JR      XLA20          ;GO BACK & DO IT AGAIN
71B9 19       02580 XLA30    ADD     HL,DE          ;HL=VALUE BEFORE CARRY
71BA C630     02590          ADD     A,30H          ;CHANGE TO ASCII VALUE
71BC DD7700   02600          LD      (IX),A         ;SAVE ASCII VALUE
71BF 7B       02610          LD      A,E            ;GET LSB OF POWER
71C0 FE01     02620          CP      1              ;IS IT 1?
71C2 C8       02630          RET     Z              ;RETURN IF IT IS
71C3 DD23     02640          INC     IX             ;ELSE IX==>NEXT CHAR.
71C5 FD23     02650          INC     IY             ;BUMP IY TWICE TO
71C7 FD23     02660          INC     IY             ;  POINT TO NEXT POWER
71C9 18DF     02670          JR      XLA10          ;AND DO IT AGAIN
              02680 ;
71CB 3AE671   02690 PAUSE    LD      A,(DIFBUF)     ;GET DIFFICULTY VALUE
71CE 47       02700          LD      B,A            ;PUT IN B REGISTER
71CF C5       02710 PAUS10   PUSH    BC             ;SAVE VALUE
71D0 010001   02720          LD      BC,100H        ;GET DELAY VALUE
71D3 CD6000   02730          CALL    DELAY          ;WAIT A BIT
71D6 CD4D71   02740          CALL    GETARR         ;GET ARROW KEYS
71D9 C1       02750          POP     BC             ;RECOVER VALUE
71DA 10F3     02760          DJNZ    PAUS10         ;REPEAT UNTIL 0
71DC C9       02770          RET                    ;THEN RETURN
              02780 ;
71DD 7E       02790 PRMSG    LD      A,(HL)         ;GET CHARACTER
71DE B7       02800          OR      A              ;SET FLAGS
71DF C8       02810          RET     Z              ;RETURN IF ZERO
71E0 CD3300   02820          CALL    PRINT          ;ELSE PRINT CHAR.
71E3 23       02830          INC     HL             ;BUMP POINTER
71E4 18F7     02840          JR      PRMSG          ;AND REPEAT
              02850 ;
              02860 ;**** Buffers ****
0001          02870 DIFBUF   DEFS    1              ;1 BYTE BUFFER
0002          02880 CRSPOS   DEFS    2              ;2 BYTE BUFFER
71E9 0000     02890 SCRBUF   DEFW    0000H          ;2 BYTE BUFFER
71EB 40       02900 SAVARR   DEFB    40H            ;START GOING LEFT
71EC 0000     02910 ASCORE   DEFW    0000H          ;3 BYTE BUFFER FOLLOWED
71EE 0000     02920          DEFW    0000H          ;  BY ZERO BYTE
71F0 F871     02930 TONPTR   DEFW    TONTOP         ;START ==> 1ST TONE
              02940 ;
71F2 6400     02950 PWRTAB   DEFW    100            ;3 POWERS OF 10
71F4 0A00     02960          DEFW    10
71F6 0100     02970          DEFW    1
              02980 ;
              02990 ;**** Tone List ****
71F8 96       03000 TONTOP   DEFB    150D           ;DURATION OF 1ST NOTE
71F9 D6       03010          DEFB    214D           ;FREQ. OF 1ST NOTE
```

Listing continued

Listing continued

```
71FA C8       03020          DEFB    200D            ;DURATION OF 2ND NOTE
71FB A1       03030          DEFB    161D            ;FREQ. OF 2ND NOTE
71FC A9       03040          DEFB    169D            ;DURATION OF 3RD
71FD BF       03050          DEFB    191D            ;FREQ. OF 3RD
71FE 7E       03060          DEFB    126D            ;DURATION OF 4TH
71FF FF       03070          DEFB    255D            ;FREQ. OF 4TH
7200 0000     03080          DEFW    0000H           ;MARK END OF LIST
              03090 ;
              03100 ;**** Display Messages ****
7202 0F       03110 TMSG     DEFB    0FH             ;TURN OFF CURSOR
7203 0D       03120          DEFB    0DH             ;CARRIAGE RETURN
7204 DA       03130          DEFB    0DAH            ;TAB 26 CHARACTERS
7205 2A       03140          DEFM    '* C R A M *'
7210 0D       03150          DEFB    0DH             ;CARRIAGE RETURN
7211 DB       03160          DEFB    0DBH            ;TAB 27 CHARACTERS
7212 56       03170          DEFM    'Version 2'
721B 0D       03180          DEFB    0DH             ;CARRIAGE RETURN
721C D9       03190          DEFB    0D9H            ;TAB 25 CHARACTERS
721D 3C       03200          DEFM    '<Press ENTER>'
722A 1E       03210          DEFB    1EH             ;ERASE TO END OF LINE
722B 00       03220          DEFB    0               ;MARK END OF MESSAGE
              03230 ;
722C 53       03240 DIFMSG   DEFM    'Select Difficulty Level'
7243 0D0D     03250          DEFW    0D0DH           ;2 CARRIAGE RETURNS
7245 20       03260          DEFM    ' 0  (VERY Difficult)'
7259 0D       03270          DEFB    0DH             ;CARRIAGE RETURN
725A 20       03280          DEFM    '   TO'
725F 0D       03290          DEFB    0DH             ;CARRIAGE RETURN
7260 20       03300          DEFM    ' 9  (VERY Easy)'
726F 0D0D     03310          DEFW    0D0DH           ;2 CARRIAGE RETURNS
7271 3F       03320          DEFB    '?'
7272 5F       03330          DEFB    5FH             ;UNDERLINE CHARACTER
7273 00       03340          DEFB    00H             ;MARK END OF MESSAGE
              03350 ;
7274 55       03360 GMSG     DEFM    'USE ARROW KEYS TO MOVE'
728A 00       03370          DEFB    0               ;MARK END OF MESSAGE
              03380 ;
7000          03390          END     START           ;END OF PROGRAM
00000 TOTAL ERRORS
24953  TEXT AREA BYTES LEFT
ASCORE 71EC 02910   02430 02490
CHKEND 717A 02260   00640
CLS    01C9 00220   00380 01500
CRSPOS 71E7 02880   01850 02070 02230 02260 02310
DELAY  0060 00240   00450 00760 02730
DIFBUF 71E6 02870   01610 02690
DIFMSG 722C 03240   01510
DISPLY 7181 02310   00550 00660
END    7066 00720   00340
END10  7068 00730   00780
FL10   7094 00950   01090
FL20   70A0 01050   00970 00990
FLASH  7091 00940   00420 00740
FPO10  7167 02140   02120
FPO20  716C 02170   02150
FPO30  7171 02200   02180
FPO40  7176 02230   02210
FPOSN  7157 02070   00630
FR10   7117 01680   01700
FR20   7120 01730   01770
FR30   7128 01790   01810
FRAME  7110 01650   00530 00590
GAME   7004 00310   00350
GAMMSG 713B 01910   00560
GD10   70F8 01530   01550 01570
GETARR 714D 02010   00620 02740
```

Listing continued

Pause routine puts a time interval between moves. Without the pause, the game would be impossibly fast. Pause is similar to Delay, but alters the amount of time between moves depending on the difficulty level you select.

After Pause, a relative jump in line 700 continues the game. Unless you use the conditional return in RET (line 650), to exit the Play driver, the game continues until CHKEND reports a collision.

The End Routine

The shortest sub-driver is END (line 720). The program loads the register with 0A hex (decimal 10), the number of times the routine repeats the END10 loop. Since the program also uses the B register as a delay counter, it stores the B register on the stack with the PUSH BC command. Then the routine calls Flash, which flips the graphics on the screen when a collision occurs.

Line 750 loads BC with 1800 hex, and uses the Delay routine to slow things down. The program then recovers the original value in B from the stack with a POP BC command and a DJNZ loop (which you learned to use last month) and sends control back to END10 until B equals zero. At that point, control returns to the main program driver.

The END10 loop demonstrates an important programming technique: Loops are easiest to program using the DJNZ instruction, which decrements the B register and jumps unless B is zero.

You can also use the B register for something else inside the loop (both Flash and Delay use the B register here). To resolve the conflict, save BC on the

HOWE SOFTWARE PRODUCT GUIDE

SYSTEM DIAGNOSTIC

Is your computer working correctly? *Are you sure?* Find out for certain with System Diagnostic, which has complete diagnostic tests for every component of your TRS-80™ Model 1, 3 or 4. Separate tests for:

- RAM: three separate tests incl. every location and data value.
- Video Display: character generator, video RAM, and video signal.
- Disk Drives: disk controller, drive select, track seek, read sectors, formatting, read/write/verify data with or without erasing, disk drive timer, disk head cleaner. Single or double density, any number of tracks.

- Line Printer: character tests, adjustable platen length.
- ROM: checksum tests.
- Keyboard: all key contacts tested.
- Cassette Recorder: read, write, verify data.
- RS-232-C Interface: connector fault, data transmission, framing, data loop,baud rate generator.

Individual tests or continuous testing mode, reports optionally written to line printer.

System Diagnostic $99.95

TYPITALL

The SCRIPSIT™ Compatible Word Processor

TYPITALL is a word processor which is upward compatible with SCRIPSIT™ for the TRS-80™ Models 1, 3 and 4. TYPITALL includes many powerful features, too numerous to list here. Following are some of them:

- Assign any sequence of keystrokes to a single control key.
- Print text on the screen before going to the printer.
- Send the formatted text to disk file for later printing.
- Merge data from a disk file during printing.
- Send **any** control or graphic character to the printer.
- Use the same version on the Model 1, 3 or 4.
- Reenter the program with all text intact if you accidentally exit without saving the text.
- Read your old SCRIPSIT™ files.

"If you like SCRIPSIT™, you'll probably love TYPITALL." **80 Micro.**

TYPITALL (disk only) $129.95

SMALL BUSINESS MANAGEMENT SYSTEM

An integrated package that allows you to use your computer to manage a small business.

- Order Entry: enter orders, print invoices, and recall and change invoices from past orders. Use preprinted forms or plain paper.
- Bookkeeping: complete general ledger by user-defined categories.
- Accounts Receivable: interacts with order entry by invoice number.
- Inventory: complete summary of products sold by part number and price invoiced.
- Installation package tailors the programs to your business.

Requires minimum 2 floppy disk drives or hard disk, 80-column printer. Available for: TRS-80™ Models 1/3/4, 2/12/16, IBM/PC, Kaypro II, and many other CP/M computers.

All versions $250.00

HOME BUDGET & CHECKBOOK ANALYST

Analyzes your income and expenses, maintains checkbook register, computes monthly and year to date summaries and projections.

Model 1/3/4 disk version $59.95
Model 1/3/4 cassette version .. $29.95
Mod. 2/12/16 or IBM/PC version $69.95

TRS-80™ MODEL III ASSEMBLY LANGUAGE

A complete course in assembly language, written for the **beginner.** Contents include:

- The Z-80 instruction set
- TRS-80™ Model III ROM and RAM
- Using the Editor/Assembler
- Reading, printing, and moving data
- Arithmetic operations with integers
- Floating-point and BCD numbers
- Logical and bit operations
- Cassette input and output
- USR subroutines in BASIC
- RS-232-C data communications
- Disk input and output
- The TRSDOS 1.3 disk operating system

With the book you can also purchase **Monitor #5,** a comprehensive machine language monitor and debugging program, which allows disassembling and single stepping through machine operations.

Book only $16.95
Monitor #5 only $22.95
Book and Monitor #5 on disk ... $29.95

SMART TERMINAL

The *intelligent* communications program. Allows you to use your computer as a terminal to information systems, time-sharing computers, or for data transmission.

- Memory buffer holds data to be transmitted or received.
- Automatic transmission of data from buffer
- Automatic storage of incoming data in buffer
- Character translations
- Data files compatible with Electric Pencil™ and SCRIPSIT™
- True BREAK key
- Create "personalized" back-up copies
- Same program supports both cassette and disk systems.

Model 1, 3 or 4 version $74.95
Model 2/12/16 (CP/M) version . .$79.95

SMALL BUSINESS ACCOUNTING

Based on Dome Bookkeeping Record #612, this program keeps track of income, expenses, and payroll for a small business (payroll not included in cassette version). Computes monthly, through last month, and year to date summaries.

Model 1/3/4 disk version $59.95
Model 1/3/4 cassette version .. $29.95
Mod. 2/12/16 or IBM/PC version $69.95

Howe Software
14 Lexington Rd., New City, NY 10956
*TRS-80 is a trademark of Tandy Corp

175

24-Hour TOLL-FREE Order Number:
Outside California call:
(800) 428-7825, ext. 169
Inside California call:
(800) 428-7824, ext. 169
Visa, Master Card, or COD orders only.

For information call
(914) 634-1821

Add $3.00 postage & handling
New York residents add sales tax

VOLCANO HUNTER©

over 200 screens

The dreaded Druts have robbed the city of its precious fuel supply.
YOUR MISSION:

Infiltrate the volcano and retrieve the stolen fuel, while killing the Drut monsters who use the fuel to mine the gold in the volcano. Explore the mine caverns and grab up as many fuel cannisters and as much gold as possible. Avoid the heat from the volcano's lava and leap across the treacherous gaps along the way. Equipped with bombs and your own wit, out maneuver the murderous Druts and return the fuel and gold to the city.

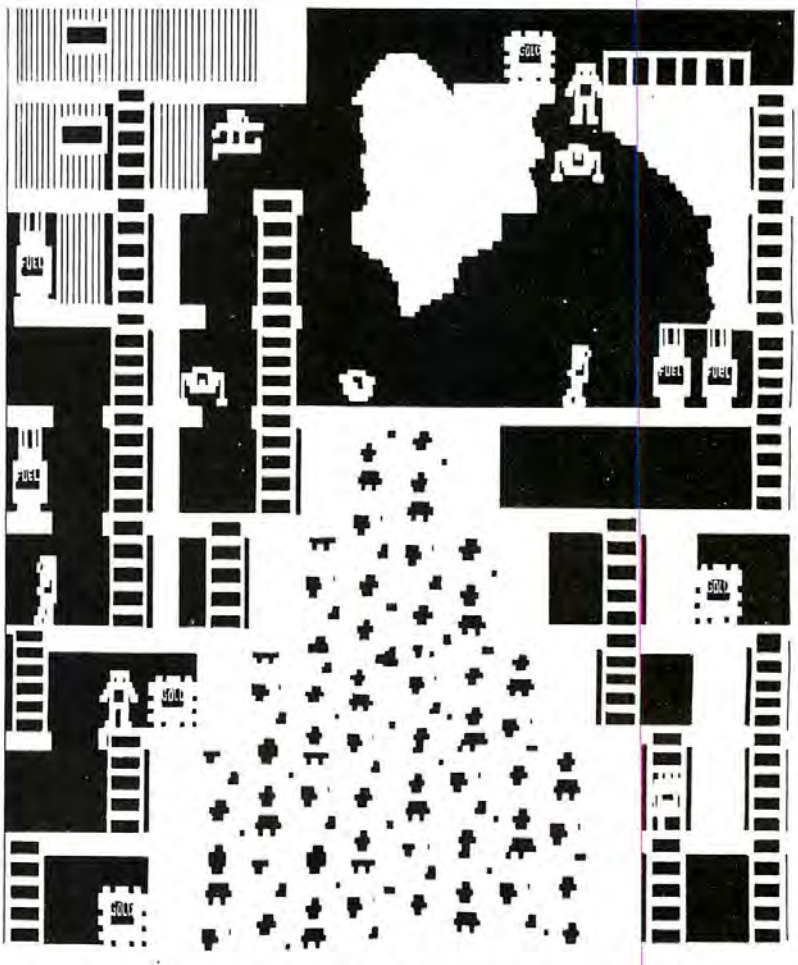
- 100% Machine Language
- Real Time Arcade Adventure
- Sound Effects
- For Models 1 and 3
- Top Ten High Scores
- Original Game
- Joystick Compatible
- Pause Feature
- Over 200 Screens in Memory
- Disk Version Has Added Features

ORDERING INFORMATION:
For TRS-80 Level 2 Model I and III

PRICES:
16K Tape $19.95
32K Disk $22.95

PLUS:
Shipping and Handling $2.00
Please specify model and tape or disk.

Lap Video Entertainment ↩130
Post Office Box 1736
Mississippi State University, MS 39762



The picture above shows 6 of Volcano Hunter's 200+ screens.

stack at the beginning of the loop and recover it just before DJNZ; the solution seems obvious, but has stumped more than one novice programmer.

The Main Subroutines

Lines 800 to the end contain third-level routines that do the real work of the program. Most of these routines are complete in themselves, although a few call fourth-level routines.

Since each routine handles a single function, most are simple to program and understand. Only two of them, Flash and Xlate, contain code that is at all difficult.

The first subroutine, GRPHCL, fills the screen with graphics spaces, CHR$(128). Line 810 loads HL with the address of the beginning of the display, and loads DE with the address of the second position. Line 830 loads BC with the number of positions on the display minus one. Line 840 loads 80 hex (128 decimal) into the first position.

The next instruction, LDIR, runs the routine by moving blocks of data from one location to another in memory. Before you use LDIR, HL must point to the beginning of the source block, DE must point to the beginning of the destination area, and BC must contain the number of bytes to move.

When the Z80 executes an LDIR instruction, it moves a byte from the HL location to the DE location, decrements BC, and increments HL and DE. Then it repeats the entire process until BC equals zero.

This instruction and the similar LDDR instruction (which acts the same but decrements HL and DE after the transfer of each byte) are two of the most powerful instructions in the Z80 instruction set. They perform in a single instruction what would take a long loop of instructions on other processors.

GRPHCL, however, doesn't move a block of bytes from one location to another; it uses LDIR to fill a block of memory with a single byte, demonstrating a second popular use of this instruction. When LDIR first operates, it moves the 80 hex byte in the first screen location (pointed to in HL) into the byte that DE points to, in the second screen location.

The program increments HL and DE so HL points to the second location and DE points to the third. The program decrements BC, and since it still doesn't equal zero, the process repeats. The program moves the byte from the second screen location to the third, incrementing HL and DE and decrementing BC. The Z80 still performs a block move, but the destination block is only 1 byte higher in memory than the origin block.

Because of the order of events, the program places the required byte in each location just in time to be copied to

Listing continued

```
GETDIF 70EF 01500   00490
GETENT 70DC 01370   00460
GETKEY 7148 01970   00570
GMSG   7274 03360   01930
GRPHCL 7076 00810   00390 00520
KEYIN  0049 00250   01530
MAK10  70C5 01240   01330
MAK20  70CC 01270   01270
MAK30  70D5 01310   01310
MAKTON 70C1 01220   01160
NOKEY  70E2 01410   00480 01620
NOKY10 70E5 01430   01450 01460
PAUS10 71CF 02710   02760
PAUSE  71CB 02690   00690
PLAY   704E 00620   00330 00700
PRINT  0033 00230   02820
PRMSG  71DD 02790   00910 01520 01940 02440 02840
PTITLE 7084 00880   00400
PWRTAB 71F2 02950   02480
SAVARR 71EB 02900   02040 02090
SCORE  7187 02350   00670
SCRBUF 71E9 02890   01870 01880 02350 02370 02470
SET10  7045 00570   00580
SETCRS 712D 01840   00540
SETUP  7036 00520   00320
SHWSCR 718F 02400   00410 00680
START  7000 00280   00280 03390
TITL10 701E 00420   00470
TITLE  7012 00380   00310
TMSG   7202 03110   00900
TONE   70A7 01110   00430
TONE10 70B5 01160   01140
TONPTR 71F0 02930   01110 01190
TONTOP 71F8 03000   01150 02930
VIDEO  3C00 00210   00810 00820 00880 00940 01650 01840 02410
XLA10  71AA 02500   02670
XLA20  71B1 02530   02570
XLA30  71B9 02580   02550
XLATE  719F 02470   02400
```

the next one. When the entire process is finished, HL points to the last screen location, DE points to the first location past the screen, and BC equals zero. The screen is full of graphics spaces and the subroutine is finished. You need only five instructions to fill the screen with a given byte.

The second subroutine, PTITLE, prints the title messages on the screen. It loads HL with the address of the first character to print, and loads that address into location 4020 hex.

A ROM routine handles actual screen printing by keeping track of the current print position. It does so by saving the print position at memory location 4020 hex. Loading a screen address into 4020 hex is comparable to using a PRINT@ statement in Basic, because it sets the next print location for the ROM Print routine.

PTITLE next loads HL with the address of the message to print (all print messages are at the end of the program) and then calls the PRMSG subroutine, which sends the characters to the ROM Print routine.

You can save 1 byte of instructions by JP PRMSG instead of Call PRMSG, because the RET at the end of the PRMSG routine sends control back to line 400, which calls PTITLE. However, such a jump weakens the structure of the general program (albeit not by much in this circumstance) by destroying the general logic of each routine calling other routines.

The Flash Subroutine

Though it is only 16 instructions long, the Flash subroutine is the most complex in the entire program and introduces four new Z80 instructions. It's a good general-purpose screen flash, and has many other uses (for examples see "The Next Step," July 1983, p. 24).

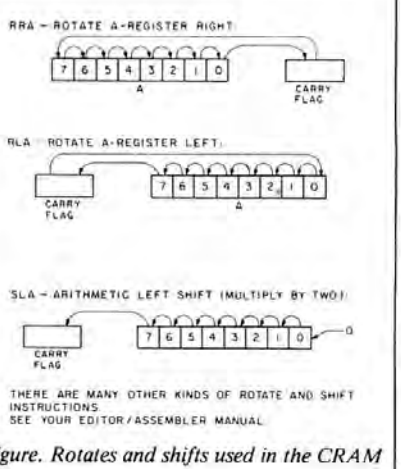


Figure. Rotates and shifts used in the CRAM 2 game program.

The first two instructions, in lines 940 and 950, are easy to understand. The program loads HL with the first screen address, and then places the character in that position in the A register.

The purpose of Flash is to change the state of every graphics block on the screen to its exact opposite. Every pixel that was originally on is turned off and vice versa. Any screen location without a graphics character remains unchanged.

Graphics characters range from CHR$(128) to CHR$(191). By representing each graphics value in binary, the characters are all put in the form 10XXXXXX, where the X's represent individual pixels in each character block.

Flash first tests each screen character to see if the 2 leftmost bits are 10 in binary. When it finds a graphics byte, it changes each of the last 6 bits to its opposite and checks that the 2 leftmost bits are still 10. To do so, it uses a series of Rotate instructions (see the Figure). There are many other kinds of Rotate and Shift instructions in your editor/assembler manual.

Line 960 rotates the A register containing the character from the screen to the left. If the program has not set the carry flag, the first bit of the character cannot be a 1. Control then passes to FL20 to get the next character.

When the first bit is a 1, the program performs another rotate to test the second bit of the character. If this bit is also a 1, the present value represents one of the Model III's special characters, and control again passes to FL20. If it is zero, the program has found a graphics character.

Line 1000 performs a CPL instruction to reverse every bit in the A register. This turns all zeros to 1's and all 1's to zeros to complement the original bits zero through 5.

Next, the program restores the first 2 bits (bits 6 and 7). A right-rotate restores bit 6 in line 1010. Then SCF (set carry flag) in line 1020 sets the carry flag to 1 before the second RRA command in line 1030 completely restores the value in A to its graphics complement. Finally, line 1040 places the new value on the screen.

The remainder of Flash is simple. It increments HL to point to the next screen location, then loads the value in H into the A register and compares (CP) it with 40 hex. Since the last screen location is 3FFF hex, as soon as H holds 40 hex and HL holds 4000 hex, the routine ends. In that case, the CP command shows a true result by setting the Z flag, and the program peforms the conditional return in line 1080. Otherwise, the relative jump in line 1090 starts the process again.

Making a Noise

The Tone and MAKTON routines create the game's sound. CRAM 2 originally had Sound routines during Setup, Play, and End, as well as during Title. They were too distracting and slowed down program execution, however, so I removed them from all sections except Title.

Line 1110 loads the IX register with the present value in the TONPTR buffer, IX, and then points to the next location in the list of tone frequencies and durations. The program loads the present value of that list into A to see if it equals zero, the last value in the list. If it does, line 1150 points IX back to the beginning of the tone list (TONTOP).

Line 1160 calls the MAKTON routine to produce the actual tone through the cassette port. The program then increments IX twice to point to the next tone's values, and stores it back in TONPTR to provide the next tone in the series.

MAKTON (starting in line 1220) is essentially the same as the Sound routine I presented in my last installment. If you wish to add other sets of tones to the program, you can establish new tone lists and a new TONPTR, and write a subroutine similar to Tone. To have more tones in the series, extend the TONTOP tone list.

The next subroutine, GETENT, is very short. It checks to see if you have pressed the enter key. If so, it resets the zero flag to NZ; if not, it returns with the Z flag set.

To accomplish this, the program first looks at the address in keyboard memory that contains the enter key, 3840 hex, and loads the current value in that location into the A register. The keyboard matrix is simple to read (see the Table for the location of each key).

Since GETENT is only interested in the enter key, it masks out the bits representing all other keys. Notice in the Table that the enter key is wired to bit zero, the only bit the routine considers.

Line 1380 performs the test by ANDing the present value in A with 01 hex. Written in binary, 01 hex is 0000 0001. Because of the way a logical AND works, the result of the AND is 0000 0001B if you pressed the enter key, or 0000 0000B if you didn't.

Also, the program sets the Z flag to show whether or not the result value

COMPUTERS ARE CREATING JOBS FOR NRI-TRAINED PEOPLE.

IF YOU'RE SERIOUS ABOUT MAKING MONEY IN MICROCOMPUTERS, NRI IS SERIOUS ABOUT SHOWING YOU HOW.

The U.S. Department of Labor projects job openings for qualified computer technicians will soon double. International Resource Development, Inc., estimates a 600% increase in these jobs in a decade. And most of these will be new jobs, created by the expanding role of computers.

NEVER HAS THERE BEEN A FASTER-GROWING FIELD OF TECHNOLOGY.

Many people are afraid of losing their jobs to computers, but thousands of jobs will be created for those who are prepared to meet the challenge.

With NRI training, you'll be prepared. You can have a profitable, exciting future as an expert who can handle the operational, programming and technical aspects of all kinds of microcomputers and microprocessors.

Your NRI course will include the new TRS-80 Model 4 with Disk Drive or the TRS-80 Color Computer with NRI Computer Access Card...plus a professional LCD multimeter, NRI Discovery Lab and hundreds of demonstrations and experiments. It's all yours to keep.

LEARN IN YOUR SPARE TIME.

NRI trains you in your own home, at your convenience...no classroom schedules to meet, no need to quit your job. As a class of one with complete course materials and the backing of a staff of professional electronics instructors, you'll get extraordinary hands-on training on the latest, most popular microcomputer: the new TRS-80™ Model 4 with Disk Drive. Designed to perform diverse personal and business functions and accept more software, the TRS-80 is yours to keep.

LEARN HOW TO USE, PROGRAM AND SERVICE STATE-OF-THE-ART MICRO-COMPUTERS.

Through your carefully designed NRI course, you'll get a wealth of practical experience. You'll build circuits... from the simplest to the most advanced...with your NRI Discovery Lab.® You'll use a professional 4-function LCD digital multimeter for analysis and troubleshooting. With NRI training you'll explore your computer's registers, memory and input-output ports. You'll even write programs to control the circuits you've designed and built. You'll perform hundreds of challenging experiments, always backed by a full-time faculty ready to help you personally.

When your NRI training is complete, you'll be a computer technician, ready for your first job — servicing, testing or programming all types of microcomputers — in a rewarding and challenging new career.

THE CATALOG IS FREE. THE TRAINING IS PRICELESS.

Send the coupon today for your FREE 104-page catalog. It's a valuable guide to opportunities and training in the high-tech revolution. You'll see how easily you become part of the growing high-tech world of microcomputers. If the coupon has been removed, write: NRI, 3939 Wisconsin Ave., Washington, DC 20016.

TRS-80 is a trademark of the Radio Shack division of Tandy Corp.

All career courses approved under GI bill. ☐ Check for details.

NRI School of Electronics
McGraw-Hill Continuing Education Center
3939 Wisconsin Avenue
Washington, D.C. 20016

We'll give you tomorrow.
The catalog is free.
The training is priceless.

Please check for one free catalog only.
☐ Industrial Electronics
☐ Computer Electronics including Microcomputers
☐ Color TV, Audio, and Video System Servicing
☐ Electronics Design Technology
☐ Digital Electronics
☐ Communications Electronics • FCC Licenses • Mobile • CB • Aircraft • Marine
☐ Basic Electronics
☐ Small Engine Servicing
☐ Appliance Servicing
☐ Automotive Servicing
☐ Auto Air Conditioning
☐ Air Conditioning, Heating, Refrigeration & Solar Technology
☐ Building Construction

Name (Please Print) Age

Street

City/State/Zip

Accredited by the Accrediting Commission of the National Home Study Council 174-024

was zero (enter key not pressed). The program ignores all the other keys at 3840 hex by using the AND mask.

NOKEY (line 1410) saves the BC register pair on the stack so the routine does not destroy the current value in BC. It then loads B with 0FF hex so the NOKY10 loop repeats 255 times. At NOKY10, the program loads the A register with the value at 3BFF hex. That value is zero only if you aren't pressing any keys. If you press a key, the program loops back to NOKY10 until you release that key. Once you release it, the DJNZ loop takes over and repeats NOKY10 255 times before it retrieves the original BC value and returns.

NOKY10 repeats 255 times (you can use a smaller number) to avoid keybounce. When you press or release a key, it makes and loses contact several times before settling down. Keybounce was so bad in the Model I that Radio Shack gave out a special software program to overcome it. Similar debounce routines are part of the ROMs in later Model I's and Model IIIs.

When your program reads the keyboard directly, it must deal with occasional bounces without frustrating the operator. Also, some people create more keybounce than others by the way they press and release keys—what works satisfactorily for you may only frustrate someone else using your program.

Address	Bits							
	7	6	5	4	3	2	1	0
3801 hex	G	F	E	D	C	B	A	@
3802 hex	O	N	M	L	K	J	I	H
3804 hex	W	V	U	T	S	R	Q	P
3808 hex						Z	Y	X
3810 hex	7	6	5	4	3	2	1	0
3820 hex	/	.	—	,	;	:	9	8
3840 hex	SPACE	RT ARW	LT ARW	DN ARW	UP ARW	BREAK	CLEAR	ENTER
3880 hex							*	SHIFT*

*Note: On the Model III, bit zero is the left shift key and bit 1 is the right shift key. On the Model I, bit zero is both shift keys.

Table. Organization of the keyboard matrix.

Getting Into Difficulties

Starting on line 1500, the next subroutine requests and inputs a difficulty level. It uses the ROM CLS routine to clear the screen, and PRMSG to print a message that asks for a number between zero and nine.

Line 1530 uses the ROM KEYIN routine to wait for and report your choice. The program then does some error checking to make sure the player entered a legal value.

Line 1540 compares the input key with zero (30 hex). The Compare instruction (CP) sets flags the same way a subtract does: If a borrow occurs, the program sets the carry flag. If the two values are equal, the program sets the zero flag. If the input key has an ASCII value of less than 30 hex (for example, ! has a value of 21 hex), then the CP in line 1540 sets the carry flag, and sends the program back to look for another input key in line 1550.

If the test on line 1540 passes and the ASCII value of the input key is high enough, the program performs another test on line 1560 to be sure the input key is higher than nine (ASCII value of 39 hex).

The comparison on line 1560 sets a carry flag unless the ASCII value of the input key is higher than 39 hex. In that case, the program goes back looking for the proper input. You should test every input in a program to assure values that the program expects and can handle.

After the program makes sure that the input is between zero and nine, it subtracts 2F hex from the key value to generate a number in the range of one to 10 (01 hex to 0A hex). The program then performs two left shifts to multiply the value by four (just as adding a zero at the end of a decimal number is the same as multiplying by 10, adding a zero at the end of a binary number multiplies it by two).

The easiest way to add a zero is to use the Shift instruction. Since the original value has to be between one and 10, the final value must be between four and 40, and an overflow into the carry flag is impossible.

After the multiplications, the program saves the final value in the 1-byte buffer, DIFBUF. Then the NOKEY routine checks to make sure you release

all keys before the program control returns.

The Frame routine, beginning on line 1650, is the same as that in Program Listing 2 from my last installment. It's followed by four short subroutines.

The program sorts the initial game cursor in CRSPOS, and SETCRS sets it to the middle of the screen (line 1850) and fills the score counter, SCRBUF, with zeros. GAMMSG prints the game instructions on the screen by setting the system print cursor in line 1920 and using the PRMSG routine. GETKEY checks to see if you are pressing any key and, by ORing the A register with itself, GETKEY sets the zero flag to NZ when you hold down any key.

GETARR reads the arrow row on the keyboard and masks out all possible non-arrow bits by ANDing the value found at 3840 hex with 78 hex (0111 1000 binary). If the result is zero (no arrows pressed), the program takes a conditional return at line 2030.

If you're holding down at least one arrow, the program stores the value in the A register in the SAVARR buffer before returning in line 2050. If you press no arrow keys, the direction indicator in SAVARR remains unchanged and the game cursor continues moving in the same direction.

Finding a New Position

The FPOSN routine beginning in line 2070 finds the next screen location that the moving cursor sets. First, the program loads the HL register with the present game cursor address as stored in CRSPOS. It loads DE with 40 hex (64 decimal), the distance between any screen location and the location immediately above or below it. The program loads A with the value in SAVARR, indicating which arrow(s) you pressed last.

The OR A instruction in line 2100 does not set the zero flag according to the value in A. Instead, it automatically resets the carry flag so the possible SBC (subtract with carry) in line 2130 provides a true result. Starting with line 2110, the program tests each of the possible bits representing an arrow key.

The Z80 supports three instructions affecting individual bits: Set, RES, and Bit. Set turns on any individual bit, Reset turns any bit off, and Bit tests any bit to find its current status.

In a sense, these three instructions are similar to Basic's Set, Reset, and Point commands, although the Z80 instructions operate on any bit instead of only on pixels on the screen. Each of the tests in the FPOSN routine uses the Bit instruction to see if you pressed a particular arrow key, and the Z flag holds the result of each test.

The first check, BIT 3, is for the up-arrow key (you can see the correspondence between arrows and bits in the Table). When the bit equals zero, control passes to FPO10. In any other case, the program subtracts the 40 hex value in DE from the current game cursor position in HL, and the new position is one line higher.

Then the program tests the down-arrow key with a BIT 4,A instruction and adds 40 hex to the current game cursor if you have pressed the down-arrow key. In a similar manner, the program tests the left- and right-arrow keys and moves the cursor accordingly. Finally, line 2230 stores the new cursor value in CRSPOS and the subroutine ends.

The next four routines are also short. CHKEND takes the value in CRSPOS and checks if the screen location is still a graphics blank. When it is, the program sets the Z flag with the CP instruction. In either case, control immediately returns to the play driver.

DISPLY loads a solid graphics block into the next cursor location before returning to Play, and Score gets the current score from SCRBUF, increments it by one, and stores it back in the buffer.

SHWCR calls XLATE to translate the current binary score in ASCII decimal, then sets up the system print cursor to display the score in the middle of the top display line. Finally, it uses the PRMSG subroutine to print the current score on the screen.

Translating Numbers

XLATE is the last complex routine in the program, and demonstrates an easy way to change from binary to another number base. It translates the current binary value in SCRBUF into an ASCII string so the program can display the score on the screen. Scores range from zero to 867, so the program provides three print positions in the ASCORE storage buffer and on the screen.

XLATE loads the HL register pair with the current binary score in line 2470, then loads IY with the address of a table of powers of 10, and IX with the address of the ASCORE buffer.

Line 2500 loads the DE register pair with the current power of 10. The Z80 stores all 2-byte values in the order of least significant byte followed by most

✓ See List of Advertisers on Page 211

significant byte (LSB/MSB order). Therefore, the program must load the E register containing the LSB from the lower of the two memory addresses that hold the power of 10.

Starting in line 2540, the program repeatedly subtracts the current power of 10 from HL until HL drops below zero (indicated by the carry flag). The A register counts the number of successful subtractions, incrementing by one each time.

When HL finally drops below zero, the program jumps to XLA30 to restore the last positive value in HL. At this time, the A register counter holds a value between 00 hex and 09 hex. Adding 30 hex to that value changes it into the value of an ASCII character between zero and nine. The program loads the resulting character into the ASCORE buffer, and then ends in 2610 and 2620, by testing the current power of 10 in DE to see if it is one.

If XLATE is not completed, the program increments the IX register by one to point to the next position in the ASCORE buffer (line 2640). Then it increments the IY register twice to point to the next power of 10, since the computer stores each number in 2 successive bytes of memory. In line 2670, a relative jump starts the whole process over again.

The XLATE routine has many uses. If there are more than 999 possible values to translate, more values have to be added to the power table, but the routine is the same. In my program, XLATE only works on values that the program can store in 2 bytes, so its upper limit is 0FFFF hex or 65535 decimal.

The Pause routine begins in line 2960 and sets the time between cursor moves. First, the program loads the stored value in DIFBUF (04 hex to 28 hex) into the B register to act as a counter for a DJNZ loop. It then saves the B value on the stack by PUSHing BC, just as in the End routine.

Each loop performs two operations. Line 2720 loads BC with the small value of 100 hex for a call to the ROM Delay routine. If you want to change the range of difficulty in the program, change that value. Next, the program calls the GETARR routine to check for keyboard responses.

In testing the program, I found that it often missed short taps on the arrow keys. My solution was to read the keyboard more often by including the arrow scan as part of the Pause routine. Now it's impossible to tap a key so quickly that the program misses it.

The last routine of the program is PRMSG on line 2790. This routine assumes that HL points to a printable message, and that the message terminates with a byte of 00 hex. It first reads a character into the A register and checks it. If it is 00 hex, the routine ends. Otherwise, the routine sends the character to the Print ROM routine to increment the pointer and repeat the process.

The ROM routine at 2B75 hex does the same thing as PRMSG. Unfortunately, it doesn't work unless you initialize Basic. A tape-based TRS-80 always initializes Basic during power-up and reset, so it uses the 2B75 hex routine in Assembly-language programs written without disks.

A disk system doesn't initialize Basic until it loads Disk Basic. It then returns to DOS Ready, and the 2B75 hex ROM routine (as well as several others) works.

If you copy an Assembly-language program from a magazine and find it doesn't work on a disk-based machine, try loading Disk Basic, returning to DOS, and running the program again. If it works, either replace any calls to 2B75 hex with PRMSG, or load Disk Basic before each use of the program.

Buffers, Tables, and Messages

Most Assembly-language programmers define buffers, tables, and messages at the end of a program (or program section) for several reasons. First, when grouped together, these non-instructions are easier to find with a monitor during debugging and they don't interfere with the flow of the program. Finally, messages are usually not worked out until debugging is complete.

CRAM 2 defines six short buffers, or storage areas, beginning in line 2870. Some have a specific value at the beginning of the program; others only need a certain amount of memory space set aside.

DIFBUF (line 2870) is a 1-byte buffer that holds the selected difficulty value. Since it doesn't hold any specific value, the program sets aside space with a DEFS (define space) pseudo-op.

CRSPOS, which holds the location of the game cursor, also doesn't need definition, and CRAM 2 allocates 2 bytes to it with the DEFS pseudo-op. The assembler sets no values for these two buffers: Once the game starts, they contain the values previously held in their memory locations.

SCRBUF is a 2-byte buffer that holds the current score in binary form. Since the initial score in the game is zero, the program defines this buffer as a word with a value of 0000 hex. The program holds the value from the last scan of the arrow keys in the 1-byte buffer, SAVARR. It must initialize this buffer with some possible value for the arrow keys: I chose to use 40 hex, the value that the program would read if you pressed the left-arrow key.

ASCORE contains the ASCII representation of the player's score. This includes the 3 bytes necessary to hold the maximum score of 867, plus a final byte of 00 hex to mark the end of the ASCII message. By initializing ASCORE to 4 bytes (two words) of 00 hex, the program displays no score the first time the title screen appears.

TONPTR holds the address of the next tone values for the MAKTON routine. CRAM 2 initializes the one-word buffer to the beginning of the tone list so the system can find the correct location of the tone table.

PWRTAB is a list of powers of 10 for the XLATE routine. The program loads each power into DE for processing, so each one must be one word long. The tone list in line 3000 holds both the frequency and duration values of each tone (see the last installment in this series for the method to calculate these values).

Because the Print routine handles control codes as well as ASCII values, I liberally sprinkled individual bytes and ASCII strings in the first two messages. TMSG (line 3110) is the next line printed on the screen. It starts with a 0F hex byte that turns off the system cursor and a 0D hex byte that sets the cursor to the beginning of the next line and clears the line. CRAM 2 uses the TRS-80's space compression codes for tab values: 0C1 hex provides one space and 0FF hex creates 255 spaces. The program centers each line of the title message by calculating the number of spaces to tab before it on the screen.

After the last character of TMSG, the program uses a control byte of 1E hex to erase to the end of the line. Remember, CRAM 2 fills the entire screen with graphics spaces before printing the title message.

The program converts each graphics space to a full graphics block with the Flash routine. In order to leave the middle screen lines clear for the title, the program fills them with ASCII characters and spaces that are unaffected by Flash.

The second message, DIFMSG, is more straightforward. ASCII strings (defined with the DEFM pseudo-op) and carriage returns make up most of

Computer Tutor™

VIDEO INSTRUCTION TAPES

STEP BY STEP INSTRUCTIONS

VHS or BETA FORMAT

PICTURES ARE WORTH THOUSANDS OF WORDS AND SAVE HOURS OF FRUSTRATION

Use your VCR side by side with your computer to learn disk operating systems, how to program, and how to use programs. Your VCR along with your computer serve as your personal tutor. Pause your VCR to review and learn at your own pace.

SPECIAL OFFER

Lazy Writer word processor and video instruction tape complete with documentation and registration card for model 1, 3 or the model 4, (the model 4 has 80 character display).

ONLY $119.00

VERBATIM DATALIFE

Single side double density 10 in a hard plastic storage box.

ONLY $25.00

C.O.D. Orders Add $3.00
$3.00 Shipping Per Order
Illinois Residents Add 6% Sales Tax

Specify either VHS or BETA Tape Format

TAPES FOR THE TRS 80

CAT #	TOPIC	APPROX.	PRICE
DOS-1	TRSDOS 1.3 MOD 3	1 HR 30 MIN	**$39.95**
DOS-6	TRSDOS 6.0 MOD 4	1 HR 45 MIN	**$39.95**
DOS-3	NEWDOS 80 2.0	1 HR 45 MIN	**$39.95**
DOS-2	DOSPLUS 3.5	1 HR 45 MIN	**$39.95**
DOS-5	LDOS	1 HR 45 MIN	**$39.95**
DOS-7	CPM 2.2 MOD 4 MONTEZUMA	1 HR 45 MIN	**$39.95**
DOS-4	MULTIDOS	1 HR 45 MIN	**$39.95**
EW-1	VISICALC (TRS80)	1 HR 30 MIN	**$39.95**
WP-1	LAZY WRITER	1 HR 45 MIN	**$39.95**
WP-2	SUPERSCRIPSIT	1 HR 45 MIN	**$39.95**
WP-3	SCRIPSIT	1 HR 30 MIN	**$39.95**
WP-4	NEWSCRIPT	1 HR 45 MIN	**$39.95**
DB-1	PROFILE III PLUS	1 HR 45 MIN	**$39.95**
DB-2	PFS:FILE	1 HR 30 MIN	**$39.95**
DIO-3	TRSDOS 1.3 DISK I/O	1 HR 45 MIN	**$39.95**
DIO-4	TRSDOS 6.0 DISK I/O	1 HR 45 MIN	**$39.95**
BP-1	LEARNING MOD 3, 4 BASIC	3 HR	**$49.95**
BP-2	LEARNING MOD 1 BASIC	3 HR	**$49.95**
TC-1	MTERM	45 MIN	**$29.95**
UT-1	DOTWRITER	1 HR 15 MIN	**$29.95**

TAPE FOR THE TRS80 COLOR COMPUTER

BP-5	EXTENDED COLOR BASIC	3 HR	**$49.95**

TAPES FOR THE COMMODORE 64 AND VIC 20

BP-3	LEARNING C-64 BASIC	2 HR	**$49.95**
BP-4	LEARNING VIC-20 BASIC	2 HR	**$49.95**
DIO-1	COMMODORE 64 DISK I/O	1 HR 45 MIN	**$49.95**
DIO-2	VIC 20 DISK I/O	1 HR 45 MIN	**$49.95**

TAPES FOR THE IBM PC

EW-2	VISICALC (IBM PC)	1 HR 30 MIN	**$39.95**
TC-2	MTERM (IBM PC)	45 MIN	**$29.95**

LYNN COMPUTER SERVICE

115

6831 West 157th Street Tinley Park, Illinois 60477 (312) 429-1915

master charge BANKAMERICARD VISA

VISICALC IS A TRADEMARK OF VISICORP INC. LAZY WRITER IS A TRADEMARK OF ALPHA BIT COMMUNICATIONS. SCRIPSIT, SUPERSCRIPT, PROFILE III PLUS AND TRSDOS ARE TRADEMARKS OF TANDY CORP. LDOS IS A TRADEMARK OF LOGICAL SYSTEMS INC. NEWDOS IS A TRADEMARK OF APPARAT INC. DOS PLUS AND MTERM ARE TRADEMARKS OF MICRO SYSTEMS SOFTWARE INC. MULTIDOS IS A TRADEMARK OF COSMOPOLITIAN INC. VIC 20 AND COMMODORE 64 ARE TRADEMARKS OF COMMODORE BUSINESS MACHINES INC. IBM IS A TRADEMARK OF INTERNATIONAL BUSINESS INC. CP/M IS A TRADEMARK OF DIGITAL EQUIPMENT CORP. NEWSCRIPT IS A TRADEMARK OF PROSOFT. DOTWRITER IS A TRADEMARK OF RCM COMPUTERS. TM INDICATES A TRADEMARK OF LYNN COMPUTER SERVICE.

See List of Advertisers on Page 211

the message. However, on a separate line at the end of the message, the program uses a two-character prompt so the player knows he is to make a keyboard entry. GMSG displays the complete instructions for the game, and is the simplest routine of all: one ASCII string plus a terminating byte of 00 hex.

Other Assembler and Debug Techniques

When you look at output from the assembler, either on the screen or on a printout, you see more than the source code you typed in. The computer adds two new columns on the left: one with assembly addresses and one with the hex code stored at those addresses.

A printed copy of this part of the assembler output is valuable for debugging because it lets you verify the instructions in memory as you single-step through a program. Also, the assembled code helps you catch number-base errors—if you meant ORG 7000H but entered ORG 7000, the assembler shows instructions starting at address 1B58 hex. No matter how hard you try, you can't run a program you have loaded into ROM.

At the end of the Program Listing is a table of all the labels that you use in the program, the address or value associated with each, and the source code lines that use each label. Some assemblers alphabetize the label listing and some list the labels in the order they were defined.

This table can also assist you during debugging. You might, for example, have both a LOOP1 and a LOOP10 in your program. You know that your program uses LOOP1 only twice, but it refers to LOOP10 several times. If the label reference table shows several references to LOOP1, you know you have made a typing mistake somewhere in your program.

Almost all the ways to debug a program involve some type of monitor program that lets you view and change memory and lets you single-step through your program. If you have structured your program like CRAM 2, try executing each of the calls in the main driver in full. After you find which calls don't work, you can move down into the layers of the program to examine the sub-drivers and the subroutines.

When debugging a straight-line program, set a breakpoint near the middle of the program. If the bug occurs before that point, reset the breakpoint halfway to the beginning. If one occurs after the breakpoint, move it halfway to the end. Using this type of binary search, you will quickly locate all bugs.

You can often use the monitor to change an instruction in memory and fix a bug. Keep notes during your debugging process, because eventually you'll want to return to the source code, fix your errors, and reassemble a perfected version of the object code.

Don't get discouraged by program bugs; the debugging and recoding process takes almost as long as the original coding. With practice, your debugging skills and your intuitions about where the bugs are actually hidden will improve. ■

Readers who subscribe to CompuServe are invited to take part in open discussions of the topics covered. GO PCS-117 to the Software and Authors Special Interest Group (SASIG) and leave your questions or comments addressed to me on the message board.

You can reach Hardin Brothers at 280 N. Campus Ave., Upland, CA 91786. Please include a stamped, self-addressed envelope for a reply.

SOFTWARE PRICE WAR

$19.95

Not cheap programs but complete software systems. I believe most software is pirated because it is overpriced and I'm taking this gamble to prove it.

HELP STAMP OUT SOFTWARE PIRATES

I advertised these Systems in **80MICRO** for **$79.00**. I rather sell 1000 systems at **$19.95** than 100 at **$79.00.** These are complete software systems supplied to you on a **TRSDOS 1.3** system diskette (I paid Tandy Corp. a royalty fee for this privilege). All systems fully menu driven — user friendly just insert diskettes, reset and go. Systems are supplied with full documentation (although most users report that they don't read it).

LYNN'S PAYROLL SYSTEM

Complete record keeping & payroll calculation. To include your state's unemployment tax & forms, detailed 941 form, prints paychecks & W-2's.

LYNN'S ACCOUNTS RECEIVABLE SYSTEM

Single entry — open invoice system. Prints invoice and statements. Built in aging report.

LYNN'S CHECK REGISTER SYSTEM

Prints checks. Register for any month, reconciliation statement.

LYNN'S TRSDOS UTILITY PROGRAMS

For TRSDOS 1.3 includes short directory (file names only), cursor directory (allows cursor access to CMD, BASIC, and "DO" Files), and a 2K Print Spooler. Menu installs utilities on existing TRSDOS Diskettes.

LYNN'S E-Z MAIL SYSTEM

Sort by name, zip code, state. 2,500 names. E-Z edit.

All systems are written for Model III or Model IV in Model III Mode.

AND NOW THE BEST PART

ANY SYSTEM ONLY $19.95 EACH.

Plus $3.00 shipping per order. Illinois residents add 5¼% sales tax.

(The **TRSDOS** alone is worth **$14.95**)

All systems phone supported call 815/436-4477 and ask the author JOE LYNN what you want to know. I will assist all original purchasers any way I can.

master charge
THE INTERBANK CARD

Joe Lynn ⌐112

Computer Services
23501 W. Gagne Lane
Plainfield, Illinois 60544
815/436-4477

BANKAMERICARD
VISA

⌐ See List of Advertisers on Page 211

Synthetically Speaking—Part II

by David L. Engelhardt

After building last month's speech board for your Model I/III, program it with applications for education and the visually handicapped.

Last month I showed you how to build your own speech board for a Model I or III ("Synthetically Speaking—Part I," p. 142). Although once an expensive undertaking, adding synthetic speech capability to your computer is now easy and inexpensive due to the Votrax speech chip.

My previous article covered the construction and testing of the speech board. This month I present four programs that let you use the speech board in educational applications and as an aid for the visually handicapped.

Program Listing 1 is a Basic program that converts text to speech on key entry via Program Listings 2 and 3—the Assembly-language routines that make up the Text-to-Speech program. Program Listing 4 converts, through the Text-to-Speech program, strings of text entered through the keyboard into speech.

The programs run on a 16K RAM or larger computer. However, if you own a 16K machine you'll have to get someone with a bigger system to assemble the controlling program due to its large size.

Text-to-Speech Program

Listing 2 consists of the Text-to-Speech program and Listing 3 is the Rules Table that controls the conversion of ASCII text to speech. I separate the two programs due to the size of the Rules Table. The commented source listing of the Text-to-Speech program is 17,153 bytes long and the object code 1,680 bytes long. It is complex due to the many steps required to convert ASCII text to simulated speech.

The program takes each character in an English-language string and scans the appropriate rules one at a time. When it finds a matching rule, it inserts the proper phoneme code into an output buffer, which it later sends to the SC-01 chip.

If it doesn't find a matching rule, the program enters an Error routine, and displays the deciphered portion of the input string along with its phoneme equivalents. The program then outputs an error message to the Votrax chip. This routine gives you a chance to see if you need any additional rules in the table.

The Text-to-Speech program consists of many routines controlled by special characters within the rules. Not only does the program examine a specific character in a word, it also examines what precedes and follows the character. A character's location and use dictate which routine the program calls. The Text-to-Speech program checks for inflection codes, nonalphabetic characters, one or more vowels,

Symbol	Definition
<	Causes jump to the Left-Scan routine.
>	Causes jump to the Right-Scan routine.
!	Causes call to a routine that attempts to match any non-alphabetic character in the input string pointed to by the IX register pair. (Blanks, numbers, etc.)
#	Causes call to a routine that attempts to match one or more vowels. (A,E,I,O,U,Y)
:	Causes call to a routine that attempts to match zero or more consonants. There is always a match. (B,C,D,F,G,H,J,K,L,M,N,P,Q,R,S,T,V,W,X,Z)
"	Causes call to a routine that attempts to match one consonant.
'	Causes call to a routine that attempts to match a voiced consonant. (B,D,G,J,L,M,N,R,V,W,Z)
+	Causes call to a routine that attempts to match front vowels. (E,I,Y)
%	Causes call to a routine that attempts to match a suffix at the end of a word. (E,ER,ES,ED,ING,ELY)
&	Causes call to a routine that attempts to match sibilants. (S,C,G,Z,X,J,CH,SH)
$	Causes call to a routine that attempts to match influencing consonants. (T,S,R,D,L,Z,N,J,TH,CH,SH)

Table 1. Text-to-Speech special symbols.

The Key Box

Models I and III
16K RAM Cassette Basic
32K RAM Disk Basic
Assembly Language
(If 16K—object code only)
Hardware project

```
10 '************************************************
20 '              L I S T I N G  1
30 '************************************************
40 '
50 '       KEY ENTRY PRONOUNCIATION PROGRAM
60 '                      BY
70 '                 DAVID ENGELHARDT
80 '
100 '
110 ' DEPENDING UPON THE OPTION SELECTED, THE PROGRAM WILL
120 ' EITHER PRONOUNCE EACH KEY AS IT IS PRESSED OR NOT
130 ' PRONOUNCE THE SELECTED KEY.
140 ' THE PROGRAM WILL PRONOUNCE EACH WORD THAT IS CREATED
150 ' UPON THE ENTRY OF THE SPACEBAR.
160 ' WHEN THE -ENTER- KEY IS HIT, THE ENTIRE SENTENCE WILL BE
170 ' CREATED TO SPEECH.
180 '
190 '************************************************
200 '
210 CLS:CLEAR 1000
220 DEF USR0=&HE950:CMD"L","TTSPRG/CMD":CMD"L","TABLE/CMD"
230 ON ERROR GOTO 720
240 A$="DO YOU WANT TO REPEAT E CH KEY" :
250 A=VARPTR(A$)
260 VV=USR0(A):
                'SPEAK ENTRY CHOICE MESSAGE
270 A$=INKEY$:IFA$=""GOTO270       'GET CHOICE
280 IFA$="Y" AN$="Y":GOTO 310
290 IFA$="N" AN$="N":GOTO 310
300 GOTO 270
310 C$="YOU HAVE REACHED THE MAXIMUN NUMBER OF ENTRYS.  THIS IS WH
AT YOU HAVE TYPED IN SO FAR" :
320 D$="ENTER PLEASE" :
330 CT=1:                          'BACKSPACE COUNTER
340 A=VARPTR(D$):VV=USR0(A):       'SPEAK ENTER PLEASE MESSAGE
350 A$=INKEY$:IFA$=""GOTO350
360 IFA$=CHR$(13) TS$=TS$+SB$:B$=TS$:GOTO 660  : 'ENTER KEY
370 IF A$<>CHR$(32) GOTO 460  :     'SPACEBAR
380 SB$=SB$+A$:                    'CREATE WORD STRING
390 TS$=TS$+SB$:                   'CREATE TOTAL STRING
400 IFLEN(TS$)>255 B$=TS$:GOTO 620
410 A=VARPTR(SB$)
420 VV=USR0(A):                    'SPEAK CREATED WORD
430 SB$=""
440 CT=0:                          'CLEAR BACKSPACE COUNTER
450 GOTO530
460 IFA$=CHR$(8) AND CT=1 GOTO350  :  'CHECK FOR BACKSPACE
470 IFA$<>CHR$(8) GOTO 510  :      'CHECK FOR MAX BACKSP COUNT
480 CT=CT-2
490 SB$=LEFT$(SB$,CT):             'RE-CREATE BACKSPACED WORD
500 GOTO530
510 SB$=SB$+A$:                    'CREATE WORD STRING
520 IFLEN(SB$)>255 B$=SB$:GOTO 620
530 PRINTA$;
540 IFAN$="N"GOTO590  :            'TEST FOR CHOICE QUESTION
550 ZZ$=TS$:                       'MUST USE TO RETAIN STRING
560 A=VARPTR(A$)
570 VV=USR0(A):                    'SPEAK THE ENTERED KEY
580 TS$=ZZ$:                       'RESTORE SAVED STRING
590 CT=CT+1
600 IFCT<255 GOTO 650  :           'TEST FOR MAX STRING COUNT
610 B$=SB$
620 A=VARPTR(C$)
630 VV=USR0(A):                   'SPEAK MAX STRING COUNT MESS
640 GOTO 660
650 GOTO350
660 PRINT:PRINTB$
670 A=VARPTR(B$):                  'SPEAK TOTAL CREATED STRING
680 VV=USR0(A)
690 B$="":TS$="":SB$="":          'CLEAR VARIABLES FOR NEXT STRG
700 GOTO310
710 '
720 B$="ERROR CONDITION...PLEASE REDUE";
730 RESUME 320
740 END
```

Program Listing 1. Basic application program.

consonants, voiced consonants, front vowels, suffixes, sibilants, influencing consonants, and ASCII characters. Table 1 defines the Text-to-Speech program's special symbols.

Listing 3 includes the rules index, rules pertaining to each letter of the alphabet, and number and special character rules. I also incorporated all basic math operators so you can use simple math functions in education programs.

How It Works

After saving all of the register contents on entry to the Text-to-Speech program, lines 230–360 condition the string input buffer with zeros and the output buffer with Votrax stop codes. This clears out any of the buffers' old string text and sets the stop character (00), which indicates the end of the conversion process. The stop code (3F) clears the output buffer and indicates a finished output condition. Notice that the output buffer is 145 bytes longer than the input buffer. This is because there isn't always a one-to-one relationship of ASCII letters to phoneme codes and, therefore, the phoneme buffer always contains more codes.

Lines 410–430 check for a string to convert, exiting if there is no string. This section then inserts a blank at the first location in the INPBUF buffer for speech conversion alignment purposes. Lines 530–600 convert everything to uppercase, and transfer the string from where it resides in memory to INPBUF.

The next routine, CHTYPE, finds the location in the Rules Table that corresponds to each character in a string. The program then scans and compares for matches. The Rules Table dictates what action the Text-to-Speech program takes. If it can't find a rule location, the program drops down to the Inflection Check routine and tests for the special # symbol. If it doesn't find the symbol, the program branches to the Error routine.

On a rule match, the program jumps to the RULSCN routine. This routine double-checks the HL and IX pointers to ensure that they recognize the same character, but in different locations. The program increments the Rules Table pointer and the routine decides which side of the character to check, if any. If the direction symbols are present, control passes to either the Left or Right Scan routines according to the control symbols < (left) and > (right).

If neither one of the direction symbols is in the rules, this section assumes that an ASCII character exists and checks for a match between the input string and rule character. Match-checking continues until the program accesses any of the special control characters. An equals sign within a rule indicates a match.

At a match the program branches to the TXFR routine that transfers the appropriate phoneme codes to the output buffer (OUTBUF). After this the GNXTCH routine checks for input string code 00. If the program doesn't locate this code, it jumps back to CHTYPE to scan the next character.

The rest of the program routines make specific tests for conditions that surround the character in check, such as vowels, influential consonants, and so on. Some of the routines that scan the left side sometimes check for two characters instead of one. For example, the SIBCON routine makes a test for influencing consonants. To keep a correct alignment, for example, when making CH and SH checks, I put them into the table in reverse order (HC, HS). Since the scan goes to the left, the first character checked is the H and the program checks the second character if the first matches.

The Output routine sends the phoneme codes to the Votrax chip. After the program converts the input string buffer to phonemes, it branches to this routine. Output handles both the string and error message outputs. The only difference between the two is the location to which the HL register pair points. The routine continues to send out phoneme codes until it reaches the stop character 3F. At the stop code, the program restores all registers and returns to the calling program (Basic Listings 1 or 4 in this case).

The Output routine also checks for the special inflection codes, which the INFLT routine inserts into the output buffer in response to the special symbol, #. You must enter the value of 1, 3, 5, or 7 from the keyboard following the # symbol.

Program Listing 2. Text-to-Speech source code.

```
00010 ;*********************************
00020 ;       L I S T I N G  2
00030 ;*********************************
00040 ;
00050 ;     TEXT-TO-SPEECH PROGRAM
00060 ;              BY
00070 ;        DAVID ENGELHARDT
00100 ;
00110 ;*********************************
00120 ;
00130           ORG     0E950H          ;ORG'ED FOR 48K SYSTEM
00140 CHTBLE    EQU     0F000H          ;STRT LOOKUP TABLE
00150 ENDTBL    EQU     0F0C0H          ;LENGTH OF LOOKUP TBL
00160 ;
00170 VOTRAX    PUSH    HL              ;SAVE ALL REGISTORS
00180           PUSH    DE
00190           PUSH    BC
00200           PUSH    AF
00210           PUSH    IX
00220           PUSH    IY
00230           LD      HL,INPBUF       ;GET START OF INPUT BUFF
00240           LD      DE,OUTBUF       ;GET START OF OUTPUT BUFF
00250           LD      B,255           ;BYTE COUNTER
00260           LD      A,3FH           ;BYTE TO CONDITION BUFF
00270           LD      C,0             ;BYTE TO CONDITION BUFF
00280 CLRBF1    LD      (HL),C          ;CONDITION INPUT BUFF
00290           LD      (DE),A          ;CONDITION OUTPUT BUFF
00300           INC     HL
00310           INC     DE
00320           DJNZ    CLRBF1          ;REDUE UNTIL B=0
00330           LD      B,145           ;LOAD COUNTER
00340 CLRBF2    LD      (DE),A          ;CONTINUE TO CONDITION
00350           INC     DE
00360           DJNZ    CLRBF2          ;DO IT AGAIN
00370 ;
00380           CALL    0A7FH           ;GET PASSED VALUE
00390           LD      B,0             ;CLR B REG
00400           LD      C,(HL)          ;GET LENGTH OF STRING
00410           XOR     A               ;CLR A REG
00420           CP      C               ;BAIL OUT IF C=0
00430           JP      Z,RETURN        ;ERROR-BAIL OUT
00440           INC     HL
00450           LD      E,(HL)          ;GET LSB OF STR. LOC
00460           INC     HL
00470           LD      D,(HL)          ;GET MSB OF STR. LOC
00480           LD      HL,INPBUF       ;GET STRT OF INPUT BUFF
00490           LD      (HL),' '        ;FILL 1ST LOC WITH BLANK
00500           PUSH    HL              ;SAVE STRT OF BUFF
00510           INC     HL              ;STRT TO PUT DATA
00520           EX      DE,HL           ;SWAP PTRS
00530 ML1       LD      A,(HL)          ;GET CHAR FROM STRNG BUFF
00540           SUB     61H             ;TEST UPPER/LOWER CASE
00550           JP      M,ML2           ;SKIP IF UPPER CASE
00560           LD      A,-20H          ;LOAD CONVERT VALUE
00570           ADD     A,(HL)          ;MAKE LOWER CASE UPPER
00580           LD      (HL),A          ;STORE VALUE BACK
00590 ML2       LDI                     ;MOVE CHARACTER
00600           JP      PE,ML1          ;IS THE MOVE DONE?
00610           INC     HL
00620           LD      (HL),' '        ;LOAD SPACE TO END
00630           POP     IX              ;INPBUF LOC TO IX
00640           LD      IY,OUTBUF       ;SET IY TO OUTPUT BUFF
00650 ;
00660 ;*************************
00670 ; CHARACTER LOOKUP ROUTINE
00680 ;*************************
00690 ;
00700 CHTYPE    LD      HL,CHTBLE       ;GET START OF TABLE
00710           LD      A,(ENDTBL)      ;GET LENGTH OF TABLE
00720           ADD     A,-3            ;OBTAIN CORRECT LENGTH
00730           LD      B,0             ;CLR B
00740           LD      C,A             ;LOAD LOOP CTR
00750           LD      A,(IX)          ;GET STRG CHA
00760           INC     HL              ;INC TO NEXT ASCII CH
00770 LOOP      INC     HL              ;INC TWICE PAST WORD
00780           INC     HL              ;  LOCATION OF RULE
00790           CPI                     ;HL=HL+1  BC=BC-1
00800           JR      Z,MATCH         ;FOUND MATCH
00810           DEC     C               ;SET TO TRACK HL
00820           DEC     C               ;DEC LENGTH CTR
00830           JR      NZ,LOOP         ;SCAN AGAIN
00840           JP      INFLT           ;CHECK FOR INFLECT CHAR
00850 ;
00860 MATCH     LD      E,(HL)          ;GET LSB OF CH. RULES
00870           INC     HL
00880           LD      D,(HL)          ;GET MSB OF CH. RULES
00890           LD      (SAVEIX),IX     ;SAVE IX PTR
00900           EX      DE,HL           ;PUT STRT OF RULE IN HL
00910           JP      RULSCN          ;COMPARE RULES
00920 ;
00930 ;*********************************
00940 ; INFLECTION CHANGE ROUTINE = 1
00950 ;*********************************
00960 ;
00970 INFLT     CP      '1'             ;INFLECTION CODE?
00980           JP      NZ,ERROR        ;NOT RECOGNIZABLE CHAR
00990           INC     IX              ;GET INFLECTION PITCH
01000           LD      A,(IX)          ;  TO A REG FOR TEST
01010           CP      '1'             ;1=NORMAL BASE
01020           JR      NZ,THREE        ;TEST FOR A 3
01030           LD      A,'O'           ;LOAD 1 CODE
01040           JR      CODE            ;GOTO CODE ROUTINE
01050 THREE     CP      '3'             ;IS CODE SET TO A 3?
01060           JR      NZ,FIVE         ;GO TEST FOR A 5
01070           LD      A,'T'           ;LOAD 3 CODE
01080           JR      CODE
01090 FIVE      CP      '5'             ;IS CODE SET TO A 5?
01100           JR      NZ,SEVEN        ;GO TEST FOR A 7
01110           LD      A,'F'           ;LOAD 5 CODE
01120           JR      CODE
01130 SEVEN     CP      '7'             ;IS CODE SET TO A 7
01140           JP      NZ,ERROR        ;ILLEGAL CODE ENTERED
01150           LD      A,'S'           ;LOAD 7 CODE
01160 ;
01170 CODE      LD      (IY),A          ;LOAD CODE TO OUTPUT BUFF
01180           INC     IY
01190           JP      G1              ;GET NEXT CHARACTER
01200 ;
01210 ;*******************
01220 ; PROGRAM VARIABLES
01230 ;*******************
01240 ;
01250 INFLEC    DEFB    1               ;INFLECT CODE SET TO 1
01260 LRFLAG    DEFB    0               ;USED TO CONTROL L/R ROUT
01270 SAVEIX    DEFW    00              ;SAVES CURR IX PTR
01280 IXPTR     DEFW    00              ;HOLDS CURR IX PTR
01290 ;
01300 ;*******************
01310 ; RULE SCAN ROUTINE
01320 ;*******************
01330 ;
01340 RULSCN    PUSH    IX
01350           POP     DE              ;MAKE DE PT TO STRT OF IX
01360 RULSN1    CP      (HL)            ;HL PTS TO RULE
01370           JR      Z,OK            ;DOUBLE CHECK
01380           JP      NEXTRL          ;RULE FAILURE..CHECK NEXT
01390 OK        INC     HL              ;PT TO NEXT RULE CH
01400           LD      A,(HL)          ;GET RULE CH
01410           CP      '<'             ;LEFT SYMBOL
01420           JP      Z,LEFT          ;GOTO LEFT ROUTINE
01430           CP      '>'             ;RIGHT SYMBOL
01440           JP      Z,RIGHT         ;GOTO RIGHT ROUTINE
01450           CP      '='             ;EQUAL MEANS MATCH
01460           JP      Z,TXFR          ;TRANS. DATA TO TALK BUFF
01470           INC     IX
01480           LD      A,(IX)          ;GET NEW STR. CH
01490           JR      RULSN1          ;SCAN AGAIN
01500 ;
01510 ;*******************
01520 ; LEFT SCAN ROUTINE
01530 ;*******************
01540 ;
01550 LEFT      INC     HL              ;SET FOR NEXT CH IN RULE
01560           PUSH    HL              ;SAVE HL
01570           LD      HL,LRFLAG       ;GET RIGHT/LEFT FLAG
01580           SET     0,(HL)          ;SET FOR LEFT SCAN
01590           POP     HL              ;RESTORE HL
01600           LD      A,(HL)          ;GET NEXT CH RULE
01610           CP      '!'
01620           JP      Z,NOALPH        ;GOTO BLANK ROUTINE
01630           CP      '#'
01640           JP      Z,VOWEL         ;GOTO CHECK VOWELS
01650           CP      ':'
01660           JP      Z,CONSTS        ;CHECK 0 OR MORE CONSNTS
01670           CP      '+'
01680           JP      Z,FRNTVL        ;CHECK FRONT VOWEL
01690           CP      '^'
01700           JP      Z,CONSNT        ;ROUTINE TO MATCH 1 CONS
01710           CP      '.'
01720           JP      Z,VCNSNT        ;MATCH VOICED CONSONANT
01730           CP      '%'
01740           JP      Z,SIBLNT        ;CHECK FOR SIBILANTS
01750           CP      'S'
01760           JP      Z,INFCON        ;CHECK INFLUENCING CONSNT
01770           CP      '>'
01780           JP      Z,RIGHT         ;PERFORM RIGHT ROUTINE
01790           CP      '='
01800           JP      Z,TXFR          ;TRANSFER DATA TO BUFFER
01810           JP      ASCCHK          ;CHECK FOR ASCII CHARACT
01820 ;
01830 ;*******************
01840 ; RIGHT SCAN ROUTINE
01850 ;*******************
01860 ;
01870 RIGHT     INC     HL              ;INC TO NEXT CH RULE
01880           PUSH    HL
01890           LD      HL,LRFLAG       ;GET LEFT/RIGHT FLAG
01900           BIT     4,(HL)          ;SET AT <>IX PTR TIME
01910           JR      NZ,RPASS        ;SAVE ONLY ONCE
01920           LD      (IXPTR),IX      ;SAVE PTR
01930           SET     4,(HL)
01940 RPASS     RES     0,(HL)          ;RESET FOR RIGHT SCAN
01950           POP     HL
01960           LD      A,(HL)          ;GET NEXT RULE CH
01970           CP      '!'
01980           JP      Z,NOALPH        ;GOTO BLANK ROUTINE
01990           CP      '#'
02000           JP      Z,VOWEL         ;CHECK FOR VOWELS
02010           CP      ':'
02020           JP      Z,CONSTS        ;CHK FOR 0 OR MORE CONS
02030           CP      '^'
02040           JP      Z,CONSNT        ;CHK FOR 1 CONS MATCH
02050           CP      '+'
02060           JP      Z,FRNTVL        ;MATCH FRONT VOWEL
02070           CP      '.'
02080           JP      Z,VCNSNT        ;CHK VOICED CONSONANTS
02090           CP      '%'
02100           JP      Z,SUFFIX        ;CHECK SUFFIXES
02110           CP      '='
02120           JP      Z,TXFR          ;PUT DATA TO TALK BUFFER
02130           JP      ASCCHK          ;CHK FOR ASCII CHARACTER
02140 ;
02150 ;*********************
02160 ; NON-ALPHA TEST = !
02170 ;*********************
02180 ;
02190 NOALPH    CALL    SIDE            ;RIGHT OR LEFT SIDE CALL?
02200           JR      NZ,LNO          ;LEFT RULE
02210           INC     IX              ;RIGHT RULE..NEXT STR
02220           LD      A,(IX)          ;GET STR. CH
02230           JR      RNO             ;GO COMPARE
02240 LNO       DEC     DE              ;SET TO BEFORE IX PTR
02250           LD      A,(DE)          ;GET CH
02260 RNO       SUB     64              ;TEST FOR NON-ALPHA
02270           JP      P,NEXTRL        ;GET NEXT RULE IF ANY
02280           CALL    SIDE            ;FIND OUT WHERE TO JUMP
```

Listing 2 continued

The INFLT routine assigns these values special characters inserted into the phoneme output buffer. Since these special inflection characters are greater in value than the largest phoneme code (3F hex), lines 5670–5690 test for values above or below 40 hex. The routine passes values below 40 hex as phoneme codes, and values greater than 40 hex as inflection codes.

At the proper inflection code, the Output routine loads the value into register A and sends it to port 18. This output latches the STB line and changes at the same time the inflection pitch. The pitch remains constant until the program sends a different inflection code to the Votrax chip.

Lines 5880 and 5890 make up the set-up time delay that I described earlier. The smallest value that you can send to the time delay subroutine at 60 hex is six. (Increase this value to 10 for a Model I computer. Anything less results in erratic conditions in the creation of speech.)

Since the Text-to-Speech program and the Rules Table are separate, you must link them somehow. The first two equates in Listing 2 take care of this. The CHTBLE value is the location where the Rules Table starts in memory and the ENDTBL value is the length of the Rules Table's index. The CHTYPE routine uses the CHTBLE value to find out where it should start scanning at the rules index. If you add more rules, change the ENDTBL value accordingly.

The Rules Index

Refer to the Rules Table (Listing 3). The equates at the beginning make entering the phoneme codes easier. I felt that entering standard phoneme codes was easier than entering the hex or decimal equivalents. You may notice that some of the codes have two-letter designations instead of just one. This is so the assembler can recognize the phoneme codes as equates instead of Assembly commands that would result in assembly errors.

The character index in Listing 3 is called CHTBLE. All of the characters that pertain to the rules are included in this table. If you want to add a new character that has no existing rules, insert it at the end just before the ENDTBL label. Examine the other rules as to how to add new characters and the appropriate rules.

The @ symbol between each rule indicates the end of the rule and the start of another one. If the next value is not an up arrow (or bracket), more rules

Listing 2 continued

```
02290           JP      Z,RIGHT         ;BACK TO RIGHT SCAN ROUT
02300           JP      LEFT            ;BACK TO LEFT SCAN ROUT
02310 ;
02320 ;*******************************************
02330 ; ROUTINE TO TEST FOR ONE OR MORE VOWELS = #
02340 ;*******************************************
02350 ;
02360 VOWEL     PUSH    HL              ;SAVE TABLE RULE POINTER
02370           LD      HL,VMESS        ;GET PARAMS
02380           LD      A,(VMLEN)       ;LENGTH
02390           LD      B,A             ;LOAD B WITH VMESS LENGTH
02400           CALL    SIDE            ;CHECK DIRECTION TO SCAN
02410           JR      NZ.LVOW         ;LEFT SIDE
02420           INC     IX              ;RIGHT RULE
02430           LD      A,(IX)          ;GET CH
02440           JR      RVOW            ;RIGHT DIRECTION
02450 LVOW      DEC     DE              ;SET TO BEFORE IX PTR
02460           LD      A,(DE)          ;GET CHARACTER
02470 RVOW      CP      (HL)            ;COMPARE WITH VMESS VALUE
02480           JP      Z,VCONT         ;MATCH?
02490           INC     HL              ;INC TO NEXT VMESS VALUE
02500           DJNZ    RVOW            ;GO BACK AND CHECK
02510           POP     HL              ;RESTORE RULE POINTER
02520           JP      NEXTRL          ;NO FOUND - NEXT RULE
02530 VCONT     POP     HL              ;FOUND MATCH
02540           CALL    SIDE            ;CHECK DIRECTION OF SCAN
02550           JP      Z.RIGHT         ;GO TO RIGHT SCAN
02560           JP      LEFT            ;GO TO LEFT SCAN
02570 ;
02580 VMESS     DEFM    'AEIOUY'        ;VOWELS
02590 VMLEN     DEFB    $-VMESS         ;LENGTH OF VMESS
02600 ;
02610 ;*****************************
02620 ; ZERO OR MORE CONSONANTS = :
02630 ;*****************************
02640 ;
02650 CONSTS    CALL    SIDE            ;CHECK SCAN DIRECTION
02660           JR      NZ.LCON         ;IS IT RIGHT OR LEFT?
02670           INC     IX              ;RIGHT..INC POINTER
02680           JR      RCON            ;GO CHECK
02690 LCON      DEC     DE              ;LEFT..DEC POINTER
02700 RCON      CALL    SIDE            ;CHECK SCAN DIRECTION
02710           JP      Z.RIGHT         ;GO RIGHT
02720           JP      LEFT            ;GO LEFT
02730 ;
02740 ;**************************
02750 ; MATCH ONE CONSONANT = "
02760 ;**************************
02770 ;
02780 CONSNT    LD      A,(CONLEN)      ;TBL LENGTH
02790           LD      B,A             ;B IS LOOP LENGTH
02800           CALL    SIDE            ;CHECK SCAN DIRECTION
02810           JR      NZ.LCONS        ;LEFT RULE
02820           INC     IX
02830           LD      A,(IX)          ;GET STR. CH
02840           JR      RCONS           ;GO RIGHT
02850 LCONS     DEC     DE              ;DEC TO LEFT
02860           LD      A,(DE)          ;VALUE BEFORE IX PTR
02870 RCONS     PUSH    HL              ;SAVE RULE PTR
02880           LD      HL,CONMES       ;STRT OF CONSONANTS
02890 CLOOP     CP      (HL)            ;COMPARE
02900           JR      Z.CMATCH        ;SKIP IF FOUND MATCH
02910           INC     HL              ;INC TO NEXT CONS
02920           DJNZ    CLOOP           ;CHECK NEXT VALUE
02930           POP     HL              ;RESTORE RULE PTR
02940           JP      NEXTRL          ;CHECK NEXT RULE
02950 CMATCH    POP     HL              ;RESTORE RULE PTR
02960           CALL    SIDE            ;CHECK DIRECTION
02970           JP      Z,RIGHT         ;GO RIGHT
02980           JP      LEFT            ;GO LEFT
02990 ;
03000 CONMES    DEFM    'BCDFGHJKLMNPQRSTVWXZ'  ;CONSONANTS
03010 CONLEN    DEFB    $-CONMES        ;CONMES LENGTH
03020 ;
03030 ;*******************************
03040 ; MATCH A VOICED CONSONANT = .
03050 ;*******************************
03060 ;
03070 VCNSNT    LD      A,(VCONLE)      ;TBLE LEN
03080           LD      B,A             ;SAVE LOOP COUNTER
03090           CALL    SIDE            ;CHECK DIRECTION
03100           JR      NZ,VCLFT        ;SKIP TO LEFT CHECK
03110           INC     IX              ;INC STRING POINTER
03120           LD      A,(IX)          ;GET VALUE
03130           JR      VCRT            ;GO TEST COMPARE
03140 VCLFT     DEC     DE              ;DEC LEFT POINTER
03150           LD      A,(DE)          ;GET CHARA
03160 VCRT      PUSH    HL              ;SAVE RULE POINTER
03170           LD      HL,VMESSL       ;GET STRT OF DATA
03180 VLOOP     CP      (HL)            ;MATCH?
03190           JR      Z.VMATCH        ;SKIP IF MATCH
03200           INC     HL              ;INC VMESSL POINTER
03210           DJNZ    VLOOP           ;GO BACK AND CHECK AGAIN
03220           POP     HL              ;RESTORE RULE PTR
03230           JP      NEXTRL          ;NO MATCH...NEXT RULE
03240 VMATCH    POP     HL              ;RESTORE RULE PTR
03250           CALL    SIDE            ;CHECK DIRECTION
03260           JP      Z,RIGHT         ;GO TO RIGHT SCAN
03270           JP      LEFT            ;BACK TO LEFT ROUTINE
03280 ;
03290 VMESSL    DEFM    'BDGJLMNRVWZ'   ;VOICED CONSONANTS
03300 VCONLE    DEFB    $-VMESS         ;LENGTH OF VMESSL
03310 ;
03320 ;************************
03330 ; MATCH FRONT VOWEL = +
03340 ;************************
03350 ;
03360 FRNTVL    CALL    SIDE            ;CHECK DIRECTION
03370           JR      NZ.FVL          ;LEFT RULE
03380           INC     IX              ;PT TO NEXT CH IN STR
03390           LD      A,(IX)          ;GET CH
03400           JR      FVR             ;CHECK RIGHT SIDE
03410 FVL       DEC     DE              ;DEC LEFT POINTER
03420           LD      A,(DE)          ;GET VALUE TO CHECK
03430 FVR       CP      'E'             ;CHECK FOR FRONT VOWLS
03440           JR      Z.FVMTCH
03450           CP      'I'
03460           JR      Z,FVMTCH
03470           CP      'Y'
03480           JR      Z.FVMTCH
03490           JP      NEXTRL
03500 FVMTCH    CALL    SIDE            ;CHECK DIRECTION
03510           JP      Z.RIGHT         ;GO RIGHT
03520           JP      LEFT            ;GO LEFT
03530 ;
03540 ;************************
03550 ; TEST FOR SUFFIX'S = %
03560 ;************************
03570 ;
03580 SUFFIX    PUSH    HL              ;RIGHT SIDE SCAN ONLY
03590           INC     IX              ;INC STRING POINTER
03600           PUSH    IX              ;SAVE IT
03610           LD      A,(SMESSL)      ;GET LENGTH
03620           LD      B,A             ;SET LOOP COUNTER
03630           LD      HL,SMESS        ;GET PARAM STRT
03640 SLOOP1    LD      A,(IX)          ;GET CHAR
03650           CP      (HL)            ;COMPARE VALUE
03660           JR      Z,SECCHK        ;GO CHK 2ND VALU IF THERE
03670 SLOOP2    DEC     B               ;DEC LOOP COUNTER
03680           JR      Z.ROUTER        ;NO MATCH
03690           INC     HL              ;INC SMESS POINTER
03700           LD      A,(HL)          ;GET NEXT VALUE
03710           CP      ','             ;NEXT SET OF VALUES?
03720           JR      NZ.SLOOP2       ;GO TEST AGAIN
03730           POP     IX              ;RESTORE POINTER LOCATION
03740           PUSH    IX              ;SAVE IT AGAIN
03750           INC     HL              ;INC SMESS POINTER
03760           JR      SLOOP1          ;GO BACK & CHECK AGAIN
03770 SECCHK    INC     HL              ;CHECK FOR FINAL MATCH
03780           LD      A,(HL)          ;GET VALUE
03790           CP      ','             ;MATCH?
03800           JR      Z,FNDMTH        ;SKIP IF FOUND MATCH
03810           INC     IX              ;INC STRINP POINTER
03820           DJNZ    SLOOP1          ;CHECK NEXT VALUE
03830 ;
03840 ROUTER    POP     IX              ;RESTORE INPUT BUFF PTR
03850           POP     HL              ;RESTORE RULE TABLE PTR
03860           JP      NEXTRL          ;NEXT RULE
03870 FNDMTH    POP     BC              ;BIT BUCKET..THROW AWAY
03880           POP     HL              ;RESTORE RULE TABLE PTR
03890           JP      RIGHT           ;NEXT CHAR
03900 ;
03910 SMESS     DEFM    'E,ER,ES,ED.ING,ELY,'   ;SUFFIX'S
03920 SMESSL    DEFB    $-SMESS         ;LENGTH OF SMESS
03930 ;
03940 ;**********************
03950 ; CHECK SIBILANTS = &
03960 ;**********************
03970 ;
03980 SIBLNT    PUSH    HL              ;SAVE RULE PTR
03990           LD      HL,SIBMES       ;GET STRT OF TABLE
04000           LD      A,(SIBLEN)      ;GET LENGTH
04010           LD      B,A             ;SAVE IT
04020           CALL    SIBCON          ;GO AND COMPARE
04030           POP     HL              ;RESTORE RULE PTR
04040           JP      NZ.NEXTRL       ;SKIP IF NO MATCH
04050           JP      LEFT            ;MATCH..GO LEFT
04060 ;
04070 ;**********************
04080 ; INFLUENCING CONS = $
04090 ;**********************
04100 ;
04110 INFCON    PUSH    HL              ;SAVE RULE PTR
04120           LD      HL,CONSM        ;GET TABLE LOCATION
04130           LD      A,(CONSLN)      ;GET LENGTH OF TABLE
04140           LD      B,A             ;SET UP LOOP COUNTER
04150           CALL    SIBCON          ;GO CHECK FOR MATCH
04160           POP     HL              ;RESTORE RULE POINTER
04170           JP      NZ.NEXTRL       ;NO MATCH...NEXT RULE
04180           JP      LEFT            ;MATCH...GO LEFT SCAN
04190 ;
04200 ;*************************************************
04210 ; COMMON ROUTINE FOR SIBILANT & INFLUENCING CONS CHECK
04220 ;*************************************************
04230 ;
04240 SIBCON    DEC     DE              ;DEC LEFT POINTER
04250           PUSH    DE              ; AND SAVE IT
04260 SIBLOP    LD      A,(DE)          ;GET VALUE
04270           CP      (HL)            ;MATCH WITH TABLE VALUE?
04280           JR      Z,CHKAGN        ;SKIP IF NO AND CHK AGAIN
04290 SIB2      DEC     B               ;DEC LOOP COUNTER
04300           JR      Z.SIBERR        ;ZERO = NO MATCH
04310           INC     HL              ;INC TABLE POINTER
04320           LD      A,(HL)          ;GET VALUE
04330           CP      ','             ;READY FOR NEXT VALUE?
04340           JR      NZ.SIB2         ;SKIP IF NOT
04350           POP     DE              ;RESTORE LEFT POINTER
04360           PUSH    DE              ;SAVE IT AGAIN
04370           INC     HL              ;INC TABLE POINTER
04380           JR      SIBLOP          ;CHECK AGAIN
04390 CHKAGN    INC     HL              ;INC TABLE POINTER
04400           LD      A,(HL)          ;GET VALUE
04410           CP      ','
04420           JR      Z,FNDIT         ;IF HERE = VALUE FOUND
04430           DEC     DE              ;CHK FOR 2ND CHARCTER
04440           DJNZ    SIBLOP          ;GO BACK AND CHECK
04450 ;
04460 SIBERR    POP     DE              ;RESTORE LEFT POINTER
04470           SET     7,A             ;SET ERROR BIT.
04480           BIT     7,A             ;TEST FOR JUMP ON RETURN
04490           RET                     ;RETURN TO CALLER
04500 FNDIT     POP     DE              ;BIT BUCKET
04510           RES     7,A             ;RESET ERROR BIT
04520           BIT     7,A             ;TEST FOR JUMP ON RETURN
04530           RET                     ;RETURN TO CALLER
04540 ;
04550 SIBMES    DEFM    'S,C.G,Z.X,J,HC,HS,'    ;&..(BACKWARDS)
04560 SIBLEN    DEFB    $-SIBMES        ;LENGTH OF SIBMES
04570 ;
04580 CONSM     DEFM    'T,S,R,D,L,Z.N,J,HT,HC.HS,'     ;INFL CON
```

Listing 2 continued

INTRODUCTION TO TRS-80 DATA FILES

Send me _____ copies of **Introduction to TRS-80 Data Files** (BK7398)

at $24.97 each, plus $1.50 per system for shipping and handling.

☐ payment enclosed ☐ MasterCard ☐ Visa ☐ Am. Ex.

Card #_____ MC bank #_____

Expiration date_____ Signature_____

Name_____

Address_____

City_____ State_____ Zip_____

Wayne Green Books, Peterborough, NH 03458 **342B8DB**

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

Business Reply Card

First Class Permit No. 73, Peterborough, NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE

WAYNE GREEN BOOKS

Attn: Retail Sales

Rte. 101 and Elm St.

Peterborough, NH 03458

Listing 2 continued

```
04590 CONSLN  DEFB    S-CONSM          ;LENGTH OF CONSM
04600 ;
04610 ;*****************************************
04620 ; CHECK FOR RIGHT OR LEFT ASCII CHARACTER
04630 ;*****************************************
04640 ;
04650 ASCCHK  CALL    SIDE             ;CHECK SCAN DIRECTION
04660         JR      NZ,LASC          ;LEFT SIDE
04670         INC     IX               ;INC STRING POINTER
04680         LD      A,(IX)           ;GET CH RIGHT SIDE
04690         JR      CHECK            ;GO CHECK VALUES
04700 LASC    DEC     DE               ;DEC LEFT POINTER
04710         LD      A,(DE)           ;GET CH LEFT SIDE
04720 CHECK   CP      (HL)             ;INPUT AND RULE MATCH?
04730         JP      NZ.NEXTRL        ;NZ = NO MATCH
04740         CALL    SIDE             ;CHECK SCAN DIRECTION
04750         JP      Z.RIGHT          ;GO RIGHT SCAN ROUTINE
04760         JP      LEFT             ;GO LEFT SCAN ROUTINE
04770 ;
04780 ;*************************************************
04790 ; RIGHT OR LEFT TEST TO CHECK SCAN ROUTINE DIRECTION
04800 ;*************************************************
04810 ;
04820 SIDE    LD      A,(LRFLAG)       ;GET CONTROL WORD
04830         BIT     0,A              ;TEST BIT 0
04840         RET                      ;RETURN CALL
04850 ;
04860 ;*****************************
04870 ; SET PTRS FOR NEXT RULE SCAN
04880 ;*****************************
04890 ;
04900 NEXTRL  NOP                      ;JUMP ENTRY POINT
04910 NLOOP   LD      A,(HL)           ;GET CH
04920         CP      '@'              ;FIND RULE SEPERATOR
04930         JR      Z.FNDNXT         ;FOUND NEW RULE
04940         CP      '|'              ;MEANS END OF THESE RULES
04950         JP      Z,ERROR          ;ERROR...FOUND NO MATCH
04960         INC     HL               ;INC RULE POINTER
04970         JR      NLOOP            ;CHECK NEXT
04980 ;
04990 FNDNXT  XOR     A                ;CLEAR A REGISTOR
05000         LD      (LRFLAG),A       ;CLR FLAG WORD
05010         LD      IX,(SAVEIX)      ;RESTORE IX FOR NEXT RULE
05020         LD      A,(IX)           ;GET CH
05030         INC     HL               ;SET PTR PAST @ MARKER
05040         JP      RULSCN           ;CHECK NEXT RULE
05050 ;
05060 ;***************************
05070 ; TXFR DATA TO TALK BUFFER
05080 ;***************************
05090 ;
05100 TXFR    INC     HL               ;SET PTR PAST = SIGN
05110 TLOOP   LD      A,(HL)           ;GET VALUE
05120         CP      '@'              ;DONE TXFR?
05130         JR      Z.GNXTCH         ;Z= TEST FOR DONE
05140         LD      (IY),A           ;PUT DATA TO TALK BUFF
05150         INC     IY               ;INC TALK BUFF POINTER
05160         INC     HL               ;PT TO NEXT RULE DATA
05170         JR      TLOOP            ;DO LOOP AGAIN
05180 ;
05190 GNXTCH  LD      A,(LRFLAG)       ;GET FLAG CONTROL WORD
05200         BIT     4,A              ;IS BIT 4 SET?
05210         JR      Z,G1             ;ZERO = SKIP IX RESTORE
05220         LD      IX,(IXPTR)       ;RESTORE PTR BE4 < OR >
05230 G1      INC     IX               ;INC TO NEXT VALUE
05240         XOR     A                ;CLR A REG
05250         CP      (IX)             ;COMPARE A & IX FOR 00
05260         JR      Z,OUTPUT         ;ZERO = DONE DECODING
05270         LD      (LRFLAG),A       ;CLR FLAG FOR NEXT RULE
05280         JP      CHTYPE           ;GET NEXT STR CH RULE
05290 ;
05300 ;***************
05310 ; ERROR ROUTINE
05320 ;***************
05330 ;
05340 ERROR   LD      A,0DH            ;LOAD CARRIGE CODE
05350         CALL    33H              ;MOVE CURSOR DWN 1 LINE
05360         INC     IX               ;INC INPUT BUFF PTR
05370         LD      (IX),A           ;PUT ETX AT END (0D)
05380         LD      HL,INPBUF        ;GET STRT OF BUFF
05390         CALL    21BH             ;MOVE INP DATA TO SCREEN
05400 ;
05410         LD      HL,OUTBUF        ;GET OUTBUF LOCATION
05420         LD      DE,(4020H)       ;CURSOR POS TO PRINT ERR
05430         LD      BC.255           ;# MAX BYTES THAT CAN MOV
05440 ERLOOP  LDI                      ;MOVE OUT DATA TO SCREEN
05450         LD      A,3FH            ;STOP CODE
05460         CP      (HL)             ;COMPARE FOR STOP CODE
05470         JR      NZ.ERLOOP        ;MOVE NEXT CHARACTER
05480         LD      A,0DH            ;LOAD CARRIAGE CODE
05490         CALL    33H              ;MOVE CURSOR DWN 1 LINE
05500 ERR     LD      HL,ERRMES        ;GET ERROR DATA
05510         JP      OUT1             ;OUTPUT ERROR MESSAGE
05520 ;
05530 ERRMES  DEFW    0000H            ;ERROR PHONEME CODES
05540         DEFW    2B00H            ;  USED TO SAY = ERROR.
05550         DEFW    3E3AH
05560         DEFW    3F3EH
05570 ;
05580 ;***********************
05590 ; OUTPUT DATA TO SC-01
05600 ;***********************
05610 ;
05620 OUTPUT  LD      HL,OUTBUF        ;GET STRT OF OUTPUT BUFF
05630 OUT1    LD      A,16
05640         OUT     (236),A          ;TURN ON EXTERNAL BUS
05650 OLOOP   LD      A,144            ;SET UP 8255 FOR MODE 0
05660         OUT     (19),A           ;  OPTION NUMBER 8
05670         LD      A,(HL)           ;GET PHONEME CODE
05680         CP      40H              ;TEST TOP OF PHONEME CODE
05690         JP      M,NOINFL         ; FOR INFLECTION VALUES
05700         CP      'O'              ;IS 1 INFLECTION?
05710         JR      NZ.I3            ;GO TEST FOR 3 CODE
05720         LD      A,1              ;SET TO 1
05730         JR      SETINF           ;GO STORE VALUE
05740 I3      CP      'T'              ;IS 3 INFLECTION?
05750         JR      NZ,I5            ;GO TEST FOR 5 CODE
05760         LD      A,3              ;SET TO 3
05770         JR      SETINF           ;GO STORE VALUE
05780 I5      CP      'F'              ;IS 5 INFLECTION?
05790         JR      NZ.I7            ;GO TEST FOR 7 CODE
05800         LD      A,5              ;SET TO 5
05810         JR      SETINF           ;GO STORE VALUE
05820 I7      CP      'S'              ;IS 7 INFLECTION?
05830         JR      NZ,OL1           ;SKIP AND IGNORE VALUE
05840         LD      A,7              ;SET TO 7
05850 SETINF  LD      (INFLEC),A       ;STORE INFLECTION CODE
05860         JR      OL1              ;GO GET NEXT BUFF VALUE
05870 NOINFL  OUT     (17),A           ;SEND PHONEME TO 8255
05880         LD      BC,0006H         ;SET UP TIME DELAY
05890         CALL    60H              ; TO ALLOW SET-UP TIME
05900         LD      A,(INFLEC)       ;GET INFLECTION CODE
05910         OUT     (18),A           ;TELL SC-01 TO DO IT
05920 RESCAN  IN      A,(16)           ;READ A/R LINE OF 8255
05930         BIT     0,A              ;IS BIT SET = DONE
05940         JR      Z,RESCAN         ;GO BACK AND SCAN AGAIN
05950         LD      A,(HL)           ;GET PHONEME VALUE AGAIN
05960         CP      3FH              ;IS IT THE STOP CHARACT?
05970         JR      Z.RETURN         ;ZERO = DONE & EXIT
05980 OL1     INC     HL               ;INC TO NEXT PHONEME VALU
05990         JR      OLOOP            ;GO GET NEXT PHONEME CODE
06000 RETURN  XOR     A                ;CLEAR A REG
06010         LD      (LRFLAG),A       ;CLEAR CONTROL WORD
06020         POP     IY               ;RESTORE ALL REGISTORS
06030         POP     IX
06040         POP     AF
06050         POP     BC
06060         POP     DE
06070         POP     HL
06080         RET
06090 ;
06100 ;**************
06110 ; TALK BUFFERS
06120 ;**************
06130 ;
06140 INPBUF  DEFS    255              ;INPUT STRING BUFFER
06150         DEFB    0                ;SAFETY STOP CHARACTER
06160 OUTBUF  DEFS    400              ;OUTPUT BUFFER
06170         DEFB    3FH              ;SAFETY STOP CHARACTER
06180 ;
06190         END     VOTRAX
```

follow. The up arrow signals the end of a specific group of rules to the TTSPRG program.

Many of these rules were developed by the Naval Research Laboratory (NRL). You can find this report in "Automatic Translation of English Text to Phonetics by Means of Letter to Sound Rules" by Honey Sue Elovitz et al., United States Naval Research Laboratory Report Number 7948, 1976.

I had to modify a few of these rules to fit this application. It appears that the new Votrax system has more phonemes for better pronunciation purposes than the old Votrax system did when the NRL originally designed the rules. I rewrote some rules to enhance pronunciation.

I also added many rules to increase the capability of creating speech. The Rules Table now comprises 416 rules. I will add more in the future. To help you, the beginning of each rule has either a commented number or semicolon to the right of it. I number the rules in increments of five.

As previously mentioned, the Rules Table guides the Text-to-Speech program in its decoding routines by using special symbols to control the program's operations. If all the symbols and codes match, and the TTSPRG program finds an equals sign, the rule is complete and the program inserts the following code into the output buffer until it reaches the @ symbol. The program returns to CHTYPE and decodes the next input string character if there is one.

Keep in mind that my Rules Table isn't 100 percent accurate, nor do any exist that are. You'll find words that the Votrax can't pronounce exactly, but they'll be close and recognizable. For instance, one example pertains to the long and short *i* sounds. Currently, these rules make the board pronounce the word "like" properly, but it pronounces "life" with a short *i* sound.

TCS Heart of TEXAS COMPUTER SYSTEMS

Toll Free 1-800-433-5184 Texas 1-817-274-5625

CORVUS

25% OFF LIST PRICE!

OMNINET--A high speed multiuser network that can connect several computers for instant communication up to 4,000 feet away. **Save over $1,000 on a 20mg. Corvus Hard Disk system for TRS-80, IBM, and other computers.**

5mg. $1649 10mg. $2279 20mg. $3049

Tandon

TCS DRIVE CABINET is industrial grade heavy guage metal, safety fused, and comes with gold plated external connector with extender cable.

1 DRIVE in Cabinet

40 track single sided	$199
80 track (dual sided 40 track)	$299
160 track (dual sided 80 track	$399

2 DRIVE Double Cabinet

40 track single sided	$399
80 track (dual sided 40 tracks)	$599
160 track (dual sided 80 tracks)	$799

1 DRIVE Double Cabinet

40 track single sided	$269
80 track (dual sided 40 track)	$369
160 track (dual sided 80 track)	$449

Drives in cabinets come assembled and tested with power supply. Order cable separately.

BARE DRIVES:
TM100-1 (40-trk SS) **$159** **TM100-2** (80-trk dual 40s) **CALL**

PRINTERS

PRINTER CABLES AND INTERFACES AVAILABLE
Call for current pricing

STAR PRINTERS

- 100 - 120 - 160 CPS
- Bidirectional Logic Seeking
- Friction and Tractor
- 9X9 Dot Matrix
- True Decenders
- High Res-Bit Image Block Graphics
- Super Script-Subscript
- Underlining
- Backspacing Doublestrike
- 5, 6, 8 1/2, 10, 12 and 17 Pitch
- Programmable Line Spacing
- SIX (6) MONTH WARRANTY

Gemini 10X (9 inch, 120 cps, Friction & Tractor standard)
Gemini 15X (15 inch, 120 cps, Friction & Tractor standard)
Delta 10 o(10 inch, 160 cps, Friction & Tractor standard)
PowerType Daisywheel

STAR Printers can be interfaced with most computers on the market today, such as: Apple II,IIe,III / IBM PC / Osborne / Heath Kit H89 / TRS-80 Model I, II,III,4,12,16,100 /Zenith Z89,290,100 / TI99/4A / Kaypro / Atari 400,800 / Commodore 64,Vic 20
CALL FOR OUR LOW PRICES

CompuDisk

MINI DISKETTES $1.50

100% Certified Error Free

Meets or exceeds standards for ANSI, DIN, JIS, ECMA.
Money back guarantee.

All disks include:
- Hub Rings
- Protective Envelopes
- Write Protect Tabs
- Adhesive Labels

For orders of 1,000 diskettes or more, CALL! Dealer inquiries invited.

BOXED: 10 per sleeved box

Type	5-9 Boxes	10 plus
SSDD	$16.99	$15.99
DSDD	$21.99	$20.99

BULK: 10 per pack

Type	5-9 Packs	10 plus
SSDD	$15.99	$14.99
DSDD	$20.99	$19.99

The PRODUCER

Sensational programming breakthrough. You may never again need to buy a packaged program to store and retrieve data and perform calculations. Design your own custom programs with The PRODUCER. It writes the entire program for you, including all the BASIC code. Available now for the TRS-80 Model I, III, and IBM-PC. See our extended ad in this issue. **Just $149.99.**

TRS-80

TRS-2000

Newly introduced from Tandy. IBM compatable, dual disk drive. Call for our low price!

TRS-80 equipment comes with original 90 day Manufacturer's Limited Warranty

MODEL 12 and MODEL 16

MODEL 12, 1 drive	$CALL
MODEL 12, 2 drives	$CALL
TCS MODEL 12 version, 2 Tandon drives (like the original)	$2995

MODEL 16B...Support up to 6 users. Run your whole office with hard disk capabilities for about $1000 per user!

MODEL 16B, 1 drive	$CALL
MODEL 16B, 2 drives	$CALL

Model 12 and Model 16 Accessories

128K memory board (256K Max.)	$629
128K extra memory chips (RS)	$269
128K extra memory chips (TCS)	$189
Xenix Microsoft Multi-user Basic	$269
Xenix Accounting Software	$CALL
Xenix Multiplan Spread Sheet Software	$263
MII/12 to M16 multi-user upgrade kit	$1339
DT/1 Video Terminal	$629

MODEL IV

MODEL IV, 16K Cassette	$825
MODEL IV, 64K, 2 drives, RS-232	$CALL

MODEL 100 PORTABLE COMPUTER

Full size typewriter keyboard, 8 line 40 character display.

8K / 24K / 32K Models **CALL**

TCS Model IV, 64K, 2 Disks

Systems come with 180 Day Warranty

$1499 RS-232 included free! **$1699**

With standard 40 track double density drives. Over 340,000 bytes. Enhanced Model IV Operating System

With 2 dual headed 40 track double density drives Over 730,000 bytes Enhanced Model IV Operating System

Fully assembled and tested systems that are software compatible and functionally identical to Radio Shack units sold at computer stores for $hundreds more.

- CONTROLLER BOARDS are high quality double sided epoxy boards with gold plated contacts.
- POWER SUPPLY is the finest switching type available.
- MOUNTING HARDWARE includes power and data cables.
- DISK DRIVES are Tandon, the same ones used by Radio Shack... 40 track, double density, with a 5 millisecond stepping rate.

TCS DISK EXPANSION KITS FOR TRS-80s

	M-III	M-IV
1 Controller, Power Supply, Hardware, Instructions	$249	$329
2 Kit 1 plus one 40-track Tandon Disk Drive	$429	$479
3 Kit 1 plus two 40-track Tandon Disk Drives	$598	$649
3a Kit 3 w/2 80-track Drives (dual sided 40s)	$698	$729

Model IV needs 64K to operate. For 64K Kit, CALL!

TCS Heart of TEXAS COMPUTER SYSTEMS

P.O. Box 1327 Arlington, Texas 76004-1327

Toll Free 1-800-433-5184 Texas 1-817-274-5625

Program Listing 3. Rules Table source code. The phoneme codes begin in line 85. The length of the index is given in line 211. The rule table starts in line 213.

```
00001 ;***********************
00002 ;   L I S T I N G  3
00003 ;***********************
00004 ;
00005 ;  RULE TABLE PROGRAM
00006 ;          BY
00007 ;    DAVID ENGELHARDT
00010 ;
00011 ;***********************
00012 ;
00013          ORG     0F000H
00014 ;
00015 EH3      EQU     0
00016 EH2      EQU     1
00017 EH1      EQU     2
00018 PA0      EQU     3
00019 DT       EQU     4
00020 A2       EQU     5
00021 A1       EQU     6
00022 ZH       EQU     7
00023 AH2      EQU     8
00024 I3       EQU     9
00025 I2       EQU     10
00026 I1       EQU     11
00027 MM       EQU     12
00028 N        EQU     13
00029 BB       EQU     14
00030 V        EQU     15
00031 CH       EQU     16
00032 SH       EQU     17
00033 ZZ       EQU     18
00034 AW1      EQU     19
00035 NG       EQU     20
00036 AH1      EQU     21
00037 OO1      EQU     22
00038 OO       EQU     23
00039 LL       EQU     24
00040 K        EQU     25
00041 J        EQU     26
00042 HH       EQU     27
00043 G        EQU     28
00044 F        EQU     29
00045 DD       EQU     30
00046 S        EQU     31
00047 AA       EQU     32
00048 AY       EQU     33
00049 Y1       EQU     34
00050 UH3      EQU     35
00051 AH       EQU     36
00052 PP       EQU     37
00053 O        EQU     38
00054 I1       EQU     39
00055 U        EQU     40
00056 Y        EQU     41
00057 T        EQU     42
00058 RR       EQU     43
00059 EE       EQU     44
00060 W        EQU     45
00061 AE       EQU     46
00062 AE1      EQU     47
00063 AW2      EQU     48
00064 UH2      EQU     49
00065 UH1      EQU     50
00066 UH       EQU     51
00067 O2       EQU     52
00068 O1       EQU     53
00069 IU       EQU     54
00070 U1       EQU     55
00071 THV      EQU     56
00072 TH       EQU     57
00073 ER       EQU     58
00074 EH       EQU     59
00075 E1       EQU     60
00076 AW       EQU     61
00077 PA1      EQU     62
00078 STOP     EQU     63
00079 ;
00080 ;***********************
00081 ; CHARACTER LOOKUP TABLE
00082 ;***********************
00083 ;
00084 CHTBLE   DEFS    3
00085          DEFB    'A'
00086          DEFW    AR
00087          DEFB    'B'
00088          DEFW    BR
00089          DEFB    'C'
00090          DEFW    CR
00091          DEFB    'D'
00092          DEFW    DR
00093          DEFB    'E'
00094          DEFW    ERULES
00095          DEFB    'F'
00096          DEFW    FR
00097          DEFB    'G'
00098          DEFW    GR
00099          DEFB    'H'
00100          DEFW    HR
00101          DEFB    'I'
00102          DEFW    IR
00103          DEFB    'J'
00104          DEFW    JR
00105          DEFB    'K'
00106          DEFW    KR
00107          DEFB    'L'
00108          DEFW    LR
00109          DEFB    'M'
00110          DEFW    MR
00111          DEFB    'N'
00112          DEFW    NR
00113          DEFB    'O'
00114          DEFW    OR
00115          DEFB    'P'
00116          DEFW    PR
00117          DEFB    'Q'
00118          DEFW    QR
00119          DEFB    'R'
00120          DEFW    RRULES
00121          DEFB    'S'
00122          DEFW    SR
00123          DEFB    'T'
00124          DEFW    TR
00125          DEFB    'U'
00126          DEFW    UR
00127          DEFB    'V'
00128          DEFW    VR
00129          DEFB    'W'
00130          DEFW    WR
00131          DEFB    'X'
00132          DEFW    XR
00133          DEFB    'Y'
00134          DEFW    YR
00135          DEFB    'Z'
00136          DEFW    ZR
00137          DEFB    ' '
00138          DEFW    BLANK
00139          DEFB    '.'
00140          DEFW    PERIOD
00141          DEFB    39
00142          DEFW    APOST
00143          DEFB    ','
00144          DEFW    COMMA
00145          DEFB    '?'
00146          DEFW    QUEST
00147          DEFB    '0'
00148          DEFW    ZERO
00149          DEFB    '1'
00150          DEFW    R1
00151          DEFB    '2'
00152          DEFW    R2
00153          DEFB    '3'
00154          DEFW    R3
00155          DEFB    '4'
00156          DEFW    R4
00157          DEFB    '5'
00158          DEFW    R5
00159          DEFB    '6'
00160          DEFW    R6
00161          DEFB    '7'
00162          DEFW    R7
00163          DEFB    '8'
00164          DEFW    R8
00165          DEFB    '9'
00166          DEFW    R9
00167          DEFB    '-'
00168          DEFW    MINUS
00169          DEFB    '+'
00170          DEFW    PLUS
00171          DEFB    '='
00172          DEFW    EQUALS
00173          DEFB    '*'
00174          DEFW    MULT
00175          DEFB    '/'
00176          DEFW    DIVIDE
00177          DEFB    8
00178          DEFW    LARROW
00179          DEFB    9
00180          DEFW    RARROW
00181          DEFB    91
00182          DEFW    UARROW
00183          DEFB    10
00184          DEFW    DARROW
00185          DEFB    31
00186          DEFW    CLEAR
00187          DEFB    '<'
00188          DEFW    LTHAN
00189          DEFB    '>'
00190          DEFW    GTHAN
00191          DEFB    ':'
00192          DEFW    COLON
00193          DEFB    ';'
00194          DEFW    SEMICO
00195          DEFB    '!'
00196          DEFW    EXCLA
00197          DEFB    '"'
00198          DEFW    QUOTE
00199          DEFB    '$'
00200          DEFW    DOLLAR
00201          DEFB    '%'
00202          DEFW    PERCNT
00203          DEFB    '&'
00204          DEFW    ANDD
00205          DEFB    '('
00206          DEFW    LBRACK
00207          DEFB    ')'
00208          DEFW    RBRACK
00209          DEFB    '@'
00210          DEFW    ATSYM
00211 ENDTBL   DEFB    $-CHTBLE
00212 ;
00213 ;*************
00214 ; RULE TABLES
00215 ;*************
00216 ;
00217 AR       DEFM    'A<!>!='        ;1
00218          DEFB    A1
00219          DEFB    AY
00220          DEFB    Y
00221          DEFB    '@'
```

Listing 3 continued

Sometimes the board pronounces *y* as in "why," as opposed to the correct *ies* ending.

You can add rules to correct this; however, adding rules is tricky and you should take great care. Remember that what holds true for one expression may not hold true for another. Feel free to change some of the spelling in words to obtain accurate results in speech creation. For example, you can get the same pronunciation by spelling TODAY as 2DAY.

Hello Again

Now use all of this theory to convert the word HELLO as you did in last month's article. A Basic program calls the Text-to-Speech program via a USR function call. The USR command passes a free parameter obtained from the VARPTR(B$) command that points to the string's memory location.

On entry, the Text-to-Speech program conditions both the input and output buffers as I described earlier. It then fills the first location in INPBUF with a blank and converts any lowercase letters to uppercase. The program then transfers the entire string to the input buffer. INPBUF now has the word HELLO in it.

Register pair pointers are dedicated for use in controlling the Text-to-Speech program. The HL pair points to the conversion rules. The IX register pair points to the input buffer and the IY register pair points to the output buffer. The program uses register pair DE for left side scan manipulations and keeps the BC pair as extras.

On entry to the CHTYPE routine, HL points to the Rules Table's index and IX points to the first input string character in INPBUF, a blank rule located on line 138 in CHTBLE (Listing 3). CHTYPE then loads the HL register pair with the next two values that point to the blank rule's memory location. The program then makes a jump to the RULSCN routine at line 1340 in Listing 2. Refer to the located blank rule, number 375 in Listing 3.

RULSCN first double-checks to ensure that the value IX points to is the same as the rule's HL value. Since both values are blanks, the HL pointer increments to the next value in the rule, the equals sign. RULSCN scans the new HL value for a left or right direction pointer and then an equals sign. An equals sign indicates a rule match and the routine jumps to TXFR.

TXFR increments the HL pointer to the value past the equals sign, 03 (PA0). It transfers the value of 03 to

✓ See List of Advertisers on Page 211

Listing 3 continued

```
00222           DEFM    'A<I>H='        ;
00223           DEFB    UH2
00224           DEFB    '@'
00225           DEFM    'A>I='          ;
00226           DEFB    UH2
00227           DEFB    '@'
00228           DEFM    'ARE>I='        ;
00229           DEFB    AH
00230           DEFB    ER
00231           DEFB    RR
00232           DEFB    '@'
00233           DEFM    'AR<I>O='       ;5
00234           DEFB    UH2
00235           DEFB    RR
00236           DEFB    '@'
00237           DEFM    'AR>#='         ;
00238           DEFB    EH2
00239           DEFB    RR
00240           DEFB    '@'
00241           DEFM    'AS<"I>#='      ;
00242           DEFB    AE1
00243           DEFB    EH3
00244           DEFB    S
00245           DEFB    '@'
00246           DEFM    'A>WA='         ;
00247           DEFB    UH2
00248           DEFB    '@'
00249           DEFM    'AW='           ;
00250           DEFB    AW
00251           DEFB    '@'
00252           DEFM    'ANY<:I='       ;10
00253           DEFB    EH2
00254           DEFB    EH2
00255           DEFB    N
00256           DEFB    Y
00257           DEFB    '@'
00258           DEFM    'AGAIN='        ;
00259           DEFB    UH1
00260           DEFB    G
00261           DEFB    A2
00262           DEFB    EH1
00263           DEFB    N
00264           DEFB    '@'
00265           DEFM    'ALLY<:#='      ;
00266           DEFB    UH2
00267           DEFB    LL
00268           DEFB    EE
00269           DEFB    '@'
00270           DEFM    'AL<I>#='       ;
00271           DEFB    UH1
00272           DEFB    LL
00273           DEFB    '@'
00274           DEFM    'A>"+#='        ;
00275           DEFB    AA
00276           DEFB    AY
00277           DEFB    '@'
00278           DEFM    'AG<:#>E='      ;15
00279           DEFB    I1
00280           DEFB    DD
00281           DEFB    J
00282           DEFB    '@'
00283           DEFM    'A>"+:#='       ;
00284           DEFB    AE
00285           DEFB    '@'
00286           DEFM    'A<:I>"+I='     ;
00287           DEFB    AA
00288           DEFB    AY
00289           DEFB    '@'
00290           DEFM    'A>"&='         ;
00291           DEFB    AA
00292           DEFB    AY
00293           DEFB    '@'
00294           DEFM    'ARR<I='        ;
00295           DEFB    UH1
00296           DEFB    RR
00297           DEFB    '@'
00298           DEFM    'ARR='          ;20
00299           DEFB    AE1
00300           DEFB    EH3
00301           DEFB    '@'
00302           DEFM    'AR<:I>I='      ;
00303           DEFB    AH1
00304           DEFB    UH2
00305           DEFB    ER
00306           DEFB    '@'
00307           DEFM    'AR>I='         ;
00308           DEFB    ER
00309           DEFB    '@'
00310           DEFM    'AR='           ;
00311           DEFB    AH1
00312           DEFB    RR
00313           DEFB    '@'
00314           DEFM    'AIR='          ;
00315           DEFB    EH2
00316           DEFB    EH2
00317           DEFB    RR
00318           DEFB    '@'
00319           DEFM    'AI='           ;25
00320           DEFB    AA
00321           DEFB    AY
00322           DEFB    '@'
00323           DEFM    'AY='           ;
00324           DEFB    AA
00325           DEFB    AY
00326           DEFB    '@'
00327           DEFM    'AU='           ;
00328           DEFB    AW
00329           DEFB    '@'
00330           DEFM    'AL<:#>I='      ;
00331           DEFB    UH
00332           DEFB    LL
00333           DEFB    '@'
00334           DEFM    'ALS<:#>I='     ;
00335           DEFB    UH
00336           DEFB    LL
00337           DEFB    ZZ
00338           DEFB    '@'
00339           DEFM    'ALK='          ;30
00340           DEFB    AW
00341           DEFB    K
00342           DEFB    '@'
00343           DEFM    'AL>"='         ;
00344           DEFB    AW1
00345           DEFB    UH3
00346           DEFB    LL
00347           DEFB    '@'
00348           DEFM    'ABLE<I>I='     ;
00349           DEFB    A1
00350           DEFB    Y
00351           DEFB    BB
00352           DEFB    UH3
00353           DEFB    LL
00354           DEFB    '@'
00355           DEFM    'ABLE<:I='      ;
00356           DEFB    A1
00357           DEFB    Y
00358           DEFB    BB
00359           DEFB    UH3
00360           DEFB    LL
00361           DEFB    '@'
00362           DEFM    'ABLE='         ;
00363           DEFB    UH2
00364           DEFB    BB
00365           DEFB    UH2
00366           DEFB    LL
00367           DEFB    '@'
00368           DEFM    'ANG>+='        ;35
00369           DEFB    A1
00370           DEFB    AY
00371           DEFB    Y
00372           DEFB    N
00373           DEFB    DD
00374           DEFB    J
00375           DEFB    '@'
00376           DEFM    'A='            ;
00377           DEFB    AE1
00378           DEFB    '@'
00379           DEFB    '['
00380 ;
00381 BR        DEFM    'B<I>I='        ;
00382           DEFB    BB
00383           DEFB    E1
00384           DEFB    Y
00385           DEFB    '@'
00386           DEFM    'BE<I>"#='      ;
00387           DEFB    BB
00388           DEFB    Y
00389           DEFB    '@'
00390           DEFM    'BEING='        ;
00391           DEFB    BB
00392           DEFB    E1
00393           DEFB    I2
00394           DEFB    NG
00395           DEFB    '@'
00396           DEFM    'BOTH<I>I='     ;40
00397           DEFB    BB
00398           DEFB    O2
00399           DEFB    O2
00400           DEFB    TH
00401           DEFB    '@'
00402           DEFM    'BUS<I>#='      ;
00403           DEFB    BB
00404           DEFB    I3
00405           DEFB    I3
00406           DEFB    ZZ
00407           DEFB    '@'
00408           DEFM    'BUIL='         ;
00409           DEFB    BB
00410           DEFB    I2
00411           DEFB    I2
00412           DEFB    LL
00413           DEFB    '@'
00414           DEFM    'B='            ;
00415           DEFB    BB
00416           DEFB    '@'
00417           DEFB    '['
00418 ;
00419 CR        DEFM    'C<I>I='        ;
00420           DEFB    S
00421           DEFB    E1
00422           DEFB    Y
00423           DEFB    '@'
00424           DEFM    'CH<I>"='       ;45
00425           DEFB    K
00426           DEFB    '@'
00427           DEFM    'CH<E"='        ;
00428           DEFB    K
00429           DEFB    '@'
00430           DEFM    'CH='           ;
00431           DEFB    T
00432           DEFB    CH
00433           DEFB    '@'
00434           DEFM    'CI<S1>#='      ;
00435           DEFB    S
00436           DEFB    AH1
00437           DEFB    I3
00438           DEFB    '@'
00439           DEFM    'CI>A='         ;
00440           DEFB    SH
00441           DEFB    '@'
00442           DEFM    'CI>O='         ;50
00443           DEFB    SH
00444           DEFB    '@'
00445           DEFM    'CI>EN='        ;
00446           DEFB    SH
00447           DEFB    '@'
00448           DEFM    'C>+S1='        ;
00449           DEFB    S
00450           DEFB    II
```

Listing 3 continued

OUTBUF, to which the IY pair points, and increments HL again. TXFR tests for the @ symbol, which signifies the end. This symbol makes the program increment the IX pointer (INPBUF) and jump back to the CHTYPE routine to check the next character in the word (HELLO).

The next character rules the program locates are the H rules. The TTSPRG locates the beginning of the H rules at line 786 in the Rules Table. With the HL pointing at rule 132, it double-checks via TTSPRG and HL then increments to the letter A in HAV. The program then increments the IX pointer and the routine tests for a match. Since the A in HAV and the E in HELLO don't match, control passes to the NEXTRL routine.

The NEXTRL routine increments the HL pointer in a loop searching for the @ symbol. If it finds the up-arrow (or left bracket) symbol, it jumps to the Error routine since no rule exists for this character. If the routine finds the @ symbol, it restores the IX pair and points it to the H character. Next, it increments HL to the next rule's first character and goes back to RULSCN to test this rule. This operation continues until rule 136 matches.

Notice that this rule only has the right scan symbol (>), so control goes to the Right Scan routine. This routine increments the HL register and checks the symbol after the > symbol. The next value is a number symbol, so the routine jumps to the Vowel routine. You may notice that the special symbol for a vowel (#) is the same as that for the inflection symbol. This doesn't present a problem because the program tests vowels and inflections at different points.

The Vowel routine saves the IX pointer and increments it to test the next character after the H for a vowel. It matches because the E after H in HELLO is a vowel. The match passes control back to the Right Scan routine, which checks the next character after the # symbol. The Right Scan routine finds the equals sign, indicating another match. The equated phoneme code value HH transfers to the output buffer and the routine jumps back to CHTYPE to check the next character (the E in HELLO).

Now the program scans the E rules to find rule 198 (ELL = EH1,UH3,LL, UH3). RULSCN doesn't find any special characters so it checks the ASCII values. This routine bumps the IX string pointer and compares the L in the rule with the L in the string. Since a

DISPLAYED VIDEO IS DRIVING DOWN PRICES ON DISKS!

FREE SHIPPING!

TEC'S NEW HALF-HIGH 40 TRACK DISK DRIVE W/CASE & POWER SUPPLY AT AN INCREDIBLE LOW PRICE!

$195.00!!!!

NEW!
TRUE HALF HEIGHT DRIVES!
1 5/8 INCH NOT 2 INCH OR 2 7/16 INCH! DIRECT DRIVE!
NO DRIVE BELT! 3MS TRACK TO TRACK!

"YOU CAN BUY THE REST BUT WHY NOT BUY ONE OF THE BEST!"

TEAC's NEW SLIMLINE 40 TRACK W/CASE & POWER SUPPLY $220.00
TANDON 40 TRACK TM-100-1 W/CASE & POWER SUPPLY $205.00

(DOUBLE SIDED 40 TRACK DRIVES ADD $75.00)

TEAC NEW SLIMLINE DOUBLE SIDED 40 TRACK W/CASE & POWER SUPPLY $250.00
TEAC NEW SLIMLINE DOUBLE SIDED 80 TRACK W/CASE & POWER SUPPLY $299.00
TANDON SINGLE SIDED SLIMLINE 8" DISK DRIVE W/DUAL CASE & POWER SUPPLY $545.00
TANDON DOUBLE SIDED SLIMLINE 8" DISK DRIVE W/DUAL CASE & POWER SUPPLY $625.00

TWO DRIVE CASES AVAILABLE AT VARIED PRICES
TWO DRIVE 5 1/4" CABLE (FOR MOST COMPUTERS) ... **$23.99** WITH GOLD PLATED CONNECTORS

DV'S COLOR COMPUTER 1st DRIVE ONLY $369.00

ONE YEAR WARRANTY ON TEAC & SHUGART DRIVES/180 DAYS ON TEAC & TANDON

HAYES SMART MODEM 300 BAUD ..... $220.00 1200 BAUD $515.00
MODEL 1 DOUBLE DENSITY BOARD $89.00

PRINTER PRICES

WE'RE NOT GOING OT MAKE YOU CALL FOR PRICES — HERE THEY ARE IN BLACK AND WHITE!!!

EPSON

RX80 FT W/GRAPHTRAX PLUS $469.00
MX100 W/GRAPHTRAX PLUS $639.00
RX80 W/GRAPHTRAX PLUS $359.00
FX80 W/GRAPHTRAX PLUS $569.00
FX100 W/GRAPHTRAX PLUS $759.00

STAR

GEMINI 10X $295.00
GEMINI 15 ..... $479.00
CITOH PROWRITER 8510 ..... $375.00
STARWRITER F10 ..... $1169.00
DWP 210 ..... $649.00

PRINTER CABLE 10' LONG W/GOLD PLATED CONNECTORS STARTING AT $25.99
DAISY WHEEL PRINTERS AVAILABLE AT VARIED PRICES

Visit our two retail locations at:
886 Ecorse Road
Ypsilanti, MI 48197
(313) 426-5086/(313) 482-4424
or
111 Marshall Street
Litchfield, MI 49252
(517) 542-3280
(517) 542-3939
(517) 542-3947

WE CARRY TRS80, LNW, EPSON & FRANKLIN COMPUTERS

IMMEDIATE DELIVERY
180 Days Parts and Labor Warranty
DEALER INQUIRIES INVITED
Free Shipping in the U.S.
48 Contiguous States

TO ORDER: Call (313) 426-5086 or (313) 482-4424 or (517) 542-3280
(517) 542-3939 (517) 542-3947
OR WRITE: DISPLAYED VIDEO 111 MARSHALL ST.,LITCHFIELD, MI 49252

*TRS-80 is a trademark of the Tandy Corporation

Prices & Specifications subject to change without notice

DISPLAYED VIDEO IS DRIVING DOWN PRICES ON DISKS!

4 DRIVES!

"NO YOU'RE NOT SEEING THINGS. THAT'S FOUR INTERNAL DRIVES IN A MODEL 4

$1999.00

AVAILABLE NOW, ONLY FROM DISPLAYED VIDEO!

$1999.00

4 DRIVES!

DISPLAYED VIDEO HAS DONE IT AGAIN!

THE STORAGE CAPACITY YOU'VE ALWAYS WANTED AT THE LOWEST PRICE EVER CONCEIVED!

FOUR DRIVES IN A MODEL 4, 64K.....**$1999.00**/128K.....**$2079.00**

QUALITY BACKED UP BY DV'S SIX MONTH YOU CAN'T LOSE WARRANTY!

DISPLAYED VIDEO is offering TRS-80* MODEL 4 disk drive systems for **INCREDIBLY** low prices, quality backed up by DV's six month you can't loose warranty.

MODEL 4 with 64K dual 40 track double density disk drives with TRSDOS 6.0 and TEN DISKETTES.

*PLUG IT IN AND GO.....***$1599.00**/128K...**$1679.00**

MODEL 4 with 64K dual 40/40 track double density disk drives with TRSDOS 6.0 and TEN DISKETTES.

*PLUG IT IN AND GO.....***$1849.00**/128K.....**$1920.00**

MODEL 4 with 64K dual 80/80 track double density internal disk drives, with TRSDOS 6.0 and TEN DISKETTES.

*PLUG IT IN AND GO.....***$2199.00**/128K.....**$2279.00**

Radio Shack New MODEL 4P (Portable) $1649.00
Radio Shack's answer to the IBM P.C. New Tandy 2000 ..... $2549.00
MODEL 4 with 64K four internal drives of any configuration available to achieve up to 4 meg of disk storage ..... CALL
MODEL III/4 Internal Two Drive Kit: Includes controller board, dual drive mounting bracket, dual power supply, all hardware cable; and connectors (gold plated) & TEC Drive $399.00

4 DRIVE KIT MINUS DRIVE **$349.00**

MODEL 100 8K.....**$679.00** 24K.....**$839.00** DV'S MODEL I DOUBLE DENSITY BOARDS.....**$89.00**

WE CARRY TRS-80, IBM, LNW, EPSON & FRANKLIN COMPUTERS

Visit our retail location at:
111 Marshall Street
Litchfield, MI 49252
(313) 426-5086
(313) 482-4424
(517) 542-3280
(517) 542-3939
(517) 542-3947

DISPLAYED VIDEO

IMMEDIATE DELIVERY
DV'S SIX MONTH
PARTS AND LABOR WARRANTY

Authorized Dealership at:
111 Marshall Street, Litchfield, Michigan 49252 **P142**
To Order: Call (313) 426-5086; (313) 482-4424; (517) 542-3280
(517) 542-3939; (517) 542-3947
OR WRITE:
Displayed Video, 111 Marshall St., Litchfield, MI 49252

*TRS-80 is a trademark of the Tandy Corporation Prices subject to change without notice.

Listing 3 continued

```
00451           DEFB    ZZ
00452           DEFB    '@'
00453           DEFM    'C>+='          ;
00454           DEFB    S
00455           DEFB    '@'
00456           DEFM    'CK='           ;
00457           DEFB    K
00458           DEFB    '@'
00459           DEFM    'COM>%='        ;55
00460           DEFB    K
00461           DEFB    UH
00462           DEFB    MM
00463           DEFB    '@'
00464           DEFM    'CON<I='        ;
00465           DEFB    K
00466           DEFB    UH1
00467           DEFB    N
00468           DEFB    '@'
00469           DEFM    'C='            ;
00470           DEFB    K
00471           DEFB    '@'
00472           DEFB    '['
00473 ;
00474 DR        DEFM    'D<I>I='        ;
00475           DEFB    DD
00476           DEFB    E1
00477           DEFB    Y
00478           DEFB    '@'
00479           DEFM    'DED<:#>I='     ;
00480           DEFB    DD
00481           DEFB    I2
00482           DEFB    DD
00483           DEFB    '@'
00484           DEFM    'DG='           ;60
00485           DEFB    DD
00486           DEFB    J
00487           DEFB    '@'
00488           DEFM    'D<E.>I='       ;
00489           DEFB    DD
00490           DEFB    '@'
00491           DEFM    'D<E:"#>I='     ;
00492           DEFB    T
00493           DEFB    '@'
00494           DEFM    'DE<I>"#='      ;
00495           DEFB    DD
00496           DEFB    Y
00497           DEFB    '@'
00498           DEFM    'DIS<I='        ;
00499           DEFB    DD
00500           DEFB    I1
00501           DEFB    S
00502           DEFB    '@'
00503           DEFM    'DO<I>I='       ;65
00504           DEFB    DD
00505           DEFB    IU
00506           DEFB    U1
00507           DEFB    U1
00508           DEFB    '@'
00509           DEFM    'DOES<I='       ;
00510           DEFB    DD
00511           DEFB    UH2
00512           DEFB    UH1
00513           DEFB    ZZ
00514           DEFB    '@'
00515           DEFM    'DOING<I='      ;
00516           DEFB    DD
00517           DEFB    IU
00518           DEFB    U1
00519           DEFB    I2
00520           DEFB    NG
00521           DEFB    '@'
00522           DEFM    'DOW<I='        ;
00523           DEFB    DD
00524           DEFB    AH1
00525           DEFB    UH3
00526           DEFB    U1
00527           DEFB    '@'
00528           DEFM    'DU>A='         ;
00529           DEFB    DD
00530           DEFB    J
00531           DEFB    IU
00532           DEFB    U
00533           DEFB    '@'
00534           DEFM    'D='            ;70
00535           DEFB    DD
00536           DEFB    '@'
00537           DEFB    '['
00538 ;
00539 ERULES     DEFM    'E<I>I='        ;
00540           DEFB    E1
00541           DEFB    Y
00542           DEFB    '@'
00543           DEFM    'E<I>Q='        ;
00544           DEFB    EE
00545           DEFB    '@'
00546           DEFM    'E<I>#='        ;
00547           DEFB    EE
00548           DEFB    '@'
00549           DEFM    'E<:#>I='       ;
00550           DEFB    '@'
00551           DEFM    'E<:"I>I='      ;75
00552           DEFB    '@'
00553           DEFM    'E<:I>I='       ;
00554           DEFB    E1
00555           DEFB    '@'
00556           DEFM    'ED<#>I='       ;
00557           DEFB    I2
00558           DEFB    DD
00559           DEFB    '@'
00560           DEFM    'E<:#>DI='      ;
00561           DEFB    '@'
00562           DEFM    'ED<:>I='       ;
00563           DEFB    I2
00564           DEFB    DD
00565           DEFB    '@'
00566           DEFM    'EV>ER='        ;80
00567           DEFB    EH
00568           DEFB    V
00569           DEFB    '@'
00570           DEFM    'EX='           ;
00571           DEFB    EH1
00572           DEFB    EH2
00573           DEFB    K
00574           DEFB    PA0
00575           DEFB    S
00576           DEFB    '@'
00577           DEFM    'E<V='          ;
00578           DEFB    '@'
00579           DEFM    'E>"%='         ;
00580           DEFB    EE
00581           DEFB    '@'
00582           DEFM    'ERI>#='        ;
00583           DEFB    I1
00584           DEFB    RR
00585           DEFB    Y
00586           DEFB    '@'
00587           DEFM    'ERI='          ;85
00588           DEFB    EH1
00589           DEFB    RR
00590           DEFB    I3
00591           DEFB    '@'
00592           DEFM    'ERY>I='        ;
00593           DEFB    EH1
00594           DEFB    RR
00595           DEFB    Y
00596           DEFB    '@'
00597           DEFM    'ER<:#>#='      ;
00598           DEFB    ER
00599           DEFB    '@'
00600           DEFM    'ER>#='         ;
00601           DEFB    EH1
00602           DEFB    RR
00603           DEFB    '@'
00604           DEFM    'ER='           ;
00605           DEFB    ER
00606           DEFB    '@'
00607           DEFM    'EVEN<I='       ;90
00608           DEFB    EE
00609           DEFB    V
00610           DEFB    EH
00611           DEFB    N
00612           DEFB    '@'
00613           DEFM    'E<:#>W='       ;
00614           DEFB    '@'
00615           DEFM    'E>I='          ;
00616           DEFB    '@'
00617           DEFM    'EW<$='         ;
00618           DEFB    IU
00619           DEFB    U
00620           DEFB    '@'
00621           DEFM    'EW=Y'          ;
00622           DEFB    Y
00623           DEFB    IU
00624           DEFB    U
00625           DEFB    '@'
00626           DEFM    'E>O='          ;95
00627           DEFB    EE
00628           DEFB    '@'
00629           DEFM    'ES<&:#>I='     ;
00630           DEFB    IZ
00631           DEFB    ZZ
00632           DEFB    '@'
00633           DEFM    'E<:#>SI='      ;
00634           DEFB    '@'
00635           DEFM    'ELL='          ;
00636           DEFB    EH1
00637           DEFB    UH3
00638           DEFB    LL
00639           DEFB    UH3
00640           DEFB    '@'
00641           DEFM    'ELY<:#>I='     ;
00642           DEFB    LL
00643           DEFB    EE
00644           DEFB    '@'
00645           DEFM    'EMENT<:#='     ;100
00646           DEFB    MM
00647           DEFB    EH3
00648           DEFB    N
00649           DEFB    T
00650           DEFB    '@'
00651           DEFM    'EN<I='         ;
00652           DEFB    EH1
00653           DEFB    N
00654           DEFB    '@'
00655           DEFM    'EFUL='         ;
00656           DEFB    P
00657           DEFB    OOI
00658           DEFB    LL
00659           DEFB    '@'
00660           DEFM    'EE='           ;
00661           DEFB    EE
00662           DEFB    '@'
00663           DEFM    'EARN='         ;
00664           DEFB    ER
00665           DEFB    RR
00666           DEFB    N
00667           DEFB    '@'
00668           DEFM    'EAR<I>"='      ;105
00669           DEFB    ER
00670           DEFB    RR
00671           DEFB    '@'
00672           DEFM    'EAD='          ;
00673           DEFB    EH1
00674           DEFB    EH3
00675           DEFB    DD
00676           DEFB    '@'
00677           DEFM    'EA<:#>I='      ;
00678           DEFB    EE
00679           DEFB    UH2
00680           DEFB    '@'
```

Listing 3 continued

match occurs, this routine bumps the IX and HL pointers to test the next two ASCII values.

ASCII testing continues until the routine comes across the equals sign to signify another rule match. If the program encounters a sign other than the equals sign, control branches to the appropriate routine. Control passes again to the TXFR section where it sends the equated values of EH1, UH3, LL, and UH3 to the OUTBUF buffer. Control jumps back to CHTYPE with the IX pointer at the O in HELLO.

CHTYPE now scans the O rules and eventually finds the rule O>!=O1,U1 (line 1290 in Listing 3). It transfers these values to OUTBUF, which now contains the values PA0, HH, EH1, UH3, LL, UH3, O1, U1, and the stop code 3F. Control goes to the Output routine which sends the phonemes out to the Votrax chip. After completing the transfer, the routine restores all registers and jumps back to the calling program.

This short conversion should give you a general idea of what goes on in the TTSPRG. I didn't cover all conversion routines due to time and space limitations. I suggest you use a piece of paper to help keep track of all the pointers as you go through the various routines.

Applications Software

Now that you know how the Text-to-Speech program operates, you need to control it through a Basic program. Here I show you how to use Basic to send the Text-to-Speech program the strings you want to convert to speech. Listing 1 is the first Basic text conversion program.

You can apply Listing 1 for handicapped and educational uses. It gives sound feedback by pronouncing the letter of any key (except #) you press via the INKEY command. Single-key entries may be slow, but it takes time to pronounce each pressed key.

As it pronounces each key, the program builds a word string and speaks the word at the press of the space bar. Hitting the enter key makes the program vocalize all words created in a single string and tells you when to start typing with a spoken "Enter Please" message.

I installed the option of pronouncing or not pronouncing each key as you type it. The program verbally asks you this question on execution. It still vocalizes words individually when you hit the space bar.

The secret to this program is in its

✓ See List of Advertisers on Page 211

Listing 3 continued

```
00681           DEFM    'EA>SU='        ;
00682           DEFB    EH
00683           DEFB    '@'
00684           DEFM    'EA='           ;
00685           DEFB    EE
00686           DEFB    '@'
00687           DEFM    'EIGH='         ;110
00688           DEFB    A2
00689           DEFB    A2
00690           DEFB    Y
00691           DEFB    '@'
00692           DEFM    'EI='           ;
00693           DEFB    EE
00694           DEFB    '@'
00695           DEFM    'EYE<1='        ;
00696           DEFB    AH1
00697           DEFB    EH3
00698           DEFB    I3
00699           DEFB    Y
00700           DEFB    '@'
00701           DEFM    'EY='           ;
00702           DEFB    EE
00703           DEFB    '@'
00704           DEFM    'EU='           ;
00705           DEFB    Y
00706           DEFB    IU
00707           DEFB    U1
00708           DEFB    '@'
00709           DEFM    'E='            ;115
00710           DEFB    EH1
00711           DEFB    '@'
00712           DEFB    '['
00713 ;
00714 FR        DEFM    'F<I>I='        ;
00715           DEFB    EH1
00716           DEFB    EH2
00717           DEFB    F
00718           DEFB    '@'
00719           DEFM    'FUL='          ;
00720           DEFB    F
00721           DEFB    UH3
00722           DEFB    LL
00723           DEFB    '@'
00724           DEFM    'FY<I>I='       ;
00725           DEFB    F
00726           DEFB    AH1
00727           DEFB    EH3
00728           DEFB    Y
00729           DEFB    '@'
00730           DEFM    'F='            ;
00731           DEFB    F
00732           DEFB    '@'
00733           DEFB    '['
00734 ;
00735 GR        DEFM    'G<I>I='        ;120
00736           DEFB    DD
00737           DEFB    J
00738           DEFB    E1
00739           DEFB    Y
00740           DEFB    '@'
00741           DEFM    'GIV='          ;
00742           DEFB    G
00743           DEFB    I1
00744           DEFB    I3
00745           DEFB    V
00746           DEFB    '@'
00747           DEFM    'G<I>I"='       ;
00748           DEFB    G
00749           DEFB    '@'
00750           DEFM    'GE>T='         ;
00751           DEFB    G
00752           DEFB    EH1
00753           DEFB    EH3
00754           DEFB    '@'
00755           DEFM    'GGES<US='      ;
00756           DEFB    G
00757           DEFB    DD
00758           DEFB    J
00759           DEFB    EH
00760           DEFB    S
00761           DEFB    '@'
00762           DEFM    'GG='           ;125
00763           DEFB    G
00764           DEFB    '@'
00765           DEFM    'G<#B='         ;
00766           DEFB    G
00767           DEFB    '@'
00768           DEFM    'G>+='          ;
00769           DEFB    DD
00770           DEFB    J
00771           DEFB    '@'
00772           DEFM    'GREAT='        ;
00773           DEFB    G
00774           DEFB    RR
00775           DEFB    A1
00776           DEFB    Y
00777           DEFB    T
00778           DEFB    '@'
00779           DEFM    'GH<#='         ;
00780           DEFB    '@'
00781           DEFM    'G='            ;130
00782           DEFB    G
00783           DEFB    '@'
00784           DEFB    '['
00785 ;
00786 HR        DEFM    'H<I>I='        ;
00787           DEFB    A1
00788           DEFB    AY
00789           DEFB    Y
00790           DEFB    T
00791           DEFB    CH
00792           DEFB    '@'
00793           DEFM    'HAV<1='        ;
00794           DEFB    HH
00795           DEFB    AE1
00796           DEFB    EH3
00797           DEFB    V
00798           DEFB    '@'
00799           DEFM    'HERE<1='       ;
00800           DEFB    HH
00801           DEFB    AY
00802           DEFB    I3
00803           DEFB    RR
00804           DEFB    '@'
00805           DEFM    'HOUR<1='       ;
00806           DEFB    AH1
00807           DEFB    UH3
00808           DEFB    W
00809           DEFB    ER
00810           DEFB    '@'
00811           DEFM    'HOW='          ;135
00812           DEFB    HH
00813           DEFB    AH1
00814           DEFB    O2
00815           DEFB    U1
00816           DEFB    '@'
00817           DEFM    'H>#='          ;
00818           DEFB    HH
00819           DEFB    '@'
00820           DEFM    'H='            ;
00821           DEFB    '@'
00822           DEFB    '['
00823 ;
00824 IR        DEFM    'I<I>I='        ;
00825           DEFB    AH1
00826           DEFB    EH3
00827           DEFB    I3
00828           DEFB    Y
00829           DEFB    '@'
00830           DEFM    'IN<I='         ;
00831           DEFB    I1
00832           DEFB    N
00833           DEFB    '@'
00834           DEFM    'I<I>I='        ;140
00835           DEFB    AH1
00836           DEFB    EH3
00837           DEFB    I3
00838           DEFB    Y
00839           DEFB    '@'
00840           DEFM    'IN>D='         ;
00841           DEFB    I1
00842           DEFB    I3
00843           DEFB    N
00844           DEFB    '@'
00845           DEFM    'IDE='          ;
00846           DEFB    AH1
00847           DEFB    EH3
00848           DEFB    Y
00849           DEFB    DD
00850           DEFB    '@'
00851           DEFM    'IER='          ;
```

Listing 3 continued

```
10 '************************************************
20 '                 L I S T I N G  4
30 '************************************************
40 '
50 '       STRING ENTRY PRONOUNCIATION PROGRAM
60 '                     BY
70 '              DAVID ENGELHARDT
80 '
100 '
110 ' THIS PROGRAM MAY BE USED TO EXPERIMENT IN THE CREATION
120 ' OF WORDS OR LARGE STRINGS.
130 ' THIS PROGRAM MAY ALSO BE USED AS A SUBROUTINE IN A LARGER
140 ' PROGRAM.
150 '
160 '************************************************
170 '
180 CLS:CLEAR 1000
190 DEF USR0=&HE950:CMD"L","TTSPRG/CMD":CMD"L","TABLE/CMD"
200 LINE INPUT A$
210 A=VARPTR(A$)
220 VV=USR0(A)
230 GOTO 200
240 END
```

Program Listing 4. Basic Text-to-Speech routine.

string manipulations. You enter each key via the INKEY command. The program combines entries into a single word string that the program puts into a single string of up to 255 bytes. TS$ is the complete string while SB$ is the word string with A$ as the single-key entries.

Since the maximum string length is 255 bytes, any entry greater than this makes the program branch to an error routine. This error routine speaks a message indicating it has reached maximum count, and proceeds to speak the string created prior to the error. There is also a small error routine that keeps the program from stopping execution for any kind of error. This error is also spoken to alert you to a problem. After any error, the program prompts you with another "Enter Please" message to start string entries again.

If you want to create or change any of the messages, you must either change an existing message or define a new one in your program. For example, to hear a string message spoken, use A=VARPTR(A$) to find out where the string is located in memory. The next instruction is VV=USR(A) for tape or VV=USR0(A) for disk. This command passes the string's location in memory to the TTSPRG program for conversion to speech.

You could also adapt it to speak a screen full of ASCII text by using single-dimension arrays.

In addition to handicapped applications, Listing 1 also helps children learn the alphabet and letter pronunciations. It teaches them how to create and pronounce words. The program also vocalizes all numbers.

The board also vocalizes math operators, facilitating teaching young children math problems with voice tutoring. I am working on a routine to break a number down and pronounce the individual units. Take the number 120 for example. To date, the Text-to-Speech program pronounces this number as one, two, zero instead of one hundred twenty. I decided not to include this routine and associated rules due to the size of the program.

The only character not pronounced is the # symbol as I use this to change the inflection codes. If you want to change the inflection codes within or at the beginning of a sentence, enter the # symbol followed by a 1, 3, 5, or 7. A combination of these codes gives interesting results in a spoken sentence.

Listing 4 is a short subroutine that you can use in a bigger program to

A New Definition of Class

by

MICRO DESIGN

The SQ-5 Series REMOVABLE Cartridge Winchester Hard Disk Systems

for the TRS-80, IBM, & APPLE

- *Nine to Ten times faster than conventional floppy disk drives*
- *Combine Fixed and Removable Drives to meet your needs.*
- *Built-In Error Detection & Correction*
- *Up to 45 Megabytes of On-Line Storage*

Starting as low as **$1799.95**

TRS-80 System Upgrade for the Model III & the Model 4

Upgrade Features

- *The exceptional MDX-6 disk controller board*
- *Switching Power supply*
- *One 40-Track Disk drive*
- *ALL Necessary installation Cables & Hardware*

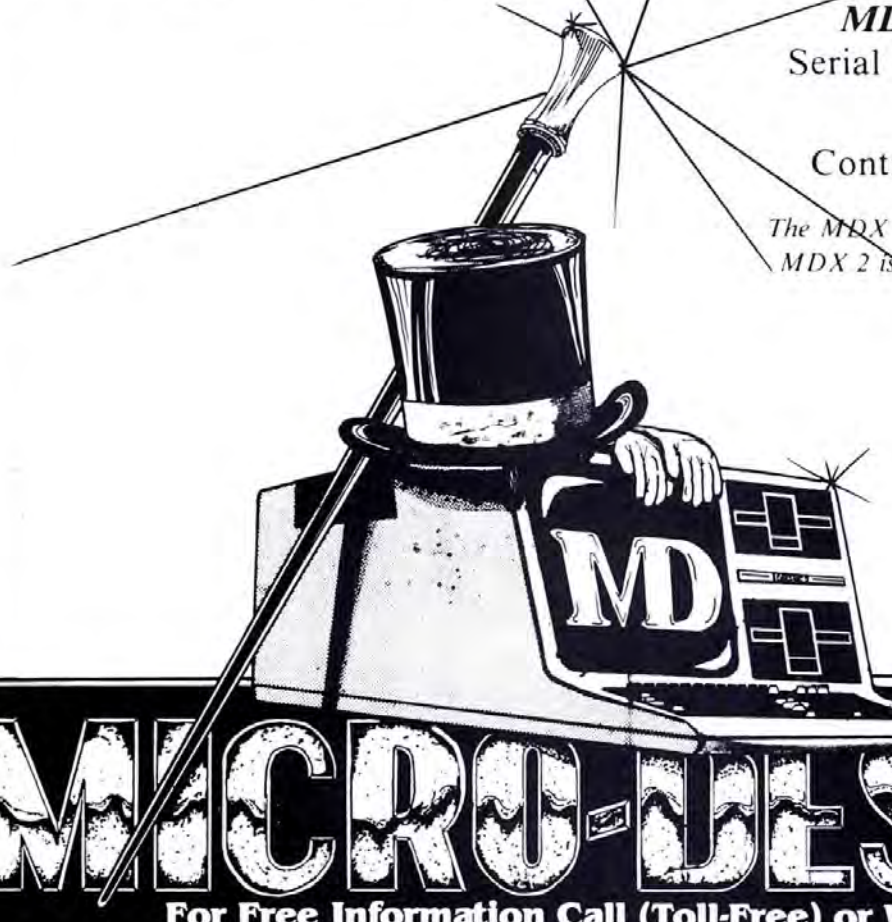
$399.00

Micro Design carries a complete line of state of the art Disk Drives, Standard and Slimline and the sixteen combination Siquest SQ-5 Series Winchester Hard Disk Systems. Additionally Micro Design Offers Expansions in the form of the fully Assembled MDX Series boards for the Model I, III, & 4, such as the

MDX-5 Phone Modem & Serial Port for only ***$159.95*** &

MDX-6 Floppy Disk Controller, your price ***$129.95***

The MDX 5 & 6 are for the model III & 4.
MDX 2 is the system for the the Model I.



MICRO-DESIGN

For Free Information Call (Toll-Free) or Write

6301 Manchaca Rd Suite B Austin Tx78745

1-800-531-5002

Visa & Mastercard Accpted ↙ 426 **Texas Res. 1-512-441-7890**

The SQ-5 Series are compatible with the new MODEL 2000. Watch for additional MODEL 2000 systems from Micro Design

↙ See List of Advertisers on Page 211

enter strings via the Input or Line Input command. If you don't have a disk system, use the Input command in place of the Line Input command. The only difference between the two is that the Input command doesn't allow insertion of commas in a string. This program also uses the inflection codes.

This program can prompt the user for an answer to a question or any other information and store it as a unique string. You can then send this unique string to the TTSPRG program when desired, such as in educational responses, games, and so on. I use this program to create words and sentences for experimental purposes.

System Adaptations

In regard to the USR command, if you run these programs on a 16K or cassette-based system, you need to change all of the VV=USR0(A) commands to VV=USR(A) in the Basic programs. The USR pointers have to be set so change line 220 in Listing 1 or line 190 in Listing 4 to read POKE 16526,240: POKE16527,105. This links USR calls to a machine-language program at the Text-to-Speech program in our case. Set memory size to 27115 and load in the speech programs with the System command.

Remember, to assemble the Text-to-Speech program and Rules Table for a 16K system, you must have someone with a bigger system do it for you unless you obtain the object code from another source. If someone does it for you, change the ORG statement in Listing 2 to 69F0 hex and the ORG statement in Listing 3 to 70A0 hex.

If you are going to assemble these programs on a 32K system, you may have to do some linking in regard to the Rules Table. The Text-to-Speech program is no problem as its source, including comments, is only 17,153 bytes long with room to spare. The Text-to-Speech program should be ORGed at A0F0 hex and the Rules Table at B0A0 hex for a 32K system.

If you have to split the Rules Table, be sure to incorporate all of the phoneme equates on the second source. This allows easy and straightforward entries of phoneme codes to the rules as opposed to the hex or decimal equivalent. To link the Rules Table sources together, patch into or equate at the beginning of the first source where the start of each rule is located in the second source. This should present no problems if you are careful.

System	Revisions
48K	TTSPRG=E950, TABLE=F000, CHTBLE=F000, ENDTBL=F0C0, MEM SIZE=59720
32K	TTSPRG=A9F0, TABLE=B0A0, CHTBLE=B0A0, ENDTBL=B160, MEM SIZE=43500
16K	TTSPRG=69F0, TABLE=70A0, CHTBLE=70A0, ENDTBL=7160, MEM SIZE=27115

Table 2. System changes.

Listings 2 and 3 as shown are for a 48K system. The ORGed values, memory size, and the two equated values at the beginning of the Text-to-Speech program are all you need to change to run it on a smaller system. Table 2 lists all value changes needed for each system.

Remember to change the DEF USR=&HXXXX statements in the Basic programs, if you use the 32K system instead of a 48K system. ■

Write to David Engelhardt at 10221 W. 101st Place, Broomfield, CO 80020.

Listing 3 continued

```
00852           DEFB    E1
00853           DEFB    I3
00854           DEFB    ER
00855           DEFB    '@'
00856           DEFM    'IED<R:#>!='    ;
00857           DEFB    EE
00858           DEFB    DD
00859           DEFB    '@'
00860           DEFM    'IED>!='        ;145
00861           DEFB    AH
00862           DEFB    E1
00863           DEFB    DD
00864           DEFB    '@'
00865           DEFM    'IEN='          ;
00866           DEFB    EE
00867           DEFB    EH
00868           DEFB    N
00869           DEFB    '@'
00870           DEFM    'IE>T='         ;
00871           DEFB    AH1
00872           DEFB    EH3
00873           DEFB    AY
00874           DEFB    I2
00875           DEFB    '@'
00876           DEFM    'IQUE='         ;
00877           DEFB    EE
00878           DEFB    K
00879           DEFB    '@'
00880           DEFM    'IR>#='         ;
00881           DEFB    AH1
00882           DEFB    EH3
00883           DEFB    AY
00884           DEFB    RR
00885           DEFB    '@'
00886           DEFM    'IR='           ;150
00887           DEFB    ER
00888           DEFB    '@'
00889           DEFM    'IGH='          ;
00890           DEFB    AH1
00891           DEFB    EH3
00892           DEFB    Y
00893           DEFB    '@'
00894           DEFM    'ILD='          ;
00895           DEFB    I2
00896           DEFB    I2
00897           DEFB    LL
00898           DEFB    DD
00899           DEFB    '@'
00900           DEFM    'IGN>!='        ;
00901           DEFB    AH1
00902           DEFB    EH3
00903           DEFB    Y
00904           DEFB    N
00905           DEFB    '@'
00906           DEFM    'IGN>"='        ;
00907           DEFB    AH1
00908           DEFB    EH3
00909           DEFB    Y
00910           DEFB    N
00911           DEFB    '@'
00912           DEFM    'IGN>$='        ;155
00913           DEFB    AH1
00914           DEFB    EH3
00915           DEFB    Y
00916           DEFB    N
00917           DEFB    '@'
00918           DEFM    'ING>!='        ;
00919           DEFB    I2
00920           DEFB    NG
00921           DEFB    '@'
00922           DEFM    'IG>%='         ;
00923           DEFB    AH1
00924           DEFB    Y
00925           DEFB    ZZ
00926           DEFB    '@'
00927           DEFM    'IS>%='         ;
00928           DEFB    AH1
00929           DEFB    Y
00930           DEFB    ZZ
00931           DEFB    '@'
00932           DEFM    'IE>!='         ;
00933           DEFB    AH1
00934           DEFB    EH3
00935           DEFB    Y
00936           DEFB    '@'
00937           DEFM    'I<L>E='        ;160
00938           DEFB    EE
00939           DEFB    '@'
00940           DEFM    'IE<.>"='       ;
00941           DEFB    AH1
00942           DEFB    EH3
00943           DEFB    Y
00944           DEFB    '@'
00945           DEFM    'I<L>F='        ;
00946           DEFB    IH
00947           DEFB    '@'
00948           DEFM    'IE='           ;
00949           DEFB    EE
00950           DEFB    '@'
00951           DEFM    'I>D%='         ;
00952           DEFB    I1
00953           DEFB    I3
00954           DEFB    '@'
00955           DEFM    'I>T%='         ;165
00956           DEFB    AH
00957           DEFB    E1
00958           DEFB    '@'
00959           DEFM    'I<"+>"+='      ;
00960           DEFB    IH
00961           DEFB    '@'
00962           DEFM    'I<:"#>"+='     ;
00963           DEFB    IH
00964           DEFB    '@'
00965           DEFM    'I<:!>%='       ;
00966           DEFB    AH
00967           DEFB    E1
00968           DEFB    '@'
00969           DEFM    'I>"+!#='       ;
00970           DEFB    AH1
00971           DEFB    EH3
00972           DEFB    '@'
00973           DEFM    'I>"+:#='       ;170
00974           DEFB    IH
00975           DEFB    '@'
00976           DEFM    'I>"+='         ;
00977           DEFB    AH2
00978           DEFB    I2
00979           DEFB    '@'
00980           DEFM    'I>%='          ;
00981           DEFB    EE
00982           DEFB    '@'
00983           DEFM    'I='            ;
00984           DEFB    I1
00985           DEFB    '@'
00986           DEFB    '['
00987 ;
00988 JR        DEFM    'J<!>!='        ;
00989           DEFB    DD
00990           DEFB    J
00991           DEFB    EH3
00992           DEFB    A1
00993           DEFB    AY
00994           DEFB    Y
00995           DEFB    '@'
00996           DEFM    'J='            ;175
00997           DEFB    DD
00998           DEFB    J
00999           DEFB    '@'
01000           DEFB    '['
01001 ;
01002 KR        DEFM    'K<!>!='        ;
01003           DEFB    K
01004           DEFB    EH3
01005           DEFB    A1
01006           DEFB    AY
01007           DEFB    Y
01008           DEFB    '@'
01009           DEFM    'K<!>N='        ;
01010           DEFB    '@'
01011           DEFM    'K='            ;
01012           DEFB    K
01013           DEFB    '@'
01014           DEFB    '['
```

Listing 3 continued on p. 148

Introducing SoftPac. A tool for modern man.

Meet **SoftPac** — the remarkable new software package that puts more raw power at your fingertips than anything yet created for the Radio Shack computer.

Integration is the key. **SoftPac** actually combines word processing, information management, spreadsheet calculations, and communications into one fully integrated system. So now you can experiment, recalculate and manipulate data, just by pressing a few keys, in an endless variety of ways. In fact, the only limit to **SoftPac** is your imagination.

SoftCalc . . . . the flexible spreadsheet!

SoftCalc . . . faster, better, smarter. Because it takes the guess work out of working with financial numbers. **SoftCalc** is a powerful electronic spreadsheet that speeds planning and budgeting.

SoftCalc's capabilities are numerous. 37 function commands . . . . nine operators . . . . repetitive task formulating . . . . and automatic graphs of your data at the touch of a key.

Whether it's investments, cash flow, inventory, cost estimates, or budgets, **SoftCalc** will help you analyze the impact of decisions before you make them.

SoftBase . . . . 5 × 7 file cards electronically!

SoftBase . . . . basically, a paper filing system without the paper. With **SoftBase** you can file, retrieve and review information in a fraction of the time it takes to use a conventional filing system. In fact, almost 50 times faster than popular disk-based filing systems!

With **SoftBase,** your allowed the freedom to customize your files to your specific needs using our exclusive full-screen cursor control. This gives you the ability to create client addresses, payment records and custom receipts, just to name a few. Manipulation of text is as easy as 1-2-3, using some 30 function commands.

And when you need to find and retrieve records, **SoftBase** sorts through your records electronically, using fast data access through special indexing. **SoftBase** allows you to be as selective as you like, with a variety of text retrieving methods.

SoftWriter . . . . Productivity at your fingertips!

SoftWriter . . . . the powerful full-screen editor with automatic formatting features. Even an inexperienced typist can sit down with **SoftWriter,** learn a few basic commands, and then quickly turn out perfect professional memos and letters.

With **SoftWriter,** you can correct misspellings or substitute one portion of text for another, even search for words using global or find commands, with just a few keystrokes!

Want to illustrate your **SoftWriter** document with a financial table or graph that's stored in a **SoftCalc** spreadsheet? You ask . . . **SoftPac** performs! In fact, merging and transferring portions of **SoftBase** to your text document is just as convenient!

And when you're through revising, **SoftWriter** shows you "on-screen" just how your document will look when it's printed. What you see is what you get.

SoftTerm . . . . Ease and speed!

SoftTerm . . . . the flexible communications program that thinks for you. With **SoftTerm** and your direct-connect modem, you can access data bases, bulletin boards, and other popular information services. **SoftTerm** saves and sends data from either disk or buffer and even transfers data from **SoftCalc, SoftBase** and **SoftWriter,** to any other source.

Other **SoftTerm** features such as flexible changing of protocols and auto dial technology are part of the Softronics tradition of high-features and high-performance.

So, there you have it. **SoftPac.** That's Softronics fully integrated package at work for only **$299.00**.

For the name and address of your nearest **SoftPac** dealer, call 1-918-749-6211.

SOFTRONICS
COMPUTER SYSTEMS
2300 East 14th Street
Suite 201
Tulsa, OK. 74104

116

See List of Advertisers on Page 211

Wayne Green Books

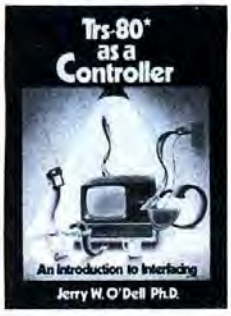
BOOKS

Introduction to TRS-80 Data Files

Learn by doing with this guide to writing a data base manager. This book, with its accompanying software, takes you through a simple mailing list program to teach you about sequential and then random access files. The construction of a DBM and the techniques for moving data to and from disks are discussed. **Book and TRS-80 disk** *$24.97 BK7398 approx. 144 pp.*

Inside Your Computer

Find out what goes on inside your TRS-80. Inside Your Computer *explains microcomputer circuits and how they work. Topics include chips, interpreters, circuits, machine language, binary numbers, algorithms, ASCII code, software, and what they all mean to the computer. Includes many photographs and schematics.* **$12.97 BK7390 108 pp.**

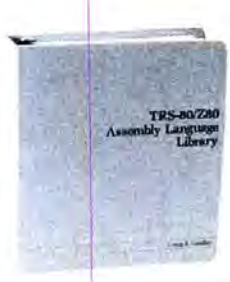


TRS-80 as a Controller

Learn to control outside devices with a TRS-80. This book is an introduction to interfacing, with simple, inexpensive projects. Applications include controlling lights and switches, building a small computer, and suggestions for more complex projects. The book applies to the Model III and, with minor conversions, to the Model I. **$12.97 BK7394 192 pp.**

The Rest of 80

Get the 80 Micro *articles so good we put them in a book. These 31 never-before-published tutorials and utilities were hand-picked for Model I and Model III users. You'll get graphics sorts, renumbering, Pascal tutorials, and more. Both BASIC and assembly-language programs are included. Complete listings are given, with photographs, schematics, and examples.* **$9.97 BK7392 232 pp.**



TRS-80/Z80 Assembly Language Library

Learn to use assembly language on the Model I to its full capacity. Two TRSDOS-compatible disks are included, with programs worth many times the book's cost. You'll learn about TRS-80 hardware and software, general Z80 routines, and TRS-80 utility programs. Examples show you how to apply the information to your everyday programming. Model III conversions are given. $4.50 shipping and handling. **$34.97 BK7395 355 pp. Disks included**

Computer Carnival

For the Models I and III. These sixty programs for beginners will entertain and educate. Children will find mazes, word games, graphics, puzzles, and quizzes. Card games, logic tests, word and number quizzes, and letter guesses make Computer Carnival *a learning experience. The* Carnival Companion *cassette of all sixty programs is also available.* **Computer Carnival and Carnival Companion** *$24.97 CC7389* **Computer Carnival** *$16.97 BK7389 218 pp.* **Carnival Companion** *$9.97 TP7389*

TEXTEDIT

Build your own word processor with the TEXTEDIT kit. This Disk BASIC system is built in modules, so you can modify them or use only the parts you need. Features include complete editing, search, replace, and count, and upper/lowercase typing on an unmodified Model I. Model III users need the TRSDOS CONVERT utility to use the disk. TEXTEDIT is compatible with any major DOS. It operates with one drive; two drives or copy utility needed to transfer programs to system disk. **Book and disk package $24.97 CC7387**

The Selectric™ Interface

You can turn an IBM Selectric I/O writer into a letter-quality printer for your TRS-80. The Selectric Interface *gives you the programs and step-by-step instructions you need for Selectric models 2740, 2980, and Dura 1041. With slight modification, the instructions will also work for other chips.* **$12.97 BK7388 124 pp.**

Annotated BASIC, vol. 1 and 2

This two-volume set teaches you the hows and whys of BASIC programming. TRS-80 Level II programs are taken apart and described in detail. Each program is accompanied by documentation, program annotation, BASIC concepts and definitions, and a flowchart. **Volume 1** *$10.95 BK7384 160 pp.* **Volume 2** *$10.95 BK7385 125 pp.*

Kilobaud Klassroom

Learn electronics with this hand-on course. This collection of electronics projects starts with simple concepts and takes you on to building your own small computer. You'll learn electronics theory and get the practice you need to master digital electronics. **$14.95 BK7386 393 pp.**

For credit card orders, call **toll-free, 1-800-258-5473.** *Or send your order on a separate piece of paper to* **Wayne Green Books, Retail Sales, Peterborough, NH 03458.** *Be sure to include the book title, order number, and price. Postage and handling is $1.50 for the first book, $1.00 for each additional book. Foreign air mail is $10.00 per book. Check, money order, or complete credit card information must accompany your order. If you have questions about your order, write customer service at the above address.*

61

I need *TRS-80/Z80 Assembly Language Library.*

Send me _____ copies of ***TRS-80/Z80 Assembly Language Library*** BK7395-01 @ $29.97 each. I have enclosed $4.50 per system for shipping and handling.

☐ MasterCard ☐ VISA ☐ AmEx ☐ Payment Enclosed

Card #_____ Expires_____

Signature_____ Interbank #_____

Name _____

Address _____

City_____ State_____ Zip_____

☐ Please send me my **FREE 1983/84 WAYNE GREEN BOOKS RETAIL CATALOG.**

TRS-80/Z80 Assembly Language Library is a Wayne Green publication.
TRS-80, TRSDOS, Model I, Model III are registered trademarks of the Radio Shack Division of Tandy Corporation.
Z80 is a registered trademark of Zilog.
Dealer inquiries invited.

342B8ZB

BOOKS

WGBooks, Peterborough, NH 03458

Please send me the following WGBooks products:

Qty.	Catalog #	Title	Unit Price	Total

Shipping and handling charges:
$1.50 1st book, $1.00 each additional book (UPS, use street address)
$4.50 for each TRS-80 ass. lang. Library order. $10.00 each book overseas airmail.

Shipping and handling _____

Total _____

Enclosed $_____ ☐ Check ☐ M.O.

Bill: ☐ AE ☐ MC ☐ VISA

Card # _____ Exp date _____

Signature _____ Interbank # _____

Name _____

Address _____

City _____ State _____ Zip _____

No C.O.D. orders accepted. Please allow 4–6 weeks for delivery.

342B8B

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 73 PETERBOROUGH, NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE

WAYNE GREEN INC.
Attn: Retail Book Sales
Rt. 101 and Elm St.
Peterborough, NH 03458

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

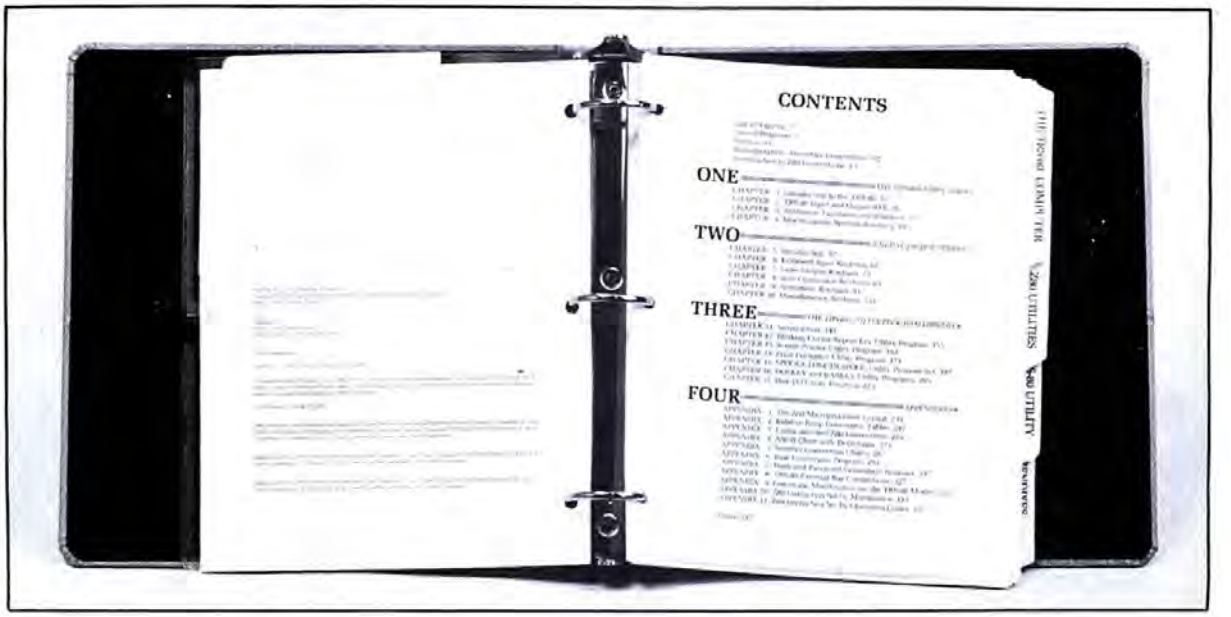
BUSINESS REPLY CARD

First Class Permit No. 73 Peterborough, NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE

WAYNE GREEN BOOKS
ATTN: RETAIL SALES
RTE. 101 AND ELM ST.
PETERBOROUGH, NH 03458

Every Z80 assembly-language programmer needs this book.



Programming in assembly language requires good tools. *TRS-80/Z80 Assembly Language Library*, a complete reference book on TRS-80 Model I assembly language, is the best tool you can find. In over 300 pages, 45 figures, and 75 program listings, author Craig A. Lindley explains the details of Model I hardware and software and shows you how to write programs that squeeze every bit of performance out of your computer. This book will teach you:

- how to use ROM and DOS routines in your own programs
- how to perform disk input/output
- how to access the video display and keyboard
- how to write easy-to-use programs
- how to perform arithmetic operations
- how to use the undocumented Z80 instructions

You'll also learn about disk operating systems, device handling, base conversion, parameter passing, and more.

TRS-80/Z80 Assembly Language Library contains a library of ready-to-run utility programs that are worth many times the book's cost. Included with the book are two Model I TRSDOS-compatible disks* containing utilities for printer formatting and spooling, single-key entry of strings, disk editing, base conversion, password encoding and decoding, and more. You get more than 15 valuable utilities, and the source code files are included, so you can examine, modify, and learn from every program.

Many of the programs also run on the Model III, and those that do not require only minor modification. One of the book's four sections is dedicated to routines and programs that run on any Z80-based system. No matter what Z80 computer you program, this book has something for you.

It's a book that's designed to be used. The text is set in large type, and the book is bound in an 8½- × 11-inch easel-backed binder that stands up next to your computer. So there's no squinting at tiny type and fighting to keep the book open.

The whole package—book, disks, and binder—is yours by mail for the **unbelievable price of only $34.97.** And if you charge it, you can even call toll-free to order.

TRS-80/Z80 Assembly Language Library. It's the assembly-language book for the '80s.

*Disks do not contain a disk operating system; two disk drives or a disk copy utility are required to transfer the files.

Call **Toll-Free 1-800-258-5473** for credit card orders or send $34.97 plus $4.50 shipping and handling to Wayne Green Books, Retail Sales, Peterborough NH 03458. Dealer inquiries invited.

TRS-80/Z80 Assembly Language Library is a Wayne Green publication
TRS-80, TRSDOS, Model I, Model III are registered trademarks of the Radio Shack Division of Tandy Corp.
Z80 is a registered trademark of Zilog

I need *TRS-80/Z80 Assembly Language Library*. ✓77

Send me _____ copies of *TRS-80/Z80 Assembly Language Library* (BK7395) @ $34.97 each. I have enclosed $4.50 per system for shipping and handling.

☐ MasterCard ☐ VISA ☐ AmEx ☐ payment enclosed

Card # _____ Expires _____

InterBank# _____ Signature _____

Name _____

Address _____

City _____ State _____ Zip _____

Wayne Green Books, Retail Sales, Peterborough, NH 03458

342B8Z

✓ See List of Advertisers on Page 211

Listing 3 continued from p. 144

```
01015 ;
01016 LR        DEFM    'L<I>I='        ;
01017           DEFB    EH1
01018           DEFB    EH3
01019           DEFB    UH3
01020           DEFB    LL
01021           DEFB    '@'
01022           DEFM    'LO>C@='        ;180
01023           DEFB    LL
01024           DEFB    UH3
01025           DEFB    O1
01026           DEFB    U1
01027           DEFB    '@'
01028           DEFM    'LO>"@='        ;
01029           DEFB    LL
01030           DEFB    O
01031           DEFB    O
01032           DEFB    '@'
01033           DEFM    'LE>I='         ;
01034           DEFB    UH3
01035           DEFB    LL
01036           DEFB    '@'
01037           DEFM    'L<L='          ;
01038           DEFB    '@'
01039           DEFM    'L<:"@>%='      ;
01040           DEFB    UH2
01041           DEFB    LL
01042           DEFB    '@'
01043           DEFM    'LEAD='         ;185
01044           DEFB    LL
01045           DEFB    E1
01046           DEFB    Y
01047           DEFB    DD
01048           DEFB    '@'
01049           DEFM    'LESS>I='       ;
01050           DEFB    LL
01051           DEFB    EH2
01052           DEFB    S
01053           DEFB    '@'
01054           DEFM    'LY>I='         ;
01055           DEFB    LL
01056           DEFB    Y
01057           DEFB    '@'
01058           DEFM    'L='            ;
01059           DEFB    LL
01060           DEFB    '@'
01061           DEFM    '['
01062 ;
01063 MR        DEFM    'M<I>I='        ;
01064           DEFB    EH1
01065           DEFB    EH2
01066           DEFB    MM
01067           DEFB    '@'
01068           DEFM    'MOV='          ;190
01069           DEFB    MM
01070           DEFB    O1
01071           DEFB    U1
01072           DEFB    V
01073           DEFB    '@'
01074           DEFM    'MENT>I='       ;
01075           DEFB    MM
01076           DEFB    EH3
01077           DEFB    N
01078           DEFB    T
01079           DEFB    '@'
01080           DEFM    'M='            ;
01081           DEFB    MM
01082           DEFB    '@'
01083           DEFB    '['
01084 ;
01085 NR        DEFM    'N<I>I='        ;
01086           DEFB    EH1
01087           DEFB    EH2
01088           DEFB    N
01089           DEFB    '@'
01090           DEFM    'NG<E>+='       ;
01091           DEFB    N
01092           DEFB    DD
01093           DEFB    J
01094           DEFB    '@'
01095           DEFM    'NG>R='         ;195
01096           DEFB    NG
01097           DEFB    G
01098           DEFB    '@'
01099           DEFM    'NG>@='         ;
01100           DEFB    NG
01101           DEFB    G
01102           DEFB    '@'
01103           DEFM    'NGL>%='        ;
01104           DEFB    NG
01105           DEFB    G
01106           DEFB    UH2
01107           DEFB    LL
01108           DEFB    '@'
01109           DEFM    'NG='           ;
01110           DEFB    NG
01111           DEFB    '@'
01112           DEFM    'NK='           ;
01113           DEFB    NG
01114           DEFB    K
01115           DEFB    '@'
01116           DEFM    'NOW<I>I='      ;200
01117           DEFB    N
01118           DEFB    AH1
01119           DEFB    UH3
01120           DEFB    U1
01121           DEFB    '@'
01122           DEFM    'NESS>I='       ;
01123           DEFB    N
01124           DEFB    EH3
01125           DEFB    S
01126           DEFB    '@'
01127           DEFM    'NON<I='        ;
01128           DEFB    N
01129           DEFB    AH1
01130           DEFB    UH3
01131           DEFB    N
01132           DEFB    '@'
01133           DEFM    'N='            ;
01134           DEFB    N
01135           DEFB    '@'
01136           DEFB    '['
01137 ;
01138 OR        DEFM    'O<I>I='        ;
01139           DEFB    O2
01140           DEFB    O1
01141           DEFB    U1
01142           DEFB    '@'
01143           DEFM    'OF>I='         ;205
01144           DEFB    UH2
01145           DEFB    V
01146           DEFB    '@'
01147           DEFM    'OROUGH='       ;
01148           DEFB    ER
01149           DEFB    IU
01150           DEFB    U1
01151           DEFB    '@'
01152           DEFM    'OR<:#>I='      ;
01153           DEFB    ER
01154           DEFB    '@'
01155           DEFM    'ORS<:#>I='     ;
01156           DEFB    ER
01157           DEFB    ZZ
01158           DEFB    '@'
01159           DEFM    'OR='           ;
01160           DEFB    O2
01161           DEFB    O2
01162           DEFB    RR
01163           DEFB    '@'
01164           DEFM    'ONE<I='        ;210
01165           DEFB    W
01166           DEFB    UH1
01167           DEFB    UH2
01168           DEFB    N
01169           DEFB    '@'
01170           DEFM    'OW<LL='        ;
01171           DEFB    AH1
01172           DEFB    UH3
01173           DEFB    U1
01174           DEFB    '@'
01175           DEFM    'OW='           ;
01176           DEFB    O1
01177           DEFB    U1
01178           DEFB    '@'
01179           DEFM    'OVER<I='       ;
01180           DEFB    O1
01181           DEFB    O2
01182           DEFB    V
01183           DEFB    ER
01184           DEFB    '@'
01185           DEFM    'OV='           ;
01186           DEFB    UH1
01187           DEFB    V
01188           DEFB    '@'
01189           DEFM    'O>"%='         ;215
01190           DEFB    O1
01191           DEFB    U1
01192           DEFB    '@'
01193           DEFM    'O>"EN='        ;
01194           DEFB    O1
01195           DEFB    U1
01196           DEFB    '@'
01197           DEFM    'O>"I@='        ;
01198           DEFB    O1
01199           DEFB    U1
01200           DEFB    '@'
01201           DEFM    'OL>D='         ;
01202           DEFB    O2
01203           DEFB    O2
01204           DEFB    LL
01205           DEFB    LL
01206           DEFB    '@'
01207           DEFM    'ON<I='         ;
01208           DEFB    UH3
01209           DEFB    N
01210           DEFB    '@'
01211           DEFM    'OUGHT='        ;220
01212           DEFB    AH1
01213           DEFB    UH3
01214           DEFB    T
01215           DEFB    '@'
01216           DEFM    'OUGH='         ;
01217           DEFB    UH
01218           DEFB    F
01219           DEFB    '@'
01220           DEFM    'OU<I='         ;
01221           DEFB    AH
01222           DEFB    O1
01223           DEFB    '@'
01224           DEFM    'OU<H>S@='      ;
01225           DEFB    AH
01226           DEFB    O1
01227           DEFB    '@'
01228           DEFM    'OUS='          ;
01229           DEFB    UH2
01230           DEFB    S
01231           DEFB    '@'
01232           DEFM    'OUR='          ;225
01233           DEFB    O
01234           DEFB    RR
01235           DEFB    '@'
01236           DEFM    'OULD='         ;
01237           DEFB    IU
01238           DEFB    IU
01239           DEFB    OO1
01240           DEFB    DD
01241           DEFB    '@'
01242           DEFM    'OU<">"L='      ;
01243           DEFB    UH
01244           DEFB    '@'
01245           DEFM    'OUP='          ;
01246           DEFB    IU
01247           DEFB    U
01248           DEFB    PP
01249           DEFB    '@'
01250           DEFM    'OU='           ;
01251           DEFB    AH
01252           DEFB    O1
01253           DEFB    '@'
01254           DEFM    'OY='           ;230
01255           DEFB    O1
01256           DEFB    E1
01257           DEFB    '@'
01258           DEFM    'OING='         ;
01259           DEFB    O1
01260           DEFB    U1
01261           DEFB    II
01262           DEFB    NG
01263           DEFB    '@'
01264           DEFM    'OI='           ;
01265           DEFB    O1
01266           DEFB    UH3
01267           DEFB    I3
01268           DEFB    AY
01269           DEFB    '@'
01270           DEFM    'OOR='          ;
01271           DEFB    O
01272           DEFB    RR
01273           DEFB    '@'
01274           DEFM    'OOK='          ;
01275           DEFB    OO
01276           DEFB    K
01277           DEFB    '@'
01278           DEFM    'OOD='          ;235
01279           DEFB    OO
01280           DEFB    DD
01281           DEFB    '@'
01282           DEFM    'OO='           ;
01283           DEFB    IU
01284           DEFB    U
01285           DEFB    '@'
01286           DEFM    'O>E='          ;
01287           DEFB    O1
01288           DEFB    U1
01289           DEFB    '@'
01290           DEFM    'O>I='          ;
01291           DEFB    O1
01292           DEFB    U1
01293           DEFB    '@'
01294           DEFM    'OA='           ;
01295           DEFB    O1
01296           DEFB    U1
01297           DEFB    '@'
01298           DEFM    'ONLY<I='       ;240
01299           DEFB    O1
01300           DEFB    U1
01301           DEFB    N
01302           DEFB    LL
01303           DEFB    EE
01304           DEFB    '@'
01305           DEFM    'ONCE<I='       ;
01306           DEFB    W
01307           DEFB    UH
01308           DEFB    N
01309           DEFB    S
01310           DEFB    '@'
01311           DEFM    'ON'            ;
01312           DEFB    39              ;'
01313           DEFM    'T='
01314           DEFB    O1
01315           DEFB    U1
01316           DEFB    N
01317           DEFB    T
01318           DEFB    '@'
01319           DEFM    'O<C>N='        ;
01320           DEFB    AH
01321           DEFB    '@'
01322           DEFM    'O>NG='         ;
01323           DEFB    AW
01324           DEFB    '@'
01325           DEFM    'O<:"I>N='      ;245
01326           DEFB    UH
01327           DEFB    '@'
01328           DEFM    'ON<I='         ;
01329           DEFB    UH2
01330           DEFB    N
01331           DEFB    '@'
01332           DEFM    'ON<:#>I='      ;
01333           DEFB    UH2
01334           DEFB    N
01335           DEFB    '@'
01336           DEFM    'ON<"@='        ;
01337           DEFB    UH2
01338           DEFB    N
01339           DEFB    '@'
01340           DEFM    'O>ST!='        ;
01341           DEFB    O1
01342           DEFB    U1
01343           DEFB    '@'
01344           DEFM    'OF>"='         ;250
01345           DEFB    AW
01346           DEFB    F
01347           DEFB    '@'
01348           DEFM    'OTHER='        ;
01349           DEFB    UH
01350           DEFB    THV
01351           DEFB    ER
01352           DEFB    '@'
01353           DEFM    'OSS>I='        ;
01354           DEFB    AW
01355           DEFB    S
01356           DEFB    '@'
01357           DEFM    'OM<:"@='       ;
01358           DEFB    UH
```

Listing 3 continued

See List of Advertisers on Page 211

Listing 3 continued

```
01359          DEFB    MM
01360          DEFB    '@'
01361          DEFM    'O='            ;
01362          DEFB    AH
01363          DEFB    '@'
01364          DEFB    '['
01365 ;
01366 PR       DEFM    'P<I>1='        ;255
01367          DEFB    PP
01368          DEFB    El
01369          DEFB    Y
01370          DEFB    '@'
01371          DEFM    'PH='           ;
01372          DEFB    F
01373          DEFB    '@'
01374          DEFM    'PEOP='         ;
01375          DEFB    PP
01376          DEFB    EE
01377          DEFB    PP
01378          DEFB    '@'
01379          DEFM    'POW='          ;
01380          DEFB    PP
01381          DEFB    AH
01382          DEFB    O1
01383          DEFB    '@'
01384          DEFM    'PUT>1='        ;
01385          DEFB    PP
01386          DEFB    OO
01387          DEFB    T
01388          DEFB    '@'
01389          DEFM    'P='            ;260
01390          DEFB    PP
01391          DEFB    '@'
01392          DEFB    '['
01393 ;
01394 QR       DEFM    'Q<I>1='        ;
01395          DEFB    K
01396          DEFB    Y1
01397          DEFB    IU
01398          DEFB    U1
01399          DEFB    U1
01400          DEFB    '@'
01401          DEFM    'QUAR='         ;
01402          DEFB    K
01403          DEFB    W
01404          DEFB    O
01405          DEFB    RR
01406          DEFB    '@'
01407          DEFM    'QU='           ;
01408          DEFB    K
01409          DEFB    W
01410          DEFB    '@'
01411          DEFM    'Q='            ;
01412          DEFB    K
01413          DEFB    '@'
01414          DEFM    '['
01415 ;
01416 RRULES   DEFM    'R<I>1='        ;265
01417          DEFB    AH1
01418          DEFB    UH2
01419          DEFB    ER
01420          DEFB    '@'
01421          DEFM    'RE<I>"#='      ;
01422          DEFB    RR
01423          DEFB    EE
01424          DEFB    '@'
01425          DEFM    'RING='         ;
01426          DEFB    RR
01427          DEFB    I2
01428          DEFB    NG
01429          DEFB    '@'
01430          DEFM    'RI='           ;
01431          DEFB    RR
01432          DEFB    AH1
01433          DEFB    EH3
01434          DEFB    I3
01435          DEFB    Y
01436          DEFB    '@'
01437          DEFM    'R='            ;
01438          DEFB    RR
01439          DEFB    '@'
01440          DEFM    '['
01441 ;
01442 SR       DEFM    'S<I>1='        ;270
01443          DEFB    EH1
01444          DEFB    EH2
01445          DEFB    S
01446          DEFB    '@'
01447          DEFM    'SH='           ;
01448          DEFB    SH
01449          DEFB    '@'
01450          DEFM    'SION<#='       ;
01451          DEFB    ZH
01452          DEFB    UH2
01453          DEFB    N
01454          DEFB    '@'
01455          DEFM    'SOME='         ;
01456          DEFB    S
01457          DEFB    UH
01458          DEFB    MM
01459          DEFB    '@'
01460          DEFM    'SUR<#>#='      ;
01461          DEFB    ZH
01462          DEFB    ER
01463          DEFB    '@'
01464          DEFM    'SUR>#='        ;275
01465          DEFB    SH
01466          DEFB    ER
01467          DEFB    '@'
01468          DEFM    'SU<#>#='       ;
01469          DEFB    ZH
01470          DEFB    IU
01471          DEFB    U
01472          DEFB    '@'
01473          DEFM    'SSU<#>#='      ;
01474          DEFB    SH
01475          DEFB    IU
01476          DEFB    U
01477          DEFB    '@'
01478          DEFM    'SED<#>1='      ;
01479          DEFB    ZZ
01480          DEFB    DD
01481          DEFB    '@'
01482          DEFM    'S<#>#='        ;
01483          DEFB    ZZ
01484          DEFB    '@'
01485          DEFM    'SAID='         ;280
01486          DEFB    S
01487          DEFB    EH
01488          DEFB    DD
01489          DEFB    '@'
01490          DEFM    'SION<"='       ;
01491          DEFB    SH
01492          DEFB    UH2
01493          DEFB    N
01494          DEFB    '@'
01495          DEFM    'S>S='          ;
01496          DEFB    '@'
01497          DEFM    'S<.>1='        ;
01498          DEFB    ZZ
01499          DEFB    '@'
01500          DEFM    'S<E.:#>1='     ;
01501          DEFB    ZZ
01502          DEFB    '@'
01503          DEFM    'S<##:"#>1='    ;285
01504          DEFB    ZZ
01505          DEFB    '@'
01506          DEFM    'S<#:"#>1='     ;
01507          DEFB    S
01508          DEFB    '@'
01509          DEFM    'S<U>1='        ;
01510          DEFB    S
01511          DEFB    '@'
01512          DEFM    'S<#:I>1='      ;
01513          DEFB    ZZ
01514          DEFB    '@'
01515          DEFM    'SCH<1='        ;
01516          DEFB    S
01517          DEFB    K
01518          DEFB    '@'
01519          DEFM    'S>C+='         ;290
01520          DEFB    '@'
01521          DEFM    'SM<#='         ;
01522          DEFB    ZZ
01523          DEFB    MM
01524          DEFB    '@'
01525          DEFM    'SN<#>1='       ;
01526          DEFB    ZZ
01527          DEFB    UH2
01528          DEFB    N
01529          DEFB    '@'
01530          DEFM    'S='            ;
01531          DEFB    S
01532          DEFB    '@'
01533          DEFB    '['
01534 ;
01535 TR       DEFM    'T<I>1='        ;
01536          DEFB    T
01537          DEFB    El
01538          DEFB    AY
01539          DEFB    Y
01540          DEFB    '@'
01541          DEFM    'THE<I>1='      ;295
01542          DEFB    THV
01543          DEFB    UH2
01544          DEFB    '@'
01545          DEFM    'TO>!='         ;
01546          DEFB    T
01547          DEFB    IU
01548          DEFB    U
01549          DEFB    '@'
01550          DEFM    'THAT>1='       ;
01551          DEFB    THV
01552          DEFB    AE
01553          DEFB    T
01554          DEFB    '@'
01555          DEFM    'THIS<I>1='     ;
01556          DEFB    THV
01557          DEFB    II
01558          DEFB    S
01559          DEFB    '@'
01560          DEFM    'THEY<1='       ;
01561          DEFB    THV
01562          DEFB    AA
01563          DEFB    AY
01564          DEFB    '@'
01565          DEFM    'THERE<1='      ;300
01566          DEFB    THV
01567          DEFB    EH
01568          DEFB    RR
01569          DEFB    '@'
01570          DEFM    'THER='         ;
01571          DEFB    THV
01572          DEFB    ER
01573          DEFB    '@'
01574          DEFM    'THEIR='        ;
01575          DEFB    THV
01576          DEFB    EH
01577          DEFB    RR
01578          DEFB    '@'
01579          DEFM    'THAN<I>1='     ;
01580          DEFB    THV
01581          DEFB    AE
01582          DEFB    N
01583          DEFB    '@'
01584          DEFM    'THEM<I>1='     ;
01585          DEFB    THV
01586          DEFB    EH
01587          DEFB    MM
01588          DEFB    '@'
01589          DEFM    'THESE>1='      ;305
01590          DEFB    THV
01591          DEFB    EE
01592          DEFB    ZZ
01593          DEFB    '@'
01594          DEFM    'THEN<1='       ;
01595          DEFB    THV
01596          DEFB    EH
01597          DEFB    N
01598          DEFB    '@'
01599          DEFM    'THROUGH='      ;
01600          DEFB    TH
01601          DEFB    RR
01602          DEFB    IU
01603          DEFB    U
01604          DEFB    '@'
01605          DEFM    'THOSE='        ;
01606          DEFB    THV
01607          DEFB    O1
01608          DEFB    U1
01609          DEFB    ZZ
01610          DEFB    '@'
01611          DEFM    'THOUGH>1='     ;
01612          DEFB    THV
01613          DEFB    O1
01614          DEFB    U1
01615          DEFB    '@'
01616          DEFM    'THUS<1='       ;310
01617          DEFB    THV
01618          DEFB    UH
01619          DEFB    S
01620          DEFB    '@'
01621          DEFM    'TH='           ;
01622          DEFB    TH
01623          DEFB    '@'
01624          DEFM    'TED<:#>1='     ;
01625          DEFB    T
01626          DEFB    II
01627          DEFB    DD
01628          DEFB    '@'
01629          DEFM    'TI<S>#N='      ;
01630          DEFB    T
01631          DEFB    CH
01632          DEFB    '@'
01633          DEFM    'TI>O='         ;
01634          DEFB    SH
01635          DEFB    '@'
01636          DEFM    'TI>A='         ;315
01637          DEFB    SH
01638          DEFB    '@'
01639          DEFM    'TIEN='         ;
01640          DEFB    SH
01641          DEFB    UH2
01642          DEFB    N
01643          DEFB    '@'
01644          DEFM    'TUR>#='        ;
01645          DEFB    T
01646          DEFB    CH
01647          DEFB    ER
01648          DEFB    '@'
01649          DEFM    'TU>A='         ;
01650          DEFB    T
01651          DEFB    CH
01652          DEFB    IU
01653          DEFB    U
01654          DEFB    '@'
01655          DEFM    'TWO<1='        ;
01656          DEFB    T
01657          DEFB    IU
01658          DEFB    U
01659          DEFB    '@'
01660          DEFM    'T='            ;320
01661          DEFB    T
01662          DEFB    '@'
01663          DEFB    '['
01664 ;
01665 UR       DEFM    'R<I>1='        ;
01666          DEFB    Y1
01667          DEFB    IU
01668          DEFB    U1
01669          DEFB    U1
01670          DEFB    '@'
01671          DEFM    'UN<I>I='       ;
01672          DEFB    Y
01673          DEFB    IU
01674          DEFB    U
01675          DEFB    N
01676          DEFB    '@'
01677          DEFM    'UN<I='         ;
01678          DEFB    UH
01679          DEFB    N
01680          DEFB    '@'
01681          DEFM    'UPON<I='       ;
01682          DEFB    UH2
01683          DEFB    PP
01684          DEFB    AW
01685          DEFB    N
01686          DEFB    '@'
01687          DEFM    'UR<$>#='       ;325
01688          DEFB    OO
01689          DEFB    RR
01690          DEFB    '@'
01691          DEFM    'UR>#='         ;
01692          DEFB    Y
01693          DEFB    OO
01694          DEFB    RR
01695          DEFB    '@'
01696          DEFM    'UR='           ;
01697          DEFB    ER
01698          DEFB    '@'
01699          DEFM    'US>!='         ;
01700          DEFB    UH1
```

Listing 3 continued

See List of Advertisers on Page 211

Listing 3 continued

```
01701           DEFB    UH2
01702           DEFB    S
01703           DEFB    '@'
01704           DEFM    'UE<">I='       ;
01705           DEFB    IU
01706           DEFB    U1
01707           DEFB    U1
01708           DEFB    '@'
01709           DEFM    'U>"I='         ;330
01710           DEFB    UH
01711           DEFB    '@'
01712           DEFM    'U>""='         ;
01713           DEFB    UH
01714           DEFB    '@'
01715           DEFM    'UY='           ;
01716           DEFB    AH
01717           DEFB    E1
01718           DEFB    '@'
01719           DEFM    'U<G!>#='       ;
01720           DEFB    '@'
01721           DEFM    'U<G>%='        ;
01722           DEFB    '@'
01723           DEFM    'U<G>#='        ;335
01724           DEFB    W
01725           DEFB    '@'
01726           DEFM    'U<N#='         ;
01727           DEFB    Y
01728           DEFB    IU
01729           DEFB    U
01730           DEFB    '@'
01731           DEFM    'U<S='          ;
01732           DEFB    IU
01733           DEFB    U
01734           DEFB    '@'
01735           DEFM    'U='            ;
01736           DEFB    Y
01737           DEFB    IU
01738           DEFB    U
01739           DEFB    '@'
01740           DEFM    '['
01741 ;
01742 VR        DEFM    'V<I>I='        ;
01743           DEFB    V
01744           DEFB    E1
01745           DEFB    AY
01746           DEFB    Y
01747           DEFB    '@'
01748           DEFM    'VIEW='         ;340
01749           DEFB    V
01750           DEFB    Y
01751           DEFB    IU
01752           DEFB    U
01753           DEFB    '@'
01754           DEFM    'VERI<I='       ;
01755           DEFB    V
01756           DEFB    EH1
01757           DEFB    RR
01758           DEFB    I3
01759           DEFB    '@'
01760           DEFM    'VE='           ;
01761           DEFB    V
01762           DEFB    '@'
01763           DEFM    'VAL<I='        ;
01764           DEFB    V
01765           DEFB    AE1
01766           DEFB    EH3
01767           DEFB    LL
01768           DEFB    '@'
01769           DEFM    'V='            ;
01770           DEFB    V
01771           DEFB    '@'
01772           DEFM    '['
01773 ;
01774 WR        DEFM    'W<I>I='        ;345
01775           DEFB    DD
01776           DEFB    UH1
01777           DEFB    BB
01778           DEFB    UH3
01779           DEFB    LL
01780           DEFB    Y1
01781           DEFB    IU
01782           DEFB    U1
01783           DEFB    '@'
01784           DEFM    'WERE<I='       ;
01785           DEFB    W
01786           DEFB    ER
01787           DEFB    '@'
01788           DEFM    'WA>S='         ;
01789           DEFB    W
01790           DEFB    AH
01791           DEFB    '@'
01792           DEFM    'WA>T='         ;
01793           DEFB    W
01794           DEFB    AH
01795           DEFB    '@'
01796           DEFM    'WHERE='        ;
01797           DEFB    HH
01798           DEFB    W
01799           DEFB    EH
01800           DEFB    RR
01801           DEFB    '@'
01802           DEFM    'WHAT='         ;350
01803           DEFB    W
01804           DEFB    UH3
01805           DEFB    UH1
01806           DEFB    T
01807           DEFB    '@'
01808           DEFM    'WHOL='         ;
01809           DEFB    HH
01810           DEFB    O1
01811           DEFB    U1
01812           DEFB    LL
01813           DEFB    '@'
01814           DEFM    'WHO='          ;
01815           DEFB    HH
01816           DEFB    IU
01817           DEFB    U
01818           DEFB    '@'
01819           DEFM    'WH='           ;
01820           DEFB    HH
01821           DEFB    W
01822           DEFB    '@'
01823           DEFM    'WAR='          ;
01824           DEFB    W
01825           DEFB    O
01826           DEFB    RR
01827           DEFB    '@'
01828           DEFM    'WOR>"='        ;355
01829           DEFB    W
01830           DEFB    ER
01831           DEFB    '@'
01832           DEFM    'WR='           ;
01833           DEFB    RR
01834           DEFB    '@'
01835           DEFM    'W='            ;
01836           DEFB    W
01837           DEFB    '@'
01838           DEFB    '['
01839 ;
01840 XR        DEFM    'X<I>I='        ;
01841           DEFB    EH1
01842           DEFB    EH2
01843           DEFB    K
01844           DEFB    PA0
01845           DEFB    S
01846           DEFB    '@'
01847           DEFM    'X<I='          ;
01848           DEFB    EH1
01849           DEFB    EH2
01850           DEFB    K
01851           DEFB    PA0
01852           DEFB    S
01853           DEFB    '@'
01854           DEFM    'X='            ;360
01855           DEFB    K
01856           DEFB    S
01857           DEFB    '@'
01858           DEFM    '['
01859 ;
01860 YR        DEFM    'Y<I>I='        ;
01861           DEFB    W
01862           DEFB    AH1
01863           DEFB    EH3
01864           DEFB    I3
01865           DEFB    Y
01866           DEFB    '@'
01867           DEFM    'YOUNG='        ;
01868           DEFB    Y
01869           DEFB    UH
01870           DEFB    NG
01871           DEFB    '@'
01872           DEFM    'YOU<I='        ;
01873           DEFB    Y
01874           DEFB    IU
01875           DEFB    U
01876           DEFB    '@'
01877           DEFM    'YES<I='        ;
01878           DEFB    Y
01879           DEFB    EH
01880           DEFB    S
01881           DEFB    '@'
01882           DEFM    'Y<!='          ;365
01883           DEFB    Y
01884           DEFB    '@'
01885           DEFM    'Y<:"#>I='      ;
01886           DEFB    EE
01887           DEFB    '@'
01888           DEFM    'Y<:"#>I='      ;
01889           DEFB    EE
01890           DEFB    '@'
01891           DEFM    'Y<:I>I='       ;
01892           DEFB    AH
01893           DEFB    E1
01894           DEFB    '@'
01895           DEFM    'Y<:I>#='       ;
01896           DEFB    AH
01897           DEFB    E1
01898           DEFB    '@'
01899           DEFM    'Y<:I>"+:#='    ;370
01900           DEFB    II
01901           DEFB    '@'
01902           DEFM    'Y<:I>"#='      ;
01903           DEFB    AH
01904           DEFB    E1
01905           DEFB    '@'
01906           DEFM    'Y='            ;
01907           DEFB    II
01908           DEFB    '@'
01909           DEFB    '['
01910 ;
01911 ZR        DEFM    'Z<I>I='        ;
01912           DEFB    ZZ
01913           DEFB    E1
01914           DEFB    Y
01915           DEFB    '@'
01916           DEFM    'Z='            ;
01917           DEFB    ZZ
01918           DEFB    '@'
01919           DEFB    '['
01920 ;
01921 BLANK     DEFM    ' ='            ;375
01922           DEFB    PA0
01923           DEFB    '@'
01924           DEFB    '['
01925 ;
01926 PERIOD    DEFM    '.='            ;
01927           DEFB    PA1
01928           DEFB    '@'
01929           DEFB    '['
01930 ;
01931 APOST     DEFB    39              ;
01932           DEFM    '<I>I='
01933           DEFB    PA0
01934           DEFB    '@'
01935           DEFB    39              ;
01936           DEFM    'S>I='
01937           DEFB    II
01938           DEFB    ZZ
01939           DEFB    '@'
01940           DEFB    39              ;
01941           DEFM    'S<I.='
01942           DEFB    ZZ
01943           DEFB    '@'
01944           DEFB    39              ;380
01945           DEFM    'S<E.:#='
01946           DEFB    ZZ
01947           DEFB    '@'
01948           DEFB    39              ;
01949           DEFM    'S<I#='
01950           DEFB    ZZ
01951           DEFB    '@'
01952           DEFB    39              ;
01953           DEFM    ' ='
01954           DEFB    PA0
01955           DEFB    '@'
01956           DEFB    '['
01957 ;
01958 COMMA     DEFM    ',='            ;
01959           DEFB    PA1
01960           DEFB    '@'
01961           DEFB    '['
01962 ;
01963 QUEST     DEFM    '?='            ;
01964           DEFB    PA1
01965           DEFB    '@'
01966           DEFB    '['
01967 ;
01968 ZERO      DEFM    '0='            ;385
01969           DEFB    ZZ
01970           DEFB    AY
01971           DEFB    I1
01972           DEFB    RR
01973           DEFB    O1
01974           DEFB    U1
01975           DEFB    '@'
01976           DEFB    '['
01977 ;
01978 R1        DEFM    '1='            ;
01979           DEFB    W
01980           DEFB    UH1
01981           DEFB    UH2
01982           DEFB    N
01983           DEFB    '@'
01984           DEFB    '['
01985 ;
01986 R2        DEFM    '2='            ;
01987           DEFB    T
01988           DEFB    IU
01989           DEFB    U1
01990           DEFB    U1
01991           DEFB    '@'
01992           DEFB    '['
01993 ;
01994 R3        DEFM    '3='            ;
01995           DEFB    TH
01996           DEFB    RR
01997           DEFB    E1
01998           DEFB    Y
01999           DEFB    '@'
02000           DEFB    '['
02001 ;
02002 R4        DEFM    '4='            ;
02003           DEFB    F
02004           DEFB    O1
02005           DEFB    O2
02006           DEFB    RR
02007           DEFB    '@'
02008           DEFB    '['
02009 ;
02010 R5        DEFM    '5='            ;390
02011           DEFB    F
02012           DEFB    AH1
02013           DEFB    EH3
02014           DEFB    Y
02015           DEFB    V
02016           DEFB    '@'
02017           DEFB    '['
02018 ;
02019 R6        DEFM    '6='            ;
02020           DEFB    S
02021           DEFB    I1
02022           DEFB    I3
02023           DEFB    K
02024           DEFB    PA0
02025           DEFB    S
02026           DEFB    '@'
02027           DEFB    '['
02028 ;
02029 R7        DEFM    '7='            ;
02030           DEFB    S
02031           DEFB    EH1
02032           DEFB    EH3
02033           DEFB    V
02034           DEFB    I2
02035           DEFB    N
02036           DEFB    '@'
02037           DEFB    '['
02038 ;
02039 R8        DEFM    '8='            ;
02040           DEFB    A2
02041           DEFB    A2
02042           DEFB    Y
02043           DEFB    T
```

Listing 3 continued

The THOROUGHBRED HANDICAPPER

When was the last time you had a good day at the track?

The THOROUGHBRED HANDICAPPER could make it your next and, except for changing your definition of good, every one thereafter.

1. Regularly and consistently picks 55-60% winners
2. TELLS YOU **WHY** IT PICKS A HORSE OVER THE FIELD
3. Produces money management betting formula
4. Projects probability and favourable odds
5. Rates on Class, Speed, Form, Jockey, plus
6. Winning GIMIX formulas

Requires: TRS-80 I, III 32K & Disc — Mod IV, Apple, IBM soon
Intro Price til 15 April, '84 — $90. — Manual alone $20 (guarantees intro)

Send check or M.O. to: **Ed Ball, B.A., B.Com.** ✓249 **Box 2063 Detroit, MI 48231**

NEW

ZOLAR ©

By Ross Virostko

You must conquer the evil warlord, Zolar. Break through his rotating walls. Destroy his relentless servants. Open a gap, and eliminate Zolar, before Zolar eliminates you.

2 player, machine language, sound, joystick compatible.

Model I & III
16K Tape 19.95
32K Disk 22.95

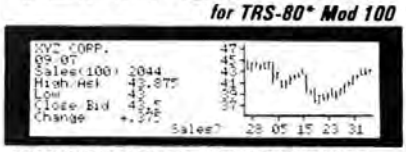
The Software Exchange
1615 Compton Road
Cleveland Heights, OH 44118
✓28

Add $1.00 per item for S & H. Overseas, FPO, APO: Add 10%. Ohio residents add sales tax. Include check or money order. COD: Add $2.00.
DEALERS & AUTHORS INVITED

INTRODUCING

TELE-STOCK™

for TRS-80 Mod 100*



STOCK MARKET DECISION ASSISTANCE

12 to 24 hours ahead of the newspaper at less than the cost of the newspaper. Your Model 100 automatically calls the Dow Jones News Service immediately following the close of the market each day, retrieves the market activity for your investments (it takes about 1 minute and cost 15-30 cents), then displays and prints for each investment:

- ***Todays Activity***
- ***45 day historical graph showing HIGH, LOW, CLOSE, VOLUME***
- ***Return on Investment***

The cost of the daily service, this software and even your TRS-80 may be tax deductible when used for this purpose (consult your tax advisor for details).

IMPROVE YOUR INVESTMENT PERFORMANCE

for only $59.95 (Ca. residents add 6% tax) on cassette with manual.
Send check, money order or VISA/MC number with expiration date to:

TELESOFT™
P.O. Box 6398
Thousand Oaks, Ca. 91359 ✓15
or call 805/499-6271

*Requires 24K Model 100 with direct connect or acoustic cable. *TRS-80 is a trademark of the Tandy Corporation.

Grammar & Style Checking for Superscripsit

Checks for 22 categories of errors, including wordy phrases, cliches, repeated words, passive voice, commonly misused words, and double negatives. Suggests improvements. Analyzes average word, sentence and paragraph lengths. Can be run from within Superscripsit and/or Electric Webster.

Also available for Newscript, Lazy Writer, Scripsit and others. For information about Electric Webster, see ad p.

Grammar & Style (Stand Alone)	$69.95
Grammar & Style (for Electric Webster)	$49.95
Correcting Electric Webster (Spelling Checker)	$149.95

Professional Software at an Affordable Price!

VolksTax

$39.95

Personal Tax Writer/Planner

As easy to use as a pencil. Displays IRS forms on the screen. You insert your income information: VolksTax fills in the blanks. Calculates wherever necessary, displays percentages of gross, income averages, and much more. Fills out 1040, SE, A, B, C, D. Other forms and schedules optional. Available for all 48K and 64K computers. Additional schedules available.

	VolksTax	$39.95
Also Available	Accounts Receivable	$39.95
	Payroll	$39.95

CORNUCOPIA SOFTWARE
P.O. BOX 6111, ALBANY, CA 94706
(415) 524-8098
✓45

See ad page 151

✓ See List of Advertisers on Page 211

Listing 3 continued

```
02044           DEFB    '@'
02045           DEFB    '['
02046 ;
02047 R9        DEFM    '9='            ;
02048           DEFB    N
02049           DEFB    AH1
02050           DEFB    EH3
02051           DEFB    Y
02052           DEFB    N
02053           DEFB    '@'
02054           DEFB    '['
02055 ;
02056 MINUS     DEFM    '-='            ;395
02057           DEFB    MM
02058           DEFB    AH1
02059           DEFB    Y
02060           DEFB    N
02061           DEFB    EH3
02062           DEFB    S
02063           DEFB    '@'
02064           DEFB    '['
02065 ;
02066 PLUS      DEFM    '+='            ;
02067           DEFB    PP
02068           DEFB    LL
02069           DEFB    UH1
02070           DEFB    UH2
02071           DEFB    S
02072           DEFB    '@'
02073           DEFB    '['
02074 ;
02075 EQUALS    DEFB    '='             ;
02076           DEFB    '='
02077           DEFB    Y
02078           DEFB    K
02079           DEFB    W
02080           DEFB    UH3
02081           DEFB    LL
02082           DEFB    ZZ
02083           DEFB    '@'
02084           DEFB    '['
02085 ;
02086 MULT      DEFM    '*='            ;
02087           DEFB    T
02088           DEFB    AH1
02089           DEFB    EH3
02090           DEFB    Y
02091           DEFB    MM
02092           DEFB    ZZ
02093           DEFB    '@'
02094           DEFB    '['
02095 ;
02096 DIVIDE    DEFM    '/='            ;
02097           DEFB    DD
02098           DEFB    I1
02099           DEFB    V
02100           DEFB    AH1
02101           DEFB    EH3
02102           DEFB    Y
02103           DEFB    DD
02104           DEFB    IZ
02105           DEFB    DD
02106           DEFB    BB
02107           DEFB    AH1
02108           DEFB    EH3
02109           DEFB    I3
02110           DEFB    Y
02111           DEFB    '@'
02112           DEFB    '['
02113 ;
02114 LARROW    DEFB    8               ;400
02115           DEFB    '='
02116           DEFB    BB
02117           DEFB    AE1
02118           DEFB    AE1
02119           DEFB    K
02120           DEFB    S
02121           DEFB    PP
02122           DEFB    A1
02123           DEFB    AY
02124           DEFB    Y
02125           DEFB    S
02126           DEFB    '@'
02127           DEFB    '['
02128 ;
02129 RARROW    DEFB    9               ;
02130           DEFB    '='
02131           DEFB    RR
02132           DEFB    UH3
02133           DEFB    AH2
02134           DEFB    Y
02135           DEFB    T
02136           DEFB    PA1
02137           DEFB    EH1
02138           DEFB    EH3
02139           DEFB    RR
02140           DEFB    O1
02141           DEFB    U1
02142           DEFB    '@'
02143           DEFB    '['
02144 ;
02145 UARROW    DEFB    91              ;
02146           DEFB    '='
02147           DEFB    UH1
02148           DEFB    UH2
02149           DEFB    PP
02150           DEFB    EH1
02151           DEFB    EH3
02152           DEFB    RR
02153           DEFB    O1
02154           DEFB    U1
02155           DEFB    '@'
02156           DEFB    '['
02157 ;
02158 DARROW    DEFB    10              ;
02159           DEFB    '='
02160           DEFB    DD
02161           DEFB    AH1
02162           DEFB    UH3
02163           DEFB    U1
02164           DEFB    N
02165           DEFB    EH1
02166           DEFB    EH3
02167           DEFB    RR
02168           DEFB    O1
02169           DEFB    U1
02170           DEFB    '@'
02171           DEFB    '['
02172 ;
02173 CLEAR     DEFB    31              ;
02174           DEFB    '='
02175           DEFB    K
02176           DEFB    LL
02177           DEFB    AY
02178           DEFB    I3
02179           DEFB    RR
02180           DEFB    '@'
02181           DEFB    '['
02182 ;
02183 LTHAN     DEFM    '<='            ;405
02184           DEFB    LL
02185           DEFB    EH1
02186           DEFB    EH3
02187           DEFB    S
02188           DEFB    THV
02189           DEFB    EH1
02190           DEFB    EH3
02191           DEFB    N
02192           DEFB    '@'
02193           DEFB    '['
02194 ;
02195 GTHAN     DEFM    '>='            ;
02196           DEFB    G
02197           DEFB    RR
02198           DEFB    A1
02199           DEFB    Y
02200           DEFB    T
02201           DEFB    ER
02202           DEFB    THV
02203           DEFB    EH1
02204           DEFB    EH3
02205           DEFB    N
02206           DEFB    '@'
02207           DEFB    '['
02208 ;
02209 COLON     DEFM    ':='            ;
02210           DEFB    PA0
02211           DEFB    '@'
02212           DEFB    '['
02213 ;
02214 SEMICO    DEFM    ';='            ;
02215           DEFB    PA0
02216           DEFB    '@'
02217           DEFB    '['
02218 ;
02219 EXCLA     DEFM    '!='            ;
02220           DEFB    PA0
02221           DEFB    '@'
02222           DEFB    '['
02223 ;
02224 QUOTE     DEFM    '"='            ;410
02225           DEFB    PA0
02226           DEFB    '@'
02227           DEFB    '['
02228 ;
02229 DOLLAR    DEFM    '$='            ;
02230           DEFB    DD
02231           DEFB    AH1
02232           DEFB    UH3
02233           DEFB    LL
02234           DEFB    ER
02235           DEFB    '@'
02236           DEFB    '['
02237 ;
02238 PERCNT    DEFM    '%='            ;
02239           DEFB    PP
02240           DEFB    ER
02241           DEFB    S
02242           DEFB    EH1
02243           DEFB    EH3
02244           DEFB    N
02245           DEFB    T
02246           DEFB    '@'
02247           DEFB    '['
02248 ;
02249 ANDD      DEFM    '&='            ;
02250           DEFB    AE1
02251           DEFB    EH3
02252           DEFB    N
02253           DEFB    DD
02254           DEFB    '@'
02255           DEFB    '['
02256 ;
02257 LBRACK    DEFM    '(='            ;
02258           DEFB    PA0
02259           DEFB    '@'
02260           DEFB    '['
02261 ;
02262 RBRACK    DEFM    ')='            ;415
02263           DEFB    PA0
02264           DEFB    '@'
02265           DEFB    '['
02266 ;
02267 ATSYM     DEFM    '@='            ;416
02268           DEFB    PA0
02269           DEFB    '@'
02270           DEFB    '['
02271 ;
02272           END     CHTBLE
```

✓ See List of Advertisers on Page 211

Got a great new program up your sleeve?

If you write programs, we need each other.
We know that some great software is being developed between the hours of midnight and 6:00 AM, in the silence of America's homes. If you're one of those disk-driven writers, buttoning up by day and hunkering down by night, take heart. Fame and fortune could be right around the corner.

You'll hear from us in 30 days.
I/O WARE is looking for innovative programs for the home and small business markets: word processing, graphic arts, finance, planning, home budgeting, "how-to", etc. Programs that will run on the Commodore 64 (Disk or Tape); IBM PC and PC Jr. (Disk); Apple II (Disk); TRS-80 I, III, IV (Disk), and Color Computer (Disk or Tape). You'll receive our decision in 30 days. If you have what we're looking for we'll immediately send you an advance of *at least* $250.

And that's just the beginning. I/O WARE can give you the kind of exposure (and royalties) you've been dreaming of.

We'll put a cash advance in your pocket.

The I/O WARE Professionals: Duane Manseau, Jim Eastman, Tom Cullity
"We're looking for great new software."

See your name up in bytes.
Every program needs professional documentation, packaging, distribution, and promotion. That's our job. We'll transform your program into a polished, and successful software package.

I/O WARE is part of the team of computer experts publishing the leading computer magazines in the country. Which means your program will have access to major national advertising, direct mail, and hundreds of retail outlets across the country.

So hit us with your best shot—today.
Because we're putting together a very strong line of software. Fast. And there's a good chance you could be a part of the team. Just call Tom Cullity at our software hotline, 603/924-9897, and let us know what you have up your sleeve.

I/O ware

I/O WARE INC.
Peterborough, NH 03458
Attention: Tom Cullity

205

✓ See List of Advertisers on Page 211

Borderline Case

by Dan Keen and Dave Dischert

Create an Assembly-language border routine that runs independently of your other programs.

Though you may not realize it, or take it for granted if you do, a master control program runs your computer. Whether you're programming in Basic or waiting for a TRSDOS READY prompt, the control program located in ROM is in charge, overseeing all the computer's operations.

You can take advantage of this feature to perform background tasks while you execute other, independently operating, programs. Our Model I/III machine-language program (see the Program Listing) creates an animated border around your screen similar to that seen on a movie marquee. Once loaded, it executes autonomously—flashing hypnotically while you type in a program or play a game.

How It Works

The TRS-80's periodic checking of the device control block (DCB) in memory makes such simultaneous execution possible.

Our program detours the computer's normally scheduled pass through the keyboard DCB, branching execution to the short machine-language routine loaded into RAM. After executing the routine, the program passes control back to the DCB and the original address stored there.

The keyboard addresses of interest are located at 4016 and 4017 hexadecimal (hex) in the Models I and III. Together they are called a vector. These locations point to the memory address where a keyboard debounce or other routine resides.

The Marquee routine is divided into two parts. The first part of the program is an initialization section establishing tables and buffer areas for screen graphics. The second part is the actual Marquee routine. By using two ORGs we were able to keep a few extra bytes between the sections available for future use. The first ORG statement is in line 130 (FD00), the second in line 360 (FE00). We chose round numbers for clarity. The pseudo-op ORG determines where the subsequent machine code resides in memory.

The Marquee routine begins on line 370. All previous lines run only once; execution begins at DISP.

The program stores the keyboard vector in lines 150 and 160. It inserts the Marquee routine and returns to the vector to which it was supposed to go before the marquee intervened.

The program creates three buffers in lines 180–250. It defines the first 3 bytes and uses the LDIR command to set up the graphics in the buffer block. This saves typing by defining a 3-byte duplicate sequence using LDIR.

Every time the ROM tells the CPU to scan the keyboard DCB, execution jumps to our routine.

Lines 370–460 act as a counter keeping track of the number of times the routine runs. This is necessary because the program can't increment screen graphics on each pass. Machine language executes so quickly that the marquee effect wouldn't be evident.

Choosing Speeds

By experimentation, we found a count of 150 to make a nicely paced display. To alter the marquee's rotation speed, change the value (currently set at 150) in line 990. Using a value of zero yields the highest count and slowest motion; the value below zero is 255, which produces a wraparound effect. A value of 1 gives the shortest time delay and moves the marquee rapidly.

STORE 3 returns execution out of the marquee program unless the routine has run 150 times. If it has, the program proceeds to line 470, and draws a line of graphics across the top of the screen.

Since the marquee lights travel in the opposite direction across the bottom of the screen, the program prints a mirror image of the top line across the bottom. It does this by reading the graphics data off the top of the screen backwards.

You can't draw lines by moving

The Key Box

Models I and III
48K RAM
Less RAM with changes
Editor/Assembler

HOME COMPUTER ACCESSORIES

VOICE SYNTHESIZER FOR APPLE AND COMMODORE

JE520AP

JE520CM

- Over 250 word vocabulary-affixes allow the formation of more than 500 words • Built-in amplifier, speaker, volume control, and audio jack • Recreates a clear, natural male voice • Plug-in user ready with documentation and sample software • Case size: 7¼"L x 3¼"W x 1-3/8"H

APPLICATIONS: • Security Warning • Teaching • Instrumentation • Telecommunication • Handicap Aid • Games

The JE520 VOICE SYNTHESIZER will plug right into your computer and allow you to enhance almost any application. Utilizing National Semiconductor's DIGITALKER™ Speech Processor IC (with four custom memory chips), the JE520 compresses natural speech into digital memory, including the original inflections and emphases. The result is an extremely clear, natural vocalization.

Coming Soon: VOCABULARY EXPANSION MODULE
- Expands to over 1000 basic words
- Allows selection of up to 6 additional word sets

Part No.	Description	Price
JE520CM	For Commodore 64 & VIC-20	$114.95
JE520AP	For Apple II, II+, and //e	$149.95

JE664 EPROM PROGRAMMER
8K to 64K EPROMS — 24 & 28 Pin Packages
Completely Self-Contained — Requires No Additional Systems for Operation

• Programs and validates EPROMs • Checks for properly erased EPROMs • Emulates PROMs or EPROMs • RS232C Computer Interface for editing and program loading • Loads data into RAM by keyboard • Changes data in RAM by keyboard • Loads RAM from an EPROM • Compares EPROMs for content differences • Copies EPROMs • Power input: 115VAC, 60Hz, less than 10W power consumption • Enclosure: Color-coordinated, light tan panels with molded end pieces in mocha brown • Size: 15¾"L x 8¼"D x 3½"H • Weight: 5¾ lbs.

The JE664 EPROM Programmer emulates and programs various 8-Bit Word EPROMs from 8K to 64K-Bit memory capacity. Data can be entered into the JE664's internal 8K x 8-Bit RAM in three ways: (1) from a ROM or EPROM, (2) from an external computer via the optional JE665 RS232C BUS, (3) from its panel keyboard. The JE664's RAMs may be accessed for emulation purposes from the panel's test socket to an external microprocessor. In programming and emulation, the JE664 allows for examination, change and validation of program content. The JE664's RAMs can be programmed quickly to all "1"s (or any values), allowing unused addresses in the EPROM to be programmed later without necessity of "UV" erasing. The JE664 displays DATA and ADDRESS in convenient hexadecimal (alphanumeric) format. A "DISPLAY EPROM DATA" button changes the DATA readout from RAM word to EPROM word and is displayed in both hexadecimal and binary code. The front panel features a convenient operating guide. The JE664 Programmer includes one JM16A Jumper Module (as listed below).

JE664-A EPROM Programmer. $995.00
Assembled & Tested (Includes JM16A Module)

JE665 — RS232C INTERFACE OPTION — The RS232C Interface Option implements computer access to the JE664's RAM. This allows the computer to manipulate, store and transfer EPROM data to and from the JE664. A sample program listing is supplied in MBASIC for CP/M computers. Documentation is provided to adapt the software to other computers with an RS232 port. 9600 Baud, 8-bit word, odd parity and 2 stop bits.

FOR A LIMITED TIME, A SAMPLE OF SOFTWARE WRITTEN IN BASIC FOR THE TRS-80® MODEL I, LEVEL II COMPUTER WILL ALSO BE PROVIDED.

JE664-ARS EPROM Prog. w/JE665 Option. $1195.00
Assembled & Tested (Includes JM16A Module)

EPROM JUMPER MODULES — The JE664's JUMPER MODULE (Personality Module) is a plug-in Module that pre-sets the JE664 for the proper programming pulses to the EPROM and configures the EPROM socket connections for that particular EPROM.

JE664 EPROM Jumper Mod. No.	EPROM	Programming Voltage	EPROM MANUFACTURER	PRICE
JM08A	2708	25V	AMD, Mostek, Nat., Intel, TI	$14.95
JM16A	2716, TMS2516 (TI)	25V	Intel, Motorola, Nat., NEC, TI, AMD, Hitachi, Mostek	$14.95
JM16B	TMS2716 (3-V's)	5V, +5V, +12V	Motorola, TI	$14.95
JM32A	TMS2532	25V	Motorola, TI, Hitachi, OKI	$14.95
JM32B	2732	25V	AMD, Fujitsu, NEC, Hitachi, Intel, Mitsubishi, Toshiba	$14.95
JM32C	2732A	21V	Fujitsu, Intel	$14.95
JM64	MCM68764, MCM68L764	21V	Motorola	$14.95
JM48	2764	21V	Intel, Hitachi, OKI	$14.95
JM64C	TMS2564	25V	TI	$14.95

4-Digit Fluorescent Alarm Clock Kit

• Bright 4-digit 0.5" high display • 10 minute snooze alarm • AM/PM indicator • Automatic display dimmer

The JE750 Clock Kit is a versatile 12-hour digital clock with 24-hour alarm. The clock has a bright 0.5" high blue-green fluorescent display. The display will automatically dim with changing light conditions. The 24-hour alarm allows the user to disable the alarm and immediately re-enable the alarm to activate 24 hours later. The kit includes all documentation, components, case and wall transformer. Size: 6¾"L x 3¼"W x 1¾"D.

JE750 Alarm Clock Kit. $29.95

KEYBOARDS — POWER SUPPLIES

21⅞"L x 9.8"W x 3⅛"H

106-Key 8-Bit Serial ASCII Keyboard
• Numeric and cursor keypad • 10 user definable keys • 7 LED function displays • Security lock • N-key rollover • Color: white with black panel • Documentation included • Weight: 6½ lbs.

KB139. . . . . $59.95

19⅞"L x 7⅛"W x 1½"H

Micro-Switch 106-Key Keyboard 8-Bit Serial ASCII
• Numeric and cursor keypad • 8 user definable keys • Japanese/English characters • 8 LED function displays • Documentation incl. • Fits DTE-22 enclosure • Wt: 3½ lbs.

KB106SD29-4. . . . . $29.95

19¾"L x 6½"W x 1½"H

SMK 103-Key Unencoded Keyboard
Numeric and cursor keypad • SPST mechanical keyswitches • 40-pin header connection • Fits DTE-22 enclosure • Weight: 3½ lbs.

KB9000. . . . . $19.95

Power/Mate Corporation REGULATED POWER SUPPLY
• Input: 105-125/210-250VAC at 47-63 Hz • Output: 5VDC @ 3.0 Amps/6VDC @ 2.5 Amps • Size: 4⅞"L x 4"W x 2⅛"H • Weight: 2 lbs.

EMA5/6B. $29.95

Power/Mate Corporation REGULATED POWER SUPPLY
• Input: 105-125/210-250VAC at 47-63 Hz • Output: 5V @ 6 amps/6V @ 5 amps • Size: 5⅛"L x 4⅞"W x 2⅛"H • Weight: 4 lbs.

EMA5/6C. $39.95

Power-One's REGULATED POWER SUPPLY
• Input: 105-125VAC, 47-440 Hz • Output: +5V @ 12 amp; +12V @ 1.7 amp; -12V @ 1.7 amp • Size: 14.25"L x 4.87"W x 2.75"H • Weight: 11 lbs.

DBB-105W. $59.95

POWER SUPPLY +5VDC @ 7.5 AMP, 12VDC @ 1.5 AMP SWITCHING
Input: 115VAC, 50-60Hz @ 3 amp/230VAC, 50Hz @ 1.6 amp. Fan volt./power supply select switches (115/230VAC). Output: 5VDC @ 7.5 amp, 12VDC @ 1.6 amp. 8 ft. blk. pow. cord. 11½"W x 13¼"D x 3¾"H. Wt. 6 lbs.

Part No. PS94V0S . . . . . $39.95 each

POWER SUPPLY 4-Channel Switching - Apple Compatible
Microprocessor, mini-computer, terminal, medical equipment and process control applications. Input: 90-130VAC 47-440Hz. Output: +5VDC @ 5A, -5VDC @ 1A, +12VDC @ 1A, -12VDC @ 1A. Line reg.: ±0.2%. Ripple: 30mV p-p. Load reg.: ±1%. Overcurrent protection. Adj. 5V main output ± 10%. 6-3/8"L x 1-7/8"W x 4-15/16"H. Wt. 1½ lbs.

Part No. FCS-604A . . . . . $69.95 each

DISKETTES AND ACCESSORIES

5¼" and 8" Diskettes

SSDD = Single Sided Double Density DSDD = Double Sided Double Density
SSQD = Single Sided Quad Density DSQD = Double Sided Quad Density

ULTRA MAGNETICS — 5¼" DISKETTES

Part No.	Description	Boxed	PRICE
UM51401	5¼" SSDD Soft Sector with Hub Ring and Envelope	10	$24.95
UM50095	5¼" SSDD Soft Sector with Hub Ring (Bulk)	100	199.95
UM52401	5¼" DSDD Soft Sector with Hub Ring and Envelope	10	32.95
UM50140	5¼" DSDD Soft Sector with Hub Ring (Bulk)	100	259.95
UM51801	5¼" SSQD Soft Sector with Envelope (96 TPI)	10	34.95
UM00214	5¼" DSQD Soft Sector (96 TPI) (Bulk)	100	259.95
UM52801	5¼" DSQD Soft Sector with Envelope (96 TPI)	10	41.95
UM00217	5¼" DSQD Soft Sector (96 TPI) (Bulk)	100	299.95

SK (ESKEI) — 5¼" DISKETTES

EX1D	5¼" SSDD Soft Sector with Hub Ring and Envelope	10	20.95
SK1DB	5¼" SSDD Soft Sector with Hub Ring (Bulk)	100	149.95
SK2D	5¼" DSDD Soft Sector with Hub Ring and Envelope	10	26.95
SK2DB	5¼" DSDD Soft Sector with Hub Ring (Bulk)	100	199.95

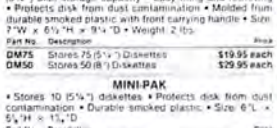
ULTRA MAGNETICS — 8" DISKETTES

UM81726	8" SSQD IBM Compatible (128 B/S, 26 Sectors) and Envelope	10	34.95
UM00825	8" SSQD IBM Compatible (128 B/S, 26 Sectors) Bulk	100	229.95
UM82726	8" DSQD Soft Sector (unformatted) with Envelope	10	49.95
UM00840	8" DSQD Soft Sector (unformatted) Bulk	100	299.95

All diskettes listed above are soft sectored. All have Hub Rings with the exception of the Quad Density and the 8" Series.

DISKETTE ACCESSORIES

Disk Minder



• Attractive, functional disk storage system • 50 (8") or 75 (5¼") disk storage capacity • Easy filing and retrieving • Protects disk from dust contamination • Molded from durable smoked plastic with front carrying handle • Size: 7"W x 6½"H x 9¼"D • Weight: 2 lbs.

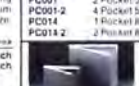
Part No.	Description	Price
DM75	Stores 75 (5¼") Diskettes	$19.95 each
DM50	Stores 50 (8") Diskettes	$29.95 each

MINI-PAK
• Stores 10 (5¼") diskettes • Protects disk from dust contamination • Durable smoked plastic • Size: 6"L x 5¾"H x 1¾"D

Part No.	Description	Price
MP-10	Stores 10 (5¼") Diskettes	$2.95 each

Diskette Envelopes
Sleeves for Bulk-Purchased Diskettes
• Tear proof • Chemically resistant liner • Anti-static protect • Write-read

Part No.	Description	Price
MPS201	10 White 5¼" Envelopes	10 for $ 1.49
MPS201-100	100 White 5¼" Envelopes	100 for $10.95
MP8201	10 White 8" Envelopes	10 for $ 1.95
MP8201-100	100 White 8" Envelopes	100 for $17.95



Vinyl Pages
For 3-Ring Binders
Protects disks from dirt, scratches, dust, spills, and other contaminants. Reduces the risk of getting dirt near your disk drive heart. 3-Ox 8½" x 11".

Part No.	Description	Price
PC001	2 Pocket 5¼" Vinyl Page	10 for $7.95
PC001-2	4 Pocket 5¼" Vinyl Page	10 for $8.95
PC014	1 Pocket 8" Vinyl Page	10 for $7.95
PC014 2	2 Pocket 8" Vinyl Page	10 for $8.95

Mail Pak™
• Holds up to 3 diskettes • Ideal for mailing and retail packaging • Dust proof and durable • Transparent sleeves allows easy identification

Part No.	Description	Price
MP-05	Holds 3 ea. 5¼" Diskettes	$2.49 each
MP-08	Holds 3 ea. 8" Diskettes	$3.95 each

$10.00 Minimum Order — U.S. Funds Only
California Residents Add 6½% Sales Tax
Shipping — Add 5% plus $1.50 Insurance
Send S.A.S.E. for Monthly Sales Flyer!

Spec Sheets — 30¢ each
Send $1.00 Postage for your *FREE 1984 JAMECO CATALOG*
Prices Subject to Change

MasterCard VISA

Mail Order Electronics - Worldwide
Jameco ELECTRONICS
1355 SHOREWAY ROAD, BELMONT, CA 94002 ✓534
2/84 *PHONE ORDERS WELCOME — (415) 592-8097 Telex: 176043*

DISK DRIVES AND CABLE

PANASONIC JA551-2
Double-Sided Half-Height 5¼" DRIVE
Shugart SA455 Equivalent
• Single or double density • 48 TPI • 40 tracks • 6ms track to track • 327KBytes formatted storage • One year warranty parts and labor
The JA551 is perfect for word processors, personal and portable computers, small business computers and terminal add-ons. Includes operating manual. Requires: +5VDC @ 9A and +12VDC @ 1A. Size: 5.88"W x 1.63"H x 8"D. Wt. 3.3 lbs.

JA551-2. . . . . $239.95

TEAC FD55A
Single-Sided Half-Height 5¼" Drive
• Single or double density • 48 TPI • 40 tracks • 6ms track to track • 5W power consumption • Brushless DC direct-drive motor • 160KBytes formatted storage • One year warranty parts and labor
Double your work space with the TEAC 5¼ FLOPPY DISK DRIVE. Because the TEAC FD55A Drive is half the height of conventional drives, you can fit up to four TEAC drives in the same space where two conventional drives fit. Or, have room for two floppy disk drives and a hard disk drive. Includes operating manual. Requires: +5VDC @ .55A and +12VDC @ .3A. Size: 5¾"W x 1¾"H x 8"D. Wt. 3 lbs.

FD55A. $249.95

SHUGART SA455
Double-Sided Half-Height 5¼" Drive
• Single or double density • 48 TPI • 40 tracks • 409KBytes formatted storage • 6ms track to track • Brushless DC direct-drive motor • One year warranty parts and labor • Media and interface compatible with the SA450
Backed by the industry's largest and most experienced engineering sales and service organization, the SA455 is the smart solution for a wide range of desktop applications. Includes operating manual. Requires: +5VDC @ .7A and +12VDC @ .75A. Size: 5.88"W x 1.63"H x 8"D. Wt. 3.3 lbs.

SA455. . . . . $259.95

UV-EPROM Eraser

8 Chips — 51 Minutes

1 Chip — 37 Minutes

Erases 2,08, 2716, 2732, 2764, 2516, 2532, 2564. Erases up to 8 chips within 51 minutes (1 chip in 37 minutes). Maintains constant exposure distance of one inch. Special conductive foam liner eliminates static build-up. Built-in safety lock to prevent UV exposure. Compact — only 9.00" x 3.70" x 2.60". Complete with holding tray for 8 chips.

DE-4 UV-EPROM Eraser . . . $79.95
UVS-11EL Replacement Bulb $16.95

5¼" APPLE™ COMPATIBLE DISK DRIVE

• Uses Shugart SA390 mechanics • 143K formatted storage • 35 tracks — compatible with Apple controller • Complete with connector and cable — just plug into your disk controller card • Size: 6"L x 3½"W x 8-9/16"D • Weight: 4¾ lbs

Part No. ADD-514 $195.95

8" FLOPPY DISK DRIVE

• Single-Sided
• 77 Tracks
• 400/800K Bytes Capacity
• Industry Standard

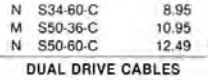
The FDD100-8 8" Floppy Disk Drive (Industry Standard) features single or double density, Recording mode: FM single, MFM double density. Transfer rate: 250K bits/sec. single density; 500K bits/sec. double density. The FDD100-8 is designed to work with the single-sided soft sectored IBM Diskette I, or eq. disk cartridge. Power: 115VAC @ 50-60Hz, + 24VDC @ 1.7 amps max., + 5VDC @ 1.2 amps max. Unit as pictured above (does not include case, power supply, or cables). Size: 8.55"W x 14"L x 4.5"H. Weighs 12 lbs. Incl. 96-pg. manual.

FDD100-8 . . $169.95 ea.

SIGNAL CABLES

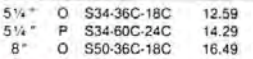
5¼" DRIVES USE 34-PIN ASSEMBLIES
8" DRIVES USE 50-PIN ASSEMBLIES
*S = SOCKET CONNECTOR
*C = CARD-EDGE CONNECTOR

SINGLE DRIVE CABLE



Drive	Style	Part No.	1-4	5+
5¼"	M	S34-36-C	7.95	7.25
5¼"	N	S34-60-C	8.95	8.39
8"	M	S50-36-C	10.95	9.95
8"	N	S50-60-C	12.49	10.95

DUAL DRIVE CABLES



Drive	Style	Part No.	1-4	5+
5¼"	O	S34-36C-18C	12.59	11.19
5¼"	P	S34-60C-24C	14.29	12.69
8"	O	S50-36C-18C	16.49	14.49
8"	P	S50-60C-24C	18.95	16.95

Power Cable Kits

Kits include: Connector shells, connector pins, and power cables.

	Price
Power Cable Kit for 5¼" Drive Part No. PCK-5	$2.95
Power Cable Kit for Full-Sized 8" Drive Part No. PCK-8	$3.95
Power Cable Kit for Qume Full-Sized 8" Drive Part No. PCK-Q	$4.95

✓ See List of Advertisers on Page 211

```
3C00            00100 VID     EQU     03C00H
0400            00110 INC     EQU     0400H
0400            00120 BYTES   EQU     0400H
FD00            00130         ORG     0FD00H
FD00 2100FE     00140 START   LD      HL,DISP ; * * * * * * * * * * * * * * * *
FD03 ED5B1640   00150         LD      DE,(04016H)
FD07 ED539EFE   00160         LD      (STORE1),DE     ;    MARQUEE
FD0B 221640     00170         LD      (04016H),HL     ;   WRITTEN BY
FD0E 014200     00180         LD      BC,66           ; DAVE DISCHERT/DAN KEEN
FD11 1128FD     00190         LD      DE,BUFF+3
FD14 2125FD     00200         LD      HL,BUFF ; * * * * * * * * * * * * * * * *
FD17 EDB0       00210         LDIR
FD19 116EFD     00220         LD      DE,BUFF2+4
FD1C 216AFD     00230         LD      HL,BUFF2
FD1F 012000     00240         LD      BC,32
FD22 EDB0       00250         LDIR
FD24 C9         00260         RET             ;       RETURN AFTER SETUP
FD25 83         00270 BUFF    DEFB    131     ;START OF BUFFER AREA
FD26 20         00280         DEFM    ' '
0042            00290         DEFS    66
FD6A 83         00300 BUFF2   DEFB    131
FD6B 8C         00310         DEFB    140
FD6C B0         00320         DEFB    176
FD6D 80         00330         DEFB    128
0010            00340         DEFS    16
0010            00350 BUFF3   DEFS    16
FE00            00360         ORG     0FE00H
FE00 F5         00370 DISP    PUSH    AF      ;START OF MAIN PROGRAM
FE01 08         00380         EX      AF,AF'
FE02 D9         00390         EXX
FE03 F1         00400         POP     AF
FE04 ED739CFE   00410         LD      (STORE),SP
FE08 ED4BA0FE   00420         LD      BC,(STORE3)
FE0C 0D         00430         DEC     C
FE0D 79         00440         LD      A,C
FE0E ED43A0FE   00450         LD      (STORE3),BC
FE12 207D       00460         JR      NZ,BACK
FE14 2125FD     00470         LD      HL,BUFF ;START OF VIDEO DISPLAY ROUTINES
FE17 11003C     00480         LD      DE,VID
FE1A 014000     00490         LD      BC,64
FE1D EDB0       00500         LDIR
FE1F 21003C     00510         LD      HL,VID
FE22 11FF3F     00520         LD      DE,03FFFH
FE25 0640       00530         LD      B,40H
FE27 7E         00540 LOOP    LD      A,(HL)
FE28 12         00550         LD      (DE),A
FE29 23         00560         INC     HL
FE2A 1B         00570         DEC     DE
FE2B 10FA       00580         DJNZ    LOOP
FE2D 060F       00590         LD      B,15
FE2F 216AFD     00600         LD      HL,BUFF2
FE32 DD21003C   00610         LD      IX,03C00H
FE36 114000     00620         LD      DE,040H
FE39 7E         00630 LOOP2   LD      A,(HL)
FE3A DD7700     00640         LD      (IX),A
FE3D DD19       00650         ADD     IX,DE
FE3F 23         00660         INC     HL
FE40 10F7       00670         DJNZ    LOOP2
FE42 060F       00680         LD      B,15
FE44 DD213F3C   00690         LD      IX,03C3FH
FE48 217EFD     00700         LD      HL,BUFF3
FE4B 7E         00710 LOOP3   LD      A,(HL)
FE4C DD7700     00720         LD      (IX),A
FE4F 23         00730         INC     HL
FE50 DD19       00740         ADD     IX,DE
FE52 10F7       00750         DJNZ    LOOP3
FE54 2126FD     00760         LD      HL,BUFF+1       ;START OF INCREMENT TABLES
FE57 1125FD     00770         LD      DE,BUFF
FE5A 014200     00780         LD      BC,66
FE5D EDB0       00790         LDIR
FE5F 3A25FD     00800         LD      A,(BUFF)
FE62 1167FD     00810         LD      DE,BUFF+66
FE65 12         00820         LD      (DE),A
FE66 216BFD     00830         LD      HL,BUFF2+1
FE69 116AFD     00840         LD      DE,BUFF2
FE6C 011000     00850         LD      BC,16
FE6F EDB0       00860         LDIR
FE71 216AFD     00870         LD      HL,BUFF2
FE74 117AFD     00880         LD      DE,BUFF2+16
FE77 7E         00890         LD      A,(HL)
FE78 12         00900         LD      (DE),A
FE79 218DFD     00910         LD      HL,BUFF3+15
FE7C 118EFD     00920         LD      DE,BUFF3+16
FE7F 011000     00930         LD      BC,16
FE82 EDB8       00940         LDDR
FE84 218EFD     00950         LD      HL,BUFF3+16
FE87 117EFD     00960         LD      DE,BUFF3
FE8A 7E         00970         LD      A,(HL)
FE8B 12         00980         LD      (DE),A
FE8C 3E96       00990         LD      A,150   ;COUNT BEFORE DISPLAY IS MOVED
FE8E 32A0FE     01000         LD      (STORE3),A
FE91 ED7B9CFE   01010 BACK    LD      SP,(STORE)      ;JUMP TO NORMAL KYBD ROUTINE
FE95 2A9EFE     01020         LD      HL,(STORE1)
FE98 E5         01030         PUSH    HL
FE99 D9         01040         EXX
FE9A 08         01050         EX      AF,AF'
FE9B C9         01060         RET
FE9C 0000       01070 STORE   DEFW    00H
FE9E 0000       01080 STORE1  DEFW    00H
FEA0 96         01090 STORE3  DEFB    150D
FD00            01100         END     START
00000 TOTAL ERRORS
```

Program Listing. Marquee routine for Models I and III.

them around the screen. The program reads buffer information to prevent keys you may have hit from appearing on the screen with the marquee graphics.

The program draws graphics on the screen's left side in lines 510–580.

Unfortunately, right-column images are a little more difficult to create. Here we're dealing with two pixels within a character position box. Due to the shape of the machine's graphics setup (two pixels wide by three pixels high), there are more pixels to worry about on the sides. This required another program table (lines 680–750).

After printing all the graphics, the program moves each character around the screen.

Relocating the Routine

The routine is located near the end of memory in both Models I and III. If you're cramped for space you can push it higher; the entire program is fewer than 300 bytes long. We located it at FD00 hex for simplicity only. ■

Contact Dan Keen and Dave Dischert at RD 1, Box 432, Cape May Court House, NJ 08210.

See List of Advertisers on Page 211

The PRODUCER

The Professional Program Writer.

59

What has your computer done for you lately? You bought it to be a powerful and time saving tool. But if lack of good software keeps you frustrated and makes your computer an expensive and idle gadget, The PRODUCER is here to solve your problem.

Now you can design and produce professional quality programs that meet your exact specifications and you don't even need to understand programming at all.

THE PRODUCER IS A SOFTWARE PACKAGE THAT WRITES PROGRAMS FOR YOU.

Even though you have no knowledge about how to write programs, you can now create impressive, sophisticated and functional software to manage your data. You answer simple English questions, draw your screen on your monitor exactly like you want it, and The PRODUCER writes the entire BASIC program by itself.

THE PRODUCER WAS DESIGNED FOR MICRO COMPUTER OWNERS WHO CAN'T FIND THE SOFTWARE PROGRAM TO DO WHAT THEY WANT IT TO DO.

You may never need to buy another computer program to store and retrieve information, perform calculations on your data and get displayed and printed reports. The PRODUCER can create customized software of truly professional quality.

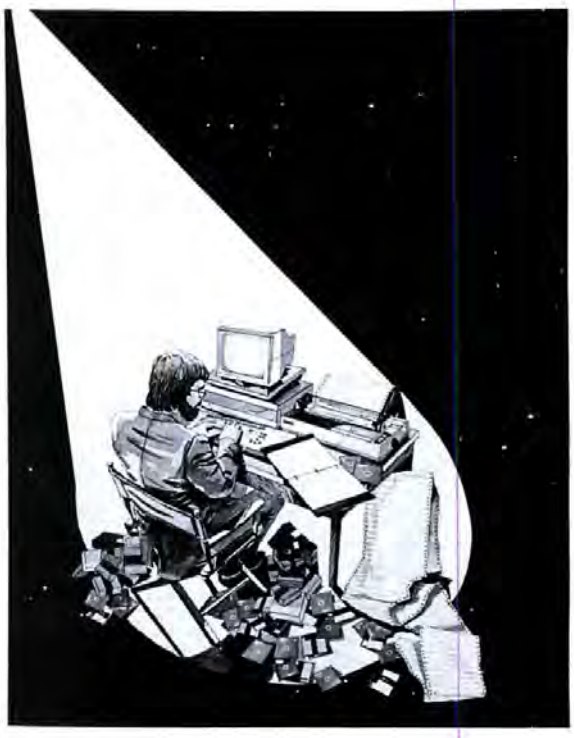
The PRODUCER makes the micro computer a useful tool to the novice and saves many hours of programming time for the experienced computer professional.

IF YOU ARE A NOVICE

The PRODUCER can make you feel like a pro. The Basic code is written for you. You push buttons, answer questions and watch the program develop in this remarkable process.

IF YOU ARE A PROFESSIONAL PROGRAMMER

The PRODUCER can be the time-saver you need to increase your productivity and make your job easier. The PRODUCER provides many of the advanced features found on products that cost many thousands of dollars more. You'll be proud to show your clients the professional quality programs created by The PRODUCER.



Listen to what one of our users wrote recently:

The PRODUCER has proven to be the greatest. I used to spend 70% of my time writing programs to create, maintain, sort, and list data. No More. Days and weeks of programming are now reduced to minutes and hours. The PRODUCER has increased the productivity of my custom software firm by 400%. This product is in a class reserved for the best.

A. Copelle, Northbrook, Illinois.

HOW DO I LEARN TO USE THE PRODUCER

In each TRS-80 version, we have provided a systematic guided tour of The PRODUCER program generator process. For the Model I and III, an audio cassette tape tutorial is part of your package. One of your fellow PRODUCER owners talks to you as you go through the step-by-step lessons. The tapes not only teach you the operating process, they enable you to actually create a program of your own design while you learn.

We have provided over 200 pages of thorough documentation in The PRODUCER Reference Manual, but we encourage you not to read the manual until after you have completed the tutorial. We've had many rave reviews from our users, like this one from S.R. Foster of Pensacola, Florida:

The tutorial was an excellent starter. It enabled me to get on with it without days and days of reading. Very helpful.

WHAT DO YOU GET WITH THE PRODUCER?

You will be impressed with the professionalism of the PRODUCER package:

DISKETTE(s) containing PRODUCER Program Development System.

REFERENCE MANUAL of over 200 pages of extensive, easy to read, well organized material. Attractive hardback 3-ring binder. Color keyed index tabs separate the chapters. Comprehensive alphabetical Index refers to specific chapter subsections.

QUICK REFERENCE CARD

REGISTRATION CARD

TUTORIAL SESSION including audio cassettes and detailed follow-along outline, written and produced by fellow PRODUCER user.

FREE HOME INVENTORY MANAGEMENT PROGRAM ($59.95 value as a sample) allowing you to use a finished program immediately.

ONE YEAR SUBSCRIPTION to the PRODUCER newsletter

TECHNICAL ASSISTANCE by phone, available only to registered PRODUCER owners.

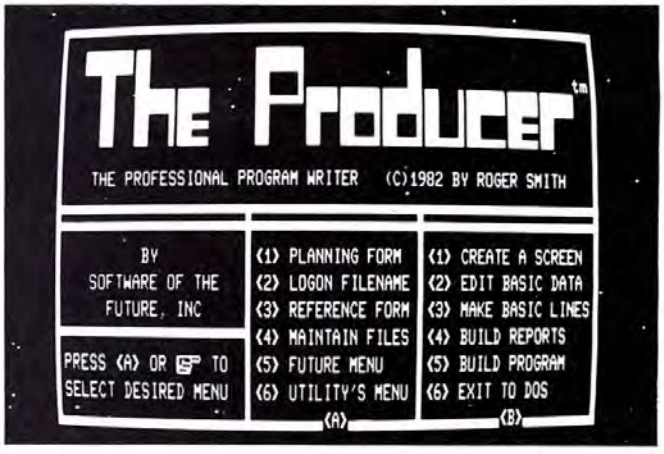
Package listed is for Model III. Format and features will vary some with other versions.

✓ 59

The PRODUCER

HOW THE PRODUCER WORKS

We think you will be impressed with the ease of operation and the amazing versatility of features you get with The PRODUCER. Here is a step by step overview of the program writing process. The screen shown is an actual photo of the Master Menu from the Model III version from which each of these steps is selected.



☐ Planning Your Program

The PRODUCER provides a helpful planning form you can print on your own printer. It helps you organize your thoughts to create a tailor made program to meet your needs.

☐ Creating The Screen

Visible on your monitor will be the screen where information will be entered, edited and displayed. There are six simple steps to follow in creating your screen.

1. Draw Your Screen

Using the arrow keys construct the screen in any configuration you desire. With single keystrokes, enter large graphic letters and borders. Edit at will until you are satisfied.

2. Define Message Areas

Select an area of your screen where The PRODUCER messages to you will appear.

3. Define Input Fields

The PRODUCER will ask you questions about the areas where you will enter the data. You specify the length of each area or field, as well as acceptable characters in each field.

4. Define Display Fields

Locate the display fields anywhere you want on your screen. These show the results of the calculations you want made on your data.

5. Define Custom Prompts

You select an area where help messages to yourself can be displayed.

6. Save Your Results

Assign a working name for your program and save it to disk.

☐ Editing Basic Data

1. Edit any part of The PRODUCER program you have created -- screen field names, lengths, prompt areas, etc.
2. Type in any help message you want as a custom prompt to help you operate the program.
3. Easily create calculations for your program using actual field names. You can use the contents of any numeric field and all math operations including logical operators.

☐ Making Basic Code

Press a key, sit back and watch The PRODUCER do all the work of creating BASIC code for your program. You can see the program lines appear on your screen. Complete error checking is done for you.

☐ Building Reports

Virtually any report is available to you thru our NEW free form report generator. It works with any size paper. You are allowed up to 100 calculations within the report. You can specify exact position of any text information to any position on your paper (even preprinted forms, checks, etc.). An amazingly versatile tool.

☐ Building The Program

Put the finishing touches on your program by selecting cursor type, size, flashing speed, auto messages, custom logos, etc. After your selections have been made, press a key and your entire finished program is created in less than 5 minutes. That's all there is to this remarkably simple program generation process.

Continued

The PRODUCER

TECHNICAL INFORMATION

The PRODUCER provides many advanced features which allow you to do "magic" with the programs you create.

The SCREEN GENERATOR

- Use the full screen (all lines and column positions)
- Create a professional well organized screen with graphics
- Save up to 9 separate screens in memory at one time and get instant access to each
- Move the cursor to any location on the screen
- Replicate bars/lines/graphics to define certain screen areas
- Access an instantly available Help Menu of all Screen Editor commands
- Insert and delete any character with a single keystroke
- Clear or erase selected areas of any screen
- Insert and delete whole lines on the screen
- Center any text on the screen
- Move any rectangular block of text anywhere on the screen (block move)
- Create titles with a single keystroke large graphic letter alphabet
- Move portions of screens between different screens (cut and paste)
- Save any number of screens to disk at any time
- Recall any screen from disk any time
- Create BASIC lines to re-create any screen

FILE and RECORD HANDLING

- Rapidly access records with BTREE File structure
- Search for a record with only the first few letters of the name or key (partial key) (Example: locate PRODUCER by typing PR)
- Recall and edit duplicate and multiple keys (Example: Several last names may be the same on a file and you can find and edit them individually
- Fully edit any part of a previously entered record
- Recover unused space automatically upon deletion of a record
- Enter data very fast with the special batch mode
- Recall immediately any record after it's been entered, eliminating time consuming sorting and indexing
- Rapidly access any record anytime (2-4 seconds average)
- Globally search and replace data in certain fields in selected record range
- Automatically rebuild any file to meet new specifications. No need to re-enter data when a file needs to be restructured.
- Balance any BTREE file automatically to reorganize and speed up file access time
- Recover from power failure and easily rebuild files that have been damaged. Avoid laborious re-entry of long data files

SCREEN ORIENTED INPUT and EDITING of DATA

- Insert and delete characters at any position in any field. No "back to start" retyping of data
- Move forward or back to previously entered fields to edit using the arrow keys. Totally non-destructive cursor. Does not require re-entering of each data field
- Move within any field using the arrow keys
- Move instantly to any field with Control G command
- Exit from input/edit mode at any point allowing immediate escape from data entry mode. Allows partial information to be entered for each record without the annoying, time consuming need to press ENTER for each blank field not used at the time of entry
- Duplicate field information from a previous record with one keystroke. No need to re-enter duplicate information, addresses, etc. on consecutive records
- View a custom prompt, your own custom reminder or help message for each field with 1 keystroke
- Verify each character typed automatically
- Enter data as fast as you want, even if you are a speed typist
- View visible display of automatic field length restrictions
- View prompts for each field showing number of characters allowed

PRINTED REPORTS

- Create up to 9 separate reports at a time in a finished program
- Generate any number of reports you want (no limit)
- Select reports by name from a report menu in the program
- Select from six different automatic report formats including custom mailing labels
- Instantly print reports by key with no time consuming sort necessary
- Sort and print any other (non key) field with the fast machine language sort
- Sort only records that meet your search criteria
- Sort on more than one field if desired
- Use any restrictions or search criteria to determine which records will be included in a report
- Use any number of multiple search criteriea (including logical) (Example: You can search for all the males who are single, and drive a car that are over 24 years old but less than 35 years old
- Send any special command to your printer before or after any report
- Specify any line length needed and any page length desired
- Select single line or multiple lines per record, even one page per record
- Total any fields during the report (running totals)

ADVANCED CALCULATIONS

- Globally recalculate any field in any or all records. (Example: If file is a list of gold assets and the spot price changes, each separate asset may be recalculated with a new value for the spot price)
- Use all math operations including exponentiation and trigonometry
- Use logical calculations such as And, Or, Not, etc.
- Use any level of parenthesis in calculation formulas
- Save results in any field and display results in any field
- Store temporary results in several extra memory slots
- Pass calculation results between records
- Determine the exact order of calculations
- Display or save results at your option in the finished record

OTHER ADVANCED FEATURES

- Edit any part of any program without starting over or redefining the entire program
- Create screen and input modules only (for professional programmers)
- Create Calculate-only programs with the easy desk-top super calculator program
- Design custom logos for your program
- Control cursor type, size, flash speed, etc.
- Design custom prompts or help info for any field

New! Optional Feature Freeform Report Generator

- Specify column and row of every heading and field
- Allow up to 100 of interfield calculations, even string calculations
- Include any text anywhere on the screen
- Keep sub-totals on any field and print at any time in any format
- Format any numeric fields anyway you wish
- Print reports on pre-printed forms, checks, etc.
- Create form letters with merged field data, with no word processing necessary
- Put any field anywhere on the page. No limitations

$49.95 **Producer owners! Upgrade your package with this versatile enhanced feature.**

The PRODUCER

WHAT ARE PRODUCER USERS SAYING?

We continue to receive testimonials from satisfied users almost every day. Here's a sampling of the feedback we are receiving:

VALUE

VERY impressive! No matter how much I use the PRODUCER, there is no doubt I got my money's worth. It is clear the program, packaging and tutorial are developed with lots of thought....Very user friendly! Congratulations!
R. N. Forbes, Los Altos Hills, California

The PRODUCER package I received was excellent. The finest software package I have ever purchased. Far beyond my expectations.
S. R. Foster, Pensacola, Florida

I think the PRODUCER will save me so much time that it will give me the time to do the more important tasks that my business calls for and the money I'll save from not having to buy canned programs that are overpriced. Now with the PRODUCER I can write a program overnight to do almost anything I want it to do and with written reports to boot. Talk about saving time and money. I feel the PRODUCER will pay for itself with my first three programs.
S. Tornatore, Canastota, New York

The PRODUCER is a very impressive software package. It is well worth the money. While other micro owners are printing mailing labels, I am now selling them programs to use. I now have more time to spend enjoying my computer.
V. E. Ryberg, Bloomington, Illinois

I'm in love with the PRODUCER. It's one of my favorite programs.
R. Selsback, Burlingame, California

It was very complete and professionally done. The packaging and program seem to have been thought out before assembly and sale. The 'value' of the deal, everything included was the best I've seen to date.
G. Slusher, Martin, Kentucky

Very professional packaging. It gave the feeling of getting your money's worth before even running the program...Very easy to use and leaves very few questions unanswered...As you can see, I like the PRODUCER and was impressed with how trouble free it is.
A. C. Vincent, Napa, California

Excellent. Above and beyond other software.
R. Hapgood, Henrietta, Texas

VERSATILITY

The PRODUCER is the best all purpose program generator I have used. (We have tried almost all of them.) The generated code is bug free, well commented and efficient.
R. A. Copella, Northbrook, Illinois

I bought the PRODUCER to save time. I feel capable of being able to write almost all programs I need. The PRODUCER generated programs will save a lot of time writing basic code and debugging. Using the PRODUCER I can write a good database type program using math calculation in about three hours. I don't have to tell you how long it would take writing the same program from scratch.
S. Tornatore, Canastota, New York

A special thanks to Roger and all of you. You've made my computing life easier and better. My 10 year old can't wait to get his hands on the PRODUCER.
J. D. Konkler, Columbus, Ohio

DOCUMENTATION

The Reference Manual is a work of art. Not only is it attractive and easy to use, it is so well organized, documented and logically written that the manual is a rarity in the software market place.
S. R. Foster, Pensacola, Florida

One of the best I've seen. We write about 20 volumes of material per year. Take it from a 'pro', it's good!
J. Crespi, Sherman Oaks, California

The PRODUCER Reference Manual is professionally written to provide ready acess to easily understood answers to questions which arise during use of the PRODUCER.
R. A. Copella, Northbrook, Illinois

The Reference Manual is supreme and superior to anything I have worked with.

R. A. Neuman, Okemos, Michigan

Very well laid out and organized. One of the best I've seen.
J. D. Konkler, Columbus, Ohio

QUALITY

Thank you for an excellent program. I agree that The PRODUCER will change the entire concept of program creation in the future. But for now, you stand as the best data-base-management-system I can buy.
E. Sung, Vancouver, B.C.

Your system really is Software of the Future. Your staff has insight others of us only dream of. Congratulations on a product of extraordinary design.
S. R. Foster, Pensacola, Florida

This is an excellent program. At this point I am totally pleased. This is by far my number one software and I will use it anywhere and everywhere I possibly can both personal and business. Once again congratulations to all of the people involved.
R. A. Neuman, Okemos, Michigan

Comparison shopping indicates the PRODUCER's superiority to all others. And I already own most of the others.
R. A. Copella, Northbrook, Illinois

Glad to see you take an interest in what some of us hackers are up against. I think the PRODUCER will make the software hackers upgrade their products to this high level quality of the PRODUCER. I'm sure you realize that there is a lot of garbage on the market.
D. J. Smith, Lombard, Ilinois

I was impressed by the professional appearance of your program. Other software I have received were on copy paper and stapled into a booklet with very vague instructions.
W. J. Mahaffey, Absecon, N. J.

USE

The program is almost idiot proof.
J. Crespi, Sherman Oaks, California

It is a very friendly friend and we will be working together for some time to come.
R. A. Neuman, Okemos, Michigan

Very easy to use and leaves very few questions unanswered.
A. C. Vincent, Napa, California

The PRODUCER

59

Call your order in today.

PRODUCER
Box 1245
Arlington, Texas
76004-1245

800-433-5355

Texas 817-274-6998

Sensational Software Breakthrough

Model I Version	**$149.95**
Model III Version	**$149.95**
IBM-PC Version	**$299.95**

Freeform Report Generator: $49.95

Young Programmer's Awards

The Play's the Thing

by Stephen Roth

Listing 1 continued from p. 54
Listing 2 begins on p. 178

```
01590           ADD     HL,DE
01600           LD      A,(HL)
01610           CP      32
01620           JR      Z,FIX
01630           LD      A,(POS)
01640           LD      B,A
01650           LD      A,(FLAG9)
01660           CP      1
01670           JR      NZ,LOWA
01680           INC     B
01690           INC     B
01700           INC     B
01710 LOWA      INC     B
01720           JP      MANIP
01730 HIGH      LD      DE,64
01740           SBC     HL,DE
01750           LD      A,(HL)
01760           CP      32
01770           JR      Z,FIX
01780           LD      A,(POS)
01790           LD      B,A
01800           LD      A,(FLAG9)
01810           CP      1
01820           JR      NZ,HIGH1
01830           DEC     B
01840           DEC     B
01850           DEC     B
01860 HIGH1     DEC     B
01870           JP      MANIP
01880 SIDER     LD      A,(FLAG9)       ;MOVE TO RIGHT
01890           CP      1
01900           JP      NZ,WAIT
01910           LD      DE,16
01920           ADD     HL,DE
01930           LD      A,(HL)
01940           CP      32
01950           JP      Z,FIX
01960           LD      A,(POS)
01970           LD      B,A
01980           INC     B
01990           JP      MANIP
02000 SIDEL     LD      A,(FLAG9)       ;MOVE TO LEFT
02010           CP      1
02020           JP      NZ,WAIT
02030           LD      DE,16
02040           SBC     HL,DE
02050           LD      A,(HL)
02060           CP      32
02070           JP      Z,FIX
02080           LD      A,(POS)
02090           LD      B,A
02100           DEC     B
02110 MANIP     LD      A,B     ;PUT CURSOR OVER LETTERAND STORE LETTER
02120           LD      (POS),A
02130           PUSH    HL
02140           LD      HL,(COORD)
02150           LD      A,(MCHR)
02160           LD      (HL),A
02170           POP     HL
02180           JP      WAITA
02190 FIX       LD      HL,(COORD)
02200           JP      WAIT
02210 ;         DISPLAY MENUS
02220 OPTION    LD      (4020H),HL
02230           PUSH    HL
02240 OPTIO1    LD      A,(DE)
02250           INC     DE
02260           CP      9
02270           JP      Z,SPACE
02280           CP      10
02290           JP      Z,SPACE2
02300           PUSH    DE
02310           CALL    33H     ;DISPLAY CHARACTER
02320           POP     DE
02330           CP      13
02340           JR      NZ,OPTIO1
02350           JR      SPACE0
02360 OPTIOA    LD      A,(DE)
02370           CP      1
02380           JR      NZ.OPTION
02390           RET
02400 SPACE0    POP     HL      ;PRINT SPACES
02410           PUSH    DE
02420           LD      DE,64
02430           JR      SPACE3
02440 SPACE     POP     HL
02450           PUSH    DE
02460           LD      DE,16
02470           JR      SPACE3
02480 SPACE2    POP     HL
02490           PUSH    DE
02500           LD      DE,45
02510 SPACE3    ADD     HL,DE
02520           POP     DE
02530           JR      OPTIOA
02540 ;         DISPLAY INPUT BOX
02550 BOX       PUSH    HL
02560 BOXA      LD      A,(DE)
02570           LD      (HL),A
02580           INC     HL
02590           INC     DE
02600           CP      13
02610           JR      NZ,BOXA
02620           DEC     HL
02630           LD      A,20H
02640           LD      (HL),A
02650           POP     HL
02660           LD      DE,66   ;DRAW BOX
02670           ADD     HL,DE
02680           LD      A,191
02690           LD      (HL),A
02700           INC     HL
02710           LD      A,131
02720           LD      B,9
02730 BOX1      LD      (HL),A
02740           INC     HL
02750           DJNZ    BOX1
02760           LD      A,191
02770           LD      (HL),A
02780           LD      DE,54
02790           ADD     HL,DE
02800           LD      (HL),A
02810           LD      DE,10
02820           ADD     HL,DE
02830           LD      (HL),A
02840           LD      DE,54
02850           ADD     HL,DE
02860           LD      (HL),A
02870           LD      A,176
02880           LD      B,9
02890 BOX2      INC     HL
02900           LD      (HL),A
```

Listing continued

You've Got TOTAL ACCESS™

TO YOUR COMPUTER HARDWARE & SOFTWARE NEEDS. CALL ROSE TODAY!

AEROCOMP DISK DRIVES

External drives for TRS80 Mod. I or III, IBM PC & TI 99/4A are complete with power supply & enclosure.

40 Track Single Side (Tandon)	$199
40 Track SS "Flippy" (MPI)	239
40 Track Dual Head (Tandon/MPI)	279
80 Track SS (MPI)	299
80 Track SS "Flippy" (MPI)	329
80 Track Dual Head (Tandon)	379
Color Computer Add-On Drives (2nd & 3rd)	199

8" EXPANSION BOX

Complete system with power supply & fan (slimline)

Two 8" Single Side	$699
Two 8" Double Side	849

***All New! Half-High Drives Available Now. Call For Prices.**

BARE DRIVES

Internal drives for TRS80 Mod. III, IBM PC & TI 99/41 (controller required)

40 Track SS (Tandon)	$169
40 Track Dual Head (Tandon/MPI)	249
80 Track SS (MPI)	269
80 Track Dual Head (Tandon)	339
8 inch Single Side Thinline	260
8 inch Dual Head Thinline	375

MODEL III DRIVES

Complete internal drive kits with 40 track drives, disk controller, power supply, all hardware & cables.

Drive Kit Only (no drives)	**$199**
One Drive System Kit	369
Two Drive System Kit	**539**

MODEL I DOUBLE DENSITY CONTROLLER

AEROCOMP "DDC"	**$99**
AEROCOMP "DDC" w/LDOS	169
AEROCOMP "DDS" (Double dens. data separator for Percom Doubler, LNDoubler or Superbrain	49
AEROCOMP "SDS" (Single dens. data separator)	29

MISCELLANEOUS DRIVE STUFF

TRSDOS 2.3 Disk & Manual	$20
LDOS (Mod. I or III)	119
NEWDOS/80, 2.0 (Mod. I or III)	129
Diskettes (10 in library box)	23
MX80 Ribbons	$9
Drive Power Supply & Enclosure (5.25")	59
2-Drive cable	24
4-Drive cable	34
Extender cable	13

MICRO DECISION

$1299

64K CP/M 2.2, two serial ports
MBASIC - WordStar - Logicalc
BaZic - Correct-IT
12" Green Video 80 x 25
Double Density (200K) Drive

FRANKLIN ACE1000
Uses APPLE Software
Call, it may be cheaper by now
$999

brother
EM-1 Electronic Typewriter
HR1 PRINTER
Choose Either One
$799

EPSON FX-80
160cps Friction/Tractor ($32 Option)
10,12,16.5 cpi
+ Doublewide 6,8,12 1pi
$569

TRS-80 SPECIAL EQUIPMENT

80 x 24 Video Board for the Model III	$199
112K CP/M 2.2 for the Model III	$399
16K Memory, 200nsec, Guar 1 yr	8/$12
64K Memory, 200nsec, Guar 1 yr	8/$48
12" Green Phosphor Monitor	$87

SOFTWARE

Super Utility Plus 3.0 by Kim Watt	$59
Alcor PASCAL, Model I or III	169
P&T CP/M for the Mod II	159
P&T CP/M for the Mod 16	189
P&T CP/M for the Hard Disk	199
All SNAPPWARE	**10%** OFF

I have lots of other software. Call me now for your needs. All at discount.

MEDIA & SUPPLIES

8" disks SS DblDen, Guar. Forever	$29
8" disks DS DblDen, Same Guarantee	39
5" Flipsort, holds 75 disks	19
8" Flipsort, same deal	29
5" Library boxes	$2.50
8" Library boxes	$3.50
5" or 8" Head cleaning kit	9
Tractor paper, letter size 2900 sheets	25

LNW COMPUTERS

128K LNW-80 Model II, Both NTSC & RGB outputs. RS232 and parallel printer port. Uses both 5" and 8" drives. Works on all known TRS-80 DOS's. Comes with DOSPLUS. Also works with CP/M 2.2 which is included at no extra cost. Now a FREE 12" GREEN PHOSPHOR monitor and cable included. All for the low price of $1695

PRINTERS & OTHER ACCESSORIES

ANADEX DP-9500A or 9501A	$1239
ANADEX DP-9620A, 200 cps	1399
ANADEX WP-6000, 284 cps, NLQ	2695
PROWRITER, 120 cps, 10" Friction/Tractor	375
PROWRITER 2, 120 cps, 15" Fric/Trac	669
STARWRITER F-10, 40 cps Daisy Wheel	1250
PRINTMASTER F-10, 55 cps Daisy	1499
Rutishauser Bidirectional Tractor	199
Rutishauser Sheet Feeder	895

PERIPHERALS –

32K LNW Expansion Interface w/RS232	$329
Mod III RS232 complete, ready to install	79
Signalman Mk I Modem 300 baud, direct connect	89

Please add $5 handling charge to all orders under $24

ORDER NOW! TOLL FREE 800-527-3582

Write or call. Toll free lines are for orders only. Texas residents call 214/458-1966 and deduct $2.00 from your order. If you need technical information or service use the Texas number. Prices are subject to change without notice and are mail order only. I accept VISA or MASTERCARD, you can send a check or money order (allow a couple of weeks for personal or company checks to clear) or order COD (cash, certified check or money order only). Shipping is not included unless otherwise indicated Please add $5 handling charge to all orders under $24. Texas orders add 5% tax. No tax added on shipments outside Texas. Order up – I need the money!

TOTAL ACCESS™
P.O. BOX 790276
DALLAS, TX 75379
214-458-1966 ↙81

NEXT DAY SHIPMENT on all in stock Merchandise.

↙ See List of Advertisers on Page 211

Listing 1 continued

```
02910           DJNZ    BOX2
02920           INC     HL
02930           LD      A,191
02940           LD      (HL),A
02950           LD      DE,73
02960           SBC     HL,DE
02970 BOX21     LD      A,'-'
02980           LD      (HL),A
02990           LD      A,(LEN)
03000           LD      B,A
03010           LD      (SCRIN),HL
03020 BOX3      CALL    49H       ;WAIT FOR INPUT
03030           CP      0DH
03040           RET     Z
03050           CP      8
03060           JR      Z,BOX4
03070           LD      (HL),A
03080           INC     HL
03090           DJNZ    BOX3
03100           RET
03110 BOX4      LD      HL,(SCRIN)
03120           LD      A,20H
03130           LD      B,7
03140 BOX5      LD      (HL),A
03150           INC     HL
03160           DJNZ    BOX5
03170           LD      HL,(SCRIN)
03180           JP      BOX21
03190 ;         COMPARE STRING6
03200 COMPAR    LD      A,(DE)
03210 COMPR1    CP      0DH
03220           RET     Z
03230           CP      20H
03240           RET     C
03250           CP      (HL)
03260           RET     NZ
03270           INC     HL
03280           INC     DE
03290           JP      COMPAR
03300 STRING    PUSH    HL
03310           PUSH    DE
03320           CALL    COMPAR  ;COMPARE (HL) WITH (DE)
03330           JP      Z,STRIN3        ;IF THE SAME, JUMP
03340           POP     HL      ;IF NOT,CONTINUE
03350           LD      DE,18
03360           ADD     HL,DE
03370           EX      DE,HL
03380           POP     HL
03390           DJNZ    STRING
03400           LD      A,(FLAG8)
03410           CP      0
03420           RET     NZ      ;ITS AN ERROR
03430 STRIN1    PUSH    DE      ;STORE NEW BACKG/CHR/MV IN MEMORY
03440           LD      B,8
03450 STRIN2    LD      A,(HL)
03460           LD      (DE),A
03470           CP      20H
03480           JR      Z,STRINA
03490           INC     DE
03500           INC     HL
03510           DJNZ    STRIN2
03520           JR      STRINB
03530 STRINA    INC     HL
03540           INC     DE
03550           DJNZ    STRINA
03560 STRINB    DEC     DE
03570           LD      A,13
03580           LD      (DE),A
03590           INC     DE
03600           POP     BC      ;CONTAINS ADDRESS TO STORE NAME
03610           DEC     BC
03620           LD      A,(BC)
03630           LD      H,A
03640           DEC     BC
03650           LD      A,(BC)
03660           LD      L,A
03670           LD      BC,(BYTE)
03680           ADD     HL,BC
03690           LD      A,L
03700           LD      (DE),A
03710           INC     DE
03720           LD      A,H
03730           LD      (DE),A
03740           CP      0FFH
03750           RET     NZ
03760 STRIN3    POP     HL
03770           LD      DE,8
03780           ADD     HL,DE
03790           EX      DE,HL
03800           LD      A,(DE)
03810           LD      L,A
03820           INC     DE
03830           LD      A,(DE)
03840           LD      H,A
03850           POP     IX      ;RESTORE STAK
03860           LD      A,(FLAG8)
03870           CP      1
03880           RET     Z
03890           PUSH    HL
03900           LD      HL,(ALLBUF)
03910           DEC     (HL)
03920           POP     HL      ;HL=CONTENTS OF BACK/CHR/MV IN MEM
03930           XOR     A
03940           CP      0
03950           RET     Z       ;RET WITH Z FLAG
03960 COPY      LD      A,(FLAG6)
03970           OR      A
03980           JR      NZ,COPY1
03990           LD      DE,(CHRBUF)     ;FIND NAME OF CHARACTER
04000           JR      COPY2
04010 COPY1     LD      DE,(BCKBUF)     ;FIND NAME OF BACKGROUND
04020 COPY2     LD      A,(DE)
04030           LD      B,A
04040           INC     DE
04050           CALL    STRING
04060           JP      NZ,ERR2
04070           RET
04080 ;         INPUT CHARACTERS,ETC,
04090 TRANFR    LD      DE,42E8H        ;TRANSFER INPUT ON SCREEN TO A BUFFER
04100           PUSH    DE
04110           LD      B,8
04120 TRANF1    LD      A,(HL)
04130           LD      (DE),A
04140           LD      A,20H
04150           LD      (HL),A
04160           INC     HL
04170           INC     DE
04180           DJNZ    TRANF1
04190           POP     HL
04200           RET
04210 TRANP2    LD      DE,EXTRA        ;INPUT CHARACTER
04220 TRANF3    LD      HL,3C00H+598
04230           LD      A,8
04240           LD      (LEN),A
04250           CALL    BOX
04260           LD      HL,(SCRIN)
04270           LD      A,(HL)
04280           CP      '-'
04290           RET     Z       ;DO NOT WANT BASIS CHAR
04300 TRANFA    LD      A,1
04310           LD      (FLAG8),A
04320           CALL    COPY
04330           XOR     A
04340           LD      (FLAG8),A
04350           CP      1
04360           RET     NZ
04370 TRANF4    LD      DE,EXTRAB       ;INPUT BACKGROUND
04380           LD      A,1
04390           LD      (FLAG6),A
04400           LD      HL,3C00H+598
04410           CALL    BOX
04420           LD      HL,(SCRIN)
04430           LD      A,(HL)
04440           CP      '-'
04450           JP      NZ,TRANFA
04460           CALL    01C9H
04470           JR      TRANF4
04480 ;         MAKE A BACKGROUND
04490 DEFB1     CALL    TRANFR
04500           PUSH    HL
04510           CALL    01C9H
04520           LD      HL,(BCKBUF)
04530           LD      (ALLBUF),HL
04540           LD      HL,960
04550           LD      (BYTE),HL
04560           POP     HL
04570           LD      DE,(BCKBUF)
04580           LD      A,(DE)  ;A=NUMBER OF SCREENS
04590           INC     DE
04600           CP      0
04610           JP      Z,DEFB2
04620           LD      B,A
04630           LD      (MISC),A
04640           CALL    STRING
04650           JR      NZ,DEFB1A
04660           PUSH    HL      ;DISPLAY BACKGROUND IF PREVIOUSLY DEFINED
04670           LD      DE,3C00H+64
04680           LD      BC,960
04690           LDIR
04700           POP     HL
04710           JR      DEFB1B
04720 DEFB1A    LD      A,(MISC)        ;FIND # OF BCKGROUNDS
04730           CP      10      ;MAX NUMBER
04740           JP      Z,ERR1
04750 DEFB1B    PUSH    HL
04760           CALL    BCKGRO  ;CALL ROUTINE TO DRAW
04770           POP     DE
04780           LD      HL,15360+64
04790           LD      BC,1024-64
04800           LDIR    ;COPY SCREEN INTO MEM
04810           LD      HL,(BCKBUF)
04820           INC     (HL)
04830           CALL    01C9H
04840           XOR     A
04850           LD      (FLAG6),A
04860           CALL    01C9H
04870           JP      INPUT   ;RETURN TO MAIN MENU
04880 DEFB2     PUSH    HL      ;THIS IS FIRST BACKGROUND
04890           CALL    01C9H
04900           POP     HL
04910           LD      DE,(BCKBUF)
04920           INC     DE
04930           LD      BC,BEGBAK+2     ;BEG OF MEM TO STORE BCKG
04940           LD      IX,DEFB1A
04950           PUSH    IX
04960           PUSH    BC
04970           LD      B,8
04980           JP      STRIN2
04990 ;         MAKE A CHARACTER
05000 DEFC1     CALL    TRANFR
05010           PUSH    HL
05020           CALL    01C9H
05030           LD      HL,(CHRBUF)
05040           LD      (ALLBUF),HL
05050           LD      HL,144
05060           LD      (BYTE),HL
05070           POP     HL
05080           LD      DE,(CHRBUF)
05090           LD      A,(DE)  ;A = # OF CHARACTERS
05100           CP      0
05110           JP      Z,DEFC2
05120           LD      B,A
05130           LD      (MISC),A
05140           INC     DE
05150           CALL    STRING
05160           JR      Z,$+10
05170           LD      A,(MISC)
05180           CP      18
05190           JP      Z,ERR1
05200           LD      (MISC),HL
05210           PUSH    AF
05220           LD      HL,(ALLBUF)
05230           INC     (HL)
05240           CALL    TRANF2  ;SEE IF WANT BASIS CHARACTER
05250           PUSH    AF
05260           PUSH    HL
05270           LD      HL,(ALLBUF)
05280           DEC     (HL)
05290           CALL    01C9H
05300           POP     HL
05310           POP     AF
05320           JR      NZ,DEFC10       ;WANT A BASIS CHAR
05330           POP     AF
05340           LD      HL,(MISC)
05350           JP      NZ,DEFC1A
05360           JR      DEFC12
05370 DEFC10    POP     AF
05380 DEFC12    LD      DE,3C00H+277
05390           EX      DE,HL
05400           CALL    TRANS   ;COPY CHARACTER TO SCREEN IF ALREADY DEFINED
05410           LD      HL,(MISC)
05420 DEFC1A    PUSH    HL
05430           CALL    CHRCTR
05440           POP     DE
05450           LD      HL,3C00H+277
05460           CALL    TRANS3  ;ASK FOR BASIS CHAR
05470           LD      HL,(CHRBUF)
05480           INC     (HL)
05490           CALL    01C9H
05500           JP      INPUT
05510 DEFC2     PUSH    HL
05520           CALL    01C9H
05530           POP     HL
05540           LD      DE,(CHRBUF)
05550           INC     DE
05560           LD      BC,BEGCHR+2
05570           LD      IX,DEFC1A
05580           PUSH    IX
05590           PUSH    BC
05600           LD      B,8
05610           JP      STRIN2
05620 ;         MAKE THE MOVES
05630 DEFM1     CALL    TRANFR
05640           LD      (MISC),HL
05650           LD      HL,(MVBUFF)
05660           LD      (ALLBUF),HL
05670           LD      HL,256
05680           LD      (BYTE),HL
05690           LD      DE,(MVBUFF)
```

Listing 1 continued

↙ See List of Advertisers on Page 211

Listing 1 continued

```
05700           LD      A,(DE)  ;A=NUMBER OF MOVES
05710           PUSH    AF
05720           LD      (MISC1),DE
05730           CALL    TRANP2  ;GET INPUT FOR CHR
05740           JR      Z,$-3
05750           LD      (CPYCHR),HL
05760           CALL    01C9H
05770           CALL    TRANP4  ;GET INPUT FOR BACKGROUND
05780           JR      Z,$-3
05790           XOR     A
05800           LD      (FLAG6),A
05810           PUSH    HL
05820           CALL    01C9H
05830           POP     HL
05840           LD      DE,15360+64
05850           LD      BC,1024-64
05860           LDIR    ;COPY BACKGROUND TO SCREEN
05870           LD      DE,(MISC1)
05880           POP     AF
05890           LD      HL,(MISC)       ;RESTORE NAME OF MOVE
05900           LD      (DE),A
05910           CP      0
05920           JP      Z,DEFM2
05930           LD      B,A
05940           LD      (MISC),A
05950           INC     DE
05960           CALL    STRING
05970           JR      NZ,DEFM1A
05980           LD      B,(HL)
05990           LD      A,B
06000           CP      0
06010           JR      Z,DEFM1D
06020           PUSH    HL
06030           INC     HL
06040           INC     HL
06050 DEFM1B    LD      E,(HL)
06060           INC     HL
06070           LD      D,(HL)
06080           LD      A,'-'
06090           LD      (DE),A
06100           INC     HL
06110           DJNZ    DEFM1B
06120           LD      (PLACE1),HL
06130           POP     HL
06140           LD      (PLACE),HL
06150           JR      DEFM1C
06160 DEFM1A    LD      A,(MISC)
06170           CP      10
06180           JP      Z,ERR1
06190 DEFM1D    LD      (PLACE),HL
06200           INC     HL
06210           INC     HL
06220           LD      (PLACE1),HL
06230 DEFM1C    CALL    DEFMV
06240           LD      HL,(MVBUFF)
06250           INC     (HL)
06260           CALL    01C9H
06270           JP      INPUT
06280 DEFM2     LD      DE,(MVBUFF)
06290           INC     DE
06300           LD      BC,BEGMOV+2
06310           LD      IX,DEFM1A
06320           PUSH    IX
06330           PUSH    BC
06340           LD      B,8
06350           JP      STRIN2
06360 ;         DISPLAY NAMES OF CHARS/MVS/BCKGROUND
06370 CHOOSE    LD      A,1
06380           LD      (POS),A
06390           LD      HL,3C00H+192
06400           LD      A,(DE)  ;HL=ADRS OF BACK,CHR,OR MOVE BUFF
06410           CP      0
06420           JP      Z,CHOOS6
06430           INC     DE
06440           LD      B,A
06450 CHOOS1    PUSH    BC
06460           PUSH    DE
06470           LD      B,16
06480 CHOOS2    LD      A,(DE)
06490           CP      13
06500           JR      Z,CHOOS3
06510           LD      (HL),A
06520           INC     HL
06530           INC     DE
06540           DJNZ    CHOOS2
06550           JR      CHOOS4
06560 CHOOS3    LD      A,(PLAG11)
06570           CP      1
06580           CALL    Z,SHONUM        ;DISPLAY # OF MOVES
06590 CHOOSA    LD      A,20H
06600           LD      (HL),A
06610           INC     HL
06620           DJNZ    CHOOSA
06630 CHOOS4    POP     DE
06640           LD      B,10
06650 CHOOS5    INC     DE
06660           DJNZ    CHOOS5
06670           POP     BC
06680           DJNZ    CHOOS1
06690           LD      A,(FLAG10)
06700           CP      1
06710           RET     Z
06720           LD      HL,15360+192
06730           LD      A,1
06740           LD      (FLAG9),A
06750           CALL    CURSOR
06760           XOR     A
06770           LD      (FLAG9),A
06780           CALL    01C9H
06790           RET
06800 CHOOS6    LD      A,(FLAG10)
06810           OR      A
06820           RET     NZ
06830           POP     IX
06840           LD      HL,(EPROG)
06850           XOR     A
06860           LD      (HL),A
06870           CALL    01C9H
06880           JP      INPT1A  ;SKIP IT SINCE NO CHAR/MV/OR BCKG
06890 SHONUM    LD      A,B
06900 SHONM1    INC     DE
06910           DEC     A
06920           DEC     B
06930           INC     HL
06940           CP      8       ;POINT TO ADD WHERE MOVES STORED
06950           JR      NZ,SHONM1
06960           PUSH    BC
06970           PUSH    HL
06980           LD      A,(DE)
06990           LD      C,A
07000           INC     DE
07010           LD      A,(DE)
07020           LD      B,A
07030           LD      A,(BC)
07040           LD      L,A
07050           LD      H,0
07060           CALL    LISNU0
07070           POP     HL
07080 SHONU2    LD      A,(BC)
07090           CP      0
07100           LD      (HL),A
07110           INC     HL
07120           INC     BC
07130           JR      NZ,SHONU2
07140           POP     BC
07150           DEC     B
07160           DEC     B
07170           DEC     B
07180           DEC     B
07190           RET
07200 ;         ACTION!!
07210 RUN1      CALL    01C9H
07220           LD      HL,1000
07230           LD      (RATE),HL
07240           XOR     A
07250           LD      (DATFLG),A
07260           LD      HL,8900H        ;RESET LOOP BUFFER
07270           LD      (HL),A
07280           LD      DE,8901H
07290           LD      BC,255
07300           LDIR
07310           LD      HL,(MVBUFF)     ;RESET MOVES
07320           LD      B,(HL)
07330           LD      A,B
07340           OR      A
07350           JR      Z,RUN12
07360           LD      DE,256
07370           INC     HL
07380           XOR     A
07390 RUN11     ADD     HL,DE
07400           LD      (HL),A
07410           DJNZ    RUN11
07420 RUN12     LD      HL,(BPROG)
07430           LD      A,(HL)
07440           CP      0
07450           JP      Z,RUN1B
07460 RUN1A     PUSH    HL
07470           CALL    RUN3    ;FIND PLACE IN MEM
07480           LD      BC,(RATE)
07490           CALL    60H     ;PAUSE
07500           POP     HL
07510           INC     HL
07520           INC     HL
07530           INC     HL
07540           CALL    0028H
07550           CP      1
07560           JP      Z,RUN1B
07570           LD      A,(HL)
07580           CP      0
07590           JR      NZ,RUN1A
07600           CALL    49H
07610 RUN1B     CALL    01C9H
07620           JP      INPUT
07630 RUN3      LD      A,(HL)  ;FIND PLACE IN TABLE
07640           DEC     A
07650           PUSH    HL
07660           LD      HL,2
07670           CALL    444EH   ;MULTIPLY
07680           LD      L,A
07690           LD      DE,TABLE
07700           ADD     HL,DE   ;FIND TOKEN ADDRESS IN TABLE
07710           LD      E,(HL)
07720           INC     HL
07730           LD      D,(HL)
07740           PUSH    DE
07750           POP     HL
07760           JP      (HL)    ;JUMP TO THAT LOCATION
07770 CLR       LD      A,1
07780 CLR0      LD      (HL),A
07790           JP      POINT
07800 ;         CLEAR SCREEN
07810 CLR1      POP     IX
07820           CALL    01C9H
07830           RET
07840 CURT      LD      A,2
07850 CURT10    LD      (HL),A
07860           INC     HL
07870           PUSH    HL
07880           CALL    01C9H
07890           LD      HL,EXTRAB       ;MSSGE TO ASK FOR BCKGRND
07900           CALL    021BH
07910           LD      DE,(BCKBUF)
07920           CALL    CHOOSE
07930           LD      A,(POS)
07940           POP     HL
07950           LD      (HL),A
07960           JP      POINT
07970 ;         DISPLAY BCKGR WITH CURTAIN
07980 CURT2     POP     HL
07990           INC     HL
08000           LD      A,(HL)
08010           LD      HL,10
08020           CALL    444EH
08030           LD      DE,(BCKBUF)
08040           CALL    POSIT
08050           PUSH    HL
08060           LD      HL,3C00H
08070           LD      A,191
08080           LD      (HL),A
08090           LD      DE,3C01H
08100           LD      BC,1023
08110           LDIR
08120           POP     HL
08130           LD      DE,32
08140           ADD     HL,DE
08150           PUSH    HL
08160           LD      DE,3C00H+96
08170           SBC     HL,DE
08180           LD      (ADD),HL
08190           LD      DE,0
08200           POP     HL
08210 CURT3     PUSH    HL
08220           PUSH    DE
08230           ADD     HL,DE
08240           CALL    CURT40
08250           POP     DE
08260           POP     HL
08270           PUSH    HL
08280           PUSH    DE
08290           SBC     HL,DE
08300           CALL    CURT40
08310           LD      BC,8000
08320           CALL    60H
08330           POP     DE
08340           POP     HL
08350           INC     DE
08360           LD      A,E
08370           CP      32
08380           JP      NZ,CURT3
08390           SBC     HL,DE
08400           CALL    CURT40
08410           RET
08420 CURT40    LD      B,15
08430           LD      A,20H
08440           LD      DE,(ADD)
08450           PUSH    HL
08460           SBC     HL,DE
08470           LD      DE,64
08480           SBC     HL,DE
08490           LD      (HL),A
08500           POP     HL
```

Listing 1 continued

OPEN UP AN ACCOUNTING DEPARTMENT FOR $395.00

GENERAL LEDGER integrated postings from A/R, A/P and Payroll. **Prints 13 detailed reports** • Company or departmental Income Statements • **Comparative financial statements** with current, YTD, budget, and last year (month and YTD) • Presents everything you, your bookkeeper, and your accountant need to know • G/L reconciles all accounts and maintains extensive, detailed audit trails • Trial Balance includes all transactions • **Flexible Chart of Accounts** • True double-entry bookkeeping • Master File capacity: 400 accounts • Monthly Transaction capacity: 1,000 with 200K diskette; 3,500 with 500K diskette; 7,000 per Megabyte with a hard disk.

ACCOUNTS RECEIVABLE provides **instant, on-line customer account information (both current and aged),** complete invoicing (open-item or balance forward) and statement capabilities on optional preprinted forms give your company a professional image • Quickly identify overdue accounts, speed collections, help control cash flow • Detailed and summary customer activity and aging reports • Produces 8 reports • Automatic periodic customer/client billing option • Itemized monthly transactions • Master File capacity: 400 Customers • Monthly Transactions capacity: 800 with 200K diskette; 3,500 with 500K diskette; 7,000 per Megabyte with a hard disk.

ACCOUNTS PAYABLE maintains complete vendor/voucher history and includes **check-writing** capabilities • **Current and aged payable reports** • Cash flow/cash requirements report • Prints checks with comprehensive check stubs • **Produces 11 reports and documents** • Automatic pay selection program allows payment by due date or by discount date • **Manual and automatic checkwriting** • Check register • Master File capacity: 400 Vendors • Monthly Transactions capacity: 800 with 200K diskette; 3,500 with 500K diskette; 7,000 per Megabyte with a hard disk.

PAYROLL—Be the office hero each week when the checks come out on time! • **Calculates payroll for every type of employee** (hourly, salaried, and commissioned) and **prints payroll checks** (with popular, comprehensive check stubs) with an absolute minimum of input • Maintains monthly, quarterly, and yearly totals for reporting in **multiple states** • **User-maintainable Federal, State, and local tax tables** • W-2 printing • 941 Reporting • Produces 10 reports • Master File capacity: 400 employees.

• California residents add 6½% Sales Tax • Payment by VISA/MasterCard/COD/MO/Cashier's Check • All Brand Names are manufacturers' registered trademarks • No sales to Dealers • Foreign orders please call or write before ordering •

MasterCard VISA

Why staff up? With the Desktop Accountant™, all the accounting help your office needs can be at your fingertips!

Open up a wide new range of possibilities for your microcomputer! No matter what type of business you're in, Desktop Accountant will let you manage the financial end of it more professionally than ever before.

A Complete System with Support. Desktop Accountant includes General Ledger, Accounts Receivable, Accounts Payable and Payroll programs, along with comprehensive user manuals and training aids. We've even prepared an audio cassette tape to make learning the system fast and fun. And our telephone "hotline" means personalized support whenever you need it.

Desktop Accountant's fully integrated accounting system is a complete package of software, training aids, manuals and user's newsletter.

Produces 42 Reports. "Keeping the books" has never been so easy! Desktop Accountant prepares every bookkeeping and accounting report your growing business requires: from invoicing to statements to aged A/R listings; from cash distribution to A/P checks to vendor activity reports; from **complete** payroll checks and stubs to W-2 forms; from the chart of accounts to balance sheet and income statement, as well as many others so vital to efficient management.

Desktop Accountant is available for nearly every portable, personal and desktop computer. The system requires either CP/M® or MS-DOS™ (PC-DOS), Microsoft BASIC™, 64K RAM, two disk drives or hard disk, and a 132-column printer (or an 8½" x 11" printer with compressed print mode).

You won't find better quality software at such a low price—a price we can offer now because development costs were recovered years ago. Just **$395.00 for most CP/M® formats ($495.00 for IBM® and some CP/M formats) complete.** *Call for available formats.*

Order Desktop Accountant today! You'll soon see why we call it **INTELLIGENT SOFTWARE FOR INTELLIGENT PEOPLE.**

ROCKY MOUNTAIN SOFTWARE SYSTEMS
Business Microcomputers and Software

To order Desktop Accountant and for more comprehensive literature, call toll-free:

☎1-800-832-2244
(In California call 1-800-732-2311)

or send orders to:
DESKTOP ACCOUNTANT
1280-C Newell Avenue, Suite 1221
Walnut Creek, CA 94596

Listing 1 continued

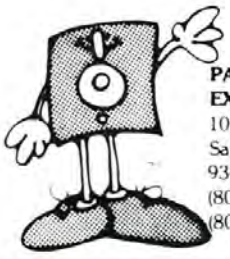
```
08510 CURT4   LD      DE,(ADD)        ;OVERLAY BACKGROUND OVER WHITE SCREEN: HORIZO
08520         LD      A,(HL)
08530         PUSH    HL
08540         SBC     HL,DE
08550         LD      (HL),A
08560         LD      DE,64
08570         POP     HL
08580         ADD     HL,DE
08590         DJNZ    CURT4
08600         RET
08610 LPLIN   LD      A,5
08620         LD      (HL),A
08630         INC     HL
08640         PUSH    HL
08650         LD      HL,3C00H+726
08660         LD      DE,LINNUM
08670         LD      A,3
08680         LD      (LEN),A
08690         CALL    BOX
08700         LD      HL,(SCRIN)
08710         CALL    1E5AH    ;DECODE #
08720         LD      A,E
08730         POP     HL
08740         LD      (HL),A
08750         INC     HL
08760         PUSH    HL
08770         LD      HL,3C00H+726
08780         LD      DE,TIMNUM
08790         CALL    BOX
08800         LD      HL,(SCRIN)
08810         CALL    1E5AH
08820         LD      A,E
08830         POP     HL
08840         LD      (HL),A
08850         JP      POINT
08860 LOOP2   POP     HL       ;LOOP ROUTINE FOR LOOPING
08870         INC     HL
08880         LD      A,(HL)
08890         PUSH    HL
08900         LD      E,A
08910         LD      D,0
08920         LD      HL,8900H
08930         DEC     DE
08940         ADD     HL,DE
08950         LD      A,(HL)   ;FIND # OF TIMES TO LOOP
08960         LD      (MISC1).HL
08970         CP      0
08980         JR      NZ,LOOP2A
08990         POP     HL
09000         INC     HL
09010         LD      A,(HL)
09020         DEC     A
09030         CP      255
09040         JR      Z,LOOP21
09050         CP      0
09060         JR      NZ,$+3
09070         INC     A
09080         LD      HL,(MISC1)
09090         LD      (HL),A
09100         JR      LOOP2C
09110 LOOP2A  CP      1        ;IS LOOP DONE?
09120         JR      NZ,LOOP2B
09130         POP     HL
09140 LOOP21  LD      HL,(MISC1)
09150         XOR     A
09160         LD      (HL),A
09170         RET
09180 LOOP2B  POP     HL
09190         LD      HL,(MISC1)
09200         DEC     (HL)
09210 LOOP2C  LD      A,E      ;FIND LINE # IN MEMORY
09220         LD      HL,3
09230         CALL    444EH
09240         LD      E,A
09250         LD      D,0
09260         LD      HL,(BPROG)
09270         ADD     HL,DE
09280         DEC     HL
09290         DEC     HL
09300         DEC     HL
09310 LOOP4   POP     IX       ;IX=RET TO 'RUN' ROUTINE
09320         POP     DE       ;POP OFF PROGRAM POINTER
09330         PUSH    HL       ;STORE NEW PROG POINTER
09340         PUSH    IX       ;RESTORE RETURN ADDRESS
09350         RET
09360 MOVCHR  LD      A,6
09370         LD      (HL),A
09380         INC     HL
09390         PUSH    HL
09400         CALL    01C9H
09410         LD      HL,EXTRA
09420         CALL    021BH
09430         LD      DE,(CHRBUF)
09440         CALL    CHOOSE
09450         LD      A,(POS)
09460         POP     HL
09470         LD      (HL),A
09480         INC     HL
09490         JP      RESTR1
09500 MOVE2   POP     HL       ;MOVE ROUTINE
09510         INC     HL
09520         PUSH    HL
09530         LD      A,(HL)
09540         LD      HL,10
09550         CALL    444EH
09560         LD      DE,(CHRBUF)
09570         CALL    POSIT
09580         LD      (CPYCHR),HL
09590         POP     HL
09600         INC     HL
09610         LD      A,(HL)
09620 MOVE2A  PUSH    AF
09630         PUSH    HL
09640         LD      HL,144
09650         DEC     A
09660         CALL    444EH
09670         LD      H,L
09680         LD      L,A
09690         LD      DE,(SAVBUF)
09700         ADD     HL,DE
09710         LD      (SCRBUF),HL
09720         POP     HL
09730         POP     AF
09740         LD      HL,10
09750         CALL    444EH
09760         LD      DE,(MVBUFF)
09770         CALL    POSIT
09780         LD      B,(HL)   ;GET # OF MOVES
09790         INC     HL
09800         LD      A,(HL)   ;GET # OF MOVES COMPLETED
09810         PUSH    AF
09820         LD      A,(FLAG13)
09830         OR      A
09840         JR      NZ,MOVE21
09850         POP     AF
09860         CP      B
09870         RET     Z
09880         JR      $+3
09890 MOVE21  POP     AF
09900         PUSH    HL
09910         LD      HL,2
09920         CALL    444EH
09930         LD      D,0
09940         LD      E,A
09950         POP     HL
09960         PUSH    HL
09970         INC     HL
09980         ADD     HL,DE    ;POINT TO CORRECT MOVE
09990         LD      E,(HL)
10000         INC     HL
10010         LD      D,(HL)
10020         PUSH    DE
10030         LD      (MISC1),A
10040         LD      A,(FLAG13)
10050         OR      A
10060         JR      NZ,MOVE22
10070         LD      A,(MISC1)
10080         CP      0        ;IS THIS FIRST MOVE
10090         JR      Z,MOVE2B
10100 MOVE22  DEC     HL
10110         DEC     HL
10120         LD      D,(HL)
10130         DEC     HL
10140         LD      E,(HL)
10150         PUSH    DE
10160         POP     HL
10170         LD      DE,(SCRBUF)      ;BUFFER TO STORE SCREEN
10180         CALL    TRANS    ;RESTORE PREVIOUS CONTENTS OF SCREEN
10190 MOVE2B  POP     HL
10200         LD      A,(FLAG13)
10210         OR      A
10220         JR      Z,$+4
10230         POP     DE
10240         RET
10250         PUSH    HL
10260         LD      DE,(SCRBUF)      ;BUFFER TO STORE PART OF SCREEN
10270         CALL    TRANS3
10280         POP     HL       ;RESTORE COORD OF SCREEN
10290         LD      DE,(CPYCHR)
10300         CALL    TRANS6   ;COPY CHARACTER TO SCREEN
10310         POP     HL       ;RESTORE PLACE IN MEMORY
10320         INC     (HL)
10330         RET
10340 DIALOG  LD      A,3      ;STATEMENT TO PRINT SCRIPT
10350         JP      CLR0
10360 DIALG1  POP     HL
10370         LD      HL,(BSCRIP)
10380         LD      A,(DATFLG)
10390         INC     A
10400         LD      (DATFLG),A
10410         DEC     A
10420         OR      A
10430         JR      Z,DIALG3
10440         LD      B,A
10450 DIALG2  INC     HL
10460         LD      A,(HL)
10470         CP      0
10480         RET     Z
10490         CP      3
10500         JR      NZ,DIALG2
10510         DJNZ    DIALG2
10520         INC     HL
10530         LD      A,(HL)
10540         CP      0
10550         RET     Z
10560 DIALG3  LD      A,(HL)
10570         CP      0
10580         RET     Z
10590         LD      B,63
10600         LD      DE,3C00H
10610 DIALG4  LD      A,(HL)   ;DISPLAY IT!!!
10620         CP      3
10630         JR      Z,DIALG5
10640         LD      (DE),A
10650         INC     HL
10660         INC     DE
10670         DJNZ    DIALG4
10680         RET
10690 DIALG5  LD      A,20H
10700 DIALG6  LD      (DE),A
10710         INC     DE
10720         DJNZ    DIALG6
10730         RET
10740 DISPL   LD      A,4
10750         JP      CURT10
10760 DISPL1  POP     HL       ;DISPLAY BACKGROUND
10770         INC     HL
10780         LD      A,(HL)
10790         LD      HL,10
10800         CALL    444EH
10810         LD      DE,(BCKBUF)
10820         CALL    POSIT
10830         LD      DE,3C00H+64
10840         LD      BC,960
10850         LDIR
10860         RET
10870 RESTOR  LD      A,7
10880         LD      (HL),A
10890         INC     HL
10900 RESTR1  PUSH    HL
10910         CALL    01C9H
10920         LD      HL,MOVES
10930         CALL    021BH
10940         LD      DE,(MVBUFF)
10950         CALL    CHOOSE
10960         LD      A,(POS)
10970         POP     HL
10980         LD      (HL),A
10990         JP      POINT
11000 ;       RESTORE MOVES
11010 RESTR2  POP     HL
11020         INC     HL
11030         LD      A,(HL)
11040         LD      HL,10
11050         CALL    444EH
11060         LD      DE,(MVBUFF)
11070         CALL    POSIT
11080         INC     HL
11090         XOR     A
11100         LD      (HL),A
11110         RET
11120 SPEED   LD      A,8
11130         LD      (HL),A
11140         INC     HL
11150         PUSH    HL
11160         LD      DE,SPEEDS
11170         LD      HL,3C00H+726
11180         LD      A,1
11190         LD      (LEN),A
11200         CALL    BOX
11210         LD      HL,(SCRIN)
11220         CALL    1E5AH    ;DECODE INTO HEX
11230         LD      A,E
11240         OR      A
11250         JR      NZ,SPEEDA
11260         INC     A
11270 SPEEDA  POP     HL
11280         LD      (HL),A
11290         JP      POINT
11300 SPEED2  POP     HL       ;SPEED OF PROG EXECUTION
11310         RST     10H
```

Listing 1 continued

MEMOREX
FLEXIBLE DISCS

WE WILL NOT BE UNDER-SOLD!! Call Free **(800)235-4137** for prices and information. Dealer inquiries invited and C.O.D.'s accepted.

VISA



PACIFIC EXCHANGES
100 Foothill Blvd.
San Luis Obispo, CA
93401. In Cal. call
(800)592-5935 or
(805)543-1037

207

PROGRAM eXchange / ORION SYSTEMS INC.

TRADE YOUR USED, UNWANTED (GOOD CONDITION) SOFTWARE FOR SOFTWARE NOT IN YOUR LIBRARY. WE HAVE GAMES, EDUCATION, AND BUSINESS SOFTWARE ☆ IN STOCK ☆ DON'T SPEND $20 $30 $40 OR MORE WHEN YOU CAN EXCHANGE PROGRAMS FOR ONLY

☆☆☆ $7.95 ☆☆☆

For more information:
ORION SYSTEMS INC. 108
12275 59th STREET N.
ROYAL PALM BEACH, FL 33411

The Phone Line

Hwy. 11 South Trenton, GA
1-404-657-6948

New! Low Prices!

RADIO SHACK:
(All equip is 100% pure RS)

Model 100 8k ram	$ 650
NEC Version. 16k ram	$ 650
Model 12. 2-drive	$3250
Model 4. 64k. 2-drive	$1650
5 meg hard drive	$1658
DMP-2100 printer	$1599
DMP-200 printer	$ 599

All Radio Shack Software 10% Off
Call for other
LOW. LOW RS prices not listed.

TeleVideo

Portable. 1 drive	$1439
Portable. 2 drive	$1795

Comes with spreadsheet. word processing. graphics. and CP/M

Miscellaneous

DaisyWriter w/48k buffer & cable	$1295
ProWriter 8510 printer	$ 399
ProWriter 1550 printer	$ 675
Okidata ML-92P printer	$ 525
Okidata ML-80P printer	$ 350
Okidata ML-82PS printer	$ 450
Box 10 Verbatim 5¼" diskettes (SS DD)	$ 29
Box 10 Verbatim 8" diskettes (SS DD)	$ 49

QUANTITY PRICING AVAILABLE ON DISKETTES . . . CALL!

5¼" flip file. holds 50	$ 25
8" flip file. holds 50	$ 35
5¼" flip-pak. holds 10	$ 5
8" flip-pak. holds 10	$ 6
Disk drive head cleaner kit (Verbatim)	$ 5
9½" x 11" paper. 1250 shts. disaperf	$ 21
14⅞" x 11" green bar. 1500 shts	$ 30
ProWriter Ribbons	$ 6
J-Cat Direct Connect Modem	$ 125

Complete line of accessories — CALL. CALL. CALL
(Prices subject to change.)

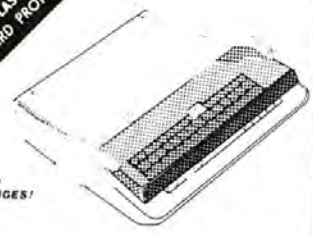
We accept American Express. VISA. MasterCard (Add 3%)

THIS MONTH'S SPECIAL!
Verbatim Verex 5¼" Diskettes (SS,DD)..... $24

509

The Phone Line
Hwy. 11 South Trenton, GA
1-404-657-6948

K-COVER
RIGID SMOKED PLASTIC COMPUTER KEYBOARD PROTECTION



NOW WITH EASY-TILT HINGES!

INTRODUCING **K-COVER** AN ANTI-STATIC UNBREAKABLE COMPUTER KEYBOARD PROTECTOR MADE OF ATTRACTIVE SMOKED GREY PLASTIC WITH MAR-RESISTANT RUBBER FEET. **K-COVER** PROTECTS YOUR KEYBOARD FROM DUST. DIRT. AND WANDERING FINGERS. **K-COVER** ALSO DOUBLES AS A TILT RISER FOR COMPUTER OR MONITOR AND COMES WITH LIFETIME WARRANTY AGAINST BREAKAGE. ONLY **$7.95 PLUS $1.00 POSTAGE AND HANDLING**. CHECK. MONEY ORDER. VISA AND MASTER CARD ACCEPTED.

PENGUIN PRODUCTS
P.O. BOX 7008
ROSEVILLE. MICHIGAN 48305-7008
1 (800) 732-0614
1(313) 293-5210

127

Sure it's insured?

SAFEWARE™ Insurance provides full replacement of hardware. media and purchased software. As little as $35/yr covers:
• Fire • Theft • Power Surges
• Earthquake • Water Damage • Auto Accident

For information or immediate coverage call:

1-800-848-3469

In Ohio call (614) 262-0559

SAFEWARE

368

SAFEWARE. THE INSURANCE AGENCY INC.

Trend Software Package

1. Bar Charts
2. Linear Regression
3. Statistics
4. Histograms

$99.95

(Add $1.50 for S & H)
(PA Residents Add 6% Sales Tax)
Make Checks or Money Orders Payable To:

Selective Software Systems

SSS

4757 Liberty Avenue
Pittsburgh, PA 15224

53

DATA COMPACTOR

Increase data storage capacity 25% to 100% without using machine-code!

These routines are written in BASIC SOURCE CODE which may be integrated into user's programs in direct sequence or as subroutines. This manual will show you how to do it yourself.
Send $49.95 & $2.00 USA shipping
Demo disks available with mergeable routines for CP/M or TRSDOS compatible routines for TRS-80, II, or III.
Specify CPM or TRSDOS & add $19.95 for 5¼" disk or $24.95 for 8" disk.

VISA **Dealers Invited**

EDWARDS & ASSOCIATES
P.O. BOX 42158
COLUMBIA, S.C. 29240

435

See List of Advertisers on Page 211

Listing 1 continued

```
11320           LD      HL,1000
11330           CALL    444EH
11340           LD      H,L
11350           LD      L,A
11360           LD      (RATE),HL
11370           RET
11380 DELET     LD      A,9
11390           LD      (HL),A
11400           INC     HL
11410           PUSH    HL
11420           CALL    01C9H
11430           LD      HL,MVCHR        ;PROMPT
11440           CALL    021BH
11450           LD      DE,(MVBUFF)
11460           CALL    CHOOSE
11470           LD      A,(POS)
11480           POP     HL
11490           LD      (HL),A
11500           JP      POINT
11510 ;         CLEAR CHARACTER
11520 DELET2    POP     HL
11530           INC     HL
11540           LD      A,1
11550           LD      (FLAG13),A
11560           LD      A,(HL)
11570           CALL    MOVE2A
11580           XOR     A
11590           LD      (FLAG13),A
11600           RET
11610 SCRIPT    CALL    01C9H
11620           LD      A,1
11630           LD      (POS),A ;RESTORE POSITION
11640           LD      A,(FLAG12)
11650           OR      A
11660           JP      NZ,INPT1A
11670           LD      HL,(SCRMAX)
11680           LD      DE,(ESCRIP)
11690           RST     18H
11700           JP      Z,INPT1A
11710           LD      HL,SCRIPS
11720           CALL    021BH
11730 SCRIPA    LD      HL,3C00H+128
11740           LD      (4020H),HL
11750           LD      DE,3C00H+129
11760           LD      A,'-'
11770           LD      (HL),A
11780           LD      BC,60
11790           LDIR
11800           LD      HL,0FD00H
11810           LD      B,60
11820           CALL    40H
11830           JP      C,SCRIP3
11840           LD      A,(HL)
11850           CP      0DH
11860           JP      Z,SCRIPA
11870           LD      DE,(ESCRIP)
11880 SCRIP2    LD      A,(HL)
11890           LD      (DE),A
11900           INC     DE
11910           INC     HL
11920           PUSH    HL
11930           LD      HL,(SCRMAX)
11940           RST     18H
11950           JR      Z,SCRIP4
11960           POP     HL
11970           DJNZ    SCRIP2
11980           LD      A,3
11990           LD      (DE),A
12000           INC     DE
12010           LD      (ESCRIP),DE
12020           XOR     A
12030           LD      (DE),A
12040           INC     DE
12050           LD      (DE),A
12060           JP      SCRIPA
12070 SCRIP3    CALL    01C9H
12080           JP      INPT1A
12090 SCRIP4    POP     HL
12100           JR      SCRIP3
12110 POSIT     INC     DE      ;FIND ADDRESS OF BACK/CHR/MOVE
12120           DEC     A
12130           DEC     A
12140           LD      L,A
12150           LD      H,0
12160           ADD     HL,DE
12170           LD      E,(HL)
12180           INC     HL
12190           LD      D,(HL)
12200           PUSH    DE
12210           POP     HL
12220           RET
12230 POINT     LD      A,(FLAG12)
12240           CP      1
12250           RET     Z
12260           LD      HL,(EPROG)      ;END OF PROG POINTER
12270           LD      DE,3
12280           ADD     HL,DE
12290           LD      (EPROG),HL
12300           CALL    01C9H
12310           LD      A,1
12320           LD      (POS),A
12330           JP      INPT1A
12340 LIST      LD      A,0C9H
12350           LD      (41BEH),A
12360           XOR     A
12370           LD      HL,(EPROG)
12380           LD      (HL),A  ;CLEAR LAST BYTE OF PROG
12390           CALL    01C9H
12400           LD      DE,(BPROG)      ;GET BEG OF PROGRAM
12410           PUSH    DE
12420           LD      A,(DE)
12430           CP      0
12440           JP      Z,INPT1A
12450           LD      HL,1
12460 LIST01    PUSH    HL
12470           PUSH    DE
12480           CALL    LISNU0
12490           POP     DE
12500           PUSH    DE
12510           LD      A,(DE)  ;TOKEN VALUE
12520           CP      1
12530           JR      NZ,LISTA
12540           LD      DE,CLRMES
12550           JP      LIST2
12560 LISTA     CP      2
12570           JR      NZ,LISTB
12580           LD      DE,CURMES
12590           JP      LIST2
12600 LISTB     CP      3
12610           JR      NZ,LISTC
12620           LD      DE,DIAMES
12630           JP      LIST2
12640 LISTC     CP      4
12650           JR      NZ,LISTD
12660           LD      DE,DISMES
12670           JP      LIST2
12680 LISTD     CP      5
12690           JR      NZ,LISTE
12700           LD      DE,LOPMES
12710           JP      LIST2
12720 LISTE     CP      6
12730           JR      NZ,LISTF
12740           LD      DE,MOVMES
12750           JP      LIST2
12760 LISTF     CP      7
12770           JR      NZ,LISTG
12780           LD      DE,RESMES
12790           JP      LIST2
12800 LISTG     CP      8
12810           JR      NZ,LISTH
12820           LD      DE,SPDMES
12830           JR      LIST2
12840 LISTH     LD      DE,DELMES
12850 LIST2     LD      A,(DE)
12860           CP      1
12870           JP      Z,LISBAK        ;DISPLAY NAME OF BACKGR
12880           CP      2
12890           JP      Z,LISCHR        ;DIS NAME OF CHR
12900           CP      3
12910           JP      Z,LISMOV        ;DIS NAME OF MOVE
12920           CP      4
12930           JP      Z,LISNUM        ;DIS NUMBER
12940 LIST2A    PUSH    DE
12950           CALL    33H
12960           POP     DE
12970           INC     DE
12980           CP      13
12990           JP      NZ,LIST2
13000           POP     DE
13010           POP     HL
13020           INC     HL
13030           POP     DE
13040           INC     DE
13050           INC     DE
13060           INC     DE
13070           PUSH    DE
13080           PUSH    HL
13090           LD      HL,(4020H)      ;FIND CURSOR POS
13100           LD      A,H
13110           CP      3EH
13120           JR      C,LIST2B
13130           CALL    49H
13140           CALL    01C9H
13150 LIST2B    POP     HL
13160           POP     DE
13170           PUSH    DE
13180           LD      A,(DE)
13190           CP      0
13200           JP      NZ,LIST01
13210           LD      A,0C3H
13220           LD      (41BEH),A
13230           POP     DE
13240           CALL    49H
13250           CALL    01C9H
13260           JP      INPT1A
13270 LISBAK    LD      HL,(BCKBUF)
13280           JR      FINDIT  ;FIND NAME IN MEM
13290 LISCHR    LD      HL,(CHRBUF)
13300           JR      FINDIT
13310 LISMOV    LD      HL,(MVBUFF)
13320           JR      FINDIT
13330 LISNUM    LD      (CPYCHR),DE
13340           POP     DE
13350           INC     DE
13360           PUSH    DE
13370           LD      A,(DE)
13380           LD      H,0
13390           LD      L,A
13400           CALL    LISNU0
13410           JP      LIST3
13420 LISNU0    CALL    0FAFH
13430 LISNUA    POP     IX
13440           POP     IX
13450           POP     IX
13460           POP     IX
13470           LD      BC,4132H
13480           LD      A,(FLAG11)
13490           CP      1
13500           RET     Z
13510 LISNU1    LD      A,(BC)
13520           CALL    33H
13530           CP      0
13540 LISNU2    INC     BC
13550           JR      NZ,LISNU1
13560           LD      A,20H
13570           CALL    33H
13580           RET
13590 FINDIT    LD      (CPYCHR),DE
13600           POP     DE
13610           INC     DE
13620           PUSH    DE
13630           LD      A,(DE)
13640           DEC     A
13650           PUSH    HL
13660           LD      HL,10
13670           CALL    444EH   ;FIND PLACE IN MEM
13680           LD      E,A
13690           LD      D,0
13700           POP     HL
13710           INC     HL
13720           ADD     HL,DE
13730 FINDT1    LD      A,(HL)
13740           CP      13
13750           JP      Z,LIST3
13760           CALL    33H
13770           INC     HL
13780           JR      FINDT1
13790 LIST3     LD      DE,(CPYCHR)
13800           INC     DE
13810           JP      LIST2
13820 STAT      CALL    01C9H
13830           LD      A,1
13840           LD      (FLAG10),A
13850           LD      HL,NUMBCK
13860           CALL    021BH
13870           LD      DE,(BCKBUF)
13880           CALL    CHOOSE
13890           CALL    PAUSE   ;WAIT TO CONTINUE
13900           LD      HL,NUMCHR
13910           CALL    021BH
13920           LD      DE,(CHRBUF)
13930           CALL    CHOOSE
13940           CALL    PAUSE
13950           LD      HL,NUMMOV
13960           CALL    021BH
13970           LD      DE,(MVBUFF)
13980           LD      A,1
13990           LD      (FLAG11),A
14000           CALL    CHOOSE
14010           CALL    PAUSE
14020           LD      HL,SCRMES
14030           CALL    021BH
14040           LD      A,13
14050           CALL    33H
14060           LD      HL,(BSCRIP)
14070 STAT1A    LD      A,(HL)
14080           CP      0
14090           JR      Z,STAT2
14100           CALL    021BH
14110           LD      A,13
14120           CALL    33H
14130           PUSH    HL
14140           LD      HL,(4020H)
14150           LD      A,H
```

Listing 1 continued

RUN BASIC PROGRAMS AT SUPER SPEED

WITH ZBASIC 2.2.

THE WORLDS FASTEST TRS-80 BASIC COMPILER from SIMUTEK

BELIEVE IT OR NOT WE'VE ADDED MORE NEW FEATURES to the ONLY INTERACTIVE BASIC COMPILER for the TRS-80!

1. *Speed increases of 10-100 times are typical after compilation.*
2. *Compiled code can be RELOCATED to run anywhere in memory. Code is even ROMable!*
3. *ZBASIC 2.2 NOW SUPPORTS BOTH RANDOM and SEQUENTIAL DISK I/O.*
4. *ZBASIC 2.2 is now a super tool for business programmers: RANDOM ACCESS FILES, and PRINT USING statements are supported as well as a HIGH PRECISION MATH package (with no rounding problems).*
5. *Special BUILT-IN MACHINE LANGUAGE COMMANDS to increase program operation by as much as 1000 times! Special commands are implemented for fast memory searching (CPDR, CPIR), block memory moves (LDIR, LDDR), inputting and printing HEX numbers, inserting MACHINE LANGUAGE into COMPILED CODE, disabling and enabling interrupts, inverting memory, 16 bit PEEKs and POKEs, and stack control, debug and much more.*
6. *ZBASIC 2.2 compiles the ENTIRE PROGRAM into Z-80 machine language. (Not 8080 code or a combination of BASIC and machine language like some other compilers.) Clumsy LINKING LOADERS, and RUNTIME MODULES are not needed; ZBASIC 2.2 creates a ready to run MACHINE LANGUAGE program.*
7. *NO ROYALTIES imposed on registered ZBASIC owners.*
8. *Typical COMPILATION TIME is TWO SECONDS for a 4K program.*
9. *Use TRS-80 Basic to write ZBASIC programs!*
10. *Compile some existing programs with only minor changes. (BASIC programming experience is required.)*
11. *Fully compatible with both the Model I and the Model III. Mod I compiled programs work on a MODEL III, and vice-versa. ZBASIC works with NEWDOS-80, NEWDOS+, DOSPLUS, LDOS, MULTIDOS, ULTRADOS, TRSDOS etc. (Not TRSDOS Mod I double density)*
12. *BUILT-IN and much improved MUSIC and SOUND EFFECTS commands.*
13. *Improved CHAINING for disk users.*
14. *TIME$ now available on DISK version. (Mod I only)*
15. *ZBASIC 2.2 now has an INPUT @ command (similar to PRINT @).*
16. *The TAB function will now tab 255 columns on a printer. (BASIC cannot tab past column 64.)*
17. *NEWDOS 80 2.0 USERS can use the CMD "dos command" function! (DOSPLUS may use name "dos command")*
18. *NEW and EASIER to use USR COMMANDS.*
19. *New math functions to calculate XOR and INTEGER REMAINDERS of a DIVISION.*
20. *Logical STRING COMPARISONS are now supported.*
21. *The disk commands INSTR, MID$ ASSIGNMENT are now supported on both DISK AND TAPE ZBASIC.*
22. *DEFSTR is now supported.*
23. *Eight disk files may be opened simultaneously; random, sequential or mixed.*
24. *LINE INPUT#, is now supported.*
25. *Invoke the compiler by simply hitting these two keys: "—"*
26. *NEW 60+ PAGE MANUAL WITH DESCRIPTIONS AND EXAMPLE.*
27. *ZBASIC 2.2 Comes with CMDFILE/CMD program from MISOSYS, to allow appending or merging compiled programs and machine language programs from tape or disk.*

ZBASIC 2.2 DOES NOT SUPPORT THESE BASIC COMMANDS:

1. ATN, EXP, COS, SIN, LOG, TAN, and exponentiation. (However, subroutines are included in the manual for these functions.)
2. ERROR, ON ERROR GOTO, ERL, ERR RESUME.
3. No direct commands like AUTO, EDIT, LIST, LLIST ETC, although these commands may be used when writing programs.
4. Others NOT supported: CDBL, CINT, CSNG, DEFFN, FIX, FRE.
5. Normal CASSETTE I/O. (ZBASIC supports it's own SPECIAL CASSETTE I/O statements.)
6. SOME BASIC COMMANDS MAY DIFFER IN ZBASIC. For instance, END jumps to DOS READY, STOP jumps to BASIC READY etc.
7. MEMORY REQUIREMENTS: to approximate the largest BASIC program that can be compiled in your machine (at one time), enter BASIC and type: PRINT (MEM-6500)/2. Remember, you can merge compiled programs together to fill memory.

ZBASIC 2.2 SPEED COMPARISON DEMO

To help give you an idea how fast compiled programs are, we have included this demo program:

ZBASIC 2.2 DEMO PROGRAM

Time to compile and run complete program	**: 0 MIN. 2 SEC.**
BASIC Execution speed MOD 1, LEVEL II	**: 7 MIN. 34 SEC.**
ZBASIC Execution speed MOD 1, LEVEL II	**:0 MIN. 18 SEC.**
BASIC Program size (WITHOUT VARIABLES)	**: 895 BYTES**
ZBASIC Program size (WITHOUT VARIABLES)	**: 2733 BYTES**

(Remember that the ZBASIC program includes an 1879 byte subroutine package.) Program shown exactly as compiled and run in BASIC and ZBASIC.

```
10 '========= ZBASIC 2.2 EXAMPLE PROGRAM AND TIME TEST========
20 CLS:CLEAR100:DEFINT A-X:DEFSTR Z:DIM AA(64,24),Z(50):RANDOM
30 AA=100:BB=-1000:CC=3:DD=-3:EE=-9999:ST$="START TIME "+TIME$
40 FOR I=1TO127STEP2 :FOR J=47TO1STEP-3:XX=POINT(I,J):SET(I,J)
50 XX=(I-J)/CC*(7+I+J) :XX=ABS(INT(RND(1*J)-AA)+7) :RESET(I,J)
60 XX=PEEK(I+J) :POKE15360+I+J,J :OUT255,J AND (3*J):XX=INP(1)
70 AB$=STR$(I+J) :BA$=LEFT$(AB$,2) :AA(I/2,J/2)=VAL(BA$)+AA*3
80 BA$=BA$+RIGHT$(BA$,RND(3)) :XX=INSTR(1,BA$,"9"):XX=SQR(I*J)
90 BA$=MID$(BA$,2,2) :MID$(BA$,1,1)=Z :IF XX THEN 100 ELSE CLS
100 IF LEN(BA$))3 OR SGN(XX)=1 AND ASC(BA$)=32 THEN PRINT"+++";
110 IFPOS(0))62 THEN TRON:TROFF:PRINT ELSE XX=NOT(RND(99))+100
120 A$=INKEY$:IF A$="Y" OR A$="y" AND I)120 THEN PRINT"TRUE.."
130 RESTORE :READA,C,Z(J),D:GOSUB170:GOSUB170:GOSUB170:GOTO210
140 NEXT :PRINT"*";:NEXTI:CLS:PRINT@512,ST$,"STOP TIME ";TIME$
150 STOP'============== END OF MAIN TEST LOOP ================
160 DATA 12345,-1,"TEST",-9999
170 ON RND(6) GOTO 180,190,200,180,190,200
180 RETURN
190 RETURN
200 RETURN
210 ON RND(9) GOSUB 180,190,200,180,190,200,180,190,200
220 GOTO140
```

NOTICE ZBASIC 2.0 OWNERS: you can upgrade your ZBASIC 2.0 for no charge. Just send us your original diskette/cassette and $15.00 with your registered serial number and copy of your invoice. We will send your ZBASIC 2.2 and updates to your manual.

VISA, MASTERCARD, AMERICAN EXPRESS, C.O.D. ORDERS CALL

800 528-1149 order line

ZBASIC 2.2 DISK VERSION AND MANUAL	89.95
ZBASIC 2.2 TAPE VERSION AND MANUAL	79.95
ZBASIC 2.2 DISK & TAPE VERSION AND MANUAL	99.99
MANUAL ONLY...(APPLIES TO PURCHASE)	25.00

SIMUTEK COMPUTER PRODUCTS INC.

TECHNICAL QUESTIONS PLEASE CALL (602) 323-9391
4897 E. SPEEDWAY, TUCSON, ARIZONA 85712 ↙12

TRS-80 is tm of Radio Shack, a Tandy Corp.

↙ See List of Advertisers on Page 211

Listing 1 continued

```
14160          CP     3EH      ;IS THERE 8 LINES ON SCREEN
14170          CALL   Z,PAUSE
14180          POP    HL
14190          JR     STAT1A
14200 STAT2    CALL   PAUSE
14210          XOR    A
14220          LD     (FLAG10),A
14230          LD     (FLAG11),A
14240          JP     INPT1A
14250 PAUSE    LD     HL,3C00H+915
14260          LD     (4020H),HL
14270          LD     HL,WAITIN
14280          CALL   021BH
14290          CALL   49H
14300          CALL   01C9H
14310          RET
14320 ;        EDIT ROUTINE
14330 EDIT     CALL   01C9H
14340          LD     A,1
14350          LD     (FLAG12),A
14360          LD     (POS),A
14370          LD     HL,3C00H+147
14380          LD     DE,MENU3
14390          CALL   OPTION
14400          LD     HL,3C00H+147+192
14410          CALL   CURSOR
14420          LD     A,(POS)
14430          CP     2
14440          JP     Z,EDIT1 ;DELETE SCRIPT
14450          LD     HL,(BPROG)
14460          LD     A,(HL)
14470          OR     A
14480          JP     Z,ENDED
14490 EDIT0    LD     A,1
14500          LD     (POS),A
14510          CALL   01C9H
14520          LD     HL,3C00H+147
14530          LD     DE,MENU4
14540          CALL   OPTION
14550          LD     HL,3C00H+147+192
14560          CALL   CURSOR
14570 ;        ROUTINE THAT EDITS LINE #
14580          LD     HL,15360+598
14590          LD     DE,PROGNM       ;LINE NUMBER
14600          LD     A,4
14610          LD     (LEN),A
14620          CALL   BOX
14630          LD     HL,(SCRIN)
14640          CALL   1E5AH    ;DECODE #
14650          LD     A,E
14660          OR     A
14670          JP     Z,EDIT
14680          LD     (ADD),A
14690          DEC    A
14700          LD     HL,3
14710          CALL   444EH
14720          LD     L,A
14730          LD     H,0
14740          LD     DE,(BPROG)
14750          ADD    HL,DE
14760          LD     A,(HL)   ;TEST IF OVER PROG POINTERS
14770          OR     A
14780          JP     Z,EDIT
14790          LD     A,(POS)
14800          CP     1
14810          JP     NZ,EDITA
14820 ;        REPLACE LINE #
14830          PUSH   HL
14840          CALL   01C9H
14850          POP    HL
14860          CALL   INPT1A
14870          CALL   01C9H
14880          JP     ENDED
14890 EDITA    PUSH   AF
14900          LD     A,1
14910          LD     (POS),A
14920          POP    AF
14930          CP     2       ;INSERT LINE
14940          JP     NZ,EDITB
14950          PUSH   HL
14960 ;        ROUTINE TO INCREMENT LOOPS
14970          LD     HL,(BPROG)
14980          LD     DE,3
14990 EDITAA   LD     A,(HL)
15000          CP     0
15010          JR     Z,EDITAC
15020          CP     5
15030          JR     NZ,EDITAB
15040          INC    HL
15050          LD     A,(ADD)
15060          CP     (HL)
15070          JR     Z,EDITA1
15080          JR     NC,EDITAB
15090 EDITA1   INC    (HL)
15100          DEC    HL
15110 EDITAB   ADD    HL,DE
15120          JR     EDITAA
15130 EDITAC   POP    HL
15140          PUSH   HL
15150          PUSH   HL
15160          POP    DE      ;DE=START
15170          LD     HL,(EPROG)       ;HL=END OF DATA BLOCK
15180          PUSH   HL
15190          POP    BC
15200          INC    BC
15210          INC    BC
15220          INC    BC      ;BC=END OF DESTINATION BLOCK
15230          CALL   195BH   ;MOVE DATA BLOCK ROUTINE
15240          CALL   01C9H
15250          POP    HL      ;RESTORE HL
15260          CALL   INPT1A
15270          XOR    A
15280          LD     (FLAG12),A
15290          JP     POINT
15300 EDITB    PUSH   HL
15310          LD     HL,(BPROG)
15320          LD     DE,3
15330 EDITBA   LD     A,(HL)
15340          CP     0
15350          JR     Z,EDITBC
15360          CP     5
15370          JR     NZ,EDITBB
15380          LD     A,(ADD)
15390          INC    HL
15400          CP     (HL)
15410          JR     NC,EDITBB
15420          DEC    (HL)
15430          DEC    HL
15440 EDITBB   ADD    HL,DE
15450          JR     EDITBA
15460 EDITBC   POP    DE
15470          PUSH   DE
15480          INC    DE
15490          INC    DE
15500          INC    DE      ;DE=START OF DATA BLOCK
15510          POP    BC      ;BC=START OF DESTINATION BLOCK
15520          LD     HL,(EPROG)       ;HL=END OF DATA BLOCK
15530 EDITC    RST    18H
15540          LD     A,(DE)
15550          LD     (BC),A
15560          INC    BC
15570          INC    DE
15580          JR     NZ,EDITC
15590          LD     HL,(EPROG)
15600          DEC    HL
15610          DEC    HL
15620          DEC    HL
15630          LD     (EPROG),HL
15640 ENDED    XOR    A
15650          LD     (FLAG12),A
15660          CALL   01C9H
15670          LD     A,1
15680          LD     (POS),A
15690          JP     INPT1A
15700 EDIT1    LD     HL,(ESCRIP)      ;DELETE LAST LINE OF SCRIPT
15710          LD     DE,(BSCRIP)
15720          RST    18H
15730          JR     Z,ENDED
15740          DEC    HL
15750 EDIT1A   DEC    HL
15760          LD     A,(HL)
15770          CP     3
15780          JR     NZ,EDIT1A
15790          XOR    A
15800          INC    HL
15810          LD     (HL),A
15820          LD     (ESCRIP),HL
15830          INC    HL
15840          LD     (HL),A
15850          JP     ENDED
15860 ;SAVE AND LOAD
15870 MEMOR    CALL   01C9H
15880          LD     A,1
15890          LD     (POS),A
15900          LD     HL,(EPROG)
15910          LD     (0B6E0H),HL
15920          LD     HL,(ESCRIP)
15930          LD     (0B6E2H),HL
15940          LD     HL,3C00H+147
15950          LD     DE,MENU5
15960          CALL   OPTION
15970          LD     HL,3C00H+147+192
15980          CALL   CURSOR
15990          LD     A,8
16000          LD     (LEN),A
16010          LD     HL,15891
16020          LD     DE,NAME
16030          CALL   BOX
16040 ;SAVE ROUTINE
16050          LD     HL,(SCRIN)
16060          LD     DE,0FF00H
16070          LD     BC,8
16080          LDIR
16090          LD     A,13
16100          LD     (DE),A
16110          LD     HL,0FE00H
16120          LD     DE,0FF00H
16130          LD     B,0
16140          LD     A,(POS)
16150          CP     2
16160          JP     Z,LOAD
16170          CALL   4420H
16180          CP     8
16190          JP     NZ,TRSERR
16200          LD     HL,(BPROG)
16210          LD     B,6
16220          CALL   WRITE
16230          LD     HL,(CHRBUF)
16240          LD     A,(HL)
16250          INC    A
16260          LD     B,A
16270          CALL   WRITE
16280          LD     HL,(MVBUFF)
16290          LD     B,(HL)
16300          INC    B
16310          CALL   WRITE
16320          LD     HL,(BCKBUF)
16330          LD     A,(HL)
16340          SLA    A
16350          SLA    A
16360          LD     B,A
16370          INC    B
16380          CALL   WRITE
16390 CLOSE    LD     DE,0FF00H
16400          CALL   4428H
16410          CALL   01C9H
16420          LD     A,1
16430          LD     (POS),A
16440          JP     INPUT
16450 WRITE    PUSH   BC
16460          LD     DE,0FE00H
16470          LD     BC.256
16480          LDIR
16490          PUSH   HL
16500          LD     DE,0FF00H
16510          CALL   4439H
16520          CP     0
16530          JP     NZ,TRSERR
16540          POP    HL
16550          POP    BC
16560          DJNZ   WRITE
16570          RET
16580 LOAD     CALL   4424H
16590          CP     0
16600          JP     NZ,TRSERR
16610          LD     HL,(BPROG)
16620          LD     (MISC),HL
16630          LD     B,7
16640          CALL   READ
16650          LD     HL,(CHRBUF)
16660          LD     A,(HL)
16670          CP     0
16680          JR     Z,LOAD1
16690          LD     B,A
16700          LD     DE,256
16710          ADD    HL,DE
16720          LD     (MISC),HL
16730          CALL   READ
16740 LOAD1    LD     HL,(MVBUFF)
16750          PUSH   HL
16760          LD     (MISC),HL
16770          LD     B,1
16780          CALL   READ
16790          POP    HL
16800          LD     A,(HL)
16810          CP     0
16820          JR     Z,LOAD2
16830          LD     B,A
16840          LD     DE,256
16850          ADD    HL,DE
16860          CALL   READ
16870 LOAD2    LD     HL,(BCKBUF)
16880          PUSH   HL
16890          LD     (MISC),HL
16900          LD     B,1
16910          CALL   READ
16920          POP    HL
16930          LD     A,(HL)
16940          SLA    A
16950          SLA    A
16960          CP     0
16970          JR     Z,LOAD3
16980          LD     DE,256
16990          ADD    HL,DE
```

Listing 1 continued

↙ See List of Advertisers on Page 211

Listing 1 continued

```
17000           LD      B,A
17010           CALL    READ
17020 LOAD3     LD      HL,(0B6E0H)
17030           LD      (EPROG),HL
17040           LD      HL,(0B6E2H)
17050           LD      (ESCRIP),HL
17060           JP      CLOSE
17070 READ      PUSH    BC
17080           LD      DE,0FF00H
17090           CALL    4436H
17100           LD      HL,0FE00H
17110           LD      DE,(MISC)
17120           LD      BC,256
17130           LDIR
17140           LD      (MISC),DE
17150           POP     BC
17160           CP      0
17170           JP      NZ.TRSERR
17180           DJNZ    READ
17190           RET
17200 ERR1      CALL    ERROR
17210           LD      HL,ERROR1
17220           JP      ERR
17230 ERR2      POP     IX
17240           POP     IX
17250           POP     IX
17260           XOR     A
17270           LD      (FLAG6),A
17280           LD      (FLAG8),A
17290           CALL    ERROR
17300           LD      HL,ERROR2
17310           JP      ERR
17320 TRSERR    OR      0C0H
17330           LD      HL,16147
17340           LD      (4020H),HL
17350           CALL    4409H
17360           CALL    49H
17370           JP      CLOSE
17380 ERR       CALL    021BH
17390           CALL    49H
17400           CALL    01C9H
17410           JP      INPUT
17420 ERROR     LD      HL,(SCRIN)      ;SCREEN POSITION
17430           LD      DE,181
17440           ADD     HL,DE
17450           LD      (4020H),HL
17460           RET
17470 MMENU     DEFM    '  Play-Byte Menu'
17480           DEFB    13
17490           DEFM    '  ---------------'
17500           DEFB    13
17510           DEFB    13
17520           DEFM    'Draw Chr/Bckgrnd/Mves'
17530           DEFB    13
17540           DEFM    'Construct Play'
17550           DEFB    13
17560           DEFM    'Action !!!'
17570           DEFB    13
17580           DEFM    'Return to TRSDOS'
17590           DEFB    13
17600           DEFB    1
17610 MMENU1    DEFM    'Drawing Menu'
17620           DEFB    13
17630           DEFM    '------------'
17640           DEFB    13
17650           DEFB    13
17660           DEFM    'Background'
17670           DEFB    13
17680           DEFM    'Character'
17690           DEFB    13
17700           DEFM    'Moves'
17710           DEFB    13
17720           DEFB    1
17730 MMENU2    DEFM    'Construction Menu'
17740           DEFB    13
17750           DEFM    '------------------'
17760           DEFB    13
17770           DEFB    10
17780           DEFM    'List      '
17790           DEFB    9
17800           DEFM    'Status    '
17810           DEFB    9
17820           DEFM    'Edit      '
17830           DEFB    9
17840           DEFM    'Memorize  '
17850           DEFB    9
17860           DEFM    'Clr Screen'
17870           DEFB    9
17880           DEFM    'Curtain   '
17890           DEFB    9
17900           DEFM    'Erase Chr '
17910           DEFB    9
17920           DEFM    'Dialogue  '
17930           DEFB    9
17940           DEFM    'Displ Bkgr'
17950           DEFB    9
17960           DEFM    'Loop      '
17970           DEFB    9
17980           DEFM    'Move Char '
17990           DEFB    9
18000           DEFM    'Restre Mvs'
18010           DEFB    9
18020           DEFM    'Speed     '
18030           DEFB    9
18040           DEFM    'Script    '
18050           DEFB    9
18060           DEFB    1
18070 MENU3     DEFM    'Editor Menu'
18080           DEFB    13
18090           DEFM    '-----------'
18100           DEFB    13
18110           DEFB    13
18120           DEFM    'Edit Program Lines'
18130           DEFB    13
18140           DEFM    'Delete Script'
18150           DEFB    13
18160           DEFB    1
18170 MENU4     DEFM    'Editor Menu'
18180           DEFB    13
18190           DEFM    '-----------'
18200           DEFB    13
18210           DEFB    13
18220           DEFM    'Replace Program Line'
18230           DEFB    13
18240           DEFM    'Insert  Program Line'
18250           DEFB    13
18260           DEFM    'Delete  Program Line'
18270           DEFB    13
18280           DEFB    1
18290 MENU5     DEFM    'Memorize Menu'
18300           DEFB    13
18310           DEFM    '-------------'
18320           DEFB    13
18330           DEFB    13
18340           DEFM    'Save'
18350           DEFB    13
18360           DEFM    'Load'
18370           DEFB    13
18380           DEFB    1
18390 TABLE     DEFW    CLR1
18400           DEFW    CURT2
18410           DEFW    DIALG1
18420           DEFW    DISPL1
18430           DEFW    LOOP2
18440           DEFW    MOVE2
18450           DEFW    RESTR2
18460           DEFW    SPEED2
18470           DEFW    DELET2
18480 NAME      DEFM    'Input Name:'
18490           DEFB    13
18500 EXTRA     DEFM    'Base Character?'
18510           DEFB    13
18520 EXTRAB    DEFM    'Base Background?'
18530           DEFB    13
18540 MVCHR     DEFM    'Input MOVE which CHAR is using'
18550           DEFB    13
18560 PROGNM    DEFM    'Input Appropriate Line #'
18570           DEFB    13
18580 MOVES     DEFM    'Base Move?'
18590           DEFB    13
18600 LINNUM    DEFM    'Line Number?'
18610           DEFB    13
18620 TIMNUM    DEFM    'Number Of Loops?'
18630           DEFB    13
18640 SPEEDS    DEFM    'Speed?'
18650           DEFB    13
18660 SCRIPS    DEFM    'Enter One Line of Script'
18670           DEFB    13
18680 NUMBCK    DEFM    'Backgrounds:'
18690           DEFB    13
18700 NUMCHR    DEFM    'Characters:'
18710           DEFB    13
18720 NUMMOV    DEFM    'Moves:'
18730           DEFB    13
18740 SCRMES    DEFM    'Script:'
18750           DEFB    13
18760 WAITIN    DEFM    'Press ANY key to continue'
18770           DEFB    13
18780 ERROR1    DEFM    'Too Many Bcks/Chrs/or Mves'
18790           DEFB    13
18800 ERROR2    DEFM    'No such Bckgr/Chr/ or Mve'
18810           DEFB    13
18820 CLRMES    DEFM    'Clear Screen'
18830           DEFB    13
18840 CURMES    DEFM    'Background: '
18850           DEFB    1
18860           DEFM    ' with Curtain'
18870           DEFB    13
18880 DIAMES    DEFM    'Dialogue'
18890           DEFB    13
18900 DISMES    DEFM    'Background: '
18910           DEFB    1
18920           DEFB    20H
18930           DEFB    13
18940 LOPMES    DEFM    'Loop to Line '
18950           DEFB    4
18960           DEFM    ' , '
18970           DEFB    4
18980           DEFM    ' Times
18990           DEFB    13
19000 MOVMES    DEFM    'Move Char: '
19010           DEFB    2
19020           DEFM    ' with Move: '
19030           DEFB    3
19040           DEFB    13
19050 RESMES    DEFM    'Restore Move: '
19060           DEFB    3
19070           DEFB    13
19080 SPDMES    DEFM    'Speed = '
19090           DEFB    4
19100           DEFB    13
19110 DELMES    DEFM    'Erase Char which is using MOVE: '
19120           DEFB    3
19130           DEFB    13
19140 FLAG6     DEFB    0H      ;FLAG TO GET ADDRESS OF BACKGRND
19150 FLAG7     DEFB    0       ;FLAG TO PRINT '# OF MOVES'
19160 FLAG8     DEFB    0       ;FLAG TO RETURN TO PROG EXECUTION
19170 FLAG9     DEFB    0       ;FLAG TO HAVE DIFFERENT CURSOR MOVEMENT ON MENUES
19180 FLAG10    DEFB    0       ;FLAG FOR USE OF 'STAT'
19190 FLAG11    DEFB    0       ;USED TO PRINT # OF MOVES
19200 FLAG12    DEFB    0       ;FLAG FOR EDITOR TO INTERCEPT
19210 FLAG13    DEFB    0       ;FLAG FOR ERASE CHARACTER
19220 DATFLG    DEFB    0       ;FLAG FOR DATA STATEMENTS
19230 MCHR      DEFB    0
19240 POS       DEFB    1
19250 LEN       DEFB    0
19260 BPROG     DEFW    9000H   ;ADDRESS OF BEG OF PROGRAM
19270 EPROG     DEFW    9000H   ;ADDRESS OF END OF PROGRAM
19280 BSCRIP    DEFW    9301H
19290 ESCRIP    DEFW    9301H
19300 SCRMAX    DEFW    95FFH
19310 COORD     DEFW    0000
19320 BCKBUF    DEFW    0B600H
19330 BEGBAK    DEFW    0B340H
19340 CHRBUF    DEFW    09600H
19350 BEGCHR    DEFW    09640H
19360 MVBUFF    DEFW    0A100H
19370 BEGMOV    DEFW    0A180H
19380 ALLBUF    DEFW    0000H
19390 CPYCHR    DEFW    0000H
19400 BYTE      DEFW    0000H
19410 ADD       DEFW    0000H
19420 RATE      DEFW    0001H
19430 SCRIN     DEFW    0000H
19440 MISC      DEFW    0000H
19450 MISC1     DEFW    0000H
19460 SAVBUF    DEFW    0E000H
19470 SCRBUF    DEFW    0000H
19480 PRGEND    DEFW    92FDH
19490 TRANS     EQU     859AH
19500 TRANS3    EQU     85AFH
19510 TRANS6    EQU     85C4H
19520 BCKGRO    EQU     81CEH
19530 CHRCTR    EQU     83DEH
19540 DEFMV     EQU     843DH
19550 PLACE     EQU     8677H
19560 PLACE1    EQU     8679H
19570           END     6000H
```

Program Listing 2. Play-Byte, part 2.

LOAD 80

```
00100           ORG     81CEH
00110 ;         DISPLAY STATUS ON TOP OF SCREEN
00120 ;         MAKE BACKGROUND
00130 BCKGRO    JP      Z,BCKGRA
00140           CALL    01C9H
00150 BCKGRA    CALL    INTRO
00160 BCKGRB    LD      HL,3000
00170           LD      (SPEED),HL
00180           LD      A,1
00190           LD      (FLAG1),A
00200 BCKGR1    LD      A,0
00210           LD      (X),A
00220 BCKGR2    LD      A,8
```

Listing 2 continued

ENTER MY 1 YEAR SUBSCRIPTION TO *RUN* AT THE LOW CHARTER RATE OF $17.97...10% OFF THE BASIC SUBSCRIPTION PRICE.

With payment enclosed or credit card order I will receive a FREE issue totaling 13 issues for $17.97.

☐ 2 YRS $29.97 *(25 issues with payment)* ☐ 3 YRS $41.97 *(37 issues with payment)*

CHECK/MO MC VISA AE BILL ME

Card #_____ Exp. Date_____

Signature _____

Name _____

Address _____

City _____ State_____ Zip_____

Canada and Mexico: $20.97; Foreign surface: $37.97. One year only. US funds drawn on US bank. Foreign airmail, please inquire. This offer expires February 29, 1984. Please allow 4 to 6 weeks for delivery.

RUN • BOX 954 • FARMINGDALE, NY 11737 342F8B

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

First Class Permit No. 73 Peterborough, NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE

Wayne Green, Inc.

PO Box 954
Farmingdale, NY 11737

CHARTER OFFER
SAVE $2.00 OFF THE BASIC RATE!

RUN

THESE COULD BE THE KEYS TO YOUR FUTURE

Unlock *all* the potential of your Commodore 64 and VIC-20* with **RUN**.

Explore...Experiment...Enjoy... Beginner and expert alike will be taken beyond the manual to the limits of their abilities. Enter your own game programs. Construct a simple hardware add-on. Broaden your scope with unique applications...And...get a 13th issue FREE!

Enjoy key features like these:

- Games for fun & strategy.
- Programming tips help you learn short cuts.
- Candid reviews help you make money-saving decisions.
- Programs to add to your library.
- Instructions & tutorials to increase your skills.
- Hardware & software modifications help your machine work smart.
- Unique applications broaden your scope.

Here's a system-specific magazine written with you in mind. Written by and for the reader to give time-saving, money-saving hints. You'll get instructions and tutorials to increase your skills, and candid reviews to help you make the right decisions. Most of all though, you'll have fun.

*Commodore 64 and VIC-20 are registered trademarks of Commodore Business Machines, Inc.

Commodore 64 and VIC-20 owners are one of the largest groups of computerists today. Enjoy the benefits of this with your own magazine. Be in control like never before. Order **RUN** today and get a 13th issue free with your prepaid order (check or credit card) of only $17.97. Send in the coupon or call toll free *1-800-258-5473.* In N.H. call 1-924-9471.

Send me a subscription to ***RUN*** *for only $17.97 per year. I understand that with payment enclosed or credit card order I will receive a FREE issue making a total of 13 issues for $17.97. Save $2.00 off the basic rate!*

☐ *CHECK/MO* ☐ *MC* ☐ *AE* ☐ *VISA* ☐ *BILL ME*

card #_____ exp. date _____

signature _____

name _____

address _____

city _____ state _____ zip _____

Canada & Mexico $20.97; Foreign Surface $37.97. 1 year only. US funds drawn on US bank. Foreign airmail, please inquire. Please allow 6 to 8 weeks for delivery.

RUN • Box 954 • Farmingdale, NY 11737

342F8

Listing 2 continued

```
0023Ø          LD      (Y),A
0024Ø          CALL    SET3A
0025Ø ;        ACCEPT INPUT FROM USER
0026Ø LOOP     CALL    HALT
0027Ø          LD      A,(3840H)
0028Ø          BIT     3,A
0029Ø          JR      NZ,UP
0030Ø          BIT     4,A
0031Ø          JR      NZ,DOWN
0032Ø          BIT     5,A
0033Ø          JR      NZ,LEFT
0034Ø          BIT     6,A
0035Ø          JR      NZ,RIGHT
0036Ø          BIT     2,A
0037Ø          JP      NZ,ENDIT
0038Ø          BIT     0,A
0039Ø          JP      NZ,MOVE
0040Ø          BIT     1,A
0041Ø          JP      NZ,ERASE
0042Ø          LD      A,(3810H)
0043Ø          BIT     1,A
0044Ø          CALL    NZ,FAST
0045Ø          BIT     2,A
0046Ø          CALL    NZ,MED
0047Ø          BIT     3,A
0048Ø          CALL    NZ,SLOW
0049Ø          LD      A,(3801H)
0050Ø          BIT     0,A
0051Ø          JP      NZ,MOVE1
0052Ø          JR      LOOP
0053Ø ;        MOVE CURSOR WHICH WAY?
0054Ø LEFT     LD      A,(X)
0055Ø          DEC     A
0056Ø          CALL    SET
0057Ø          JR      LOOP
0058Ø DOWN     LD      A,(Y)
0059Ø          INC     A
0060Ø          CALL    SET1
0061Ø          JR      LOOP
0062Ø RIGHT    LD      A,(X)
0063Ø          INC     A
0064Ø          CALL    SET
0065Ø          JR      LOOP
0066Ø UP       LD      A,(Y)
0067Ø          DEC     A
0068Ø          CALL    SET1
0069Ø          JP      LOOP
0070Ø ;        SET AND RESET PARTS OF SCREEN
0071Ø SET      CP      80H
0072Ø          JP      NC,PAUSE
0073Ø SETA     CP      0H
0074Ø          JP      C,PAUSE
0075Ø          LD      (X),A
0076Ø          JR      SET3
0077Ø SET1     CP      30H
0078Ø          JP      NC,PAUSE
0079Ø SET2     CP      4
0080Ø          JP      C,PAUSE
0081Ø          LD      (Y),A
0082Ø SET3     LD      A,(CHAR)
0083Ø          LD      (DE),A
0084Ø SET3A    LD      A,(Y)
0085Ø          LD      D,0FFH
0086Ø SET4     INC     D
0087Ø          SUB     3
0088Ø          JR      NC,SET4
0089Ø          ADD     A,3
0090Ø          LD      C,A
0091Ø          LD      A,(X)
0092Ø          ADD     A,A
0093Ø          LD      E,A
0094Ø          LD      B,2
0095Ø SET5     LD      A,D
0096Ø          RRA
0097Ø          LD      D,A
0098Ø          LD      A,E
0099Ø          RRA
0100Ø          LD      E,A
0101Ø          DJNZ    SET5
0102Ø          LD      A,C
0103Ø          ADC     A,A
0104Ø          INC     A
0105Ø          LD      B,A
0106Ø          XOR     A
0107Ø          SCF
0108Ø SET6     ADC     A,A
0109Ø          DJNZ    SET6
0110Ø          LD      C,A
0111Ø          LD      A,D
0112Ø          OR      3CH
0113Ø          LD      D,A
0114Ø          LD      A,(DE)
0115Ø          OR      A
0116Ø          JP      M,SET7
0117Ø          LD      A,80H
0118Ø SET7     LD      B,A
0119Ø          LD      A,(FLAG2)
0120Ø          CP      1
0121Ø          JR      Z,SET9
0122Ø          LD      A,B
0123Ø          LD      (DE),A
0124Ø          OR      C
0125Ø          LD      (DE),A
0126Ø          LD      (CHAR),A
0127Ø          LD      A,(FLAG1)
0128Ø          CP      1
0129Ø          RET     NZ
0130Ø SET8     LD      A,B
0131Ø          LD      (CHAR),A
0132Ø          RET
0133Ø SET9     LD      A,B
0134Ø          PUSH    AF
0135Ø          PUSH    BC
0136Ø          LD      A,C
0137Ø          CPL
0138Ø          LD      C,A
0139Ø          LD      A,(DE)
0140Ø          AND     C
0141Ø          LD      (CHAR),A
0142Ø          POP     BC
0143Ø          POP     AF
0144Ø          LD      (DE),A
0145Ø          OR      C
0146Ø          LD      (DE),A
0147Ø          RET
0148Ø ;        PAUSE
0149Ø HALT     LD      BC,(SPEED)
0150Ø          CALL    60H
0151Ø          RET
0152Ø PAUSE    POP     IY
0153Ø          JP      LOOP
0154Ø ;        ENTER MOVE MODE
0155Ø MOVE     XOR     A
0156Ø          LD      (FLAG1),A
0157Ø          LD      (FLAG2),A
0158Ø          LD      A,1
0159Ø          LD      (FLAG4),A
0160Ø          LD      HL,15360+7
0161Ø          LD      (4020H),HL
0162Ø          LD      HL,DRAWME
0163Ø          PUSH    DE
0164Ø          CALL    021BH
0165Ø          POP     DE
0166Ø          CALL    SET3
0167Ø          JP      LOOP
0168Ø MOVE1    LD      A,1
0169Ø          LD      (FLAG1),A
0170Ø          XOR     A
0171Ø          LD      (FLAG2),A
0172Ø MOVE2    LD      HL,15360+7
0173Ø          LD      (4020H),HL
0174Ø          LD      HL,MOVEME
0175Ø          PUSH    DE
0176Ø          CALL    021BH
0177Ø          POP     DE
0178Ø          JP      LOOP
0179Ø ;        ENTER ERASE MODE
0180Ø ERASE    XOR     A
0181Ø          LD      (FLAG4),A
0182Ø          LD      A,1
0183Ø          LD      (FLAG2),A
0184Ø          LD      HL,15360+7
0185Ø          LD      (4020H),HL
0186Ø          LD      HL,ERASEM
0187Ø          PUSH    DE
0188Ø          CALL    021BH
0189Ø          POP     DE
0190Ø          CALL    SET3
0191Ø          LD      HL,(PLACE1)
0192Ø          INC     (HL)
0193Ø          JP      LOOP
0194Ø ;        ENTER DRAW MODE
0195Ø DRAW     XOR     A
0196Ø          LD      (FLAG2),A
0197Ø          JR      MOVE2
0198Ø ;        DO IT FAST!!
0199Ø FAST     LD      HL,15360+22
0200Ø          LD      (4020H),HL
0201Ø          LD      HL,FASTME
0202Ø          PUSH    DE
0203Ø          CALL    021BH
0204Ø          POP     DE
0205Ø          LD      HL,500
0206Ø          JR      SPEED1
0207Ø ;        DO IT MEDIUM
0208Ø MED      LD      HL,15360+22
0209Ø          LD      (4020H),HL
0210Ø          LD      HL,MEDME
0211Ø          PUSH    DE
0212Ø          CALL    021BH
0213Ø          POP     DE
0214Ø          LD      HL,3000
0215Ø          JR      SPEED1
0216Ø ;        DO IT SLOWLY
0217Ø SLOW     LD      HL,15360+22
0218Ø          LD      (4020H),HL
0219Ø          LD      HL,SLOWME
0220Ø          PUSH    DE
0221Ø          CALL    021BH
0222Ø          POP     DE
0223Ø          LD      HL,8000
0224Ø SPEED1   LD      (SPEED),HL
0225Ø          RET
0226Ø ;        DISPLAY STATUS ON TOP OF SCREEN
0227Ø INTRO    LD      HL,MODE
0228Ø          CALL    021BH
0229Ø          LD      HL,SPEED2
0230Ø          CALL    021BH
0231Ø          LD      HL,EXIT
0232Ø          CALL    021BH
0233Ø          LD      HL,15360+7
0234Ø          LD      (4020H),HL
0235Ø          LD      HL,MOVEME
0236Ø          CALL    021BH
0237Ø          CALL    MED
0238Ø          RET
0239Ø ;        WANT TO END SESSION,CONTINUE, OR ERASE SCREEN?
0240Ø ENDIT    LD      B,A
0241Ø          LD      A,140
0242Ø          LD      (15360+63),A
0243Ø          LD      A,(CHAR)
0244Ø          LD      (DE),A
0245Ø ENDIT1   PUSH    DE
0246Ø          CALL    49H
0247Ø          POP     DE
0248Ø          CP      'E'
0249Ø ENDITA   JP      Z,ENDIT3
0250Ø          CP      'C'
0251Ø ENDITB   JP      Z,ENDIT2
0252Ø          CP      'M'
0253Ø ENDITC   JP      Z,ENDIT0
0254Ø          JR      ENDIT1
0255Ø ENDIT0   LD      A,(FLAG3)
0256Ø          CP      1
0257Ø          JP      Z,CHAR1
0258Ø          RET
0259Ø ENDIT2   LD      A,32
0260Ø          LD      (15360+63),A
0261Ø          JP      LOOP
0262Ø ENDIT3   CALL    01C9H
0263Ø          CALL    INTRO
0264Ø          LD      A,(FLAG3)
0265Ø          CP      0
0266Ø          JP      Z,BCKGRB
0267Ø          CALL    01C9H
0268Ø ;        DRAW THE CHARACTER
0269Ø CHRCTR   LD      A,1
0270Ø          LD      (FLAG3),A
0271Ø          LD      A,88
0272Ø          LD      (SET+1),A
0273Ø          LD      A,42
0274Ø          LD      (SETA+1),A
0275Ø          LD      (BCKGR1+1),A
0276Ø          LD      A,30
0277Ø          LD      (SET1+1),A
0278Ø          LD      A,12
0279Ø          LD      (SET2+1),A
0280Ø          LD      (BCKGR2+1),A
0281Ø          LD      HL,15360+213
0282Ø          LD      A,65
0283Ø          LD      (HL),A
0284Ø          LD      B,22
0285Ø CHAR2    INC     HL
0286Ø          INC     A
0287Ø          LD      (HL),A
0288Ø          DJNZ    CHAR2
0289Ø          LD      HL,15360+276
0290Ø          LD      A,31H
0291Ø          LD      (HL),A
0292Ø          LD      B,5
0293Ø          LD      DE,64
0294Ø CHAR3    ADD     HL,DE
0295Ø          INC     A
0296Ø          LD      (HL),A
0297Ø          DJNZ    CHAR3
0298Ø          JP      BCKGRA
0299Ø CHAR1    XOR     A
0300Ø          LD      (FLAG3),A
0301Ø          LD      A,88H
0302Ø          LD      (SET+1),A
0303Ø          XOR     A
```

Listing 2 continued

See List of Advertisers on Page 211

Listing 2 continued

```
03040           LD      (SETA+1),A
03050           LD      (BCKGR1+1),A
03060           LD      A,30H
03070           LD      (SET1+1),A
03080           LD      A,4
03090           LD      (SET2+1),A
03100           LD      A,8
03110           LD      (BCKGR2+1).A
03120           RET
03130 ;         DRAW THE MOVES
03140 DEFMV     CALL    INTRO
03150           LD      A,1
03160           LD      (FLAG1),A
03170           LD      A,0C9H
03180           LD      (SET3),A
03190           LD      A,0C3H
03200           LD      (LOOP).A
03210           LD      HL,LOOP1
03220           LD      (LOOP+1),HL
03230           LD      HL,RETURN
03240           LD      (ENDITC+1),HL
03250           LD      HL,DEFMV1
03260           LD      (ENDITA+1),HL
03270           LD      BL,15360+129
03280           LD      (X),HL
03290           LD      DE,SCREEN
03300           PUSH    HL
03310           INC     HL
03320           CALL    TRANS3
03330           POP     HL
03340           LD      A,(15360+129)
03350           LD      (CHAR).A
03360 LOOP1     CALL    HALT
03370           LD      HL,(X)
03380           LD      A,(3840H)
03390           BIT     3,A
03400           JR      NZ,UPM
03410           BIT     4,A
03420           JR      NZ,DOWNM
03430           BIT     5,A
03440           JR      NZ,LEFTM
03450           BIT     6,A
03460           JR      NZ.RIGHTM
03470           BIT     2,A
03480           JP      NZ,ENDIT
03490           BIT     0,A
03500           JP      NZ.MOVE
03510           BIT     1,A
03520           JP      NZ,ERASE
03530           LD      A,(3810H)
03540           BIT     1,A
03550           CALL    NZ,FAST
03560           BIT     2,A
03570           CALL    NZ,MED
03580           BIT     3,A
03590           CALL    NZ,SLOW
03600           LD      A,(3801H)
03610           BIT     0,A
03620           JP      NZ,MOVE1
03630           JR      LOOP1
03640 DEFMV1    CALL    01C9H
03650           JP      DEFMV
03660 UPM       CALL    COORD2
03670           LD      DE,64
03680           OR      A
03690           SBC     HL,DE
03700           JP      COORD
03710 DOWNM     CALL    COORD2
03720           LD      DE,64
03730           ADD     HL,DE
03740           JP      COORD
03750 LEFTM     CALL    COORD2
03760           DEC     HL
03770           JP      COORD
03780 RIGHTM    CALL    COORD2
03790           INC     HL
03800           JP      COORD
03810 COORD     LD      A,H
03820           CP      3CH
03830           JR      Z,COORD1
03840           CP      62
03850           JP      P,COORDB
03860 COORDA    INC     HL
03870           CALL    DRWCHR
03880           DEC     HL
03890           LD      (X).HL
03900           LD      A,(HL)
03910           LD      (CHAR),A
03920           LD      A,'-'
03930           LD      (HL),A
03940           JP      LOOP1
03950 COORD1    LD      A,L
03960           CP      41H
03970           JP      NC,COORDA
03980           LD      DE,64
03990           ADD     HL,DE
04000           JP      COORDA
04010 COORDB    LD      A,L
04020           CP      167
04030           JP      C,COORDA
04040           LD      DE,64
04050           OR      A
04060           SBC     HL,DE
04070           JP      COORDA
04080 COORD2    LD      A,(FLAG2)
04090           CP      1
04100           JR      Z,ERASM
04110           LD      A,(FLAG1)
04120           CP      1
04130           JR      Z,MOVEM
04140           LD      A,(FLAG4)
04150           CP      1
04160           JR      Z.DRAWM
04170           JP      LOOP1
04180 ;         DRAW DASHES WHERE MOVES ARE
04190 DRAWM     PUSH    HL
04200           LD      HL,(PLACE)
04210           LD      A,(HL)
04220           CP      125
04230           JR      Z,DRAWM1
04240           INC     (HL)
04250           POP     HL
04260           LD      DE,(PLACE1)
04270           LD      A,L
04280           LD      (DE),A
04290           INC     DE
04300           LD      A,H
04310           LD      (DE),A
04320           INC     DE
04330           LD      (PLACE1).DE
04340           LD      A,'-'
04350           LD      (HL),A
04360           RET
04370 DRAWM1    POP     HL
04380           POP     IX
04390           LD      A,'-'
04400           LD      (CHAR),A
04410           JP      MOVE1
04420 MOVEM     LD      A,(CHAR)
04430           LD      (HL),A
04440           RET
04450 ;         ERASE DASHES
04460 ERASM     LD      A,(CHAR)
04470           CP      '-'
04480           LD      (HL),A
04490           RET     NZ
04500           LD      A,20H
04510           LD      (HL),A
04520           LD      DE,(PLACE1)
04530           DEC     DE
04540           DEC     DE
04550           LD      (PLACE1).DE
04560           PUSH    HL
04570           LD      HL,(PLACE)
04580           DEC     (HL)
04590           POP     HL
04600           RET
04610 ;         ROUTINE TO ERASE PORTION OF SCREEN AND
04620 ;         RECORD CONTENTS OF NEXT PORTION OF SCREEN
04630 ;         AND DISPLAY CHARACTER
04640 ;         APPEARS AS IF IT IS MOVING
04650 DRWCHR    PUSH    HL
04660           LD      HL,(X)
04670           INC     HL
04680           LD      DE,SCREEN
04690           CALL    TRANS
04700 DRWCH1    POP     HL
04710           PUSH    HL
04720           LD      DE,SCREEN
04730           CALL    TRANS3
04740           LD      DE,(CPYCHR)
04750           POP     HL
04760           PUSH    HL
04770           CALL    TRANS6
04780           POP     HL
04790           RET
04800 TRANS     LD      B,6
04810 TRANS1    PUSH    BC
04820           LD      B,24
04830 TRANS2    LD      A,(DE)
04840           LD      (HL),A
04850 TRANSA    INC     HL
04860           INC     DE
04870           DJNZ    TRANS2
04880           PUSH    DE
04890           LD      DE,40
04900           ADD     HL,DE
04910           POP     DE
04920           POP     BC
04930           DJNZ    TRANS1
04940           RET
04950 TRANS3    LD      B,6
04960 TRANS4    PUSH    BC
04970           LD      B,24
04980 TRANS5    LD      A,(HL)
04990           LD      (DE),A
05000           INC     HL
05010           INC     DE
05020           DJNZ    TRANS5
05030           PUSH    DE
05040           LD      DE,40
05050           ADD     HL,DE
05060           POP     DE
05070           POP     BC
05080           DJNZ    TRANS4
05090           RET
05100 TRANS6    LD      B,6
05110 TRANS7    PUSH    BC
05120           LD      B,24
05130 TRANS8    LD      A,(DE)
05140           CP      128
05150           JP      Z,TRANS9
05160           CP      20H
05170           JP      Z,TRANS9
05180           LD      (HL),A
05190 TRANS9    INC     HL
05200           INC     DE
05210           DJNZ    TRANS8
05220           PUSH    DE
05230           LD      DE,40
05240           ADD     HL,DE
05250           POP     DE
05260           POP     BC
05270           DJNZ    TRANS7
05280           RET
05290 ;         RESET EVERYTHING
05300 RETURN    XOR     A
05310           LD      (FLAG1),A
05320           LD      (FLAG2),A
05330           LD      A,3AH
05340           LD      (SET3),A
05350           LD      A,0CDH
05360           LD      (LOOP),A
05370           LD      HL,HALT
05380           LD      (LOOP+1),HL
05390           LD      HL,ENDIT3
05400           LD      (ENDITA+1),HL
05410           LD      HL,ENDIT0
05420           LD      (ENDITC+1),HL
05430           CALL    01C9H
05440           RET
05450 MODE      DEFM    'Mode:          '
05460           DEFB    03
05470 EXIT      DEFM    'Exit: (Era, Cont, or Mem)'
05480           DEFB    03
05490 SPEED2    DEFM    'Speed:                 '
05500           DEFB    03
05510 MOVEME    DEFM    'MOVE  '
05520           DEFB    3
05530 DRAWME    DEFM    'DRAW  '
05540           DEFB    3
05550 ERASEM    DEFM    'ERASE '
05560           DEFB    3
05570 FASTME    DEFM    'FAST'
05580           DEFB    3
05590 MEDME     DEFM    'MED '
05600           DEFB    3
05610 SLOWME    DEFM    'SLOW '
05620           DEFB    3
05630 X         DEFB    0
05640 Y         DEFB    0
05650 PLAG1     DEFB    0       ;MOVE FLAG
05660 FLAG2     DEFB    0       ;ERASE FLAG
05670 FLAG3     DEFB    0
05680 FLAG4     DEFB    0       ;DRAW FLAG
05690 CHAR      DEFB    0
05700 SPEED     DEFW    0
05710 PLACE     DEFW    0000H
05720 PLACE1    DEFW    0000H
05730 SCOUNT    DEFB    0
05740 CPYCHR    EQU     81BCH
05750 BEGIN     LD      HL,9000H
05760           LD      DE,9001H
05770           XOR     A
05780           LD      (HL),A
05790           LD      BC,11520
05800           LDIR
05810           JP      7000H
05820 SCREEN    DEFB    01
05830           DEFS    193
05840           END
```

Three Good Reasons To Buy Software From Mumford Micro Systems:

1. Quality. Mumford Micro has been selling software for the TRS-80 since 1978. Nobody survives in this competitive market for that long on poorly written software. Our best references are thousands of satisfied and repeat customers, but if you haven't spoken to one of them, our 10 day money-back guarantee might secure your confidence. Our rate of return? Less than one in 200.

2. Value. Quality software at exorbitant prices is no bargain. Mumford Micro is known for true value in addition to high quality.

3. Service. We are not distributors or "middle men". We are the original producers of every package we sell. Orders are filled promptly and if you have a problem with one of our programs, you can often call and talk to the author. On those occasions when a problem cannot be resolved on the phone, prompt technical help is available from a programmer who is intimately familiar with the program you purchased.

INSTANT ASSEMBLER

The Instant Assembler is a powerful assembly language development system for the TRS-80. If you are already an assembly language programmer, its unique design will greatly increase your productivity. If you are just getting started, there is no better assembler to help you learn machine language programming. Some of its unique features are immediate assembly, which detects syntax errors as source is entered, and a compact source format that allows you to write programs nearly three times as large as other assemblers in the same amount of memory. It will assemble to disk, tape, or directly to memory for immediate debugging with the built-in debugger, and also produces relocatable code modules that can be saved on disk or tape and linked together in memory for large or modular assemblies. You can quickly switch from assembler to debugger without losing your source. The built-in debugger will step though your programs one instruction at a time, showing each disassembled instruction and its effect on the registers and memory. It will load or save both conventional source files and its own condensed source format.

The Instant Assembler package includes six separate programs. The assembler itself includes the editor and built-in debugger. The **Linking Loader** is included in several versions for different memory sizes. A stand-alone version of the debugger (**MicroMind**) is also included. **MicroMind** can be relocated in memory and has commands to single-step, set breakpoints, display or alter registers or memory, find bytes or words, disassemble to screen or printer, convert between hex and decimal numbers, and write SYSTEM tapes. The Instant Assembler comes with a comprehensive 65 page instruction manual with many examples.

Specify Model I or Model III. TAPE INTASM 2.1 ..... $39.95 on tape
Specify Model I or Model III. DISK INTASM 2.1 ..... $49.95 on disk

TELCOM

Mumford Micro offers two telecommunications programs. **TELCOM I** has most of the features needed to communicate with bulletin boards, time share systems, or for file transfers between two disk-based micros over modems or direct wire. It is menu driven and extremely simple to use. Functions include transmit a disk file, receive a disk file, save received data on disk, examine and modify UART parameters, 8 programmable log-on messages, automatic checksum verification of accurate transmission and reception, and many more user conveniences. Supports line printers, lowercase characters, Xon/Xoff protocol, and programmable character keys.

TELCOM II is an expanded version of this program for the most demanding telecommunications applications. The terminal mode has a help menu and large printer spooler for high baud rates. From within the terminal mode you can load disk files into the memory buffer, type into the buffer, transmit the buffer, or view the buffer or data that has already scrolled off the screen. It has 10 different programmable messages that can each be sent with a single command for auto log-on or auto dialing, and 5 different character translation tables.

TELCOM II also includes an error correction file transfer mode which is compatible with the LYNC program available on CP/M systems and the IBM PC. **TELCOM II** will exchange disk files with any computer running this protocol (including another TRS-80 running TELCOM II), and will automatically detect and correct errors in transmission. Files can be sent to or fetched from an unattended computer. Both versions of TELCOM come with complete instruction manuals, which are available separately for $5 to help you decide which program is best suited to your needs.

Specify Model I or Model III. TELCOM I ..... $39.95 on disk
Specify Model I or Model III. TELCOM II ..... $69.95 on disk

MODEL I SPEED UP

The SK-2 clock modification allows CPU speeds to be switched between normal, an increase of 50% or 100%, or a 50% reduction. Speeds may be changed with a toggle switch (not included) or on software command. It can also be configured to return to normal speed any time a disk is active. It mounts inside the keyboard unit with only 4 necessary connections and is easily removed if the computer ever needs service. The SK-2 has been field proven by 3 years of use, and comes fully assembled.

Model I only. SK-2 ..... $29.95

TYPESETTERS

Call or write for information on our TRS-80 to Compugraphic support package that will allow you to set type with your word processor and create disks on a Model 3 or 4 that will run in your Compugraphic typesetter.

DISK INDEX VERSION 3.2

DISK INDEX will assemble a master index of your entire program library by automatically reading the program names and free space from each disk. The index may then be alphabetized or searched for any disk, program, or extension. It will alphabetize 2000 programs in less than 60 seconds and will find any program out of 2000 in less than 3 seconds. Disks or programs may be added or deleted manually, and the whole index or any selected part may be printed on paper in several different formats. You can quickly display the free space left on each disk and the number of copies of each program in the index. The index itself may also be stored on disk for future access and update. A 48K machine will hold up to 255 disks and over 2000 programs in each file, and you may build as many files as you need. There is no limit to the number of filenames it can read on any one disk. It will run on either a Model I or Model III and catalog disks for either machine regardless of which one is running it, though Model I owners must have double density to catalog Model III disks. It will automatically recognize any DOS and disk density, and has special modes to support unusual configurations of LDOS and NEWDOS. **DISK INDEX** works with any operating system written for the Model I or Model III except CP/M, and is extremely fast and easy to use.

Specify Model I or Model III. DISK INDEX VERSION 3.2 ..... $39.95 on disk

INSIDE LEVEL II

INSIDE LEVEL II is a comprehensive reference guide to the Model I and Model III ROMs which allows the machine language or Basic programmer to easily utilize the sophisticated routines they contain. Concisely explains set-ups, calling sequences, and variable passage for number conversion, arithmetic operations, and mathematical functions, as well as keyboard, tape, and video routines. Part II presents an entirely new composite program structure which loads under the SYSTEM command and executes in both Basic and machine code with the speed and efficiency of a compiler. In addition, the 18 chapters include a large body of other information useful to the programmer. 80 Micro said "The book has no flaws; it is a perfect gem." Byte Magazine said "I recommend this book to serious machine language programmers."

Includes updates for Model III. INSIDE LEVEL II $15.95

DEMON DEBUGGER

DEMON (for DEbugger and MONitor) is a sophisticated tool with which you can explore and debug machine language programs. In the STEP mode, it "emulates" the operation of the Z-80 and allows you to step through any machine language program one instruction at a time, showing you the address, hexadecimal value, Zilog mnemonic, register contents, and step count for each instruction. The 19 different STEP mode commands include step, step to a branch, run in step mode at a variable rate, run for a specified number of steps, change flags or registers, execute a CALL or RST, set breakpoints in RAM or ROM, and break when a number in a defined range appears in any double register. The 26 commands in the MONITOR mode include hex arithmetic, hex to decimal conversion, block move, fill memory, find bytes, jump to address, disassemble to screen, printer, disk, or tape, load memory from disk or tape, write memory to disk or tape, full screen memory edit in hex or ASCII, and relocate other programs or itself. Screen displays may be routed to your line printer for hard copy. **DEMON** includes a comprehensive 40 page manual with many examples.

Specify Model I or Model III. DEMON ..... $39.95 on tape or disk

8748 ASSEMBLER

Assemble programs for the complete Intel MCS-48 family of microcontrollers including the 8741, 8742, 8748, and 8749 on your Model I, III, or IV. Assembles from a source file written on your text editor directly to an object file on disk. It supports the standard Intel mnemonics and features conditional assembly and listing, complete expression evaluation, ten significant characters for symbols, a complete range of pseudo-ops, and informative error messages. It comes with a comprehensive instruction manual which includes the instruction set for each component and sample listings for arithmetic and I/O subroutines. We also offer plans, schematic, and software to help you build your own inexpensive 8748 programmer. The 8748 is an inexpensive ($15) single chip computer that contains RAM, EPROM, clock oscillator, a counter/timer, and 27 I/O lines in a single 40 pin package. A complete computer controller can be built with this one chip, a crystal, three capacitors, and a five volt power supply.

Specify Model I or Model III. CASM48 ..... $125.00 on disk
Plans and software for an 8748 programmer ..... $29.95 on disk

MUMFORD MICRO SYSTEMS

ORDERING: Add $2.50 for postage and handling. California residents add 6% sales tax. Visa, Mastercard, and COD orders are accepted. If not completely satisfied, return your purchase within 10 days for a full refund. Be sure to specify Model I or Model III format.

Box 400-E, Summerland, California 93067 (805) 969-4557
Quality software since 1978

Aerocomp's Proven Best-By Test! The "DDC"

Double Density Controller

$99.00 WINTER SPECIAL

★ Technical Superiority

At last! A double density controller for Model I with HIGHER PROBABILITY OF DATA RECOVERY THAN WITH ANY OTHER DOUBLE DENSITY CONTROLLER ON THE MARKET TODAY! The "DDC" from Aerocomp. No need to worry about the problems that keep cropping up on existing products. AEROCOMP'S new analog design phase lock loop data separator has a wider capture window than the digital types currently on the market. This allows high resolution data centering. The finest resolution available with digital circuitry is 125 ns (nano seconds). The "DDC" analog circuit allows infinately variable tuning. Attack and settling times are optimum for 5-1/4 inch diskettes.
The units presently on the market use a write precompensation circuit that is very "sloppy". Board to board tolerance is extremely wide - in the order of ± 100 ns. The "DDC" is accurate to within ± 20 ns.
The bottom line is state of the art reliability!

★ Test Proven

Tests were conducted on AEROCOMP'S "DDC", Percom's "Doubler A"* and "Doubler II"* and LNW's "LNDoubler"** using a Radio Shack TRS80*** Model I, Level 2, 48 K with TRS80 Expansion Interface and a Percom TFD100* disk drive (Siemens Model 82). Diskette was Memorex 3401. The test diskette chosen was a well used piece of media to determine performance under adverse conditions. The various double density adapters were installed sequentially in the expansion interface.
The test consisted of formatting 40 tracks on the diskette and writing a 6DB6 data pattern on all tracks. The 6DB6 pattern was chosen because it is recommended as a "worst case" test by manufacturers of drives and diskettes. An attempt was then made to read each sector on the disk once - no retrys. Operating system was Newdos/80, Version 1.0, with Double Zap, Version 2.0. Unreadable sectors were totalled and recorded. The test was run ten times with each double density controller and the data averaged. Test results are shown in the table.

★ Features

TRS80 Model I owners who are ready for reliable double density operation will get (1) 80% more storage per diskette, (2) single and double density data separation with far fewer disk I/O errors, (3) single density compatibility and (4) simple plug-in installation. Compatible with all existing double density software.

SUMMER SPECIAL $99.00
for the Best DD Controller on the market.

"DDC" and LDOS	$169.95
"DDC" and Newdos 80	$179.95

★ TEST RESULTS ★

MFR & PRODUCT	SECTORS LOCKED OUT (AVG)
AEROCOMP "DDC"	0
PERCOM "DOUBLER II"	18
PERCOM "DOUBLER A"	250
LNW "LNDOUBLER"	202

Note: test results available upon written request. All tests conducted prior to 8-25-81

Aerocomp's 14 day money back guarantee applies to hardware only. Specials will be prorated. Shipping $2.00 in Cont. U.S. See opposite page for details. Add $4.00 shipping & handling for DDC & DOS.

Data Separators

The advances that make the "DDC" great are incorporated in the new AEROCOMP Single Density Data Separator ("SDS") and Double Density Data Separator ("DDS").

★ Has your original manufacturer left you holding the bag?

If you already own a Percom "Doubler A", "Doubler II" or LNW "LNDoubler" or Superbrain, the AEROCOMP "DDS" will make it right. Look at the test results:

MFR. & PRODUCT	SECTORS LOCKED OUT	
	WITHOUT "DDS"	WITH "DDS"
PERCOM "DOUBLER II"	18	1
PERCOM "DOUBLER A"	250	0
LNW "LNDOUBLER"	202	0

Note: Same test procedures as "DDC".
* Trademark of Percom Data Co.
** Trademark of LNW
*** Trademark of Tandy Corporation

★ "DDS".....$49.95
(Use 1791 chip from your DD Controller)

★ "DDS" with disk controller chip included.....$79.95

★ Disk controller chip.....$34.95

(Shipping $2.00 Cont. US - see opposite page for details)

Plugs directly into your existing Double Density Controller.

Do you need a Single Density Data Separator?

The internal data separator in the WD1771 chip (R/S Expansion Interface) is NOT recommended by WD for reliable data transfer. Do you have any of these problems: Lost data, tracks locked out, CRC errors, disk retry? YOU NEED ONE!

★ "SDS".....$29.95
(For Mod. I; shipping $2.00)

See opposite page ▸▸▸▸▸▸

DISK DRIVES
40 & 80 TRACK
SINGLE & DOUBLE SIDED
as low as $169

NEW!

PACESETTERS

Aerocomp leads the way to the BEST value in disk drives on the market. Quality, performance, reliability, warranty, service plus free trial – that's what you get from the leader. AEROCOMP!

BEST FEATURES

- ★ Fast 5 ms. track-track access
- ★ Single or double density
- ★ Easy entry door
- ★ "Flippy" feature allows read-write to the back of the diskette to cut media cost in half! (MPI)
- ★ Disk ejector (MPI)
- ★ External drive cable connection (no need to remove the cover to hook up the cable)

COMPLETE DRIVES

TRS80 Mod. I & III, IBM PC & TI 99/4A. Power supply & enclosure. Specify silver or almond. 5.25 inch.

- ★ 40 track single side (Tandon) $199
- ★ 40 track SS "Flippy" (MPI) $239
- ★ 40 track Dual Head (either) $279
- ★ 80 track SS (MPI) $299
- ★ 80 track SS "Flippy" (MPI) $329
- ★ 80 track Dual Head (Tandon) $379

Shipping & Handling $5.00 per drive

BARE DRIVES

Internal drives for TRS80 Mod. III, IBM PC, TI 99/4A, 5.25 in. (controller required)

- ★ 40 track Single Side (Tandon) $169
- ★ 40 track Dual Head (either) $249
- ★ 80 track SS (MPI) $269
- ★ 80 track Dual Head (Tandon) $339

Shipping & Handling $4.00 Per Drive

8 INCH DRIVES

Drive expansion box complete with power supply and fan. Tandon Slimline.

- ★ Two (2) 8" Single Side $699
- ★ Two (2) 8" Double Side $849
- ★ 8" Bare Slimline, SS $260
- ★ 8" Bare Slimline, DS $375

Shipping & Handling $5.00 Per Drive

NEW!

MODEL III & 4 CONTROLLER and DRIVES

Convert cassette Model III or 4 to disk. Complete internal drive kits with 40 track SS drives (Tandon), Aerocomp disk controller board (will take up to 4 drives), power supply, mounting towers, all hardware & cables.

- ★ DRIVE KIT (no drives) $199
- ★ ONE DRIVE SYSTEM $369
- ★ TWO DRIVE SYSTEM $539

Shipping & handling $8.00 per system

- ★ AEROCOMP DISK CONTROLLER ... $119

Shipping & handling $2.00 per controller

- ★ MOUNTING KIT w/o Drives & Controller .. $99

Shipping & handling $4.00 per kit

MODEL I STARTER PACKAGE

One 40 track SS drive, 2-drive cable, TRSDOS 2.3 disk & manual, freight & insurance (Tandon).

$249

MISCELLANEOUS GOODIES

- ★ TRSDOS 2.3 disk & manual $20
- ★ LDOS (Mod. I or III) $119
- ★ NEWDOS/80, 2.0 (Mod. I or III) $129
- ★ Diskettes (10 in library box) $23
- ★ 5.25" Drive Power Supply & case $59
- ★ 2-Drive Cable $24
- ★ 4-Drive Cable $34
- ★ Extender Cable $13

Shipping & Handling $2.00

PERSONAL CHECKS WELCOME

We'll be happy to accept your personal check with any mail order without any shipping delay.

FREE TRIAL OFFER

Use your AEROCOMP drive for up to 14 days. If you are not satisfied for ANY REASON (except misuse or improper handling), return in the original shipping container for a full purchase price refund. Applies to hardware only. Sorry, we cannot refund on software. We have confidence in our products and we know you will be satisfied.

WARRANTY

We offer a six months warranty on parts and labor against defects in materials and workmanship. In the event service becomes necessary for any reason, our service department is fast, friendly and cooperative. Our goal is 48 hour turnaround on all warranty or repair drives!

100% TESTED

AEROCOMP disk drives are 100% subjected to burn-in and bench test. We even enclose a copy of the test check list, signed by the test technician, with each drive. AEROCOMP means reliability!

ORDER NOW!

Order by mail or call TOLL FREE TO THE NUMBERS BELOW. Please note toll free lines will accept orders only. We accept VISA or MASTERCARD. Be sure to include card number and expiration date. We will not charge your card until the day we ship. Order by mail with credit card or send check or money order. Personal checks welcome. No delay. Order COD. No deposit required but all COD's will arrive cash, certified check or money order only. We'll send a card showing the exact COD amount before your shipment arrives. Shipping is **not** included in the prices shown. Texas residents add 5% sales tax.

CALL TOLL FREE FOR FAST SERVICE
(800) 824-7888, OPERATOR 24
FOR VISA/MASTERCHARGE/C.O.D. ORDERS
California dial (800) 852-7777, Operator 24. Alaska and Hawaii dial (800) 824-7919, Operator 24.
TOLL FREE LINES WILL ACCEPT ORDERS ONLY!
For information only call (214) 339-8324.
For TECHNICAL ASSISTANCE call (214) 337-4346.

Dealer inquiries invited

AEROCOMP
Redbird Airport, Bldg. 8
P.O. Box 24829
Dallas, TX 75224 ⌐82

⌐ See List of Advertisers on Page 211

Seven 100 Percent Solutions

by Rolf A. Deininger

Once upon a time, there was darkness in Tandyland. The apples and oranges and other fruits blossomed and the Big Blue grapes grew bigger and bluer each day, but the models numbered one through four would not sell.

And John, prophet of Tandyland, looked throughout the warehouses at the other items, the works of the mighty Archer and others, and chose batteries, saying, "If we cannot sell computers, then let us sell batteries."

"But how do we do this?" asked the sages and shopkeepers.

And John said, "Realistically. Let us find a toy which everyone on this earth needs, and one which devours batteries. Let it consume two hundred AA batteries in a year."

Now the shopkeepers were conservative, and were abashed by the boldness of John. They thought they could sell but 100 batteries per year, and thus the name Model 100 was born.

The wise men searched high and low for such a product, until Jon, aide of John, prophesied, "Let us look to the land of the rising sun, and the software bedlam of Bellevue."

So they looked to Japan and Washington, and men there created a portable computer. And John, Jon, and the sages and shopkeepers rejoiced, because they saw that the toy was good and that it would sell many batteries.

And later, Jon left for Bellevue himself, to be received by open Gates. But here endeth the story of John and Jon. And our story begins.

The Problem

In my first two weeks with the Model 100, I went through three sets of alkaline AA batteries at a cost of roughly $10. Scared by the prospect of a $200 annual battery bill, I bought an ac adapter ($5.95)—which freed me from alkalines, but kept me within a 75-inch radius of the nearest wall outlet.

A 50- or 100-foot extension cord would give me some leeway, but not enough to write a story at an airport. The gangways are long, the doors of the plane crimp the cord, and when the plane taxis from the gate, I lose both the power and the cord.

What I needed was an auxiliary power supply capable of giving my 32K RAM Model 100 the 60 milliamperes (mA) it needs for longer than the 20 hours available on alkaline batteries (20 hours × 60 milliamps = 1,200 mAhrs or milliamperehours).

The Solutions

I came up with seven possible solutions. Solution 1 was to replace the Model 100's four alkaline batteries with rechargeable nickel-cadmium batteries—about $10 for four NiCads, plus $7 for a battery charger. Unfortunately, this doesn't work very well.

The NiCads produce 5 volts when fresh and then drop to 4.8 volts, triggering the Model 100's low-power light. At this voltage, the Model 100 draws 75 mA, and up to 120 mA while dialing the telephone.

Five NiCads would provide the necessary 6 volts, but there's no room for a fifth cell in the battery compartment.

Solution 2, by contrast, works like a charm: Put only three NiCads into the 100 and tape a two-battery holder (Radio Shack part number 270-382, 79 cents) to the top of the case to hold a few more NiCads (see Photo 1).

With the outside battery pack properly connected to the partially empty battery compartment, the NiCads produce about 450 mAhrs, powering the computer for seven to eight hours between recharges.

Fitting five batteries into a recharge holder for four is a problem, but so is this solution's ugly appearance.

Solution 3 is a similarly simple, clumsy kludge. A 6-volt

Photo 1. Model 100 with two AA batteries taped next to the display.

Photo 2. Gel cells—from a 9.5-Ahrs Sears cell to a 1-Ahrs cell.

lantern battery ($3 to $13) and a homemade cable using Radio Shack's coaxial adapter plugs (Radio Shack part number 274-1551, $1.69 for two) provide from three to 40 ampere-hours (Ahrs) of power. A 20-Ahrs Duracell MN908 battery ($6), for instance, should power your Model 100 for over 300 hours; a 40-Ahrs MN918 battery ($13) for over 650.

Solution 4 is a compromise: Attach the plug described in Solution 3 to a six-cell battery holder for C cells. Put five rechargeable NiCads (about $16, plus $12 for a charger) in the holder. For the sixth cell space, use any dead C battery. Solder a wire from the bottom to the tip of the dead battery so it serves as a space filler only.

A fully charged set should last between 15 and 20 hours. If you don't want to use a dead cell to fill the holder, you can use six NiCads; 7.2 volts won't hurt the 100 and the current will go down to about 50 mA.

Gel Cells and Solar Power

Solutions 5 and 6 are more exotic. Gel cells are normal electrolyte batteries (like your car battery), except that their electrolyte solution is gelled; they are excellent power supplies, a bit heavy, but with ample reserve for remote computing.

Photo 2 shows a 9.5-Ahrs Sears cell (toy catalog part number 49 N 86522, $15) and three Globe Industries cells (7.5, 1.8, and 1 Ahrs). You'll have to make the appropriate cables and buy a charger ($6 to $10).

Returning to Radio Shack, Tandy's solar panel (part number 277-1250, $25) delivers 80 mA at 6 volts in bright sunlight (see Photo 3). If you can keep the panel in the sun and yourself and the 100 in the shade, it's okay. However, it doesn't work at night, and it makes it hard to take notes in a lecture hall.

Solar power may not be practical, but my other solutions are bargains compared to alkaline AA cells. With 20 hours' computing per week, you'd need 52 sets (208 batteries) per year—at about $3 per set, a $156 total. By contrast, just three Duracell MN908 lantern batteries will provide a year's worth of use for $18.

Rechargeables are even cheaper. Solution 4's five NiCad C cells will survive several hundred weekly recharges, or at least three years' service. If the cells, charger, and cable cost $30, that's a modest $10 per year. And the Sears gel cell mentioned in Solution 5, about $21 with charger and cable, will run your 100 for $7 a year.

Photo 3. A solar power panel for the Model 100.

The Ideal Solution

The best bargain of all? Solution 7: Enroll 18 friends or relatives in Radio Shack's Battery-of-the-Month Club (see p. 150 of the 1984 Radio Shack catalog). Each month, have your friends pick up the give-away AA battery. Except for the gas they use trotting to participating Radio Shack dealers, it's free. ■

Rolf A. Deininger is a professor of environmental health at the University of Michigan's School of Public Health, Ann Arbor, MI 48109.

The Shadow Knows

by Richard Ramella

Lamont Cranston, or The Shadow as old-time radio listeners will remember, had the ability to cloud men's minds. I've named this short program in his honor. LAMONT.BA protects sensitive text material in the Model 100 by encoding it (see Program Listing 1).

The encrypted files thwart the casual sneakpeek as well as the most determined snoop. In addition, LAMONT.BA befuddles many line printers' efforts to print the coded version. And the same program both encodes and decodes text files.

LAMONT works by increasing the ASCII value of each of your old file's characters by 100, then sending them to a new .DO file. The resulting program code looks like gibberish because it comprises both pictographic and non-English characters instead of conventional ASCII character values.

Using the Program

First, create a short .DO file to test the system. Call it Test.DO.

Then run LAMONT. Answer the first prompt, "Will you be 1-Coding? 2-Decoding?", by typing 1 and pressing the enter key. Answer the next prompt, "File to be coded?", by typing TEST and hitting the enter key. Answer the last prompt, "Name of new file?", with TEST2 and tap the enter key.

Once the program writes the new encoded or decoded file, it displays "File transferred," and beeps 20 times. It also reminds you to destroy the plaintext version by stating: To kill old file, type "KILL TEST.DO" and tap enter.

To see the coded version of your text, press the F8 key to enter the menu mode. Put the cursor over the file named TEST2 and press the enter key.

The Key Box

The programs in "The Shadow Knows," "Backing Up the 100," and "Write Away" run in 8K RAM.

Decoding Material

To decode your text, run LAMONT.BA again. This time answer the first prompt by pressing 2 and then the enter key. The old file is now TEST2, the new file, TEST3, and the program has restored the Text file to plaintext.

To store a Basic program file in plaintext, first run the program, then stop it by pressing the shift and the break keys simultaneously. Note that all subsequent references to the word program in commands denote the name you gave the program, not the word itself. Type SAVE"PROGRAM.DO" and press the enter key.

To reclaim the program in Basic, decode the coded file, then type SAVE"PROGRAM" and wait until it begins to run. Stop the run by pressing the shift and break keys together, then type SAVE"PROGRAM" and the program returns to a Basic file.

Further Security

You can, of course, enhance this method in order to provide more security. Line 280 is a likely place to customize LAMONT.BA. Note that 100 appears twice in that line. You can substitute any single number from one to 133 to yield a different set of encryption symbols. Be sure to use the same number in both places.

If you're really paranoid, you can store an encoded text file or program in two different files, sending every other character to alternate files, then have the same program reassemble the file in plaintext. ■

```
100 REM * Lamont * TRS-80 Model 100 8K *
Richard Ramella
110 MAXFILES=2
120 CLS
130 PRINT "Will you be...
140 PRINT "1 - Coding
150 PRINT "2 - Decoding
160 INPUT Z
170 IF Z<>1 AND Z<>2 THEN 120
180 PRINT "File to be ";
190 IF Z=2 THEN PRINT "de";
200 INPUT "coded";A$
210 INPUT "Name of new file";B$
220 IF LEN(B$)>6 THEN CLS: PRINT "New
file name must be in 6 or fewer
characters.": GOTO 210
230 BEEP
240 PRINT "Translation begun. Patience,
please."
250 OPEN A$ FOR INPUT AS 1
260 OPEN B$ FOR OUTPUT AS 2
270 C$=INPUT$(1,1)
280 IF Z=1 THEN C$=CHR$(ASC(C$)+100)
ELSE C$=CHR$(ASC(C$)-100)
290 PRINT #2,C$;
300 IF EOF(1) THEN 320
310 GOTO 270
320 PRINT "File transferred."
330 FOR T=1 TO 20
340 BEEP
350 NEXT
360 PRINT "To kill old file "A$",
370 PRINT "Type " CHR$(34)"KILL
"A$".DO"CHR$(34)" and tap Enter
380 END
```

Program Listing 1. LAMONT.BA.

You can reach Richard Ramella at 1493 Mt. View Ave., Chico, CA 95926.

Backing Up the 100

by Bryan R. Leipper

In spite of the Model 100's constant memory feature, you still need to back up your files. If you maintain several document files, it's inconvenient to save them all to tape by going in and out of Text or by typing the file names. Here's a way to automate that process.

BKUPDO (Program Listing 2) PEEKs into memory to find the file names for all of the document files in the Model 100 menu. The program asks you whether you want to store each file to tape. It notes the files you select for tape back-up by typing a Y, stores them in a cassette file, and then reads the files from memory and stores them to tape.

To ensure that the 100 properly stores the files on cassette, BKUPDO has a verify function that compares the tape files with the RAM files. It tells you if you changed a line in the file, changed the length of a file, or if there is an input/output error in the file. One error that doesn't show up directly is when the 100 cannot find a file (probably due to a recording problem). In this case the tape machine continues to run, and you have to hit the break key or control/C to interrupt the program.

When you want to reload the programs, BKUPDO first reads the list of file names from cassette and then reads the files and places them in memory. Be careful to avoid overwriting updated files that you haven't backed up.

How It Works

The 100 stores its directory of user files starting at memory location 63930 (F98A hexadecimal (hex)). Each entry is 11 bytes long. The first byte identifies the type of file, with 80 hex indicating a Basic program file and a C0 hex (192 decimal) indicating a document file. This is followed by a 2-byte starting address for the file. Then the Model 100 stores the file name as 6 bytes, padded on the right with blanks. The last 2 bytes are the ASCII codes for the extension, which should be .DO for document files.

Knowing this, you can write a Basic program to PEEK at the addresses looking for the proper codes in order to extract a file name. The computer then reads and stores this file on tape.

When you run the program, you must first decide whether you want to read files from the tape, save files to tape, or verify what you've previously saved. If you select R to read the tape, make sure you have a proper back-up tape in the cassette machine ready to play. The program looks for cassette file FINM to read a list of names and then will load each file just as it was recorded. All you have to do is wait a few minutes until the program loads all of your document files to RAM.

Typing S makes the program search memory for document file names. As it finds them, the program asks if you want the file saved to cassette. An upper- or lowercase Y sets the file name in an array for later processing. Any other response bypasses back-up of the file named. After the program ex-

Program Listing 2. BKUPDO.

```
10 CLEAR2E3 :DIMF$(19) :MAXFILES=2
15 INPUT"(R)ead, (S)ave, or (V)erify";B$
:IF B$="R" THEN 700 ELSE IF B$="S" THEN
500 ELSE IF B$="V" THEN 300 ELSE 15
20 OPEN"RAM:"+F$(J%)+".DO" FOR INPUT AS
1 :RETURN
30 OPEN"CAS:"+F$(J%) FOR INPUT AS 2
:RETURN
60 OPEN"CAS:FINM" FOR INPUT AS 2 :X%=0
:PRINT"Reading File Names from cassette"

70 IF EOF(2) THEN 90 ELSE X%=X%+1
:LINEINPUT#2,F$(X%) :IF F$(X%)="" THEN
X%=X%-1
75 PRINT F$(X%);"- ";
80 GOTO70
90 PRINT:CLOSE:RETURN
100 I=I+1:A=PEEK(I)+256*PEEK(I+1)
120 N$=""
130 FOR I=I+2 TO I+5
140 J%=PEEK(I) :IF J%>32 THEN N$=N$+CHR$
(J%)
150 NEXT
160 EX$=CHR$(PEEK(I))+CHR$(PEEK(I+1))
170 IFEX$<>"DO"THENPRINT"Bad
extension":RETURN
180 PRINT"File: ";N$;".DO at
address:";A:PRINT"store on cassette (y/n
)? ";
190 B$=INKEY$:IFB$=""THEN190
200 IFB$="Y"ORB$="y"THENX%=X%+1:F$(X%)
=N$:PRINT"save ";N$ELSEPRINT"skip ";N$
210 RETURN
300 PRINT"Failure to find file will
cause": PRINT"cassette to run until tape
out.":PRINT"Looking for FINM":ON ERROR
GOTO 450
310 GOSUB60
320 FOR J%=1 TO X%
330 PRINTUSING"Attempting to verify: \
  \";F$(J%);
340 GOSUB30:PRINT" Found ";
350 GOSUB20
360 IFEOF(1)ANDEOF(2)THEN PRINT" ok"
:GOTO410
370 IFEOF(1)OREOF(2)THENPRINT" length
mismatched":GOTO410
380 A$=INPUT$(1,1):B$=INPUT$(1,2)
390 IFA$<>B$ THEN PRINT"char mismatch"
400 GOTO360
410 CLOSE:NEXT
420 PRINT:PRINT"Verify Completed":END
450 MOTOR OFF:PRINT:PRINT"Machine error
at file: ";F$(J%)
460 RESUME 410
500 I=63930:X%=0
510 IF PEEK(I)=192THENGOSUB100:GOTO510
520 I=I+1:IF I<64140 THEN 510
530 IF X%<1 THEN END ELSE INPUT"prepare
cassette and hit <ENTER>";B$
540
OPEN"CAS:FINM"FOROUTPUTAS2:PRINT"Saving
FIle NaMes":FORJ%=1TOX%:PRINT#2,F$(J%)
:NEXT:CLOSE
550 FOR J%=1TOX%
560 PRINT"Saving: ";F$(J%)
570 GOSUB20
```

Listing 2 continued

Listing 2 continued

```
580 OPEN"CAS:"+F$(J%)FOROUTPUTAS2
590 IFEOF(1)THEN610ELSEB$=INPUT$(1,1)
600 PRINT#2,B$;:GOTO590
610 CLOSE:NEXT
620 CSAVE"BKUPDO":END
700 GOSUB 60
710 FOR J%=1 TO X%
720 PRINT"Reading: ";F$(J%)
730 OPEN"RAM:"+F$(J%)+".DO"FOROUTPUTAS1
740 GOSUB 30
750 IFEOF(2)THEN770ELSEB$=INPUT$(1,2)
760 PRINT#1,B$;:GOTO750
770 CLOSE:NEXT
780 CLOSE:END
```

amines all file names, it writes file FINM to cassette, followed by each file in the back-up list. Finally, a CSAVE"BKUPDO" command saves a copy of the program as a follower to the data files. This command stops program execution.

Because the 100 requires a .DO extension for files opened in Basic, it cannot save Basic and command program files by reading them from memory. This isn't usually a nuisance as working programs are not often updated. You can make backups by the usual loading and saving routine with little inconvenience. ■

Contact Bryan R. Leipper at 714 Terra Court, Reno, NV 89506.

Write Now

by Ronald F. Balonis

While not a full-blown word processor with fancy editing features, Writer.BA is a print processor that complements the Model 100's text editor (see Program Listing 3).

Writer.BA provides the most basic formatting features (see Table 1): page length (PL), left margin (LM), right margin (RM), top margin (TM), bottom margin (BM), line spacing (LS), and page numbering (PG). The program left-justifies the printed text and determines line length using the last space in the text before the right margin value. The only in-text editing command is the down-arrow key used to start a new page.

Program Operation

I chose variable names that relate closely to the variables' functions. Program lines 0–30 initialize the variables and the screen. In lines 35–75, the program reads the directory with the text files printed on the screen. Lines 80–90 prompt for the name of the file you want printed. Then lines 100–180 open the file, read the format line, and set the print format parameters. Lines 190–450 read and print the text file to your format specifications. Lines 500–600 error-trap for file or format errors; all other errors cause an abort to the menu.

Using the Program

To save memory and attain an acceptable printing speed (about 30 characters per second), I limited the program's functions to those I usually need. You must start a Writer text file with a format line: begin with a greater-than sign and end with an end-of-text marker (by pressing the enter key).

See List of Advertisers on Page 211

101 Business Letters

Ready to Use... On Disk...

- NO WRITING!
- NO TYPING!
- NO DICTATION!
- PRINT A PROFESSIONALLY WRITTEN LETTER IN SECONDS!

FINALLY . . .

- Save time
- Keep Business correspondence up to date
- Improve company image
- Speed up collections
- Increase mailing response
- Respond promptly to all customer and prospect queries
- Promote new business
- Keep in touch with clients and vendors

ORDER *GOLD LETTERS* TODAY!

One Hundred and One professionally written business letters on diskette, allowing you to automatically answer all your business correspondence needs.

Twelve major categories are included, covering every conceivable business situation.

No typing needs, because each letter is complete and ready for addressing.

Personal revisions are easy, because Gold Letters is accessible by all major word processing programs.

Ready to use in minutes, with no programming or typing necessary.

Your complete 101 Gold Letters software package includes a 5¼" or 8" diskette, a binder containing hard copy versions of all 101 letters by category, and instructions for using the Gold Letters.

Available in CP/M, MS/DOS and PCDOS.

Price for entire package: Diskette binder, 101 categorized letters and instructions. $159.00

Add $5.00 shipping & handling costs. CA residents add 6% sales tax.

SPECIAL
Gold Letters and WordStar™
$395.00

Letter Listing

- 9 credit letters
- 4 marketing letters
- 13 customer relations letters
- 8 employee letters
- 12 contracting products letters
- 6 follow up letters
- 9 sales letters
- 4 media letters
- 7 collection letters
- 2 sympathy letters
- 9 services letters
- 4 procurement letters
- 21 other business letters!

FOR IMMEDIATE DELIVERY CALL 800-922-5555

Visa and MasterCard accepted.

To order by mail, write to:

Data Base Industries -63

330 W. Felicita Avenue
Suite D-6
Escondido, CA 92025

Yes, send me ____ copies of Gold Letters for $159 each.

Enclosed is $ _____

Name _____

Company _____

Address _____

City/State/Zip _____

Telephone _____

Visa/MC # _____

Exp: _____

Mail to:
Data Base Industries
330 W. Felicita Ave., Suite D-6
Escondido, CA 92025
619/480-9616

Format	Default	Low	High
Page Length	PL-66	1	90
Left Margin	LM-12	0	131
Right Margin	RM-72	1	132
Top Margin	TM-6	1	89
Bottom Margin	BM-60	2	90
Line Spacing	LS-1	1	90
Page Numbering	PG-0 (off)	0	(on) 1

Table 1. Text format values.

The program tests the format values to ensure that the given range of values exist; the rest is left up to you. The format line need only contain those format values that differ from the default values listed in Table 1.

A typical format line might look like this:

>PL=66 LM=12 RM=72 TM=6 BM=60 LS=1 PG=0

The program consists of two screens: The first, or initialization, screen lists your text files and prompts you to type in the name of a file you want printed or press the enter key to exit the program. The second screen notifies you of format or file errors or that the program is printing a file.

After it prints a file, the program redisplays the first screen so you can either select and print another text file or exit the program.

Initially, it might seem difficult to visualize how the text on the Model 100 screen would appear once printed, but with experience you'll become proficient at it. ■

You can reach Ronald F. Balonis at 118 Rice St., Trucksville, PA 18708.

Program Listing 3. Writer.BA.

```
0 ' WRITER.BA  MEMO SCRIBE
2 ' A TEXT PRINT PROCESSOR
5 'BY RON BALONIS
10 CLS:CLEAR500:DEFINTA-Z:N=0:NX=0
12 ON ERROR GOTO 500
15 NPAGE$="":CHAR$="":LINE$=""
20 TITLE$="---- Memo Scribe ----"
25 LF$=CHR$(10):CR$=CHR$(13):SPAC$=" "
30 CLS:PRINT@10,TITLE$:PRINT:FNME$=""
32 '
35 '--READ MENU FOR .DO FILES ONLY--
40 FOR I!=63930 TO 64139 STEP 11
45  B$="":IF PEEK(I!)=0 THEN 75
50  FOR II=3 TO 10
55   B$=B$+CHR$(PEEK(I!+II))
60  NEXT II
65  IF INSTR(B$,"DO")<1 THEN 75
70  PRINTLEFT$(B$,6)"  ";
75 NEXT I!:PRINT
80 PRINT@240,"<FILE NAME> OR <> TO
EXIT";:INPUT FNME$
85  IF FNME$="" THEN BEEP: MENU
90   FNME$=LEFT$(FNME$,6)
95 '
98 '--GET FILE AND PRINTING FORMAT--
100 OPEN FNME$ FOR INPUT AS 1
```

Listing 3 continued

See List of Advertisers on Page 211

Listing 3 continued

```
105  INPUT#1,FØRMT$:PAGE=1
110  CLS:PRINT@10,TITLE$:RESTORE:ERX=0
115  IF LEFT$(FØRMT$,1)<>">" THEN 600
120   FOR I=1 TO 7
125    READ T$,L,H,DFLT
130    II=INSTR(FØRMT$,T$)
135    IF II=0 THEN 150
140     DFLT=VAL(MID$(FØRMT$,II+3,2))
145     IF DFLT<L OR DFLT>H THEN ERX=-1
150      X0(I)=DFLT
155   NEXT I
160   PL=X0(1):LM=X0(2):RM=X0(3)
165   TM=X0(4):BM=X0(5):LS=X0(6)
170   PG=X0(7):LCNT=0:LGTH=RM-LM:PN=0
175   IF LM>=RM OR TM>=BM OR BM>PL THEN
ERX=-1
180    IF ERX=-1 THEN 600
185 '
190     PRINT@120,"  -- PRINTING TEXT
FILE < "FNME$" > --"
200     L1NE$="":N=0:LCNT=0:PN=PN+1
210     IF TM=1 THEN 300
220      FOR I=1 TO TM
230       FOR II=1 TO LS
240        LPRINT"":LCNT=LCNT+1
250       NEXT II
260      NEXT I
290 '
300  CHAR$=INPUT$(1,1):IF EOF(1) THEN
350
305  IF CHAR$=NPAGE$ THEN 410
310   IF CHAR$=CR$ THEN 350
315    IF CHAR$=LF$ THEN 300
320     N=N+1:L1NE$=L1NE$+CHAR$
325     IF CHAR$=SPAC$ THEN NX=N
330 '
335     IF N<LGTH THEN 300 ELSE 360
340 '
350   LPRINTTAB(LM)L1NE$;
355   L1NE$="":N=0:NX=0:GOTO370
360   LPRINTTAB(LM)LEFT$(L1NE$,NX);
365   L1NE$=RIGHT$(L1NE$,N-NX):N=N-NX
370   FOR I=1 TO LS
375    LPRINT"":LCNT=LCNT+1
380   NEXT I
390 '
400   IF LCNT<BM AND NOT EOF(1) THEN 300

410    FOR I=LCNT TO PL-1
415     IF I<>PL-3 THEN 425
420      IF PG=1 THEN LPRINTTAB(((RM-LM)
/2)-7+LM)USING"Page ##";PN;
425       LPRINT""
430    NEXT I
440     IF EOF(1) THEN CLOSE 1 ELSE 200
450      GOTO 30
490 '
500 CLOSE 1
510  IF ERL<>100 OR ERL<>105 OR ERL<>300
THEN MENU
520  PRINT@250,"---- FILE ERROR ----"
530  FOR I=1 TO 500:NEXT I:RESUME 30
540 '
600 CLOSE 1
610  PRINT@250,"--- FORMAT ERROR ---"
650  FOR I=1 TO 500:NEXT I: GOTO 30
800 END
890 'FORMAT,LOW,HIGH,DEFAULT--VALUES
900 DATA PL,1,90,66,LM,0,131,12
905 DATA RM,1,132,72,TM,1,89,6
910 DATA BM,2,90,60,LS,1,90,1
915 DATA PG,0,1,0
```

Pin Pals

by Ronald F. Balonis

You can transfer Basic and Text files between two Model 100s with a homemade cable. I'll describe the construction of such a cable and asynchronous file transfers using the Model 100 RS-232 ports for file input/output (I/O). See Table 2 for a list of parts necessary for this project.

Constructing the Cable

Since the Model 100's RS-232 port is flush with its case, you first have to trim the plastic lips that cover the pin flange on the cable covers (see Photo 4). Lay the covers on a flat surface and cut through the molded groove with a hacksaw blade.

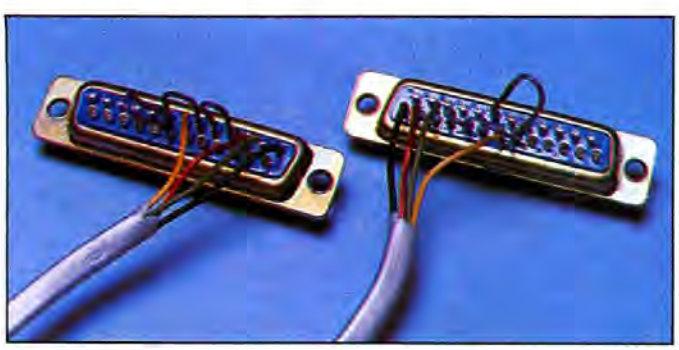


Photo 4. Pin connections for null modem RS-232 connectors.

Next, cut six 1-inch jumper wires from some hook-up wire; trim 1/16 inch of insulation from the ends of each and tin the exposed ends with solder.

Then, on both ends of the cable, cut the outside insulation back 1 inch and trim 1/16 inch of insulation from each cable wire and tin the ends with solder.

On each 25-pin connector, solder jumper wires as follows (see the Figure): one from pin 4 to 5, one from pin 6 to 8, and one from pin 8 to 20.

Push the cable ends through the covers.

To one 25-pin connector and cable end, solder the wires as follows: black to pin 1, green to pin 2, red to pin 3, and yellow to pin 7.

To the other connector and cable end, solder the wires as follows: black to pin 1, red to pin 2, green to pin 3, and yellow to pin 7.

Quantity	Part Description	Manufacturer	Part Number
5 feet	conductor cable	Radio Shack	278-365
2	25-pin D submini connectors	Radio Shack	276-1547
2	25-pin D submini hoods	Radio Shack	276-1549

Table 2. Parts list.

INTRODUCING

Mt. Olympus SOFTWARE

INTEGRATED ACCOUNTING SYSTEM

IS YOUR SYSTEM BARE? Did you purchase or plan to purchase your first computer system and find that you did not select the right software to run on it. MT. OLYMPUS SOFTWARE can dress up your system with applications for over 30 different types of businesses or professions. These applications all come complete with all the programs you will ever need to run your system. Best of all, you get a 5 megabyte capacity hard drive at no extra charge. If you purchased these two items separately, you would pay over $ 2500.00 for only the software and $ 1500.00 or more for the hard drive unit. Yet we do not sacrafice quality for price. Most of our integrated packages with the hard drive are only $ 2495.00 for the TRS-80 and $ 2995.00 for the IBM-PC and IBM clones. If you already own a hard drive or IBM-PC XT then deduct $ 1000. Just add $ 500.00 for a 10 megabyte drive or $ 2800. for a 30 megabyte drive. Several packages like our Medical Office, Tax Preparation, Materials Requirement Planning, Hotel Management, and a few others are priced slightly higher. But you eliminate disk swapping. A 5 meg drive can hold the capacity of about 30 floppy disks of 175k each. Plus your system runs faster and more reliably. And our features far surpass those of our competition, who give you only run-of-the-mill AP, AR, GL, Payroll. So when you need a system that has already been customized to your needs and has been proven in the field, just drop us a line telling us your specific needs and we will send you information on your particular application or put you in touch with the dealer nearest you. Of course we will not be able to help everyone, but new applications are being added each month.

APPLICATIONS FOR

ACCOUNTANTS
ALARM SERVICE
AUTO PARTS
BOOK STORES
BUSINESS BROKERS
CONTRACTORS
ENGINEERING FIRMS
FINANCIAL CONSULTANTS
FUEL OIL DISTRIBUTORS
GIFT SHOPS
HOTELS
LABOR UNIONS
LAWYERS
LEASING/RENTAL FIRMS
LOAN/FINANCE COMPANIES
MAIL ORDER FIRMS
MAILING SERVICE
MEDICAL OFFICE
MANUFACTURING INVENTORY
OFFICE MANAGEMENT
OIL & GAS WELL MGT.
PERSONNEL AGENCIES
PHARMACIES
PROPERTY MANAGEMENT
REALTORS
RESTAURANTS
RETAILERS
SCHOOL SYSTEMS
TAX PREPARERS
TIRE DEALERS
UTILITY BILLING
VENDING/ROUTE SALES
VIDEO STORES
WHOLESALERS
. . PLUS MANY MORE
. . NOT LISTED HERE

CALL **(203) 828-0027** **(203) 828-0359**

OR WRITE TO: **MT. OLYMPUS SOFTWARE**
869 MILL ST.
EAST BERLIN, CT 06023

DEALER INQUIRIES INVITED

↩23

↩ See List of Advertisers on Page 211

Fasten the connectors to the covers with the screws provided and tighten the strain relief screws. Your null modem cable is ready for use.

File Transfer Procedure

On the Model 100, communications files require an RS-232 transmission configuration (refer to p. 125 of the reference manual for more detailed information).

The configuration I use, 88E1E, is 9,600 baud, 8-bit word length, even parity, 1 stop bit, and XON/XOFF enabled.

Transferring Basic Programs

To transfer a Basic program file, connect the null modem cable to both machines. Load Basic on the destination machine, type LOAD "COM:88E1E" and press the enter key.

Then load Basic on the source machine, load the file to transfer, then type SAVE"COM:88E1E" and press the enter key. When the cursors on both machines reappear, indicating the transfer is complete, type SAVE"PROGRAM NAME" on the destination machine, and press the enter key.

Transferring Text Files

To transfer a text file first interconnect the machines with the null modem cable. Load the 100's built-in Text file on the destination machine, create a text file with the same file name, and press the F2 key (Load). Then type COM:88E1E and press the enter key.

Load Text on the source machine, load the text file to transfer, press the F3 key (Save); then type COM:88E1E and press the enter key. When the cursors reappear, the transfer is complete.

Both cable and procedure have worked well for me. With the configuration of 88E1E, small programs of 1 to 2 kilobytes require just a few seconds. ■

Contact Ronald F. Balonis at 118 Rice St., Trucksville, PA 18708.

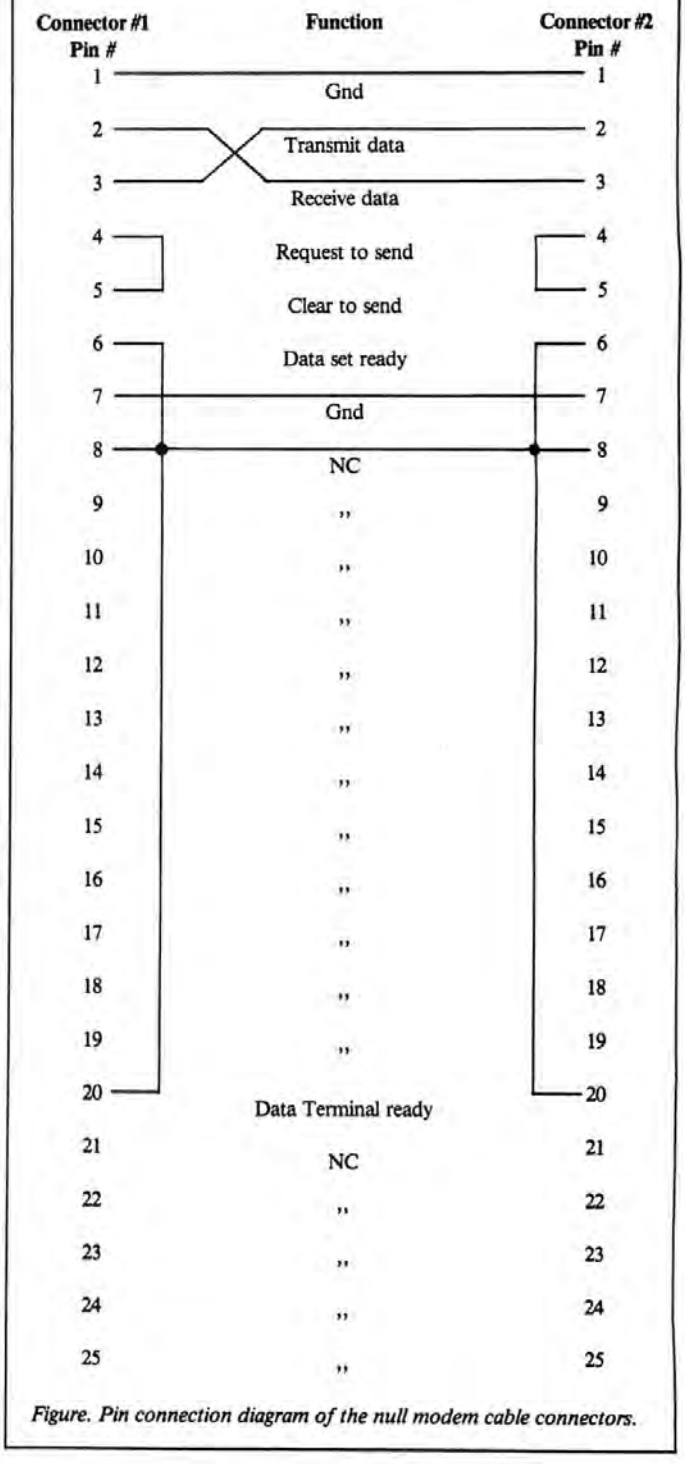


Figure. Pin connection diagram of the null modem cable connectors.

Peripheral Problems

While on my way through a Model 100-to-peripheral routine for the serial port, I tried without success to use STR$ for the conversion of values to string form as the instruction manual description suggests.

I traced the peripheral's resulting fits and starts to the computer's unexpected insertion of a space character just before the numerals in each output string.

If your unit has the same characteristics as mine (serial number 301005984), you may see the problem by typing and running the following routine, based on the manual's example:

```
10 BAL = 133
20 B$ = "$" + STR$(BAL) + ".00"
30 PRINT B$
```

Note that the unwanted space occurs between the dollar sign and the numeral 1 on the screen display.

One way to handle this is to substitute a synthetic command for STR$, automatically removing the leading space:

```
RIGHT$(STR$(n.e.),LEN(STR$(n.e.)) - 1)
```

where n.e. is the numeric expression you want to convert.

This command is long enough to qualify for subroutine status when used frequently in a program. You can see its effectiveness by running a revised version of the above program that includes the solution as part of two additional lines:

```
40 B$ = "$" + RIGHT$(STR$(BAL),LEN(STR$(BAL)) - 1) + ".00"
50 PRINT B$
```

The screen printout provides results both before and after running this program for quick performance confirmation.

John M. Hicks
117 Presidio Court
Verona, PA 15417

Mighty Write Debug

In the PRNTXT.BA program in "Mighty Write" (C•Notes, October 1983, p. 266) the N option to print the next page of text only works the first time you use it. This creates a problem when text is longer than two pages.

To correct the problem, replace lines 200, 230, and 340 with:

```
200 LC = LC + 1:IF LC = 57 THEN 320
230 IF LC<57 THEN 140 ELSE GOTO 320
340 FORQ = 1 TO 50:NEXT Q:CLS:LC = 1
```

—Eds.

See List of Advertisers on Page 211

POWERMAIL PLUS
By Powersoft

Retail 150.00
Special 112.50

SAVE 25%

Features Over the Original PowerMAIL

Separates any category for you in any manner you wish. This new version can span drives and may use multiple floppies. The program will sort on any eight levels if you wish. Separates your flags and puts them into another file. Print Options: Labels & Listing. One choice is the simple telephone book. Type/listing. Two of the other modes give you not only the listing of names, but also either FLAG numbers or the actual FLAG descriptions. The final print mode is what we call THE SALESMAN'S NOTEBOOK™. It prints out the whole list or whatever parts you specify with all info including full flag descriptions in each record. Then the word 'Notes' is printed with several lines of blank space for making those notes in the field.

MOD I/III/4 2/12/16
SPECIFY

CP/M 2.2

TRS-80 MODEL 4
Only 199.95

Now, run CP/M on your Model 4 and unleash the powerful features resident in your computer.

● Includes INTERCHANGE™, a utility that allows reading, writing and copying 20 different manufacturers' disk formats such as IBM, KAYPRO, OSBORNE, XEROX, etc. ● Includes MEMLINK™, a unique feature that uses the optional 64K RAM memory as a fast disk drive. ● Complete with all these CP/M utilities: ASM, DDT, DUMP, ED, LOAD, PIP, STAT and SYSGEN. ● Includes MODEM 7, a powerful public domain communications program. ● Supports 80 x 24 video, reverse video, direct cursor addressing and more. ● Utilizes the Model 4 function keys and allows user defined keys. ● FORMAT utility permits up to 52 disk formats to be constructed at menu driven. ● Ready to run in the standard 64K Model 4. The additional extra cost 64K RAM upgrade not required.

MISOSYS

Software for the Model 4

PRO-CESS	37.95
PRO-CREATE	94.95
PRO-CURE	46.95
PRO-DUCE	37.95
PRO-LC	139.95

SEE MISOSYS AD FOR DETAILS

ELECTRIC WEBSTER
by Cornucopia Software

149.95 SAVE 20% WITH CORRECTING FEATURE

ONLY 119.95

SYSTEM DIAGNOSTIC
By Howe Software

ONLY $99.95

Complete diagnostic tests for components of your TRS-80. Separate tests for: ● ROM ● RAM ● VIDEO DISPLAY ● KEYBOARD ● LINE PRINTER ● CASSETTE RECORDER ● RS-232-C INTERFACE ● DISK DRIVES ● DISK CONTROLLER ● INDIVIDUAL TESTS AND CONTINUOUS SYSTEM TESTS.

MOD I or MOD III

6.0 PLUS
by MicroSystems Software

Enhance Your Model 4 TRSDOS 6.0 and Disk Basic

39.95

A great value from the DOSPLUS folks, these utilities for your MOD 4 DOS add some really important features. Enhancements include:

DISKZAP	MAP	REF
DISKDUMP	RESTORE	RESOLVE
DIRCHECK	SORT	BE1
	SR	RF2
	MOD 4	

EYE EASE GREEN CRT SCREEN FOR
TRS-80 MODEL III

SAVE OVER 30%
MOD I, II, III/4 NOW 16.95

$24.95
GREEN WINDOW

PREFERRED BY COMPUTER PROFESSIONALS EVERYWHERE

OPTICOM

ENIGMA
by Powersoft

ONLY 39.95

This encryption program is based on the ENIGMA cypher machine used by the Germans in the Second World War. It incorporates the improvements which by the admission of Allied Intelligence who participated in cracking the Enigma code, would have made the system almost completely impenetrable. A 29-character key is used for encrypting textual material yielding over TRILLIONS of possible keys and encoding combinations. A technical description of the German ENIGMA machine and how it is simulated (and improved upon) is given in the manual. The perfect program for students of cryptography, business persons who want to transmit sensitive data over open data links (voice or modem) or TELEX lines, and anyone else who wishes to maintain confidential information.

MOD I/III

PowerDRAW
by Powersoft

Retail 39.95
Special . . . 34.95

SAVE OVER 10%

A full screen graphics editor, POWERDRAW is 100% assembly language. You can create screens of graphics, save to disk, merge them, run in sequence like a movie, merge text with graphics, and write your own game or business application screens! PowerDRAW saves graphics to disk or tape so that they can be recalled at a later time in the following formats:

0) CONDENSED TOKENS
1) EDAS Source file format
2) EDTASM Source file format
3) BASIC data statements
4) BASIC Strings
5) BASIC arrays (Apex 4 version 1.3)
6) Load File Format

THE TOOLBELT MOD 4
by Powersoft

Powersoft does it again with a collection of 16 of the most popular LDOS TOOLBOX utilities re-published for TRSDOS 6.0 running on the MODEL 4. Now enhance your MOD 4 TRSDOS with this powerful set of utilities. Complete with detailed manual on how to use each utility program to do the many things the TOOLBOX for LDOS was famous for.

49.95

Model 4

GRAPHIT
by Powersoft

ONLY 39.95

A simple, easy-to-use program for portraying graphs on a microcomputer. Creates and uses files in Visicalc DIF format, so it will be able to graph Visicalc data as well! Graphs may be changed at the touch of a key; graph scales can be compressed or expanded. Display and print just one portion of the graph or all of it at once. Extremely user-friendly and contains multiple HELP screens. GRAPHIT will print the raw data and produce graphs on any lineprinter. No special lineprinter options required.

MOD I/III

EPSON™ Driver Compiler
by Powersoft

29.95

This is a program which allows you to construct your own printer character table for an EPSON bit-graphics printer, and compile a relocating printer driver which will produce those characters. Character tables can be saved to disk and manipulated in a variety of ways (including rotation of each character 90, 180 or 270 degrees) before the actual compile process. Several example files are included on the disk. This program requires an EPSON MX-80 with GRAFTRAX or GRAFTRAX-Plus, or an FX-80.

Mod I/III

BBS-80
by MicroSystems Software

ONLY 74.95

Your computer becomes a mini-communications network. BBS-80 allows a person to exchange messages or information via the telephone. Access can be completely public, private to authorized users or private to certain individuals.

BBS-80 can leave messages on another (remote) acting as a 'dead' machine. Further:

BBS-80 provides some of the most advanced and most convenient features ever incorporated in a microcomputer bulletin board system including:

Users may send private messages that may be read only by the addressee. Multi-level System security allows up to 15 levels of security in both the message base and the upload/download database.

BBS-80 remembers each regular user and stores personal information, security access level, last message reviewed during last call an individual user password.

Users may upload (or transmit) data and programs (text) to BBS-80. These programs may then be 'downloaded' to other BBS-80 users.

BBS-80 is self maintaining. Message space is automatically reclaimed when messages are deleted.

MOD I or III

IMPAKT
by Powersoft

39.95

An outstanding value for Basic Programmers. IMPAKT is a relocatable machine language module that requires no modification to basic but will do the following: search & replace ● remove remarks ● remove spaces ● remove colons ● remove gotos following then ● compress or decompress ● renumber ● trace by line # or step without destroying screen ● set break points in trace mode ● join lines ● copy lines with auto return ● move blocks and check for validity ● display memory useage stats ● decode packed string graphics ● conv upper case to lower case strings ● auto save & load from disk and many more.

MOD I/III

MORE DISKCOUNTS

↙ 204

PROGRAM NAME	SPECIAL
BASIC S COMPILER SYSTEM I/III	39.95
MICROTERM I, III or 4	69.95
SCRIPLUS 3.0 I/III	29.95
DOSPLUS II MOD II/12 or Z-80 MOD 16	199.95
M-ZAL EDITOR ASSEMBLER RELEASE 3 MOD I or III	84.95
ZUES EDITOR ASSEMBLER MOD I/III/4	74.95
MASTER MECHANICS SET FOR LDOS (9 UTILITIES) I/III	34.99
LOAN AMORTIZATION (FULL AMORT SCHEDULE) III	29.95
E-BASIC FOR MULTIDOS MOD I OR III	49.95
SUPER UTILITY TECH MANUAL 3.(X)	11.95
INSIDER SUPER UTILITY 3.(X)	16.95
INSIDE SUPER UTILITY 2.2	14.95
THE CUSTOM TRS-80 & OTHER MYSTERIES	26.95
HOW TO DO IT ON THE TRS-80 & OTHER MYSTERIES	26.95
TRSDOS 2.3 DECODED & OTHER MYSTERIES	26.95
MICROSOFT BASIC DECODED & OTHER MYSTERIES	26.95
BASIC FASTER & BETTER & OTHER MYSTERIES	26.95
MACHINE LANGUAGE DISK I/O & OTHER MYSTERIES	26.95
BASIC DISK I/O & OTHER MYSTERIES	26.95
TRS-80 DISK AND OTHER MYSTERIES	19.95
DISKETTES 5¼ IN. SSDD, IN PLASTIC EASLE BOX	24.95
LDOS 5.1 MOD I OR III	119.95
PROFILE 3+ COMMANDS (17x22 WALL CHART)	*4.00
VISICALC COMMANDS (17x22 WALL CHART)	*4.00
SUPERSCRIPSIT COMMANDS (17x22 WALL CHART)	*4.00
MODEL III BASIC COMMANDS (17x22 WALL CHART)	*4.00

*NOT SHIPPED SEPARATELY

FREE
OFFER

Any order of over $100 or more from this ad will receive one choice of the following absolutely FREE:

Green Window I/II/III/4,
SU+3.(X) Tech Manual
Macro-Mon The Shadow

Any order over $200 will receive one choice of the following FREE:
Multidos MOD I 1.5, MOD III 1.3 or GAMEPAK 3

DiskCount Data™
214-680-8268

Send Cash, Check or Money Order. Please add $3.00 for postage and handling, additional $2.00 for C.O.D.'s

●

Foreign orders welcome, please specify air or surface. All shipping charges assumed by purchaser.

●

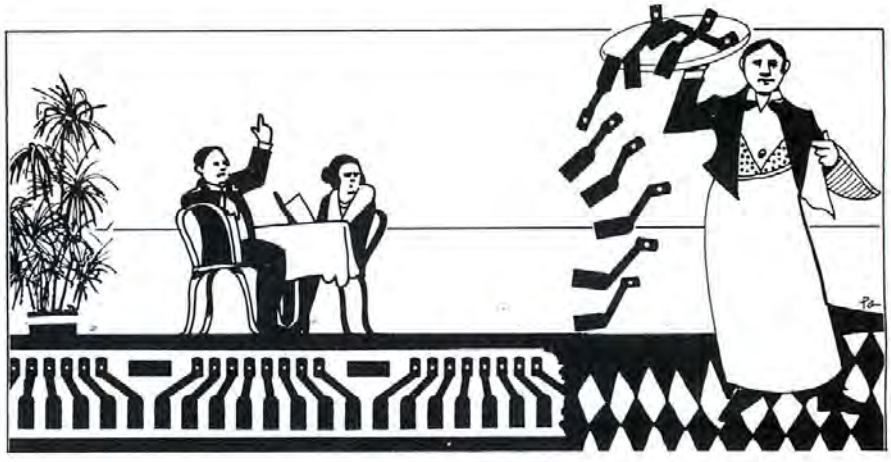
When ordering by mail, please specify computer model number (I, II, III or 4), drive configuration, and memory size.

Phone Your Order In Today Or Mail To:
DISKCOUNT DATA
2701-C W. 15th St., Suite 324
Plano, TX 75075
Office Hours Mon-Fri 10A.M. to 9P.M. C.S.T.

MasterCard AND VISA Cheerfully Accepted

↙ See List of Advertisers on Page 211

Speed Up Calculation Time With a Math Processor Board



For most applications, the TRS-80 is acceptably fast in its mathematical calculations. This month's column is for those applications where greater processing speed is necessary or desirable. I'll describe the construction and operation of a mathematical slave processor for 48K RAM Models I and III.

Adding a slave processor can decrease calculation times by well over 90 percent. You can use the slave processor with Basic programs as described below. But you may find it even more helpful in Assembly-language programs, since you can access most mathematical functions without having to write software routines which, of course, are slower than those for hardware.

Meet the Intel 8231A and 8232

Intel offers two pin-compatible mathematical slave processors. The 8231A, their arithmetic processing unit, is capable of 16-bit and 32-bit integer calculations as well as 32-bit floating-point calculations. It performs the four basic math functions as well as most trigonometric functions. Throughout this column, my examples will refer to the 8231A.

The 8232, Intel's floating-point processing unit, operates on 32- or 64-bit floating-point numbers, conforms to the proposed IEEE (Institute of Electrical and Electronics Engineers) floating-point format standard, and is limited to the four basic functions.

Integers

Integers represent the most basic data format type and are usually stored as "two's complement" values, where the binary number ranges in value from -2^{N-1} to $+2^{N-1}-1$ (N is the number of bits in the binary number). An 8-bit integer, for example, would have the range -128 to $+127$. In two's complement format, the high-order bit determines whether the value is positive (bit is zero) or negative (bit is 1). The value zero is represented by all zeros and thus is considered positive.

To determine the decimal value of a positive two's complement number, calculate the value in the normal manner, multiplying each bit value by the corresponding column weight and adding the results. For example, you can find the number 00101011 by calculating right to left: $1*2^0 + 1*2^1 + 0*2^2$ to get 43.

If the number is negative, first complement all of the bits, then add one. This gives the two's complement of the negative number, which is its absolute value. Now find the value of the new number as described above; this tells you of what the original number is a negative.

For example, the number 11101001 is a negative two's complement value. First complement all of the bits to get 00010110, then add one to get 00010111. This value can be found to be 23 using the method described above; therefore, the original value, 11101001, is -23. Be careful to note that the most negative number (for example, -128 for an 8-bit binary integer) is always its own two's complement.

Binary Floating-Point Numbers

You represent binary floating-point numbers in much the same manner as decimal numbers in scientific notation. Numbers in scientific notation are in the form $X.YYYY*10^{ZZ}$, where X is a digit from one to nine, inclusive. The Y digits represent further significant digits of the number, and the number of Y digits determines the number's accuracy or precision. The Z digits represent the exponent that determines the dynamic range of the number, that is, its magnitude range.

Notice how the number has only one non-zero digit to the left of the radix (decimal point). This is called "normalization." The number $0.00234*10^{-4}$ is not in proper scientific notation since it isn't normalized. The proper representation is $2.34*10^{-7}$. The only exception to normalization is the value zero.

The number is separated into a mantissa part and an exponent part. The binary mantissa is always normalized. The three formats I'm discussing have this much in common. But from here on, the actual normalization format varies, as do the form of the exponent and the placement of the mantissa sign bit.

A typical binary floating point number format is $0.1XXX*2^{ZZ}$, where 1XXX represents the mantissa portion, and ZZ represents the exponent portion. Both the X and Z bits are either zero or one values. Note that binary floating-point numbers are often normalized to the right of the radix (binary point).

TRS-80 Floating-Point

The TRS-80 floating-point format for a single-precision value requires 4 bytes of memory. The high-order byte is the exponent portion of the number, with the remaining 3 bytes making up the mantissa portion. The exponent byte is in excess 128 format. That is, the exponent byte is an absolute binary (*not* two's complement) byte, which is 128 greater than the actual exponent value it represents. Since an 8-bit absolute binary value can range from zero (not used as an exponent value) to 255, the actual exponent value ranges from -127 to $+127$.

The mantissa consists of the 3 low-order bytes of the 4-byte block, the low-order byte of the block being the low-order byte of the mantissa. Since the mantissa portion of the number is normalized to the right of the radix, a decimal point is assumed to be above the high-order mantissa bit. Since the mantissa is normalized, the high-order bit must be a one for any non-zero value. Because of this, the one is omitted and simply assumed. The high-order bit of the mantissa, instead of holding the 1 bit, holds the mantissa sign bit. This bit determines if the mantissa is positive (bit 0) or negative (bit 1).

The TRS-80 single-precision floating-point format as 24 bits of precision, and an 8-bit exponent range of 2^{-127} to 2^{+127}. The actual decimal range calculates out to approximately $\pm 1.701411*10^{-38}$ to $\pm 1.701411*10^{+38}$. The value zero is represented by a zero exponent byte, without regard to the bits in the mantissa portion of the number. Figure 1 shows the floating-point format the TRS-80 uses.

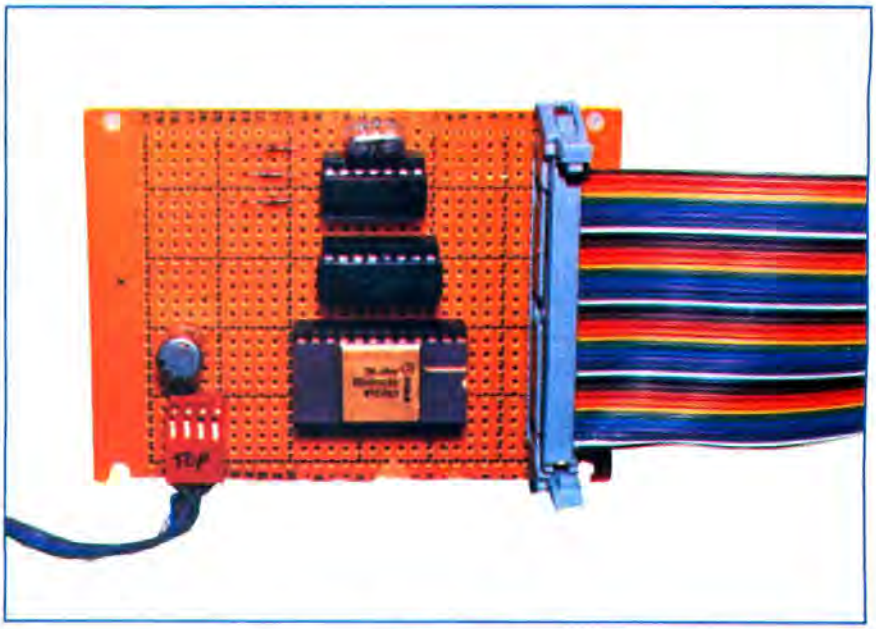


Photo. Math processor board.

Intel 8231A Floating-Point

Like the TRS-80, the 8231A uses 4 bytes to represent its value. Again, the high-order byte is the exponent byte, with the remaining 3 bytes making up the mantissa. As in the TRS-80, the mantissa is normalized to the right of the radix. Unlike the TRS-80, the high-order 1 bit must be present, even though it is a necessary part of any normalized, non-zero number and can be assumed.

Since the high-order bit must be a 1, it cannot be used as a sign bit, as is the case with the TRS-80 format. The mantissa sign bit, it turns out, is in the exponent byte. Since you move the sign bit, you still have 24 bits of mantissa precision, the same as the TRS-80 format.

The 8231A exponent portion is more different from the TRS-80 format than is the mantissa portion. The high-order bit of the exponent byte is the sign bit for the mantissa; this leaves 7 exponent bits to work with. These remaining exponent bits represent the exponent value in two's complement format. Since there are 7 bits, the exponent value can range from -64 to $+63$. This is one bit less than the TRS-80 format, giving the 8231A about half the dynamic range of the TRS-80, but the same accuracy (precision).

The value zero is represented by all 32 bits being zero. The decimal numeric range of the 8231A is approximately $\pm 2.7*10^{-20}$ to $\pm 9.2*10^{+18}$—significantly smaller than that for the TRS-80, although sufficient for most applications. Figure 2 shows the format the 8231A uses.

8232 Floating-Point

The single-precision format of the 8232 comes the closest to the TRS-80 single-precision format, and provides the same precision (24 bits) and roughly the same dynamic range.

The 8232 format may seem some-

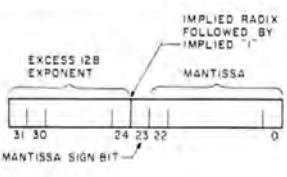


Figure 1. TRS-80's floating-point format.

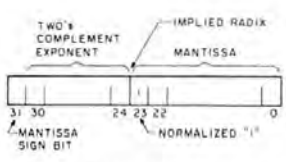


Figure 2. Intel 8231A's floating-point format.

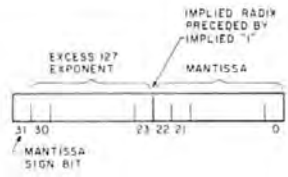


Figure 3. Intel 8232's single-precision format.

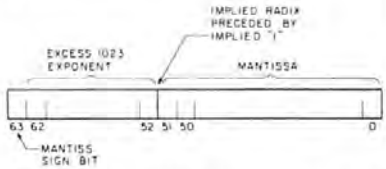


Figure 4. The 8232's double-precision format.

what awkward, since the mantissa and the exponent are not completely separated on a byte boundary. Like the TRS-80 format, the 8232 has an 8-bit exponent. The high-order byte of the 4-byte single-precision block contains most of the exponent portion; however, the low-order bit of the exponent is in the high-order bit position of the next-lower-order byte. The high-order bit of the high-order byte contains the mantissa sign bit.

The 8232 exponent byte is in excess 127 form, similar to the excess 128 form used in the TRS-80. Like the TRS-80 format, a zero exponent value represents the numeric value of zero, regardless of the value of the mantissa. In the 8232, the exponent value with all bits set to 1 is an invalid condition. The exponent can range from −126 to +127.

The mantissa portion of the 8232 single-precision format consists of the 23 bits of lower order than the exponent bits. The radix is assumed to be between the exponent and the mantissa bits. An assumed 1 bit is to the left of the radix point (instead of the right as in the previous two formats discussed). The decimal numeric range for the 8232's single-precision format is approximately $\pm 1.17*10^{-38}$ to $\pm 3.40*10^{+38}$ (see Fig. 3).

The 8232 double-precision format is a logical extension of its single-precision format, moving from 32 bits to 64 bits (8 bytes). The high-order bit again is the mantissa sign bit. The exponent consists of bits 52 through 62 (portions of the high-order 2 bytes) in excess 1023 form. Again, all zero bits in the exponent represent the numeric value zero, and all 1's represent an invalid condition. There are 52 bits of precision. The decimal numeric range of the 8232's double-precision format is approximately $\pm 2.22*10^{-308}$ to $\pm 1.80*10^{+308}$ (see Fig. 4).

Note that the 8232's and the TRS-80's double-precision formats differ greatly. The 8232 increases both its precision and its dynamic range from its single-precision format, whereas the TRS-80 increases only its precision (from 24 bits to 56 bits), leaving its dynamic range the same.

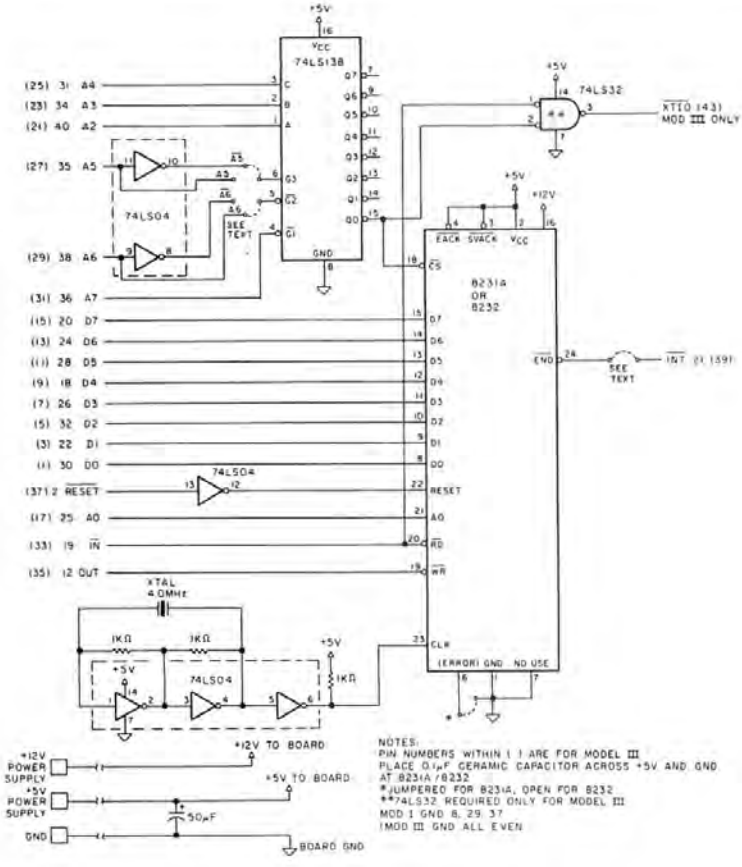


Figure 5. Schematic for math slave processor.

Quantity	Description	Manufacturer	Part Number	Price
1	74LS138 chip	DIGI-KEY		$ .66
1	74LS04 chip	DIGI-KEY		.41
1	8231A or 8232	Washtenaw		175.00
*1	math processor	Digital Systems		
1	74LS32 chip	DIGI-KEY		.43
3	4.0MHz crystal	DIGI-KEY		2.95
3	1 kΩ resistors (pkg. of 5)	DIGI-KEY		.50
1	0.1 μF capacitor	Radio Shack	272-135	.49
1	100 μF capacitor (pkg. of 2)	Radio Shack	272-1028	.79

*Model III only

See end of article for manufacturers' and suppliers' addresses.

Table 1. Parts list and ordering information.

Format Conversions

To convert from the TRS-80 single-precision format to the 8231A format, the computer follows these steps:

1. Checks the high-order TRS-80 byte for zero value; if zero, clears all four 8231A bytes, then goes to Step 7.
2. Moves the 3 low-order TRS-80 bytes to the 3 low-order bytes of the 8231A storage area.
3. Sets the high-order bit of the third byte (the high-order mantissa bit).
4. Subtracts 128 from the high-order TRS-80 byte.
5. Sees if the high-order 2 bits are the same; if not, goes to Step 8; otherwise, saves byte in 8231A high-order byte position.
6. Alters the high-order bit of the high-order 8231A byte as necessary to be the same as the high-order TRS-80 mantissa bit (the mantissa sign bit).
7. Ends.
8. Signifies error. Ends.

THE ULTIMATE COMPUTER. THE NEW

LNW80 2 +

NOW:

MODEL 1, 3, and 4 COMPATIBLE!!
CP/M 2.2 and 3.0 COMPATIBLE!!

SOON:

IBM MS-DOS and CP/M-86 COMPATIBILITY!!
PLUS add up to 1 Megabyte of RAM!!
High Speed RAMDISK is coming!

SYSTEMS START AT $1999

Includes: LNW80-2+, Hi-Res Green Monitor, One Disk Drive, CP/M 2.2, DOSPLUS, Microterm, Electric Pencil, Electric Spreadsheet, Chartex, and the LNW Small Business and Professional Accounting Software Series.

CALL US FOR THE BEST PRICES AVAILABLE
IN STOCK NOW! Ready for immediate delivery!

EXCELLONIX

(714) 973-1939 ** (213) 650-5754 ↙267

EE/EPROM PROGRAMMERS & UV ERASERS

AFFORDABLE ★ RELIABLE ★ AVAILABLE

UV ERASERS
QUV-T8 1
$49.95
*HOBBY

QUV-T8 2N
$68.95
*INDUSTRIAL

QUV-T8 2T
$97.50
*WITH TIMER & SAFETY SWITCH

GANGPRO-8™
$995.00
(GANG PROGRAMMER)

RS-232 serial, STAND ALONE, INTELLIGENT

*EASY DUPLICATION *USER FRIENDLY *128K BUFFER

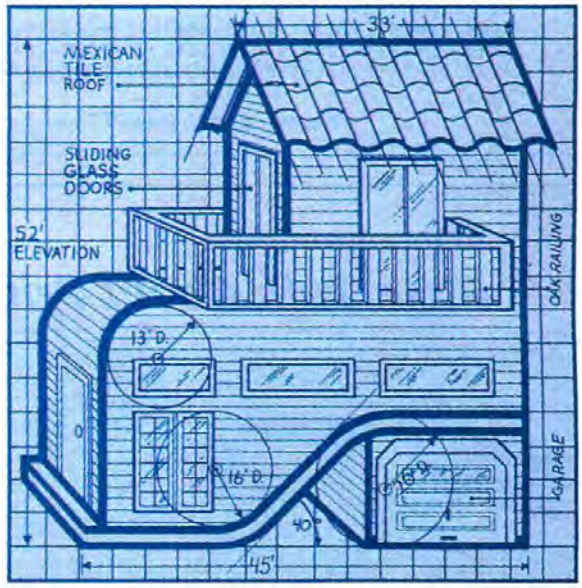
SUPPORTS MOST 8K, 16K, 32K, 64K, 128K, 256K EPROMS
PROMPRO-8 KEY PAD OPTION. EPROM SIMULATION MODE
Microcomputer Chips: 8748 (H), 8749H, 8750, 8751, 8741, 8742, 8755A
SOFTWARE DRIVERS: MDS ISIS, TEKTRONICS 8002, IBM PC, ATARI, APPLE II, CPM, FLEX, TRS-80
DIRECT HOOK UP TO ANY DUMB TERMINAL OR COMPUTER

DISTRIBUTOR INQUIRY WELCOME.

LOGICAL DEVICES INC.

1321E N.W. 65 Place, Ft. Lauderdale, FL 33309
Phone Orders (305) 974-0967 TWX 510-955-9496 ↙330

WE COULD SHOW THE OLD WOMAN WHAT TO DO WITH THE SHOE.



INVESTOR III™ makes rhyme and reason out of real estate investing.

Now you can maximize your real estate profits and avoid questionable ventures with the help of Investor III™.

Investor III™ was conceived and developed by a real estate company. So it's practical and comprehensive. Ready to meet the needs of the average investor or professional user with capabilities like pro-forma, "what-if" analysis, cash flow before and after taxes, depreciation and more.

Yet Investor III™ is as simple as Mother Goose — you can learn to use it in less than one hour.

So whether you're looking at a multi-million-dollar high rise or a cozy rent house with some unique qualities, Investor III™ can help you determine the wisdom — and profitability — of your choice.

Write or call for more details today. Investor III™...for investments that fit. And that's no fairy tale.

Good Software is proud to announce an easy-to-use amortization package — the Amortizer III™.

GOOD SOFTWARE CORP. ↙496

A Division of The Goodman Group, Inc.
12900 Preston Rd., Dallas, Texas 75230 (214) 239-6085
Compatible with Radio Shack TRS-80 and IBM PC.
Investor III™: $249.00 plus $5 shipping and handling.
Amortizer III™: $49.95 plus $5 shipping and handling.
American Express, MasterCard and Visa accepted.
Dealer inquiries invited.

Making dollars and sense out of information.

↙ See List of Advertisers on Page 211

```
20 DEFUSR0=&HF806
30 A=.2345:B=7:C=0
40 V=VARPTR(A):GOSUB1000
50 V=VARPTR(B):GOSUB1010
60 V=VARPTR(C):GOSUB1020
65 FORI=1TO5000
70 X=USR(12)
75 NEXTI
80 PRINTC:END
1000 V1=INT(V/256):POKE-2047,V1:POKE-2048,V-V1*256:RETURN
1010 V1=INT(V/256):POKE-2045,V1:POKE-2046,V-V1*256:RETURN
1020 V1=INT(V/256):POKE-2043,V1:POKE-2044,V-V1*256:RETURN
```

LOAD 80

Program Listing 1. VARPTR user function.

The only error that can occur is if the exponent value of the TRS-80 number is outside the 8231A's exponent range, which can occur since the TRS-80 uses an 8-bit exponent (the 8231A uses a 7-bit exponent). I implement this algorithm in Z80 Assembly-language code in two of my program listings.

The reverse algorithm to convert from the 8231A floating-point format to the TRS-80 single-precision floating-point format requires the computer to:

1. Check for error code from 8231A. If error, go to Step 8.
2. Check for zero flag bit set in the 8231A. If set, zero the high-order TRS-80 byte, pop the value from the 8231A stack, then go to Step 7.
3. Move all four 8231A bytes into TRS-80 storage locations.
4. Change the high-order mantissa bit (the high-order bit of the third byte) to the same value as the high-order bit of the high-order byte (this is the mantissa sign bit).
5. Make the high-order bit of the exponent (high-order) byte the same value as the next-lower-order bit.
6. Add 128 to the exponent byte.
7. End.
8. Signify error. End.

Constructing the Board

Now that you understand the relevant floating-point formats, you can construct the math processor board. The board design is actually very simple (see Photo); it requires only three chips (four for the Model III), a crystal, three resistors, and two capacitors (see Table 1). You need a +5V (at 100 mA) and +12V (at 95 mA) power supply. Place a 50-microfarad, 10-volt electrolytic capacitor across the +5V power input to supply short-term surge requirements, and an additional 0.1-microfarad ceramic capacitor across the 8231A/8232 +5V line for decoupling (not shown in the Photo).

Figure 5 shows pin numbers for both the Model I and the Model III (in parentheses). The Model III requires an extra circuit, since it must change the direction of an internal bidirectional buffer during every external I/O read. You need the 74LS32 shown in the Photo for the Model III. Of the several jumpers shown on the schematic (Fig. 5), two at the input to the

Continued on p. 207

Program Listing 2. Math processing routine.

```
                00100 ;*************************************************
                00110 ; MATH PROCESSING ROUTINE FOR INTEL 8231A MATH     *
                00120 ; PROCESSOR.                                       *
                00130 ;                                                  *
                00140 ;       Created by  Roger C. Alford                *
                00150 ;            October, 1983                         *
                00160 ;                                                  *
                00170 ; This program is entered via the Basic USR        *
                00180 ; function. The code specified during the user call *
                00190 ; determines which math function is to be executed. *
                00200 ; Currently, only floating point operations are    *
                00210 ; supported. The functions available, and their    *
                00220 ; corresponding code, are shown here:              *
                00230 ;                                                  *
                00240 ;      0 = ADDITION                                *
                00250 ;      1 = MULTIPLICATION                          *
                00260 ;      2 = INVERSE COSINE                          *
                00270 ;      3 = INVERSE SINE                            *
                00280 ;      4 = INVERSE TANGENT                         *
                00290 ;      5 = SIGN CHANGE                             *
                00300 ;      6 = COSINE                                  *
                00310 ;      7 = EXPONENTIAL                             *
                00320 ;      8 = DIVISION                                *
                00330 ;      9 = SUBTRACTION                             *
                00340 ;     10 = COMMON LOGARITHM                        *
                00350 ;     11 = NATURAL LOGARITHM                       *
                00360 ;     12 = POWER FUNCTION                          *
                00370 ;     13 = SINE                                    *
                00380 ;     14 = SQUARE ROOT                             *
                00390 ;     15 = TANGENT                                 *
                00400 ;                                                  *
                00410 ;*************************************************
                00420 ;
                00430 ;
F800            00440           ORG     0F800H
                00450 ;
0001            00460 COMAND    EQU     01H     ;8231A COMMAND PORT ADDRESS
0001            00470 STATUS    EQU     01H     ;8231A STATUS PORT ADDRESS
0000            00480 DATA      EQU     00H     ;8231A DATA PORT STACK ADDRESS
0A7F            00490 FNCTN     EQU     0A7FH   ;TRS-80 'GET USR PARAM.' ROUTINE
0010            00500 FADD      EQU     10H     ;8231A ADD COMMAND BYTE
0012            00510 FMUL      EQU     12H     ;8231A  MULTIPLY COMMAND BYTE
0006            00520 ACOS      EQU     06H     ;8231A INVERSE COSINE CMD BYTE
0005            00530 ASIN      EQU     05H     ;8231A INVERSE SINE CMD BYTE
0007            00540 ATAN      EQU     07H     ;8231A INVERSE TANGENT CMD BYTE
0015            00550 CHSF      EQU     15H     ;8231A SIGN CHANGE CMD BYTE
0003            00560 COS       EQU     03H     ;8231A COSINE CMD BYTE
000A            00570 EXP       EQU     0AH     ;8231A EXPONENTIAL CMD BYTE
0013            00580 FDIV      EQU     13H     ;8231A DIVIDE CMD BYTE
0011            00590 FSUB      EQU     11H     ;8231A SUBTRACT CMD BYTE
0008            00600 LOG       EQU     08H     ;8231A COMMON LOG CMD BYTE
0009            00610 LN        EQU     09H     ;8231A NATURAL LOG CMD BYTE
000B            00620 PWR       EQU     0BH     ;8231A POWER CMD BYTE
0002            00630 SIN       EQU     02H     ;8231A SINE CMD BYTE
0001            00640 SQRT      EQU     01H     ;8231A SQUARE ROOT CMD BYTE
0004            00650 TAN       EQU     04H     ;8231A TANGENT CMD BYTE
0018            00660 POPF      EQU     18H     ;8231A STACK POP COMMAND
033A            00670 PRCHAR    EQU     033AH   ;TRS-80 'PRINT CHAR.' ROUTINE
                00680 ;
0002            00690 PARAM1    DEFS    2       ;FIRST MATH PARAMETER ADDRESS
0002            00700 PARAM2    DEFS    2       ;SECOND MATH PARAMETER ADDRESS
0002            00710 DEST      DEFS    2       ;DESTINATION VARIABLE ADDRESS
                00720 ;
F806 F3         00730 MATH      DI                      ;INTERRUPTS ARE DISABLED
F807 CD7F0A     00740           CALL    FNCTN           ;GET MATH FUNCTION VALUE
F80A 7C         00750           LD      A,H             ;GET HIGH BYTE OF VALUE
F80B B7         00760           OR      A               ;BYTE SHOULD BE ZERO
F80C C2FFF8     00770           JP      NZ,ERROR1       ;IF NOT, ERROR
F80F 7D         00780           LD      A,L             ;GET LOW BYTE OF VALUE
F810 FE10       00790           CP      16              ;CHECK FOR VALID VALUE
F812 D2FFF8     00800           JP      NC,ERROR1       ;IF NOT, ERROR
F815 CB25       00810           SLA     L               ;MULTIPLY VALUE BY TWO
F817 116CF9     00820           LD      DE,FNCTBL       ;GET FUNCTION TABLE ADDR.
F81A 19         00830           ADD     HL,DE           ;POINT TO FNCTN ADDRESS
```

Listing 2 continued

↙ See List of Advertisers on Page 211

Listing 2 continued

```
F81B 5E        00840          LD      E,(HL)          ;GET LOW ADDRESS BYTE
F81C 23        00850          INC     HL              ;POINT TO HIGH BYTE
F81D 56        00860          LD      D,(HL)          ;GET HIGH ADDRESS BYTE
F81E EB        00870          EX      DE,HL           ;PUT ADDRESS INTO HL
F81F E9        00880          JP      (HL)            ;EXECUTE SPECIFIC FNCTN
               00890 ;
               00900 ;*
               00910 ;*** SUBROUTINE CONVRT:
               00920 ;* CONVRT gets the Radio Shack floating point value
               00930 ;* pointed at by the IX register and converts it to
               00940 ;* the Intel 8231A format, and pushes it onto the 8231A
               00950 ;* stack.
F820 DD7E03    00960 CONVRT   LD      A,(IX+3)        ;GET EXPONENT BYTE
F823 B7        00970          OR      A               ;CHECK FOR ZERO VALUE
F824 2009      00980          JR      NZ,CONV2        ;IF NOT, GO ON
F826 D300      00990          OUT     (DATA),A        ;ELSE, PUSH 00 BYTE
F828 D300      01000          OUT     (DATA),A        ;PUSH 00 BYTE
F82A D300      01010          OUT     (DATA),A        ;PUSH 00 BYTE
F82C D300      01020          OUT     (DATA),A        ;PUSH 00 BYTE
F82E C9        01030          RET                     ;DONE - RETURN
F82F DD7E00    01040 CONV2    LD      A,(IX)          ;GET LOW ORDER BYTE
F832 D300      01050          OUT     (DATA),A        ;PUT INTO 8231A
F834 DD7E01    01060          LD      A,(IX+1)        ;GET NEXT ORDER BYTE
F837 D300      01070          OUT     (DATA),A        ;PUT INTO 8231A
F839 DD7E02    01080          LD      A,(IX+2)        ;GET HIGH MANTISSA BYTE
F83C F680      01090          OR      80H             ;SET HIGH BIT
F83E D300      01100          OUT     (DATA),A        ;PUT INTO 8231A
F840 DD7E03    01110          LD      A,(IX+3)        ;GET EXPONENT BYTE
F843 D680      01120          SUB     128             ;PUT INTO REAL FORMAT
F845 4F        01130          LD      C,A             ;SAVE EXPONENT IN C REG
F846 07        01140          RLCA                    ;MOVE HIGH BIT INTO CARRY
F847 0600      01150          LD      B,00H           ;CLEAR THE B REGISTER
F849 CB18      01160          RR      B               ;PUT CARRY BIT INTO B REG
F84B E680      01170          AND     80H             ;MASK LOW 7 ACC. BITS
F84D A8        01180          XOR     B               ;CHECK FOR SAME BIT 7'S
F84E C204F9    01190          JP      NZ,ERROR2       ;IF NOT, OUT OF RANGE ERR
F851 79        01200          LD      A,C             ;ELSE, GET EXPONENT BACK
F852 E67F      01210          AND     7FH             ;CLEAR HIGH BIT
F854 DDCB027E  01220          BIT     7,(IX+2)        ;CHECK MANTISSA SIGN BIT
F858 2802      01230          JR      Z,NOCHG         ;IF POSITIVE, NO CHANGE
F85A CBFF      01240          SET     7,A             ;ELSE SET BIT FOR NEGATIV
F85C D300      01250 NOCHG    OUT     (DATA),A        ;PUT EXP. BYTE INTO 8231A
F85E C9        01260          RET                     ;DONE - RETURN
               01270 ;*
               01280 ;*** ROUTINE ADD:
               01290 ;* This routine causes the 8231A to add the two
               01300 ;* provided numbers.
F85F 3E10      01310 ADD      LD      A,FADD          ;GET FLOATING ADD CMD BYT
F861 182E      01320          JR      TWOVAL          ;TWO-VALUED OPERATION
F863 3E12      01330 MULT     LD      A,FMUL          ;GET MULTIPLY COMMAND BYT
F865 182A      01340          JR      TWOVAL          ;TWO-VALUED OPERATION
F867 3E06      01350 INVCOS   LD      A,ACOS          ;GET INV COSINE CMD BYT
F869 1843      01360          JR      ONEVAL          ;ONE-VALUED OPERATION
F86B 3E05      01370 INVSIN   LD      A,ASIN          ;GET INV SINE CMD BYT
F86D 183F      01380          JR      ONEVAL          ;ONE-VALUED OPERATION
F86F 3E07      01390 INVTAN   LD      A,ATAN          ;GET INV TANGENT CMD BYTE
F871 183B      01400          JR      ONEVAL          ;ONE-VALUED OPERATION
F873 3E15      01410 CHGSGN   LD      A,CHSF          ;GET CHANGE SIGN CMD BYTE
F875 1837      01420          JR      ONEVAL          ;ONE-VALUED OPERATION
F877 3E03      01430 COSINE   LD      A,COS           ;GET COSINE COMMAND BYTE
F879 1833      01440          JR      ONEVAL          ;ONE-VALUED OPERATION
F87B 3E0A      01450 EXPON    LD      A,EXP           ;GET EXPONENTIAL CMD BYTE
F87D 182F      01460          JR      ONEVAL          ;ONE-VALUED OPERATION
F87F 3E13      01470 DIV      LD      A,FDIV          ;GET DIVIDE CMD BYTE
F881 180E      01480          JR      TWOVAL          ;TWO-VALUED OPERATION
F883 3E11      01490 SUB      LD      A,FSUB          ;GET SUBTRACT CMD BYTE
F885 180A      01500          JR      TWOVAL          ;TWO-VALUED OPERATION
F887 3E08      01510 COMLOG   LD      A,LOG           ;GET COMM LOG CMD BYTE
F889 1823      01520          JR      ONEVAL          ;ONE-VALUED OPERATION
F88B 3E09      01530 NATLOG   LD      A,LN            ;GET NAT LOG CMD BYTE
F88D 181F      01540          JR      ONEVAL          ;ONE-VALUED OPERATION
F88F 3E0B      01550 POWER    LD      A,PWR           ;GET POWER COMMAND BYTE
F891 08        01560 TWOVAL   EX      AF,AF'          ;SAVE COMMAND BYTE
F892 DD2A00F8  01570          LD      IX,(PARAM1)     ;POINT TO FIRST PARAMETER
F896 CD20F8    01580          CALL    CONVRT          ;CONVERT TO 8231A FORMAT
F899 DD2A02F8  01590          LD      IX,(PARAM2)     ;POINT TO SECOND PARAM.
F89D CD20F8    01600          CALL    CONVRT          ;CONVRT TO 8231A FORMAT
F8A0 1814      01610          JR      EXECUT          ;EXECUTE DESIRED FUNCTIN
F8A2 18ED      01620          JR      TWOVAL          ;TWO-VALUED OPERATION
F8A4 3E02      01630 SINE     LD      A,SIN           ;GET SINE COMMAND BYTE
F8A6 1806      01640          JR      ONEVAL          ;ONE-VALUED OPERATION
F8A8 3E01      01650 ROOT     LD      A,SQRT          ;GET SQUARE ROOT CMD BYTE
F8AA 1802      01660          JR      ONEVAL          ;ONE-VALUE OPERATION
F8AC 3E04      01670 TANGNT   LD      A,TAN           ;GET TANGENT COMMAND BYTE
F8AE 08        01680 ONEVAL   EX      AF,AF'          ;SAVE COMMAND BYTE
F8AF DD2A00F8  01690          LD      IX,(PARAM1)     ;POINT TO FIRST PARAMETER
F8B3 CD20F8    01700          CALL    CONVRT          ;CONVERT TO 8231A FORMAT
F8B6 08        01710 EXECUT   EX      AF,AF'          ;GET COMMAND BYTE
F8B7 D301      01720          OUT     (COMAND),A      ;SEND COMMAND TO 8231A
F8B9 DB01      01730 LOOP1    IN      A,(STATUS)      ;CHECK FOR COMPLETION
F8BB CB7F      01740          BIT     7,A             ;STILL BUSY?
F8BD 20FA      01750          JR      NZ,LOOP1        ;IF YES, CHECK AGAIN
F8BF E61E      01760          AND     1EH             ;CHECK FOR ANY ERROR
F8C1 C20AF9    01770          JP      NZ,ERROR3       ;IF YES, CALC. ERROR
F8C4 DD2A04F8  01780          LD      IX,(DEST)       ;ELSE, POINT TO DEST.
F8C8 DB01      01790          IN      A,(STATUS)      ;GET STATUS AGAIN
F8CA CB6F      01800          BIT     5,A             ;CHECK FOR ZERO RESULT
F8CC 280F      01810          JR      Z,RESULT        ;IF NOT, GET RESULT
F8CE DD360300  01820          LD      (IX+3),0        ;ELSE STORE 00 EXP BYTE
F8D2 3E18      01830          LD      A,POPF          ;GET STACK POP COMMAND
F8D4 D301      01840          OUT     (COMAND),A      ;POP 8231A STACK
```

Listing 2 continued

PROJECT 80

Continued from p. 204

74LS138 select the address where the 8231A or 8232 appears in I/O space to the TRS-80. The dotted lines show the combination I used here and in the upcoming program listings. The four combinations give addresses as in Table 2.

The slashes following the address bit represent the inverted (barred) signals on the schematic. I assigned four addresses to each combination above, although the math processor only needs two. The math processor is therefore "double addressed" in the selected addressing space; that is, you can access it by either the lower two addresses or the upper two addresses (or a combination thereof).

The jumper between pin 6 of the 8231A/8232 and ground should be in place when you use the 8231A and removed when you use the 8232. This is the only hardware interface difference between the two devices. Pin 6 is not used on the 8231A and must be tied to ground. The 8232 uses that pin as an error output; it can generate a special interrupt if other hardware is available to support it.

G2	G3	Address Range
A6/	A5/	40 hex–43 hex
A6/	A5	60 hex–63 hex
A6	A5/	00 hex–03 hex
A6	A5	20 hex–23 hex

Table 2. Address ranges for the math processor.

The final jumper, between the END/ output of the 8231A/8232 and the TRS-80 interrupt line, goes active (low) at the completion of its command processing, and can be used to alert the TRS-80 when it's ready to be serviced. This is great for applications where the Z80 can do other useful work while the math processor cranks away, but can cause problems if not properly handled.

If you don't intend to use the interrupt feature, don't hook it up. I did hook up my interrupt line, but the upcoming software does not use the interrupt feature. When the interrupt (END/) line goes active, any access to the math processor forces it inactive.

The crystal on the schematic (Fig. 5)

↵ See List of Advertisers on Page 211

shows a 4 MHz value. There are various speed ranges available for the 8231A/8232 math processors, ranging from 2.5 MHz to 6 MHz. Only the crystal frequency needs to be changed to use a different speed math processor.

To work properly, the math processor must get a reset signal before you use it. You can guarantee this by turning on the power to the math processor board before turning on the computer power, or by pressing the TRS-80 reset button while both are on.

Listing 2 continued

```
F8D6 DB01     01850 LOOP2   IN      A,(STATUS)      ;CHECK FOR COMMAND DONE
F8D8 CB7F     01860         BIT     7,A             ;DONE?
F8DA 20FA     01870         JR      NZ,LOOP2        ;IF NOT, KEEP CHECKING
F8DC C9       01880         RET                     ;DONE - RETURN
F8DD DB00     01890 RESULT  IN      A,(DATA)        ;GET EXPONENT BYTE
F8DF 4F       01900         LD      C,A             ;SAVE VALUE TEMP.
F8E0 CB7F     01910         BIT     7,A             ;CHECK FOR NEG. MANTISA
F8E2 DB00     01920         IN      A,(DATA)        ;GET HIGH MANTISSA BYTE
F8E4 2002     01930         JR      NZ,SKIP         ;IF BIT SET, OK
F8E6 CBBF     01940         RES     7,A             ;ELSE, RESET HIGH MANT BT
F8E8 DD7702   01950 SKIP    LD      (IX+2),A        ;STORE HIGH MANT BYTE
F8EB 79       01960         LD      A,C             ;GET EXPONENT BYTE
F8EC 07       01970         RLCA                    ;ROTATE LEFT ONE BIT
F8ED CB2F     01980         SRA     A               ;DUPLICATE HIGH BIT
F8EF C680     01990         ADD     A,128           ;MAKE EXCESS 128 FORM
F8F1 DD7703   02000         LD      (IX+3),A        ;SAVE EXPONENT BYTE
F8F4 DB00     02010         IN      A,(DATA)        ;GET NEXT RESULT BYTE
F8F6 DD7701   02020         LD      (IX+1),A        ;SAVE IN RESULT AREA
F8F9 DB00     02030         IN      A,(DATA)        ;GET RESULT LSB
F8FB DD7700   02040         LD      (IX),A          ;SAVE IN RESULT AREA
F8FE C9       02050         RET                     ;DONE - RETURN
              02060 ;*
              02070 ;*** ERROR ROUTINES:
F8FF 2116F9   02080 ERROR1  LD      HL,ERMSG1       ;POINT TO ERR MSG #1
F902 1809     02090         JR      PRRTN           ;PRINT AND RETURN
F904 E1       02100 ERROR2  POP     HL              ;POP ADDR FROM STACK
F905 2131F9   02110         LD      HL,ERMSG2       ;POINT TO ERR MSG #2
F908 1803     02120         JR      PRRTN           ;PRINT AND RETURN
F90A 2154F9   02130 ERROR3  LD      HL,ERMSG3       ;POINT TO ERR MSG #3
F90D 7E       02140 PRRTN   LD      A,(HL)          ;GET CHARACTER
F90E B7       02150         OR      A               ;CHECK FOR ZERO BYTE
F90F C8       02160         RET     Z               ;IF ZERO, DONE
F910 CD3A03   02170         CALL    PRCHAR          ;ELSE, PRINT THE CHAR.
F913 23       02180         INC     HL              ;POINT TO NEXT CHAR.
F914 18F7     02190         JR      PRRTN           ;LOOP AGAIN
              02200 ;
              02210 ;
F916 0D       02220 ERMSG1  DEFB    0DH
F917 2A       02230         DEFM    '*** INVALID COMMAND CODE'
F92F 0D       02240         DEFB    0DH
F930 00       02250         DEFB    00H
F931 0D       02260 ERMSG2  DEFB    0DH
F932 2A       02270         DEFM    '*** PARAMETER VALUE OUT OF RANGE'
F952 0D       02280         DEFB    0DH
F953 00       02290         DEFB    00H
F954 0D       02300 ERMSG3  DEFB    0DH
F955 2A       02310         DEFM    '*** CALCULATION ERROR'
F96A 0D       02320         DEFB    0DH
F96B 00       02330         DEFB    00H
              02340 ;
F96C 5FF8     02350 FNCTBL  DEFW    ADD             ;ADDITION ROUTINE
F96E 63F8     02360         DEFW    MULT            ;MULTIPLICATION ROUTINE
F970 67F8     02370         DEFW    INVCOS          ;INVERSE COSINE ROUTINE
F972 6BF8     02380         DEFW    INVSIN          ;INVERSE SINE ROUTINE
F974 6FF8     02390         DEFW    INVTAN          ;INVERSE TANGENT ROUTINE
F976 73F8     02400         DEFW    CHGSGN          ;CHANGE SIGN ROUTINE
F978 77F8     02410         DEFW    COSINE          ;COSINE ROUTINE
F97A 7BF8     02420         DEFW    EXPON           ;EXPONENTIAL ROUTINE
F97C 7FF8     02430         DEFW    DIV             ;DIVISION ROUTINE
F97E 83F8     02440         DEFW    SUB             ;SUBTRACTION ROUTINE
F980 87F8     02450         DEFW    COMLOG          ;COMMON LOG ROUTINE
F982 8BF8     02460         DEFW    NATLOG          ;NATURAL LOG ROUTINE
F984 8FF8     02470         DEFW    POWER           ;POWER ROUTINE
F986 A4F8     02480         DEFW    SINE            ;SINE ROUTINE
F988 A8F8     02490         DEFW    ROOT            ;SQUARE ROOT ROUTINE
F98A ACF8     02500         DEFW    TANGNT          ;TANGENT ROUTINE
              02510 ;
F806          02520         END     MATH
```

```
10 DEF FN ACS(X)=-1*ATN(X/SQR(-1*X*X+1))+1.5708
25 PI=3.14159
30 A=2.445:B=1.44556:C=2.5:D=6:E=.456:F=2.456:G=15
65 FORI=1TO1000
70 Z=A[4+C*TAN(B*4)+2*PI*(SIN(D[7)+FNACS(E)+EXP(F))+SQR(LOG(G))
75 NEXTI
80 PRINTZ:END
```

Program Listing 3. A complex equation.

Operation

A hardware math processor is most beneficial when used strictly with Assembly-language programs, since it requires no format conversions, takes up no Basic (or other language) overhead, and since all of the common mathematical functions are available (when using the 8231A), without requiring any software math routines. You save programming time, and attain much faster speeds than executing software math routines.

But since it's likely that you use programs written in Basic, you need a way for them to access the math processor. Do this through the user (USR) function.

First create a way to pass variables to and from the Basic program. Basic provides the VARPTR function to find the addresses of specified variables. You can POKE the desired variable addresses into predetermined memory locations (reversed by the Assembly-language routine) so that the Assembly-language routine knows where to find its operands. In Program Listing 1, the variables A, B, and C are used: A or A and B are the operand(s); C is the result.

My Assembly-language routines begin at location 0F800 hex (in a 48K RAM system) reserving more memory than necessary. You can shrink the routines as well as place them at the top of memory for other memory configurations; make sure memory size is set properly to reserve the used space. I set memory size to 63487 for my routines as addressed. The six memory locations reserved for the operands are the first six in the reserved area: 0F800 hex, 0F802 hex and 0F804 hex, for the first Assembly-language routine, and the first 16 locations for the second routine.

To set up Basic to access the user function, follow line 20 of Listing 1 for a disk-based system, or POKE the proper address into locations 16526 and 16527 for a non-disk system. For the two Assembly-language routines below, the first is at 0F806 hex, requiring you to POKE the values 6 and 248, respectively. The second Assembly-language routine, starting at 0F810 hex, requires that you POKE the values 16 and 248, respectively.

The first Assembly-language routine (Program Listing 2) converts the TRS-80 operand(s) into 8231A for-

Program Listing 4. Modification of math processing routine.

```
                00100 ;*************************************************
                00110 ; MATH PROCESSING ROUTINE FOR INTEL 8231A MATH     *
                00120 ; PROCESSOR.                                       *
                00130 ;                                                  *
                00140 ;       Created by  Roger C. Alford                *
                00150 ;            October, 1983                         *
                00160 ;                                                  *
                00170 ; This program is entered via the Basic USR        *
                00180 ; function call. It executes the function:         *
                00190 ; Z=A**4+C*TAN(B*4)+2*PI*(SIN(D**7)+ACOS(E)+EXP(F)) *
                00200 ;   +SQR(LN(G))                                    *
                00210 ;                                                  *
                00220 ; Where ** is 'to the power of', LN is the natural *
                00230 ; log function, and ACOS is the inverse cosine     *
                00240 ; function.                                        *
                00250 ;                                                  *
                00260 ;*************************************************
                00270 ;
                00280 ;
F800            00290           ORG     0F800H
                00300 ;
0001            00310 COMAND    EQU     01H     ;8231A COMMAND PORT ADDRESS
0001            00320 STATUS    EQU     01H     ;8231A STATUS PORT ADDRESS
0000            00330 DATA      EQU     00H     ;8231A DATA PORT STACK ADDRESS
0A7F            00340 FNCTN     EQU     0A7FH   ;TRS-80 'GET USR PARAM.' ROUTINE
0010            00350 FADD      EQU     10H     ;8231A ADD COMMAND BYTE
0012            00360 FMUL      EQU     12H     ;8231A  MULTIPLY COMMAND BYTE
0006            00370 ACOS      EQU     06H     ;8231A INVERSE COSINE CMD BYTE
0005            00380 ASIN      EQU     05H     ;8231A INVERSE SINE CMD BYTE
0007            00390 ATAN      EQU     07H     ;8231A INVERSE TANGENT CMD BYTE
0015            00400 CHSF      EQU     15H     ;8231A SIGN CHANGE CMD BYTE
0003            00410 COS       EQU     03H     ;8231A COSINE CMD BYTE
000A            00420 EXP       EQU     0AH     ;8231A EXPONENTIAL CMD BYTE
0013            00430 FDIV      EQU     13H     ;8231A DIVIDE CMD BYTE
0011            00440 FSUB      EQU     11H     ;8231A SUBTRACT CMD BYTE
0008            00450 LOG       EQU     08H     ;8231A COMMON LOG CMD BYTE
0009            00460 LN        EQU     09H     ;8231A NATURAL LOG CMD BYTE
000B            00470 PWR       EQU     0BH     ;8231A POWER CMD BYTE
0002            00480 SIN       EQU     02H     ;8231A SINE CMD BYTE
0001            00490 SQRT      EQU     01H     ;8231A SQUARE ROOT CMD BYTE
0004            00500 TAN       EQU     04H     ;8231A TANGENT CMD BYTE
0018            00510 POPF      EQU     18H     ;8231A STACK POP COMMAND
001A            00520 PUPI      EQU     1AH     ;8231A PUSH PI ON STACK COMMAND
033A            00530 PRCHAR    EQU     033AH   ;TRS-80 'PRINT CHAR.' ROUTINE
                00540 ;
0002            00550 PARAM1    DEFS    2       ;FIRST MATH PARAMETER ADDRESS
0002            00560 PARAM2    DEFS    2       ;SECOND MATH PARAMETER ADDRESS
0002            00570 PARAM3    DEFS    2       ;THIRD PARAMETER ADDRESS
0002            00580 PARAM4    DEFS    2       ;FOURTH PARAMETER ADDRESS
0002            00590 PARAM5    DEFS    2       ;FIFTH PARAMETER ADDRESS
0002            00600 PARAM6    DEFS    2       ;SIXTH PARAMETER ADDRESS
0002            00610 PARAM7    DEFS    2       ;SEVENTH PARAMETER ADDRESS
0002            00620 DEST      DEFS    2       ;DESTINATION VARIABLE ADDRESS
                00630 ;
F810 F3         00640 MATH      DI                      ;INTERRUPTS ARE DISABLED
F811 DD2A06F8   00650           LD      IX,(PARAM4)     ;POINT TO D VARIABLE
F815 CDCFF8     00660           CALL    CONVRT          ;PUT ONTO 8231A STACK
F818 DD21B4F9   00670           LD      IX,CONST7       ;POINT TO THE CONSTNT 7
F81C CDCFF8     00680           CALL    CONVRT          ;PUT ONTO 8231A STACK
F81F CD3EF9     00690           CALL    POWER           ;RAISE TO THE POWER
F822 CD42F9     00700           CALL    SINE            ;GET SINE OF RESULT
F825 DD2A08F8   00710           LD      IX,(PARAM5)     ;POINT TO E VARIABLE
F829 CDCFF8     00720           CALL    CONVRT          ;PUT ONTO 8231A STACK
F82C CD16F9     00730           CALL    INVCOS          ;GET INVERSE COSINE
F82F CD0EF9     00740           CALL    ADD             ;ADD INTERMED. RESULTS
F832 DD2A0AF8   00750           LD      IX,(PARAM6)     ;POINT TO F VARIABLE
F836 CDCFF8     00760           CALL    CONVRT          ;PUT ONTO 8231A STACK
F839 CD2AF9     00770           CALL    EXPON           ;GET NATURAL ANTILOG
F83C CD0EF9     00780           CALL    ADD             ;ADD TO INTERMED. RESULT
F83F CD4EF9     00790           CALL    PI              ;PUSH PI ONTO 8231A STACK
F842 CD12F9     00800           CALL    MULT            ;MULTIPLY W/INTERMED. RES
F845 DD21ACF9   00810           LD      IX,CONST2       ;POINT TO CONSTNT 2
F849 CDCFF8     00820           CALL    CONVRT          ;PUT ONTO 8231A STACK
F84C CD12F9     00830           CALL    MULT            ;MULTIPLY W/INTERMED. RES
F84F DD2A0CF8   00840           LD      IX,(PARAM7)     ;POINT TO G VARIABLE
F853 CDCFF8     00850           CALL    CONVRT          ;PUT ONTO 8231A STACK
F856 CD3AF9     00860           CALL    NATLOG          ;GET NATURAL LOG
F859 CD46F9     00870           CALL    ROOT            ;GET SQUARE ROOT
F85C CD0EF9     00880           CALL    ADD             ;ADD TO INTERMED. RESULT
F85F DD2A02F8   00890           LD      IX,(PARAM2)     ;POINT TO B VARIABLE
F863 CDCFF8     00900           CALL    CONVRT          ;PUT ONTO 8231A STACK
F866 DD21B0F9   00910           LD      IX,CONST4       ;POINT TO CONSTNT 4
F86A CDCFF8     00920           CALL    CONVRT          ;PUT ONTO 8231A STACK
F86D CD12F9     00930           CALL    MULT            ;MULTIPLY B VARIABLE
F870 CD4AF9     00940           CALL    TANGNT          ;GET TANGENT
F873 DD2A04F8   00950           LD      IX,(PARAM3)     ;POINT TO C VARIABLE
F877 CDCFF8     00960           CALL    CONVRT          ;PUT ONTO 8231A STACK
F87A CD12F9     00970           CALL    MULT            ;MULTIPLY WITH INTER. RES
F87D CD0EF9     00980           CALL    ADD             ;ADD INTERMED. RESULTS
F880 DD2A00F8   00990           LD      IX,(PARAM1)     ;POINT TO A PARAMETER
F884 CDCFF8     01000           CALL    CONVRT          ;PUT ONTO 8231A STACK
F887 DD21B0F9   01010           LD      IX,CONST4       ;POINT TO CONSTNT 4
F88B CDCFF8     01020           CALL    CONVRT          ;PUT ONTO 8231A STACK
F88E CD3EF9     01030           CALL    POWER           ;DO POWER FUNCTION
F891 CD0EF9     01040           CALL    ADD             ;ADD TO INTERMEDIATE RES.
```

Listing 4 continued

✔ See List of Advertisers on Page 211

THE SECRETS OF PERFECT MEMORY: ONE AND ONE HALF EARTH DOLLARS

AT LAST: THE WHOLE TRUTH ABOUT FLOPPIES.

Amazing book reveals all!

How to keep from brainwashing your disk so it never loses it's memory.

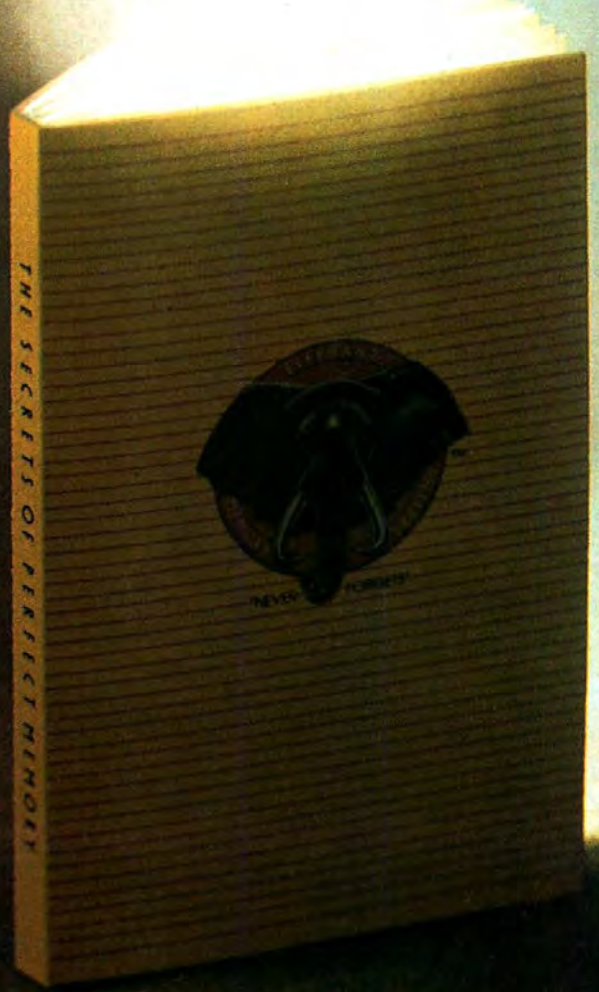
How fingerprints can actually damage disks. Unretouched Kirlian photographs of UFO's (Unidentified Floppy Objects)! The incredible importance of making copies: the Department of Redundancy Department—and what goes on when it goes on! Powerful secret methods that scientists claim can actually prevent computer amnesia! All this, and much more . . .

In short, it's an 80-page plain-English, graphically stunning, pocket-sized definitive guide to the care and feeding of flexible disks.

For The Book, ask your nearest computer store that sells Elephant™ disks, and bring along one and one half earth dollars.

For the name of the store, ask us.

Elephant Memory Systems®
Marketed exclusively by Dennison Computer Supplies, Inc., 55 Providence Highway, Norwood, MA 02062. Call toll free 1-800-343-8413. In Massachusetts, call collect (617) 769-8150. Telex 951-624.



472

READER SERVICE

This card valid until March 31, 1984

Please help us to bring you a better magazine—by answering these questions.

My vote for the best advertisement in this issue goes to _____ (company) whose Reader Service number is _____

A. How long have you been using microcomputers?
- ☐ 1. 1 year or less
- ☐ 2. 2–3 years
- ☐ 3. 4–5 years
- ☐ 4. More than 5 years

B. How often do you use each of the following operating systems? Please rate them on a scale of 1 (never use) to 5 (use most often).
- ☐ 1. TRSDOS
- ☐ 2. LDOS
- ☐ 3. NEWDOS
- ☐ 4. DOSPLUS
- ☐ 5. CP/M

C. What is your most important use for your TRS-80? Check one only.
- ☐ 1. Business Management
- ☐ 2. Accounting
- ☐ 3. Word Processing
- ☐ 4. Home Management
- ☐ 5. Science
- ☐ 6. Education
- ☐ 7. Games
- ☐ 8. Hobby
- ☐ 9. Other

D. What type of TRS-80(s) do you own?
- ☐ 1. Model I
- ☐ 2. Model II/12
- ☐ 3. Model III
- ☐ 4. Model 4
- ☐ 5. Model 4P
- ☐ 6. Model 16
- ☐ 7. Model 100
- ☐ 8. Color Computer
- ☐ 9. Pocket Computer

E. If you were planning to purchase an additional computer would you purchase another TRS-80?
- ☐ 1. Yes
- ☐ 2. No

F. Do you use CP/M on your TRS-80?
- ☐ 1. Yes
- ☐ 2. No

G. Where do you buy your peripherals and accessories? Check all that apply.
- ☐ 1. Radio Shack store
- ☐ 2. Other computer/electronics store
- ☐ 3. Retail store
- ☐ 4. Mail order
- ☐ 5. Direct from manufacturer
- ☐ 6. Office equipment dealer
- ☐ 7. Used equipment dealer
- ☐ 8. Private individual

H. What type of program would you most like to see on LOAD-80? Check one only.
- ☐ 1. Music
- ☐ 2. Graphics
- ☐ 3. Games
- ☐ 4. Utilities
- ☐ 5. Education
- ☐ 6. Home/Personal
- ☐ 7. Science
- ☐ 8. Business
- ☐ 9. Other

I. Which of the following is most important to you for the price of a LOAD-80 cassette or disk? Check only one.
- ☐ 1. Number of programs
- ☐ 2. Selection of programs
- ☐ 3. Quality of programs
- ☐ 4. Technical support
- ☐ 5. Customer service
- ☐ 6. Other

L. Which of the following types of programs would you like to see in 80 Micro? Rate them on a scale of 1 (no interest) to 5 (great interest).
- ☐ 1. Business applications
- ☐ 2. Science applications
- ☐ 3. Educational applications
- ☐ 4. Games
- ☐ 5. Utilities
- ☐ 6. Video graphics
- ☐ 7. Printer graphics
- ☐ 8. Personal investment
- ☐ 9. Home finance

M. If you are not a subscriber please cirlce 500.

Reader Service: To receive more information from any of the advertisers in this issue of 80 Micro circle the number on the Reader Service Card that corresponds with the Reader Service number on the ad in which you are interested. You will find numbers, preceded by a ✓, near the logo of each advertiser. Complete the entire card, drop into a mailbox and in 4-6 weeks you will hear from the advertiser directly.

1 6 11 16 21	151 156 161 166 171	301 306 311 316 321	451 456 461 466 471
2 7 12 17 22	152 157 162 167 172	302 307 312 317 322	452 457 462 467 472
3 8 13 18 23	153 158 163 168 173	303 308 313 318 323	453 458 463 468 473
4 9 14 19 24	154 159 164 169 174	304 309 314 319 324	454 459 464 469 474
5 10 15 20 25	155 160 165 170 175	305 310 315 320 325	455 460 465 470 475
26 31 36 41 46	176 181 186 191 196	326 331 336 341 346	476 481 486 491 496
27 32 37 42 47	177 182 187 192 197	327 332 337 342 347	477 482 487 492 497
28 33 38 43 48	178 183 188 193 198	328 333 338 343 348	478 483 488 493 498
29 34 39 44 49	179 184 189 194 199	329 334 339 344 349	479 484 489 494 499
30 35 40 45 50	180 185 190 195 200	330 335 340 345 350	480 485 490 495 500
51 56 61 66 71	201 206 211 216 221	351 356 361 366 371	501 506 511 516 521
52 57 62 67 72	202 207 212 217 222	352 357 362 367 372	502 507 512 517 522
53 58 63 68 73	203 208 213 218 223	353 358 363 368 373	503 508 513 518 523
54 59 64 69 74	204 209 214 219 224	354 359 364 369 374	504 509 514 519 524
55 60 65 70 75	205 210 215 220 225	355 360 365 370 375	505 510 515 520 525
76 81 86 91 96	226 231 236 241 246	376 381 386 391 396	526 531 536 541 546
77 82 87 92 97	227 232 237 242 247	377 382 387 392 397	527 532 537 542 547
78 83 88 93 98	228 233 238 243 248	378 383 388 393 398	528 533 538 543 548
79 84 89 94 99	229 234 239 244 249	379 384 389 394 399	529 534 539 544 549
80 85 90 95 100	230 235 240 245 250	380 385 390 395 400	530 535 540 545 550
101 106 111 116 121	251 256 261 266 271	401 406 411 416 421	551 556 561 566 571
102 107 112 117 122	252 257 262 267 272	402 407 412 417 422	552 557 562 567 572
103 108 113 118 123	253 258 263 268 273	403 408 413 418 423	553 558 563 568 573
104 109 114 119 124	254 259 264 269 274	404 409 414 419 424	554 559 564 569 574
105 110 115 120 125	255 260 265 270 275	405 410 415 420 425	555 560 565 570 575
126 131 136 141 146	276 281 286 291 296	426 431 436 441 446	576 581 586 591 596
127 132 137 142 147	277 282 287 292 297	427 432 437 442 447	577 582 587 592 597
128 133 138 143 148	278 283 288 293 298	428 433 438 443 448	578 583 588 593 598
129 134 139 144 149	279 284 289 294 299	429 434 439 443 449	579 584 589 594 599
130 135 140 145 150	280 285 290 295 300	430 435 440 445 450	580 585 590 595 600

Name _____

Address _____

City _____ State _____ Zip _____

BOOKS

2-84

Please send me the following 80 Micro products:

80 Micro
Peterborough NH 03458

Qty.	Catalog #	Title	Unit Price	Total

Shipping and handling charges:
$1.50 1st book, $1.00 each additional book (UPS, use street address)
$10.00 each book overseas airmail

Shipping/handling _____

Total _____

Enclosed $ _____ ☐ Check ☐ M.O.

Bill: ☐ AE ☐ MC ☐ VISA

Card# _____ Exp. date _____

Signature _____ Interbank# _____

Name _____

Address _____

City _____ State _____ Zip _____

No C.O.D. orders accepted. Please allow 4–6 weeks for delivery

SUBSCRIPTION

☐ New Subscription ☐ Renewal

1 year—$24.97

Enclosed $ _____ ☐ Check/ M.O.

Bill: ☐ AE ☐ MC ☐ VISA ☐ Me

Card# _____

Signature _____

Name _____

Address _____

City _____ State _____ Zip _____

Canada and Mexico $27.97/1 year only, US funds drawn on US bank.
Foreign Surface $44.97/1 year only, US funds drawn on US bank.
Please allow 6–8 weeks for delivery.

*This offer expires March 31, 1984

342R89

Place
Stamp
Here

Wayne Green Inc.
80micro®
Subscription Dept.
POB 981
Farmingdale, NY 11737

Place
Stamp
Here

Wayne Green Inc
80micro®
Attn: Mail Order
Peterborough, NH 03458

80micro®
P.O. Box 306
Dalton, MA 01226

Place
Stamp
Here

80 LIST OF ADVERTISERS

Reader Service Number		Page
208	3M Company	7
439	A & B Software	169
38	A.C.E. Systems	19
46	A-1 Computer Paper Co.	257
196	Accurate Data Services	219
568	Adams Magnetic Products	255
356	Adel Computer Mart	258
56	Adult Video Games	230
82	Aerocomp Inc.	184, 185
111	AFIPS	103
571	Alien Group	270
136	Allen Gelder/Algorix	255
140	Allied Systems Company	257
478	Alpha Bit Communications	267
17	Alpha Products Company	17, 29, 31, 33, 73
374	ALPS	209
217	American Small Business Computers	237
576	Andreasen's Electronics R & D	258
141	Anitek Software Products	65
40	Apparat Inc.	101
390	Applied Microsystems Inc.	81
514	Ashland Computer Systems	160
424	Aspen Ribbons Inc.	231
383	Astro-Star Enterprises	169
583	B.T.B. Inc.	272
300	B.T. Enterprises	107, 155
552	B.V. Engineering	264
210	Balmoral	219
152	BCCOMPCO	270
271	Becks-MFG	16
25	Beta Enterprises	230
180	Bill Cole Enterprises	121
106	Binary Devices	253
58	Binary Devices	105
542	Block Island	160
381	Bodex	259
256	Borg Industries	266
*	Bottom Line	204
382	Byte General	206
167	CDC	169
225	Chalanooga Systems Assoc.	238
569	Colorado Online Systems	273
150	Complete Computer Service	257
312	Compress Co.	124
252	CompuAdd Corporation	252
455	Compukit	256, 257
86	Compulogic	260
139	Computer Case Co.	247
387	Computer Center	257
323	Computer Council	154
357	Computer Friends	248
18	Computer Plus	19
39	Computer Shopper	238
558	Comrex International	260
356	Copley Press/Cleo	257
45	Cornucopia Software	151, 153
*	Cosmopolitan Electronic Corp.	37
190	Creative Computing	83
223	Crest Software	169
49	Custom Software	230
560	D.T.C.	254
63	Data Base Industries	192
531	Datafile Systems	238
472	Dennison MFG/Leading Edge	210
539	Desert Sound	141
282	DFW Computer Center	241
183	Diaz Enterprises	196
367	Digital Images	230
574	Discovery Design Center	268
204	Diskcount Data	198, 199
441	Diskette Connection	271
62	Displayed Video	138, 139
*	80 MICRO	
	80 MICRO Review Guide	243
	Dealer Sell	135, 267
	Foreign Dealers	177
	Load 80 Back Issues	233
	Load 80/Color Load 80	98-99
	Moving	135, 261
	Subscriptions	35
216	EAP Company	112
178	EZ Ware Corporation	219
249	Ed Ball Co.	153
85	Educational Micro Systems	188
435	Edwards and Associates	173
325	Ehlen Enterprises	188, 260
114	EJB Electronic Systems	151
544	Eicsoft	151
144	Elk-Tek Inc.	124
284	ETS Center	104
556	Evolving Technology Co.	262
267	Excellonix	203
297	Fantastic 4	160
44	FGA Software	105
214	Fort Worth Computers	135, 177
528	Future Projects Corporation	259
579	Gessler Educational Software	264
551	Gibberman Enterprises	250
523	Gibberman Enterprises	92
186	Good Lyddon Data Systems	106
496	Good Software	203
185	Gooth Software	265
9	H & E Compufronics	C III, 3, 75, 77
355	H.D.P.	141
153	Holmes Engineering	245
557	Holmes Engineering	262
*	HOT CoCo Subscription	67
564	Howard W. Sams & Co.	268
565	Howard W. Sams & Co.	260
175	Howe Software	115
198	I/O Tech	218
205	I/O Ware Inc.	156
352	Indiana Software Group	244
177	J & J Electronics	231
101	J & M Systems	12
534	Jameco Electronics	159
126	JMG Software International	39
521	Juki Industries of America	42
121	JVB Electronics	241
485	Kalglo Electronics Co.	259
580	Krell Software Corp.	250
331	KSOFT	247
354	Kuzel Computer Software	230
462	Langley St. Clair	15, 95
130	Lap Video Entertainment	116
577	Ledu Corporation	263
135	Lindbergh Systems	47
31	LNW Research Corp.	C IV
535	Lobo Drives International	87
330	Logical Devices Inc.	203
385	Logix	181
112	Lynn Computer Service	125
115	Lynn Computer Service	123
572	Macrotronics Inc.	266
527	Magicomp	114
319	Manx Software Systems	188
241	Marathon Software	262
250	Marymac	231
197	Master Stock Service	257
199	Mega-Byte	197
22	Michtron	4
132	Micro 80 Inc.	239, 268
149	Micro Architect Inc.	209
171	Micro Control Systems	59
*	Micro Data Supplies	23, 24, 25
426	Micro Design	143
313	Micro Equipment Corporation	141
157	Micro Images	213
96	Micro Management Systems	149
162	Micro Mega	267
561	Micro Peripherals	250
546	Micro Projects Engineering, Inc.	141
526	Micro Software Systems	49
157	Micro-Images	213
464	Micro-Labs Inc.	273
578	Micro-Labs Inc.	268
293	Microcomputer Applications	225
47	Microcomputer Business Systems	79
419	Microhatch	231
84	Micromatic Programming Corp.	269
312	Microsette	227
123	Microsmith Computer Tech.	181
380	Microtech Exports Inc.	230
*	Midwest Computer Wholesaler	264
137	Miller Microcomputer Service	136
107	Misosys	83, 193
563	Misosys	264
416	Montezuma Micro	55
411	Montezuma Micro	221
23	Mt. Olympus Software	195
*	Mumford Micro Systems	183
533	Nebs Computer Forms	111
255	New Classic Software	207, 231
232	Nocona Electronics	223
54	Nodvill Software	231
541	Northern Technology Corp.	177
*	NRI Schools	119
193	OFS Inc.	230
36	Omnisoft Research	105
195	Omnitek Computers International	258
151	Orion Instruments	254
108	Orion Systems	173
207	Pacific Exchanges	173, 181, 230, 231
547	PAECO	161
122	Pan American Electronics	266
20	Pasadena Technology Press	209
211	PCD	41
324	Pel/Tek	88
127	Penguin Products	173
550	Penguin Products	274
124	Perry Computers	217
176	Personal Computer Products	169, 228
509	Phone Line	173
314	Pickam Software	271
290	Pickles and Trout	129
160	Pioneer Software	57
306	Powersoft	235
260	Pro/Am Software	273
449	Professor Jones/Frogg House	214
308	Professional Tax Software	241
248	Programmer and Associates	247
98	Prosoft	11
91	Prosoft	274
1	Prosoft	62
2	Prosoft	63
13	Prosoft	64
8	Prosoft	66
194	Quant Systems	241
78	R & S Software	110
500	Radio Shack	8, 9
296	Rainbow, The	263
129	Remsoft Inc.	181
567	Rhino Robots	272
518	Rizzo Data Systems	181
452	Robert E. Litke	225
512	Rockware Data Corporation	113
*	Rocky Mountain Software	171
370	Ross Custom Electronics	253
	RUN Subscriptions	179
368	Safeware	173
277	Saturn Electronics	104
203	Scientific Engineering Labs	230
53	Selective Software Systems	173
376	Selectronics	120
35	SID	205
12	Simutek	175
163	Simutek	261, 263, 265
6	Skyline Marketing Corp.	253
554	Skyline Marketing Corp.	266
74	Skyline Software	177
245	Small Computer Company	21
57	Softshell	269
581	Softshell	270
492	Softrends	89
116	Softronics Computer Systems	145
28	Software Exchange	153
327	Software Support	90, 91
182	Something Special Sales	225
42	South Bay Software	259
92	Spiral Enterprises	157
43	Star Software	249
442	Stevens Electronics	151
71	Sublogic Communications Corp.	242
285	Sun Research Inc.	215
456	Sunlock Systems	269
559	Sweetware Inc.	273
575	Syncom	250
189	Tab Sales	160
70	Taranto and Associates	71
15	Telesoft	153
562	Telesoft	252
59	Texas Computer Systems	83, 133, 162, 163, 164, 165
72	Three State Data	205
81	Total Access	167
430	Transaction Storage Systems	C II
553	Traveling Software	252
188	Triangle Software	97
489	Triple D Software	255
227	Trisoft	225
555	Valu-tique	255
*	Vespa Computer Outlet	137
73	Videotronics of Sarasota	181
335	Virginia Micro Systems	181
10	VR Data Corp.	251
305	Wadsworth Electronics	240
309	Walonick Associates	261
*	Wayne Green Books	
66	Data Files	131
80	Rest of 80	271
*	Shelf boxes	265
77	TRS-80/Z80	147
93	TRS-80 Controller	272
61	Wayne Green Books	146
238	Western Micro Systems	117
37	Wiley Inc.	191
226	William A. Fink	89
79	Wooden Hen	205
158	XYZT Computer Dimensions Inc.	229
156	York 10	190
573	Ziyad	258

For further information from our advertisers, please use the Reader Service card.

***This advertiser prefers to be contacted directly.**

Listing 4 continued

```
F894 DD2A0EF8 01050          LD      IX,(DEST)       ;POINT TO DEST. (Z) VAR.
F898 DB01     01060          IN      A,(STATUS)      ;GET LAST STATUS BYTE
F89A CB6F     01070          BIT     5,A             ;CHECK FOR ZERO RESULT
F89C 280F     01080          JR      Z,RESULT        ;IF NOT ZERO, SKIP
F89E DD360300 01090          LD      (IX+3),0        ;ELSE STORE 00 EXP BYTE
F8A2 3E18     01100          LD      A,POPF          ;GET 8231A POP COMMAND
F8A4 D301     01110          OUT     (COMAND),A      ;SEND TO 8231A
F8A6 DB01     01120 LOOP2    IN      A,(STATUS)      ;CHECK STATUS BYTE
F8A8 CB7F     01130          BIT     7,A             ;DONE YET?
F8AA 20FA     01140          JR      NZ,LOOP2        ;IF NOT, LOOP AGAIN
F8AC C9       01150          RET                     ;ELSE, DONE - RETURN
F8AD DB00     01160 RESULT   IN      A,(DATA)        ;GET EXPONENT BYTE
F8AF 4F       01170          LD      C,A             ;SAVE VALUE TEMP.
F8B0 CB7F     01180          BIT     7,A             ;CHECK MANT. SIGN
F8B2 DB00     01190          IN      A,(DATA)        ;GET HIGH MANT. BYTE
F8B4 2002     01200          JR      NZ,SKIP         ;IF BIT SET, OK
F8B6 CBBF     01210          RES     7,A             ;ELSE, CHANGE TO NEG.
F8B8 DD7702   01220 SKIP     LD      (IX+2),A        ;SAVE HIGH MANT. BYTE
F8BB 79       01230          LD      A,C             ;GET EXPONENT BYTE
F8BC 07       01240          RLCA                    ;ROTATE LEFT ONE BIT
F8BD CB2F     01250          SRA     A               ;DUPLICATE HIGH BIT
F8BF C680     01260          ADD     A,128           ;MAKE EXCESS 128 FORM
F8C1 DD7703   01270          LD      (IX+3),A        ;SAVE EXPONENT BYTE
F8C4 DB00     01280          IN      A,(DATA)        ;GET NEXT RESULT BYTE
F8C6 DD7701   01290          LD      (IX+1),A        ;SAVE IN RESULT AREA
F8C9 DB00     01300          IN      A,(DATA)        ;GET NEXT RESULT BYTE
F8CB DD7700   01310          LD      (IX),A          ;SAVE IN RESULT AREA
F8CE C9       01320          RET                     ;DONE - RETURN
              01330 ;
              01340 ;*
              01350 ;*** SUBROUTINE CONVRT:
              01360 ;* CONVRT gets the Radio Shack floating point value
              01370 ;* pointed at by the IX register and converts it to
              01380 ;* the Intel 8231A format, and pushes it onto the 8231A
              01390 ;* stack.
F8CF DD7E03   01400 CONVRT   LD      A,(IX+3)        ;GET EXPONENT BYTE
F8D2 B7       01410          OR      A               ;CHECK FOR ZERO VALUE
F8D3 2009     01420          JR      NZ,CONV2        ;IF NOT, GO ON
F8D5 D300     01430          OUT     (DATA),A        ;ELSE, PUSH 00 BYTE
F8D7 D300     01440          OUT     (DATA),A        ;PUSH 00 BYTE
F8D9 D300     01450          OUT     (DATA),A        ;PUSH 00 BYTE
F8DB D300     01460          OUT     (DATA),A        ;PUSH 00 BYTE
F8DD C9       01470          RET                     ;DONE - RETURN
F8DE DD7E00   01480 CONV2    LD      A,(IX)          ;GET LOW ORDER BYTE
F8E1 D300     01490          OUT     (DATA),A        ;PUT INTO 8231A
F8E3 DD7E01   01500          LD      A,(IX+1)        ;GET NEXT ORDER BYTE
F8E6 D300     01510          OUT     (DATA),A        ;PUT INTO 8231A
F8E8 DD7E02   01520          LD      A,(IX+2)        ;GET HIGH MANTISSA BYTE
F8EB F680     01530          OR      80H             ;SET HIGH BIT
F8ED D300     01540          OUT     (DATA),A        ;PUT INTO 8231A
F8EF DD7E03   01550          LD      A,(IX+3)        ;GET EXPONENT BYTE
F8F2 D680     01560          SUB     128             ;PUT INTO REAL FORMAT
F8F4 4F       01570          LD      C,A             ;SAVE EXPONENT IN C REG
F8F5 07       01580          RLCA                    ;MOVE HIGH BIT INTO CARRY
F8F6 0600     01590          LD      B,00H           ;CLEAR THE B REGISTER
F8F8 CB18     01600          RR      B               ;PUT CARRY BIT INTO B REG
F8FA E680     01610          AND     80H             ;MASK LOW 7 ACC. BITS
F8FC A8       01620          XOR     B               ;CHECK FOR SAME BIT 7'S
F8FD C25EF9   01630          JP      NZ,ERROR2       ;IF NOT, OUT OF RANGE ERR
F900 79       01640          LD      A,C             ;ELSE, GET EXPONENT BACK
F901 E67F     01650          AND     7FH             ;CLEAR HIGH BIT
F903 DDCB027E 01660          BIT     7,(IX+2)        ;CHECK MANTISSA SIGN BIT
F907 2802     01670          JR      Z,NOCHG         ;IF POSITIVE, NO CHANGE
F909 CBFF     01680          SET     7,A             ;ELSE SET BIT FOR NEGATIV
F90B D300     01690 NOCHG    OUT     (DATA),A        ;PUT EXP. BYTE INTO 8231A
F90D C9       01700          RET                     ;DONE - RETURN
              01710 ;*
F90E 3E10     01720 ADD      LD      A,FADD          ;GET FLOATING ADD CMD BYT
F910 183E     01730          JR      EXECUT          ;TWO-VALUED OPERATION
F912 3E12     01740 MULT     LD      A,FMUL          ;GET MULTIPLY COMMAND BYT
F914 183A     01750          JR      EXECUT          ;TWO-VALUED OPERATION
F916 3E06     01760 INVCOS   LD      A,ACOS          ;GET INV COSINE CMD BYT
F918 1836     01770          JR      EXECUT          ;ONE-VALUED OPERATION
F91A 3E05     01780 INVSIN   LD      A,ASIN          ;GET INV SINE CMD BYT
F91C 1832     01790          JR      EXECUT          ;ONE-VALUED OPERATION
F91E 3E07     01800 INVTAN   LD      A,ATAN          ;GET INV TANGENT CMD BYTE
F920 182E     01810          JR      EXECUT          ;ONE-VALUED OPERATION
F922 3E15     01820 CHGSGN   LD      A,CHSF          ;GET CHANGE SIGN CMD BYTE
F924 182A     01830          JR      EXECUT          ;ONE-VALUED OPERATION
F926 3E03     01840 COSINE   LD      A,COS           ;GET COSINE COMMAND BYTE
F928 1826     01850          JR      EXECUT          ;ONE-VALUED OPERATION
F92A 3E0A     01860 EXPON    LD      A,EXP           ;GET EXPONENTIAL CMD BYTE
F92C 1822     01870          JR      EXECUT          ;ONE-VALUED OPERATION
F92E 3E13     01880 DIV      LD      A,FDIV          ;GET DIVIDE CMD BYTE
F930 181E     01890          JR      EXECUT          ;TWO-VALUED OPERATION
F932 3E11     01900 SUB      LD      A,FSUB          ;GET SUBTRACT CMD BYTE
F934 181A     01910          JR      EXECUT          ;TWO-VALUED OPERATION
F936 3E08     01920 COMLOG   LD      A,LOG           ;GET COMM LOG CMD BYTE
F938 1816     01930          JR      EXECUT          ;ONE-VALUED OPERATION
F93A 3E09     01940 NATLOG   LD      A,LN            ;GET NAT LOG CMD BYTE
F93C 1812     01950          JR      EXECUT          ;ONE-VALUED OPERATION
F93E 3E0B     01960 POWER    LD      A,PWR           ;GET POWER COMMAND BYTE
F940 180E     01970          JR      EXECUT          ;TWO-VALUED OPERATION
F942 3E02     01980 SINE     LD      A,SIN           ;GET SINE COMMAND BYTE
F944 180A     01990          JR      EXECUT          ;ONE-VALUED OPERATION
F946 3E01     02000 ROOT     LD      A,SQRT          ;GET SQUARE ROOT CMD BYTE
```

Listing 4 continued

mat, pushes them onto the 8231A internal stack, gives the 8231A command to start the desired operation, then processes and stores the result. It also prints appropriate error messages. If any calculation errors occur, the program displays a general Calculation Error message. I did not add the code to display the particular problem (for example, divide by zero), but you can add this since the 8231A provides specific error flags.

The Basic program indicates its choice of function by the parameter passed with the USR function. The top of Listing 2 shows the various commands and operations available. The USR function in Listing 1 is passing the value 12, which indicates the power function (raise A to the B power).

You can run some Basic benchmarks with this method, simply by running the program in Listing 1 as is (with a loop to line 5000), and with line 70 replaced with the Basic math replacement (for example, C = A*B). I did this for several functions and found only small improvement for Basic functions (5 to 9 seconds for add, subtract, multiply, and divide), but a marked improvement for more complex functions (sine, cosine, tangent, powers, etc.). For example, the power function dropped time from 5:04 (all times are in minutes:seconds) to 1:02, the sin function went from 2:49 to 0:55, and the tan function went from 5:23 to 0:56!

The program in Listing 2, as mentioned, provides the necessary interface to the math processor. The CONVRT routine beginning at line 910 converts a TRS-80-formatted variable, pointed to by the IX register, into the proper 8231A format and stores the value in the 8231A processor. The Result routine, beginning at line 1890, gets the result from the 8231A and converts it to TRS-80 format and stores it in the proper location as specified by the DEST variable at 0F804 hex.

The problem with the above method is that there is too much Basic overhead. While times are great for a lot of calculations after the program sets up the variable pointers, there is too much time required to change the variable pointers each time you want a new operation with new variables. The Basic overhead outweighs the advan-

★ FREE SHIPPING ★
WITHIN CONTINENTAL 48 STATES
ON PREPAID CASH & COD ORDERS ONLY.

TOLL FREE ORDER LINE
1-800-223-1182
ORDERS ONLY
New York Orders & Information
Call (212) 445-7124

MICRO IMAGES INDUSTRIES INC.

TRY US!

STOP & SHOP
☐ BOOKS ☐ SOFTWARE
☐ PRINTERS ☐ RIBBONS
☐ COMPUTERS ☐ ACCESSORIES
☐ SPOOLERS ☐ MODEMS
☐ DISKETTES ☐ MONITORS

VISA COD MasterCard

FREE RIBBONS
Receive a FREE RIBBON when you place an order over $100.00 and mention this ad. Specify which cartridge or reload you require when ordering. Offer includes all reloads and cartridges listed except MX100, ANADEK and CENTRONIC cartridges. Offer good Jan. 31, 1984. Limit one per order.

$124.95 DATAWRITER
NEW VERSION
SPECIFY MDL I or III

$89.95 COPYART II
MODEL I or III
SPECIFY PRINTER

$89.95 MULTIDOS
SPECIFY MDL I or III
SINGLE OR DOUBLE DENSITY

$139.95 LAZYWRITER
MDL I, III or IV
SEE LAZY OPTIONS BELOW

NEW SCRIPT 7.1 $109.95
with Mailing Label Option . . $119.95
Mailing Label Option Only . . . 27.95
See Prosoft Options Below

MAXI CRAS Model I/III **$79.95**

MAXI MGR. Model I/III **$99.95**

SUPERUTILITY +
Mod I/III Ver. 3.2 $59.95
See New Book Below

DOSPLUS Ver. 3.5 Model I or III **$119.95**

MZAL Rel. III Model I or III **$84.95**

LDOS Model I or III **$114.95**

DOTWRITER 3.0 **$69.95**

SNAPP WARE 10% Off

DOSPLUSII Model II **$199.95**

ELECTRIC WEBSTER w/corr **$129.95**

ZORLOF Model I or III **$64.95**

PROSOFT UTILITIES
TRASHMAN - Model I or III	$34.95
FASTER - Model I or III	26.95
RPM - Model I or III	21.95

LAZY OPTIONS
LAZY FONT (Epson)	44.95
LAZY MERGE	39.95
LAZY DOC	49.95
LAZY Draw, Do, Tab, Calc	34.95

MAIL LIST PROGRAMS
POWER MAIL PLUS-Mdl 2/12/16	129.95
POWER MAIL PLUS-Mdl I/III	129.95
POSTMAN MASS MAIL-Mdl I/III	74.95
MAXI MAIL-Model III	79.95
TALLYMASTER - Model I or III	69.95
MICROTERM - Model I or III	69.95
TZAL — Model III Tape	39.95
BASIC EDITOR - Model I/III Tape	24.95

MAS 80 Acctg System
Integrated G/L, A/R, A/P & Ck Reg.
Retail $599.00 ***OUR PRICE $499.00***
Overview — $5.00 Credited on Purchase

BOOKS
How To Do It on the TRS 80 IJG	27.95
Machine Lang. Disk I/O IJG	27.95
TRSDOS 2.3 Decoded IJG	27.95
Disk + Other Mysteries IJG	20.95
Basic Decoded IJG	27.95
Custom TRS-80 IJG	27.95
Basic Faster & Better IJG	27.95
Model II/16 Visicalc W C Brown	16.95
Model I/III Visicalc W C Brown	16.95
BFBLIB or BFBDEM Disk	17.95
Inside Superutility Plus 2.2 or 3.0	17.95
Electric Pencil Manual	22.95
CPM Users Guide	15.95

SPECIAL
LNW 80 Model II
With Software Bonanza!
Includes: LNW Accounting Series - G/L, A/P, A/R & Payroll, Electric Spread Sheet, Electric Pencil, Micro term, Chart ex, CPM 2.2, DOSPLUS & LNW Basic
RETAIL $1995 ***OUR PRICE $1695.00***

LNW SYSTEM EXPANSION II	349.95
LNW 5/8 DOUBLER	169.95

PRINTER STANDS

SPACE AGE		NO FRILLS	
MX80 No Frills		13.50 Bronze	15.95
MX80 Space Age		27.50 Bronze	29.95
Microline 82A Space Age	Wide	27.50 Bronze	29.95
Microline 83A Space Age	Wide	34.95 Bronze	39.95
NEC 8023A Space Age		27.50 Bronze	29.95
MX100 Space Age		34.95 Bronze	39.95
Multiuse - 18x12x6x¼		24.95 Bronze	N/A

MICROBUFFER Parallel or Serial (Epson) . . . 139.95
GEMINI 10X . . . 329.95
MICRO SPOOLER By Consolink 16K . . . 199.95
64K . . . 269.95
HAYES SMARTMODEM 300 . . . 229.95
HAYES SMARTMODEM 1200/300 . . . 579.95
LYNX AUTO DIAL/ANSWER MODEL I/III . . . 239.95

RIBBONS
ZIP BOX RELOADS

	½ Doz.	Doz.
Epson MX 70/80-20 Yds.	24.00	42.00
Epson MX 100-30 Yds	30.00	52.00
NEC/Prowriter-14 Yds	21.00	36.00
Centronics 730/737/739/779 or LP-I/II/IV-16 Yds	18.00	32.00

All ZIP BOXES are individually sealed black nylon and require no rewinding. Epson Reloads also available in red, blue, brown, green & purple. Any mix allowed.

CARTRIDGES

	Each	Doz.
Epson MX70/80	7.00	70.00
Epson MX100	12.00	125.00
Prowriter 8510 & NEC 8023A	7.50	80.00
RS LP III/V	6.50	70.00
RS LP VI/VIII	6.50	65.00
RS DSY WH II or DWP 410	6.50	70.00
RS DSY WH II - Nylon	6.50	70.00
MICROLINE 80/82A/83A/92	N/A	30.00
MICROLINE 84 ½ x 40 Yds	5.50	60.00
ANADEK - 9000 Series	13.00	135.00
Diablo Hytype II-Multi Strike	6.50	65.00
Qume-Multi Strike	5.00	50.00
NEC-5500/7700-Multi Strike	7.00	70.00
Centronic 703/04/53	11.00	120.00

Note: All cartridges black only
Minimum order 3 cartridges - any mix. For smaller quantities, add $1.50 per order. All our reloads and cartridges are manufactured by one of the oldest and most reputable ribbon manufacturers in the country.
—QUALITY GUARANTEED—

FOR THE CO CO
Sea Dragon - 32K Tape	28.95
Rearguard - 16K Tape	19.95
Diskey - 32K Disk	39.95
Eliminator - 16K Tape	19.95
Triad - 32K Tape	28.95
Earthquake - 16K Tape	19.95

Micro Images Industries Inc.
164-06 Crocheron Ave. Dept. L
Flushing, N.Y. 11358
(212) 445-7124
HOURS: Mon.-Thurs. 10AM-6PM; Fri. & Sat. 10AM-5PM

VISA MasterCard

ORDERING INFORMATION
COD Orders add $2.00. Minimum credit card order $25.00. FREE shipping on all COD and prepaid cash orders within the continental 48 states via UPS Ground. Actual shipping and insurance charges apply on all credit card orders, approved P/O's and shipments outside the continental 48 states. Certified Ck, M/O, COD and credit cards shipped immediately. Please allow 2 weeks for personal checks. Credit card orders shipped to card address only. Prices subject to change without notice. New York State residents please add appropriate sales tax.

Listing 4 continued

```
F948 1806     02010          JR      EXECUT          ;ONE-VALUE OPERATION
F94A 3E04     02020 TANGNT   LD      A,TAN           ;GET TANGENT COMMAND BYTE
F94C 1802     02030          JR      EXECUT          ;ONE-VALUED OPERATION
F94E 3E1A     02040 PI       LD      A,PUPI          ;GET PUSH PI COMMAND
F950 D301     02050 EXECUT   OUT     (COMAND),A      ;SEND COMMAND TO 8231A
F952 DB01     02060 LOOP1    IN      A,(STATUS)      ;CHECK FOR COMPLETION
F954 CB7F     02070          BIT     7,A             ;STILL BUSY?
F956 20FA     02080          JR      NZ,LOOP1        ;IF YES, CHECK AGAIN
F958 E61E     02090          AND     1EH             ;CHECK FOR ANY ERROR
F95A C264F9   02100          JP      NZ,ERROR3       ;IF YES, CALC. ERROR
F95D C9       02110          RET                     ;DONE - RETURN
              02120 ;*
              02130 ;*** ERROR ROUTINES:
F95E E1       02140 ERROR2   POP     HL              ;POP ADDR FROM STACK
F95F 2171F9   02150          LD      HL,ERMSG2       ;POINT TO ERR MSG #2
F962 1804     02160          JR      PRRTN           ;PRINT AND RETURN
F964 E1       02170 ERROR3   POP     HL              ;POP ADDR FROM STACK
F965 2194F9   02180          LD      HL,ERMSG3       ;POINT TO ERR MSG #3
F968 7E       02190 PRRTN    LD      A,(HL)          ;GET CHARACTER
F969 B7       02200          OR      A               ;CHECK FOR ZERO BYTE
F96A C8       02210          RET     Z               ;IF ZERO, DONE
F96B CD3A03   02220          CALL    PRCHAR          ;ELSE, PRINT THE CHAR.
F96E 23       02230          INC     HL              ;POINT TO NEXT CHAR.
F96F 18F7     02240          JR      PRRTN           ;LOOP AGAIN
              02250 ;
              02260 ;
F971 0D       02270 ERMSG2   DEFB    0DH
F972 2A       02280          DEFM    '*** PARAMETER VALUE OUT OF RANGE'
F992 0D       02290          DEFB    0DH
F993 00       02300          DEFB    00H
F994 0D       02310 ERMSG3   DEFB    0DH
F995 2A       02320          DEFM    '*** CALCULATION ERROR'
F9AA 0D       02330          DEFB    0DH
F9AB 00       02340          DEFB    00H
              02350 ;
F9AC 00       02360 CONST2   DEFB    00H             ;CONSTANT VALUE 2 BYTES
F9AD 00       02370          DEFB    00H
F9AE 00       02380          DEFB    00H
F9AF 82       02390          DEFB    82H
F9B0 00       02400 CONST4   DEFB    00H             ;CONSTANT VALUE 4 BYTES
F9B1 00       02410          DEFB    00H
F9B2 00       02420          DEFB    00H
```

Listing 4 continued

tage of using the hardware math processor in many applications.

A better solution is to execute an entire equation at a time by the math processor before returning from the USR call. This eliminates the overhead problem and speeds up execution significantly.

Program Listing 3 shows a short Basic program with a reasonably complex equation in line 70. I have set the program to execute the equation 1,000 times. My computer executed the program in 7:45. I then created a significantly modified version of the Assembly-language program in Listing 2 (see Program Listing 4). This program executes the same equation by obtaining the various Basic variables (in the same way as in Listing 2) and calling the various math processor functions. The USR routine calls this program from Basic (see Program Listing 5) and executes the desired equation. The program ignores the parameter passed in the USR function, although it could be used to select one of several possible equations. Notice that we are now

passing eight variable addresses, instead of three.

When I executed the Basic program in Listing 5, calling the Assembly-language program in Listing 4 1,000 times, the processing took only 25 seconds—only 1/19 of the original software-based execution time. ■

Parts for this month's project are available from DIGI-KEY Corp., P.O. Box 677, Thief River Falls, MN 56701; Radio Shack, National Parts Division, 900 East Northside Drive, Fort Worth, TX 76102; Washtenaw Digital Systems, P.O. Box 2014, Ann Arbor, MI 48106. (Add $2 per order shipping and handling for the math processor; Michigan residents, add 4 percent sales tax.)

For further information send your questions and a self-addressed stamped envelope to Roger C. Alford, c/o Washtenaw Digital Systems, P.O. Box 2014, Ann Arbor, MI 48106, or call him between 7 and 9 p.m. weeknights at 313-973-9763.

Listing 4 continued

```
F9B3 83        02430          DEFB    83H
F9B4 00        02440 CONST7   DEFB    00H              ;CONSTANT VALUE 7 BYTES
F9B5 00        02450          DEFB    00H
F9B6 3C        02460          DEFB    3CH
F9B7 83        02470          DEFB    83H
               02480 ;
F810           02490          END     MATH
```

```
20 DEFUSR0=&HF810
40 A=0:B=0:C=0:D=0:E=0:F=0:G=1:Z=0
50 V=VARPTR(A):GOSUB170
60 V=VARPTR(B):GOSUB180
70 V=VARPTR(C):GOSUB190
80 V=VARPTR(D):GOSUB200
90 V=VARPTR(E):GOSUB210
100 V=VARPTR(F):GOSUB220
110 V=VARPTR(G):GOSUB230
120 V=VARPTR(Z):GOSUB240
130 FORI=1TO1000
140 X=USR(0)
150 NEXTI
160 PRINTZ:END
170 V1=INT(V/256):POKE-2047,V1:POKE-2048,V-V1*256:RETURN
180 V1=INT(V/256):POKE-2045,V1:POKE-2046,V-V1*256:RETURN
190 V1=INT(V/256):POKE-2043,V1:POKE-2044,V-V1*256:RETURN
200 V1=INT(V/256):POKE-2041,V1:POKE-2042,V-V1*256:RETURN
210 V1=INT(V/256):POKE-2039,V1:POKE-2040,V-V1*256:RETURN
220 V1=INT(V/256):POKE-2037,V1:POKE-2038,V-V1*256:RETURN
230 V1=INT(V/256):POKE-2035,V1:POKE-2036,V-V1*256:RETURN
240 V1=INT(V/256):POKE-2033,V1:POKE-2034,V-V1*256:RETURN
```

Program Listing 5. USR routine to call program from Basic.

An Overworked Mercedes And Sad Max

The Mercedes Mach 4 microcomputer sits six stories below the streets of Cambridge in a secret bunker with 10-foot-thick, steel-reinforced concrete walls. A master cable 3 feet in diameter runs to a lead-lined control room. The computer is handled using video monitors and robot arms.

The Mach 4 has a duodecagigabyte of internal CRAM (Catalytic Random Atomic Memory), and runs at 3,000 MHz. When the microprocessor is crunching data, clocks in the immediate area run backwards.

Mercedes is proud of her achievement, and calls it her "micro opus."

"Not bad for a Ph.D. project, hey?" she said to Max and me in the MIT cafeteria.

"It makes the Cray look like a moped," I said. "But what's anybody going to do with it? I mean, you can't even take it outside that bunker."

I was referring to the fact that the government had blocked the Mach 4's early tests in the Nevada desert, claiming it would violate the SALT treaty.

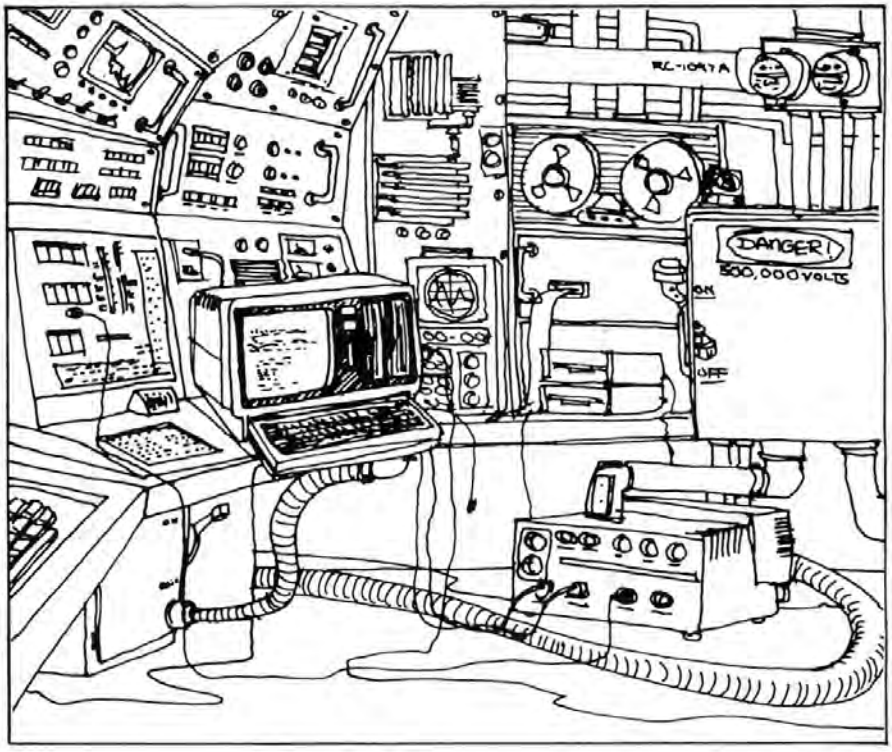
Mercedes shrugged. "Who cares? The world is full of useless gadgets. Another one won't hurt anybody."

Max and I looked at each other. We were not used to such cavalier comments from Mercedes, but they had been coming more frequently since she took the fellowship and began working on the Mach 4.

"It's just a stage," her adviser had told us. "All 11-year-old geniuses go through it."

"Well," she said, slurping down the last of her milk. "What say we start working on the operating system, Bernie?"

Bernie Washington, her student assistant, blinked behind his nearly opaque glasses. He was a computer science major, and the great-great-great-great grand-nephew of George. "Uh—sure," he said. "Uh—what do you want me to do?"



"Just get out a pad of paper and transcribe," she said. "Okay, we'll start with an ORG at 7EC0 hex."

So while Mercedes began dictating the 1 MByte program that would drive the Mach 4, Max and I turned to some mail.

*****

"You got a letter from Mr. Arcade, the King of the Arcadians," I said. "He says, 'It seems to me, Sad Max—'"

"Sad Max?!" Max growled.

"'—that someone who travels around with an 11-year-old girl and a group of misfits is nothing more than a loser.'"

"Eleven and a half. Twelve, almost," Mercedes broke in.

"'But I am giving you a chance to prove yourself. This is the challenge. You may pick any game you want, either on the market or self-written, or written by one of your comrades, and play this game until you have a high score you think can stand up to ME. (You cannot have played this game before.) Then I will play the game, and if I can't beat your score within one week, you will be King of the Arcade.

"'If I don't receive a reply in person and in your column I will assume you are chicken.'"

"Hey, I don't have to prove myself to some two-bit joystick junkie from nowhere," Max clucked. "My Bable Terror score's been on the board from the beginning."

"Easy," I said. "Here's another guy from Vermilion, OH, who says he's the original Mad Max—Mad Max Kennedy."

"Everybody's out to get you when you're number one," Max sighed.

"There's only one Mad Max," Mercedes said. "My Mad Max."

Max looked embarrassed and blushed.

PERRY COMPUTERS

CALL TOLL FREE 1-800-248-3823

TRS-80 COMPUTERS

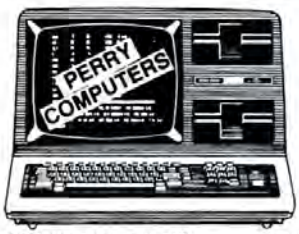
100% PURE RADIO SHACK

COLOR COMPUTERS

	LIST PRICE	OUR PRICE
26-3003 64K Color Computer	$ 399.00	$ 295.00
26-3026 16K Color II Non Extended Basic	$ 219.95	$ 149.00
26-3027 16K Color II Extended Basic	$ 319.95	$ 210.00
26-3022 O Drive Color Computer	$ 399.95	$ 310.00
26-3023 1,2,3 Color Drives	$ 279.95	$ 230.00
26-3029 O Drive Color Computer II	$ 399.95	$ 310.00
26-3025 Color Mouse	$ 49.95	$ 42.50
26-3030 OS-9 With Editor Assembler	$ 69.95	$ 59.50
26-3008 Joysticks (Pair)	$ 24.95	$ 21.00

MODEL 100 COMPUTERS

	LIST PRICE	OUR PRICE
26-3801 8K Model 100 Computer	$ 799.00	$ 650.00
26-3802 24K Model 100 Computer	$ 999.00	$ 790.00
26-3804 AC Adaptor	$ 5.95	$ 5.00
26-3805 Accoustic Coupler	$ 39.95	$ 34.00
26-3816 8K Memory Expansion	$ 119.95	$ 95.00
26-1409 Printer Cable	$ 14.95	$ 12.70
26-1410 Modem Cable	$ 19.95	$ 17.00
26-3809 Briefcase	$ 49.95	$ 42.50
26-1183 Bar Code Reader	$ 99.95	$ 85.00



MODEL 4 COMPUTERS

	LIST PRICE	OUR PRICE
26-1067 Model 4 16K Computer	$ 999.00	$ 685.00
26-1068 Model 4 64K 1 Drive Computer	$1699.00	$1400.00
26-1069 Model 4 64K 2 Drive Computer	$1999.00	$1585.00
26-1122 64K Memory Expansion	$ 149.00	$ 125.00
26-1127 0 Drive Model 4 Computer	$ 649.00	$ 550.00
26-1163 1 Drive Model 4 Computer	$ 239.00	$ 205.00
26-1130 5 Meg Hard Drive Primary	$1999.00	$1680.00
26-1131 5 Meg Hard Drive Secondary	$1799.00	$1549.00
26-1123 Model 3 to Model 4 Upgrade	$ 799.00	$ 675.00

MODEL 12 & 16 COMPUTERS

	LIST PRICE	OUR PRICE
26-4004 Model 12 80K 1 Drive Computer	$3199.00	$2575.00
26-4005 Model 12 80K 2 Drive Computer	$3999.00	$3150.00
26-6004 Model 16B 256K 1 Drive Computer	$4999.00	$4000.00
26-6005 Model 16B 256K 2 Drive Computer	$5798.00	$4510.00
26-6006 Model 16B 256K 15Meg HD Computer	$6999.00	$5950.00
26-6010 Model 12 To Model 16 Upgrade	$1499.00	$1275.00
26-4152 12 Meg Hard Drive Primary	$2995.00	$2550.00
26-4153 12 Meg Hard Drive Secondary	$1995.00	$1990.00

RADIO SHACK PRINTERS

	LIST PRICE	OUR PRICE
26-1250 DWP 410 Daisy Wheel Printer	$1295.00	$1100.00
26-1252 DMP 500 Dot Matrix Printer	$1295.00	$1100.00
26-1253 DMP 100 Dot Matrix Printer	$ 399.00	$ 300.00
26-1254 DMP 200 Dot Matrix Printer	$ 699.00	$ 575.00
26-1255 DMP 120 Dot Matrix Printer	$ 499.00	$ 400.00
26-1256 DMP 2100 Dot Matrix Printer	$1995.00	$1600.00
26-1257 DWP 210 Daisy Wheel Printer	$ 799.00	$ 660.00
26-1158 DW II Daisy Wheel Printer	$1995.00	$1600.00
26-1267 DMP 420 Dot Matrix Printer	$ 999.00	$ 850.00

PRINTERS, SOFTWARE & ACCESSORIES

Epson Printers	$ CALL
Okidata Printers	$ CALL
Gemini 10X Printer	$ 300.00
Gemini 15 Printer	$ 465.00
C. Itoh 8510 Printer	$ 400.00
C. Itoh 1550 Printer	$ 670.00
C. Itoh F10 40 cps	$1300.00
C. Itoh F10 55 cps	$1600.00
Hayes Modems	$ CALL
All Radio Shack Software	15% OFF

We have been in the Computer Business for over 5 years. Our Goal is to satisfy you the customer. Please write or call (517) 625-4161 for free price list.

Complete line of computer accessories. Please call for current prices.

PERRY COMPUTERS

FOR ORDERS CALL 1-800-248-3823

Dept No A-1 137 NORTH MAIN ST PERRY, MI 48872

124

FOR INFORMATION CALL (517) 625-4161

THE GAMER'S CAFE

The Big Board

Alien Cresta	6,913	Mike Bach, Brisbane, Australia
Alien Taxi	20,300	Dave Westfall, N. Olmsted, OH
Apple Panic	287,620	Mary Phinney, Stockbridge, MI
Arex	875,030	Rob Mitchell, Peterborough, NH
Assault	97,457	Zagros Sadjadi, Petaluma, CA
Astroball	317,240	Stefan Kunze, Moers, W. Germany
Attack Force	1,732,820	Dave Smith, Raleigh, NC
Bable Terror	8,857	Mad Max
Barricade	17,520	Troy Scrapchansky, Uncasville, CT
Caterpillar	362,883	Alvah Werner, New Albany, OH
Centipedes	94,836	Belinda Chron, Temple, AZ
Chicken	12,035	Noble Chowchuvech, Demarest, NJ
Clash	174,300	Zagros Sadjadi, Petaluma, CA
Convoy	34,770	Rick Sayre, Stockton, CA
Crazy Painter	1,087,000	Mike Beebe, Sacramento, CA
Cyborg	317,000	Robert Cavin, Laredo, TX
Danger in Orbit	69,640	Steve Sustacek, Danube, MN
Defence Penetrator	21,310	Glenn Butler, Belrose, N.S.W.
Defense Command	128,230	Bette Dufraine, Bolton, CT
Demise/Defend	165,000	David Russell, Ardrossan, Scotland
Demon Seed	103,160	Markus Blum, Ludwigshafen, W. Germany
Desert Peril	84,400	Jay McLain, Clatskanie, OR
Devil's Tower	25,700	Rick Sayre, Stockton, CA
Dungeon Escape	6,531	Donald Tindall, Littleton, CO
Firebird	185,000	Jorg Janischewski, Engelskirchen, W. Germany
Flying Saucers	2,190	Richard Stokes, Burbank, IL
Fortress	515,925	Greg Samson, Loudonville, NY
Frogger	400,900	Shawn Roberts, Oklahoma City, OK
Fury	46,120	Amy Campbell, Peterborough, NH
Galactic Empire	2,010	Mike Bach, Brisbane, Australia
Galaxy Invasion Plus	3,000,000	Shawn Lipman, Nelspruit, S. Africa
Gauntlet	58,360	David Schwartz, San Jose, CA
Ghost Hunter	43,190	Lance Smith, Auckland, N.Z.
Gobbleman	64,310	Mick Bach, Brisbane, Australia
Hamburger Sam	34,300	Mark Adams, Tampa, FL
Hoppy	70,381	Mike Bach, Brisbane, Australia
Insect Frenzy	691,156	Tommy Seniuk, Vegreville, Alta.
Invaders from Space	655,360	Darren Cotter, Oceanside, CA
Jovian	311,320	Mark Brinkman, Burlington, KS
Jungle Boy	851,900	Zagros Sadjadi, Petaluma, CA
Killer Gorilla	28,312	Alex Poon, Baton Rouge, LA
Laserball	72,530	Neil Matson, Panama City, FL
Laser Defense	1,504,610	Greg Samson, Loudonville, NY
Leaper	144,500	Tommy Seniuk, Vegreville, Alta.
Lunar Lander	18,000	Graham Williams, Ballaarat, Australia
Mad Mines	10,220	Gorman Miller, Titusville, FL
Meteor Mission 2	124,990	Andy Anderson, Orangeville, Ont.
Monster Invaders	32,620	Troy Scrapchansky, Uncasville, CT
Olympic Decathlon	10,856	Adrie van Geffen, Hoogvliet, Netherlands
Outhouse	1,000,000	Kyle Hoyt, Titusville, FL
Panik	85,075	Mark Owens, Houston, TX
Penguin	39,250	Mark Adams, Tampa, FL
Planetoids	56,450	Carl Pflanzer, Gillette, NJ
Rear Guard	195,240	John Hope, Kingston, Ont.
Rescue at Rigel	456	Glenn Butler, Belrose, N.S.W.
Sea Dragon	610,180*	Robert Fitzwilliam, Houston, TX
Sky Sweep	1,000,540	Tommy Seniuk, Vegreville, Atla.
Space Castle	69,750	Rick Sayre, Stockton, CA
Space Intruders	14,030	Ron Johnston, Emporia, KS
Space Rocks	32,600	John Dunkelberg Jr., Winston-Salem, NC
Star Blazer	52,750	Mark Adams, Tampa, FL
Stellar Escort	625,000	Kevin Josephson, Chilliwack, B.C.
Super Nova	2,138,710	Mark Fertig, Northville, MI
Swamp Wars	59,130	Farhad Abrishami, Silver Spring, MD
Temple of Apshai	390	Carl Pflanzer, Gillette, NJ
Time Bandit	14,460	Mark Adams, Tampa, FL
Time Runner	89,479	Mad Max
Voyager 1	1,177	Al Ayer, Mechanicsburg, PA
Weerd	61,180	Tommy Seniuk, Vegreville, Alta.
Wild West	47,000	Dave Westfall, N. Olmsted, OH

*Expert mode: 339,080 (David Smith, Kingwood, TX).

"We've got a new score here," I said, changing the subject. "Mark Brinkman of Burlington, KS, racked up 311,320 in Jovian on Antares sector 79. He also wants to know whether others have succeeded in Jovian's other galaxies, Belthix and Centuri, and whether anyone has finished Centuri's sector 18."

Max, meanwhile, was huffing over a letter from Richard Stokes of Burbank, IL, who set a new record for Flying Saucers.

"He asks why we don't print a program that makes Air Supply and Billy Joel songs come out of the tape recorder," he said. "And he wants to ask Mercedes out on a date. 'I'm 12 years old (born June 16, 1971), very cute and mature. If she says no then ask her again, only say please.'"

Mercedes looked flustered. "Cute and mature and he listens to Air Supply? Get serious. Besides, he's a Gemini. He'd probably change his mind and leave me for some red-haired flashdance floozy."

"Twelve years old and he's dating?" Max cried in protective, fatherly fashion. "Not our girl, he isn't. I didn't date until I was 27."

"Twenty-seven?" I asked incredulously.

"Well, not dating per se," Max said.

* * * * *

Mercedes spent three solid weeks dictating the Mach 4 operating system. She worked 16 hours a day and lived on peanut butter sandwiches and Kool-Aid. She went through student assistants like pencils—Bernie burned out after one day and was succeeded by 11 others, none of whom lasted longer than 48 hours.

Max and I spent the time hanging out in Cambridge, eating at Joyce Chen's and leafing through the magazines at the Harvard newsstand. It was a melancholy time; Max wanted to get back out on the road, and we were both worried about Mercedes.

The Mach 4 had become an all-consuming project. Mercedes rarely spoke to us, and when she did, it was in Assembly. As the days passed, her eyes took on an obsessed, wild, almost feral gleam.

"We need a change," said Max one grey winter day as we gazed across the murky Charles at the city skyline.

"Yes," I agreed. "But we can't do anything until Mercedes finishes."

"Will she ever finish?" he asked. "I mean, what happens when the Mach 4 project is done with? What will she do then? Will she start something even more monumental?"

It was a good question. Mercedes was growing up, but with no world to grow up in. She didn't belong in the world of the Gamer's Cafe. She didn't belong in the normal world of 11-year-olds. And she didn't belong in the world of adults. So she was creating one of her own, one made of circuits and chips and streams of code.

What happened to Mercedes was something neither of us was ready for. But it's a story that will have to wait until next month. ■

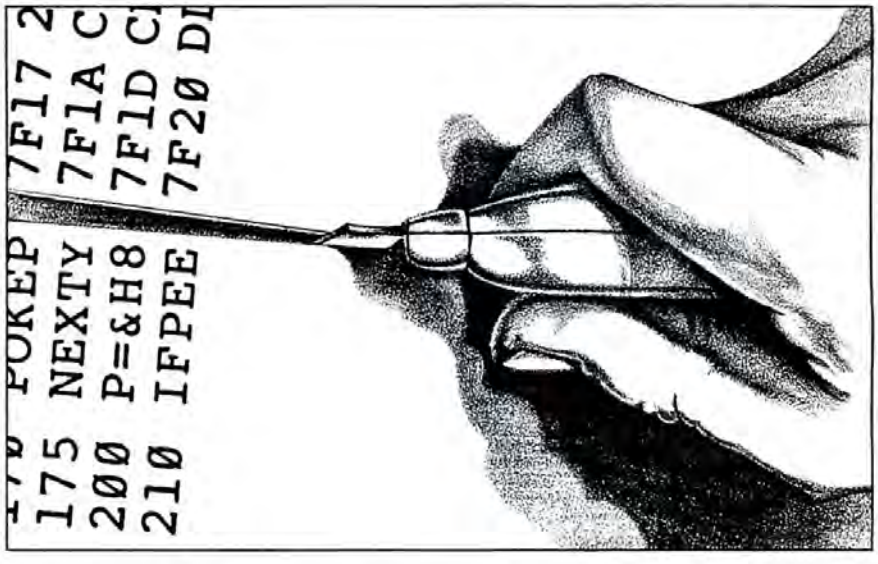
✓ See List of Advertisers on Page 211

ROM Routines Explained, Contest Winners Announced

I recently received a fantastic book in the mail—one that is quickly becoming a favorite reference manual. The book is *TRS-80 ROM Routines Documented,* by Jack Decker (Alternate Source, Lansing, MI, $19.95 + $3 s/h). It includes over 250 ROM routines and explanations of how to use them, a complete discussion of device control blocks, the RST commands, Disk Basic and DOS vectors, the differences between Model I (Level II), Model III, Model 4, and PMC ROMs, and many programming examples.

If you want to simplify your Assembly-language programs by using ROM routines, or just want to know how your TRS-80 works, I doubt you could find a better book anywhere.

When reading the book, I came across two interesting techniques that are the basis of the first half of this month's column. In the second half of the column, I'll announce the results of last October's great screen whiteout contest.



Adding Basic to Machine Language

If you regularly read this column, you should have a good grasp of the techniques of adding machine-language routines to Basic programs. During the last year, I've covered many of the normal techniques and some of the more abstruse ones. However, you may have run across some programming situations in which you'd like to do the opposite: add some lines of Basic to a machine-language program.

I use this technique to perform difficult number crunching or string manipulation. Assembly-language instructions could perform such tasks, but, when speed isn't important, it's easier to let Basic handle the work.

To make the Basic ROMs do the work, you can adopt one of three programming tactics: you can find the ROM routines that perform the necessary tasks and call them individually; if you are writing a USR routine, you can return to Basic momentarily and then enter a new USR routine to complete your task; or you can use what Jack Decker calls the "fudge-it" method of adding Basic to machine language. I'll describe this third method.

Suppose your machine-language routine needs to evaluate an expression such as:

```
X=VAL(LEFT$(A$,3))
```

A single ROM call can do it for you. First, enter Basic and type NEW. Then type:

```
10 VAL(LEFT$(A$,#))
```

Normally, that line causes a syntax error, since the results of the operation aren't returned in any variable. But you won't be running the program from Basic, only adding that function to your machine-language routine.

As soon as you press the enter key after typing line 10, the Basic ROMs take that line and tokenize it in memory, using single bytes to represent each function (like VAL and LEFT$ above). Now, without disturbing the tokenized form of the expression, you need to get it out of memory. The beginning of the Basic program is stored in memory locations 16548 and 16549, so enter the following to discover how Basic has tokenized the line:

```
X=PEEK(16548)+PEEK(16549)*256+4:
FOR Y=X TO X+100:PRINT PEEK(Y);:
NEXT Y
```

A list of numbers will appear—the decimal form of whatever is in memory. Write down the numbers up to and including the first zero value. For the example above, the numbers are:

```
245 40 248 40 65 36 44 51 41 41 0
```

Add that list of numbers as a string to your Assembly-language program, using DEFB statements. Remember, these are decimal values, not hexadecimal (hex) values. When your program needs to evaluate the expression, point the HL register to the string and CALL 2337 hex. The program evaluates the expression and places the result in Basic's accumulator between 411D hex and 4124 hex. Also, the program sets Basic's number type flag to 02, 03, 04, or 08 to indicate

THE WAIT IS OVER

CP/M®

EXCLUSIVELY FOR THE

TRS-80®

Model 4

Now, for the first time, unleash the powerful features resident in your Model 4 computer. Open up the vast store of CP/M software such as WordStar®, dBASE II and Multiplan™, along with thousands of others.

- Includes INTERCHANGE ™, a utility that allows reading, writing and copying 20 different manufacturers' disk formats such as IBM, KAYPRO, OSBORNE, XEROX, etc.
- Includes MEMLINK™, a unique feature that uses the optional 64K RAM memory as a fast disk drive.
- Complete with all these CP/M utilities; ASM, DDT, DUMP, ED, LOAD, PIP, STAT and SYSGEN.
- Operates at the 4Mhz clock in the standard Model 4 mode.
- NO HARDWARE MODIFICATIONS. Just insert the disk and boot.
- NO COPY PROTECTION. Backups may be made for your own use and protection.
- The CONFIGURATION program supports a full range of 5-1/4" disk drives: 35, 40, 77 and 80 tracks, single and dual sided in any combination as well as the standard Model 4 drives.
- Includes MODEM 7, a powerful public domain communications program for file transfer and remote data base access such as CompuServ and the Source.
- Supports 80 x 24 video, reverse video, direct cursor addressing and more.
- Utilizes the Model 4 function keys and allows user defined keys.
- Auto Execute command for turnkey applications.
- FORMAT utility permits up to 52 disk formats to be constructed, all menu driven.
- Fast backup routine with verify for mirror image copies.
- All support programs are menu driven for ease of use.
- Ready to run in the standard 64K Model 4. The additional, extra cost, 64K RAM upgrade not required.
- Complete with over 250 pages of comprehensive user documentation.

AVAILABLE NOW FOR IMMEDIATE SHIPMENT.....$ 199.95

The full line of MicroPro software is now available formatted for the Model 4 using our CP/M. Each disk is already configured and ready to run. Just install the printer of your choice and go.

WordStar® Fast memory mapped version	$250	InfoStar™ Advanced DBMS	$250
MailMerge® Multi-purpose file merging program	125	ReportStar™ Report generator & file manipulator	175
SpellStar® 20,000 word proofreader on a disk	125	DataStar™ Data entry and retrieval package	150
StarIndex™ Creates index and table of contents	95	SuperSort® Fast and flexible sorting is yours	125
WordStar Professional. All the above for only	450	CalcStar™ Advanced electronic spreadsheet	95

ORDER INFORMATION

Call now and your order will be shipped at once from our Dallas warehouse. We accept American Express, MasterCard, Visa, and most any other form of payment known to man. Credit cards are not charged until your order is shipped. Add $4 UPS surface shipping and handling on orders within the 48 States. No State Sales Tax on software or shipments delivered outside of Texas. No refunds. Defective items are replaced upon return, postpaid.

ORDER NOW . . . TOLL FREE

800-527-0347

800-442-1310

The Toll Free lines are for orders only.
Specifications subject to change without notice.

128K MEMORY UPGRADE

Our upgrade includes 64K of 150nsec RAM, genuine PAL® chip and instructions for installation. This kit will upgrade your 64K Model 4 to 128K and allow the use of our MEMLINK and TRSDOS 6.x MEMDISK. Comes with a full 1 year guarantee.

A BARGAIN AT ONLY $99.95

CP/M is a Trademark of Digital Research, Inc.; Interchange and Memlink are Trademarks of Montezuma Micro; TRS-80 is a Trademark of the Tandy Corporation; WordStar, MailMerge, SpellStar, StarIndex, InfoStar, ReportStar, DataStar, SuperSort and CalcStar are Trademarks of MicroPro International Corporation. Multiplan is a Trademark of Microsoft.

AD #883

MONTEZUMA MICRO

214-339-5104
Redbird Airport, Hangar #18
P.O. Box 32027
Dallas, Texas 75232

✓ 411

VISA MasterCard AMERICAN EXPRESS

"WE KEEP YOU RUNNING"

```
                00100 ;*********************************
                00110 ;*                               *
                00120 ;*  Demonstration USR routine     *
                00130 ;*   that directly handles       *
                00140 ;*      string variables         *
                00150 ;*                               *
                00160 ;*  Changes all lowercase        *
                00170 ;*  characters in passed string  *
                00180 ;*  to uppercase, and returns    *
                00190 ;*  to new variable.             *
                00200 ;*                               *
                00210 ;*  From Basic, call with        *
                00220 ;*    X$ = USR (Y$)              *
                00230 ;*                               *
                00240 ;*********************************
                00250 ;
7000            00260          ORG     7000H           ;COMPLETELY RELOCATABLE
7000 CDF40A     00270          CALL    0AF4H           ;BE SURE STRING IS PASSED
7003 2A2141     00280          LD      HL,(4121H)      ;GET STRING ADDR. IN HL
7006 0100FF     00290          LD      BC,0FF00H       ;SET UP BC REGISTER
7009 CD682A     00300          CALL    2A68H           ;MOVE Y$ TO HIGH MEMORY
700C DD2A2141   00310          LD      IX,(4121H)      ;GET VARPTR OF Y$ IN IX
7010 DD4600     00320          LD      B,(IX+0)        ;GET STRING LENGTH
7013 DD6E01     00330          LD      L,(IX+1)        ;GET LSB OF STRING ADDR
7016 DD6602     00340          LD      H,(IX+2)        ;GET MSB OF STRING ADDR
                00350 ;
                00360 ;Now HL ==> Y$ and B contains LEN(Y$)
                00370 ;
7019 7E         00380 LOOP     LD      A,(HL)          ;GET CHARACTER
701A FE61       00390          CP      61H             ;IS IT => a ?
701C 3807       00400          JR      C,L10           ;GO IF NOT
701E FE7B       00410          CP      7BH             ;IS IT ALPHABETIC?
7020 3003       00420          JR      NC,L10          ;GO IF NOT
7022 D620       00430          SUB     20H             ;MAKE IT UPPERCASE
7024 77         00440          LD      (HL),A          ;AND STORE IT BACK
7025 23         00450 L10      INC     HL              ;BUMP POINTER
7026 10F1       00460          DJNZ    LOOP            ;REPEAT FOR WHOLE STRING
7028 C9         00470          RET                     ;RETURN TO BASIC
                00480 ;
0000            00490          END
00000 TOTAL ERRORS
33912  TEXT AREA BYTES LEFT
```

Program Listing 1. This USR routine operates only on strings.

whether the result is an integer, string, single-precision, or double-precision value. The program further manipulates the value using other ROM routines or returns directly to Basic when it leaves the USR routine. If you want to convert the value to an integer and move it into the HL register pair (assuming it is numeric, not a string value), CALL 0A7F hex.

Be aware that this technique only works if Basic is initialized and active. On a tape-based system, that's any time the computer is turned on (unless a machine-language program has gone rampaging through Basic's reserved low memory). On a disk-based system, you are essentially limited to using this technique as part of a USR routine. And speaking of USR routines...

More Power from USR

The Radio Shack manuals, as well as others, give the impression that in the equation:

X = USR (Y)

Y must be in integer range (−32768 to +32767) if you wish to pass a value to your machine-language routine, and X will be in integer range if your routine returns a value. As Decker points out in *ROM Routines*, X and Y can be any types of variables or values: integer, single- or double-precision, or string variables. They don't even need to be of the same type!

When you follow the Radio Shack manual and begin your machine-language routine with CALL 0A7FH to get the value from Y into the HL register, you're perfoming a CINT function. When your routine ends with JP 0A9AH, it's loading the current value in HL into Basic's accumulator, marking it as an integer, and passing it to X.

The standard procedure works fine if you want to pass integers both ways, but you don't have to do this. Program Listing 1 is an example of a USR routine that operates solely on strings. Assuming you call the routine by the command:

X$ = USR (Y$)

the routine takes Y$ and changes all characters in it to uppercase, returning the new string as X$. Once you get used to the technique, it's no more difficult to use than any other USR routine.

Line 270 of Listing 1 calls 0AF4 hex to test the value that the program passes to the routine and places in Basic's accumulator. If the program passes something other than a string, control returns to Basic with a Type Mismatch error. Assuming the test is passed, the next three instructions copy Y$ to a new location in the string space in high memory. First, the program loads HL with the VARPTR of Y$, which Basic has stored in its accumulator at 4121 hex. Then the program loads B with 0FF hex (255), the maximum length of the string, and clears C to zero. The CALL to 2A68 hex copies the string into the string space, and puts the VARPTR of the new string at 4121 hex.

Using this new VARPTR, lines 310–340 transfer the address of the string into HL and its length into B. The instructions in the loop check each character to determine whether it is in lowercase (between 61 hex and 7A hex). If it is, the program changes the character to uppercase by subtracting 20 hex from its value. Otherwise, it doesn't affect the character.

At the end of the loop, the new string, whose VARPTR is still at 4121 hex, is all uppercase. The RET instruction exits the USR routine and places whatever is in the accumulator in X$. Because the VARPTR is in the accumulator and Basic's type flag is already set to 03, indicating a string, X$ becomes the new string.

That last step may need more explanation. When your USR routine returns to Basic, the program places whatever is in Basic's accumulator in the variable on the left of the equals sign. If the variable is of the wrong type, the program stops processing and generates an error. Basic always maintains a flag at 40AF hex indicating the type of the value in the accumulator: 02 indicates an integer, 03 indicates a string, 04 indicates a single-precision value, and 08 indicates a double-precision value.

Those flag values also represent the number of bytes required to store the value in either the simple or array variable table. Since the short routine has already created a new string with a VARPTR in the accumulator, the return equates that string with X$.

By judicious use of other ROM routines, you can just as easily pass single- or double-precision values between

NOCONA

ELECTRONICS

MODEL 4 **MODEL 16** **MODEL 12**

66K COLOR COMPUTER	299.00
16K MODEL 4	799.00
64K MODEL 4 2 DR RS 232	1579.00
64K MODEL 4 PORTABLE	1399.00
80K MODEL 12 1 DR	2499.00
80K MODEL 12 2 DR	3149.00
256K MODEL 16 1 DR	3898.00
256K MODEL 16 2 DR	4498.00
256K MODEL 16 1 DR W/HD	5439.00
8K MODEL 100 PORTABLE	649.00
24K MODEL 100 PORTABLE	799.00
12 MEG HD MOD 12/16	2369.00
5 MEG HD MODEL III/4	1579.00
1 DR. EXP. MOD 12/16	1049.00

DMP 100 PRINTER	299.00
DMP 120 PRINTER	399.00
DMP 200 PRINTER	499.00
DMP 420 PRINTER	699.00
DMP 500 PRINTER	1029.00
DMP 2100 PRINTER	1519.00
DWP II DAISY WHEEL	1499.00
DWP 210 DAISY WHEEL	599.00
DWP 410 DAISY WHEEL	1059.00
DWP 210 TRACTOR FEED	120.00
DWP 410 TRACTOR FEED	179.00
DWP II TRACTOR FEED	199.00
PARALLEL PRINTER SWITCH	96.00
62K PRINTER CONTROLLER	200.00

100% RS COMPONENTS NO FOREIGN DRIVES OR MEMORY FULL WARRANTY
ALL RS SOFTWARE 20% OFF CATALOG PRICE
CASHIERS CHECK OR MONEY ORDER MUST ACCOMPANY ALL ORDERS

(817) 825-4027

NOCONA ELECTRONICS • BOX 593 • NOCONA, TX 76255

232

See List of Advertisers on Page 211

```
                00100 ;****************************************
                00110 ;*                                      *
                00120 ;*     THE SHORTEST AND THE FASTEST     *
                00130 ;*                                      *
                00140 ;*      Winning routines from the       *
                00150 ;*       October Screen-Whiteout        *
                00160 ;*              Contest                 *
                00170 ;*                                      *
                00180 ;****************************************
                00190 ;
                00200 ;
                00210 ;****************************************
                00220 ;*                                      *
                00230 ;*      THE SHORTEST  (11 bytes)        *
                00240 ;*       Uses Register HL               *
                00250 ;*                                      *
                00260 ;****************************************
                00270 ;
0000 21003C     00280          LD      HL,3C00H        ;HL==> TOP OF SCREEN
0003 36BF       00290 LOOP     LD      (HL),0BFH       ;SEND ONE GRAPH. BLOCK
0005 23         00300          INC     HL              ;BUMP THE POINTER
0006 CB74       00310          BIT     6,H             ;CHECK IF H < 40H
0008 28F9       00320          JR      Z,LOOP          ;LOOP IF IT IS
000A C9         00330          RET                     ;RETURN TO CALLER
                00340 ;        End of routine
                00350 ;
                00360 ;****************************************
                00370 ;*                                      *
                00380 ;*           THE FASTEST                *
                00390 ;*        Uses registers -- HL, DE      *
                00400 ;*         5689 T-States                *
                00410 ;*   NOTE -- Disable Interrupts (DI)    *
                00420 ;*    before calling.                   *
                00430 ;*                                      *
                00440 ;****************************************
                00450 ;
000B 210000     00460          LD      HL,0000H        ;INITIALIZE HL TO 0
000E 39         00470          ADD     HL,SP           ;SAVE SP FOR RETURN
000F 310040     00480          LD      SP,4000H        ;SP==>BOTTOM OF SCREEN
0012 11BFBF     00490          LD      DE,0BFBFH       ;DE HAS 2 GRAPH. BLOCKS
0015 D5         00500          PUSH    DE              ;WHITE BOTTOM OF SCREEN
0016 D5         00510          PUSH    DE              ;  AND MOVE TOWARDS
0017 D5         00520          PUSH    DE              ;  TOP OF SCREEN
                00530 ;        .
                00540 ;        .                       ;512 PUSH DE instructions
                00550 ;        .                       ;   to fill screen
                00560 ;        .
0018 F9         00570          LD      SP,HL           ;RESTORE STACK POINTER
0019 C9         00580          RET                     ;RETURN TO CALLER
                00590 ;        End of routine
                00600 ;
NO END STATEMENT
00001 TOTAL ERRORS
33659  TEXT AREA BYTES LEFT
```

Program Listing 2. Winning subroutines.

Basic and USR routines. All necessary ROM routines are described in Decker's book. In future columns, I'll explore possible applications of some of the other routines.

Contest Results

In October, I challenged readers to find the shortest and fastest possible screen white-out subroutines. I was unprepared for the number of entries I received and surprised by the uniformity of the answers. The winning routines are shown in Program Listing 2. More than half the entries had one or both answers, though sometimes in slightly altered form.

Eleven bytes seems to be the shortest possible screen-white subroutine. (Notice the emphasis, both here and last October, on the word subroutine. Some readers tried to shorten their answers by omitting the RET instruction or making a return impossible.) The 11-byte answer is essentially the same as one of the routines that started the discussion, except that a BIT instruction makes the check for the end of the screen. I wasn't terribly surprised by this answer, and I kicked myself a bit for not having thought of it when I wrote the October column.

I *was* surprised by the "fastest" routine. A majority of those who submitted an entry for the fastest subroutine used the same algorithm:

- Save the stack pointer in the HL register pair.
- Load the stack pointer with 4000 hex, the first address above the screen memory.
- Load either DE or BC with 0BFBF hex, the code for two complete graphics blocks.
- Perform 512 successive PUSH operations to white out the screen.
- Reload the stack pointer with its original address.
- Return from the subroutine.

What surprised me most about this answer was the creative use of the stack, which I usually treat with great respect and never think of moving around in memory. Thanks to all who suggested the idea.

Many entrants, however, forgot one crucial piece of the routine. If you're using it on a Model III, or on a Model I with an expansion interface, you must disable interrupts before PUSHing bytes of 0BF hex onto the screen. If an interrupt occurs during the PUSHes, the current contents of the program counter, as well as of the general registers, are PUSHed into screen memory while the interrupt is processed.

Normally, this is no problem, because the same information is POPed off the screen when interrupt processing is complete. However, if the screen is nearly full when the interrupt occurs (or if you're using a Model I without a lowercase upgrade) some essential data may be stored below the screen in the "ghost" keyboard memory and will be unrecoverable, crashing the entire system.

From the stack of entries (no pun intended) that tied for first place for either shortest or fastest subroutines, I selected two winners at random. I awarded the prizes to George Barlow of Kirksville, MO, and Norman Watts of Logan, OH.

Along with the fastest and shortest routines, I received some ingenious entries that found a successful compromise between fast and short. I'll discuss some of those next month. Some readers asked for further contests, but until I finish answering the many questions I received, I'm in no shape to handle another flood of mail. Perhaps in a few months. ■

Readers who subscribe to CompuServe can join in open discussions of items relevant to The Next Step. GO PCS-117 to the Software and Authors Special Interest Group (SASIG), and leave messages for Hardin Brothers on Section zero of the bulletin board. Feel free to join in discussions started by other readers. You can also reach Hardin Brothers by e-mail sent to 72165,735 or you can write to him at 280 N. Campus Ave., Upland, CA 91786. Please enclose a stamped, self-addressed envelope if you want a reply.

✓ See List of Advertisers on Page 211

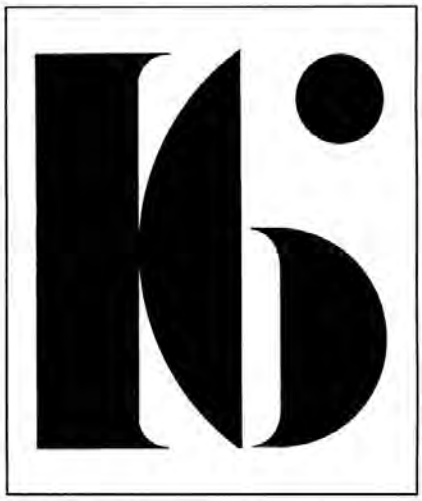
Weighing the MC68000 Against its Peers

A number of microcomputers currently on the market use the 16-bit CPU (central processing unit). But what constitutes a 16-bit microprocessor and what differences exist between those available?

In this column we'll examine Texas Instruments' 9900, Zilog's Z8001 and Z8002, Intel's 8086, and National Semiconductor Corporation's NS16032 chips to better understand the Model 16's MC68000 microprocessor.

Up front, let's define the term "16-bit." Generally, a bit value (4, 8, 16, or 32) refers to the number of data bits the microprocessor uses. These bits aren't necessarily internal to the microprocessor package; instead, they may use a chunk of dedicated RAM for their activities.

A 16-bit microprocessor doesn't have to have 16 external pins for data transmission. The number 16 doesn't necessarily denote the number of address lines. In fact, most microprocessors already have 16 address lines. The Model I, for instance, with an 8-bit Z80 CPU, has 16 address lines, and a memory of eight physical address lines. Multiplexing eight by eight address lines yields the maximum 64K RAM of directly addressable memory.



The TI9900

Texas Instruments developed one of the first 16-bit microprocessors, the 9900, in the mid-1970s. It accesses 64K of memory directly, the same as any 8-bit processor.

The 9900 is rather unique in that it has only three internal registers. Sixteen bytes of external RAM act as general-purpose registers. (Compare this with 17 registers in the Model 16's MC68000, which are extremely handy in Assembly-language programming and temporary storage.)

The original chip operated at speeds between 2 and 3.3 MHz, but a later enhancement allows 4 MHz operation. The TI9900 is a 64-pin device.

Zilog's Z8001 and Z8002

Zilog's entries into the 16-bit race comprise the Z8001 and the Z8002. Like the TI9900, the Z8002 is only capable of directly addressing 64K RAM. It has 40 pins.

Another version of that chip, the Z8001, can access 8 megabytes. Since it only has 48 pins, it must multiplex address and data bus lines. It uses the same 16 pins for data transfer (to and from memory) as it uses for address lines.

The Z8002, first out in late 1979, has 21 registers—all 16 bits wide—14 of which are general-purpose registers. Neither the Z8001 nor Z8002 have any special-purpose registers or an index register.

The 8086 and NS16032

Intel's 8086 can access 1 megabyte (1,048,576 bytes) of RAM and has 40 pins. Relatives of the 8086 are the 8088 and the 8080.

National Semiconductor Corporation's NS16032 and Motorola Corporation's MC68000 chips both have a 32-bit PC (program counter) register, making them closer to 32-bit processors. Not all of the NC16032 pins connect on the package. Only 24 pins come out of the chip. Future versions may have the potential to address an impressive 8,092 megabytes.

The NS16032 has eight general-purpose registers and a basic clock speed of 10 MHz.

Motorola's MC68000

The MC68000, the Model 16's microprocessor chip, has some unique features that make it stand out in the forest of 16-bit processors.

Physically, it's a big chip, with 64 pins on its package (like the 9900). Therefore, a full complement of data and address lines are available and it requires no multiplexing to place data and address signals on the same pins. This eliminates the need for external decoding circuits to handle such signals.

Sixteen-bit microprocessors usually provide two operating devices—simple system configuration and complex system configuration. The MC68000's 64-pin package lets it operate in either the simple or complex mode.

A big plus for the MC68000 is its ability to directly address 16 mega-

Processor	Z8001	TI9900	LSI-11	8086	NS16032	MC68000
Directly Addresses	8meg	64K	64K	1meg	16meg	16meg
Number of Pins	48	64	40	40	48	64
Number of Registers	23	3	8	14	8 gen. pur.	19

Table. 16-bit microprocessors.

bytes. This is more than most microprocessors we discuss here. It can do this because of its 24-bit address bus.

Most of the registers in the MC68000 are 32 bits wide. They are addressable as 8-bit (1 byte), 16-bit (one word), or 32-bit (one long word) registers. In contrast, the Z8000 registers are only 16 bits wide, but can be paired.

The MC68000 is available in versions using 4, 6, 8, and 10 MHz clock speeds. In your Model 16, the clock speed is 6 MHz.

The Table compares some of the characteristics of microprocessors. ■

Assembly-Language Corner

A fundamental aspect of Assembly-language programming involves displaying text on the video display. TRSDOS-16 provides two built-in routines or supervisor calls that perform this task. One routine (supervisor routine 8) displays a character, the other (supervisor routine 9) displays a line of text.

Displaying a Character

To use supervisor routine 8 to display a character, load the ASCII code of the character into byte offset 6 of the SVC block (your supervisor buffer area). The character sent to the screen prints at the current cursor position.

This routine can print control codes as well as any alphanumeric character. It's handy when you want to clear the screen or execute line feeds. (In our last column we showed you how to use this supervisor call to clear the screen by sending the ASCII code that erases the video (30) down to the supervisor routine.)

Program Listing 1 displays an A on the screen using the ASCII code for the letter A (65 decimal) at the current cursor position.

Displaying a Line of Text

To use supervisory call 9 to display a line of text on the screen, first establish the parameters for the storage area to which you want the text printed.

The Text pseudo-op lets you define an area of memory to store a string of characters. Radio Shack's Assembler-16 manual refers to pseudo-ops as directives. Whatever you call them, they are commands understood by the assembler but not by the MC68000. The assembler creates the necessary machine code that the MC68000 executes to perform an otherwise lengthy task.

Use a label to define the address of the first character in the block of text. Surround the text you want displayed with apostrophes (this is similar to using quotation marks in Basic).

Place the identifying TRSDOS supervisor call number in byte offset zero of the SVC block. Load the length of the string in byte offset 6. Count each character and space in the string as 1 byte.

This routine has a provision that sends a character or control code to

```
        LDA     .A0,SVC BLOCK  *LOAD A0 WITH BLOCK ADDRESS
        MOVW    @A0,#8         *IDENTIFYING FUNCTION CODE
        MOVW    6@A0,#65       *ASCII FOR "A" TO BE SENT
        BRK     #0             *JUMP TO SUPERVISOR CALL
SVC BLOCK                      *ESTABLISH 32 BYTE BUFFER
        RDATAB  32,0
```

Program Listing 1. A display.

See List of Advertisers on Page 211

the screen after the program finishes printing the line, much like a line feed and carriage return.

You must tell the supervisor routine the starting address of where to locate your text in memory. Load this address into byte offset 10 of the SVC block. Note that here you add the suffix L onto the Move command; the Model 16 requires a 4-byte memory address in order to store addresses. In Z80 Assembly-language programming, you need only deal with 2 bytes and the range of addresses from zero to 65535 or 0000 to FFFF hexadecimal (hex). With 4 bytes available, you can store addresses at locations up to FFFFFFFF hex.

It's best to place the storage area for your text at the end of the program to avoid a possible Odd Address Trap error in case you use text containing an odd number of characters. All instructions for the MC68000 must begin on an even memory address location.

Put it all together and you can write this routine to print the message SUITE 16.

```
        LDA     .A0,SVC BLOCK
        MOVW    @A0,#9
        MOVW    6@A0,#8
        MOVW    8@A0,#13
        MOVL    10@A0,#MES-
                SAGE
        BRK     #0
        RET
SVC BLOCK
        RDATAB  32,0
MESSAGE TEXT    'SUITE 16'
```

This is only a program module and can't run on its own.

Model 16 programs don't need ORG statements since the machine automatically places Assembly-language programs at the first available space in RAM after DOS. However, you need an End instruction as well as a label defining the beginning of the program. Whatever label you use with the End pseudo-op, be sure to also use it at the location where you want program execution to start.

If you omit the End operand label, no error message appears, but when you run the program nothing happens and you have no clue as to why.

The assembler, however, does detect the End command and generates an error message if it's missing. The assembler doesn't check the operand.

A complete stand-alone program to display a line of text on the screen and return to the TRS-DOS Ready mode is shown in Program Listing 2.

The supervisor call to display a character (number 8) sends any of the 255 ASCII codes to the screen. Therefore it can display the various graphic characters in the Model 16's video character generator. Control codes have ASCII values from zero to 31. Graphics codes extend from 128 on.

The routine to display a line (number 9) can only direct characters to the screen with an ASCII code falling between 1 and 127. You can't use it to dump graphics to the video display. ■

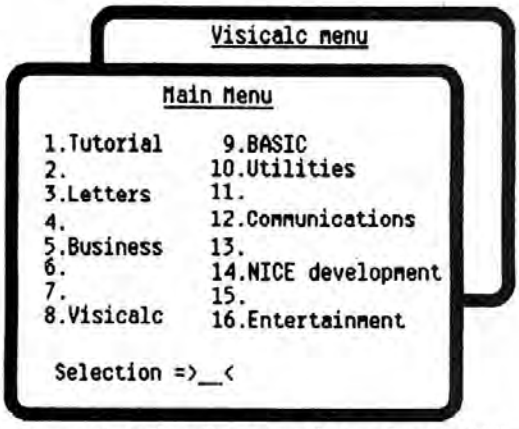
```
START   LDA     .A0,SVC BLOCK
        MOVW    @A0,#9          *IDENTIFYING SVC CODE
        MOVW    6@A0,#8         *LENGTH OF TEXT
        MOVW    8@A0,#13        *SEND CARRIAGE RETURN AT END
        MOVL    10@A0,#MESSAGE1
        BRK     #0              *EXECUTE SVC CALL #9
        MOVW    @A0,#264        *SVC CALL TO JUMP TO TRSDOS
        BRK     #0              *EXECUTE SVC CALL #264
SVC BLOCK
        RDATAB  32,0
MESSAGE1
        TEXT    'SUITE 16'
        END     START
```

Program Listing 2. Text display.

NICE (tm) New Interactive Computing Environment

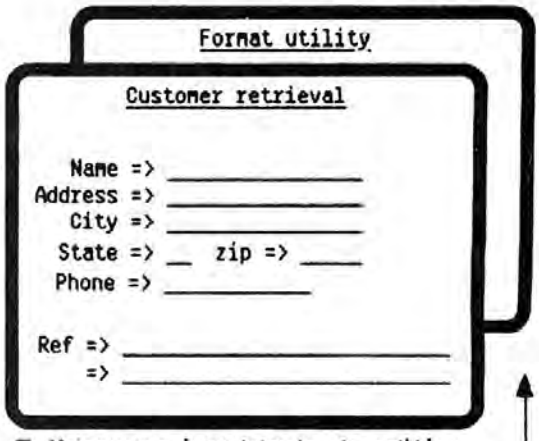
from XYZT Computer Dimensions, Inc.

Menus



The most friendly user interface utilizing a set of menus. Runs utilities, wordprocessors, games and applications directly from menus. Can be easily modified and customized.

Screens



Full screen input/output, editing, formatting and validation, PF keys, messages, full cursor control. Can be used directly from BASIC.

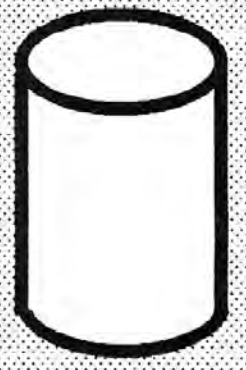
Visicalc (tm) interface

	A	B	C
1	Description	Date	Amount
2			
3	Supplies	08/15/83	$27.86
4	Parking	08/20/83	$15.00
5	Shopping	08/20/83	$73.20
6		Total:	$116.06
7			
8			
9			

The information kept in the database can be processed by Visicalc.

Special **built-in interface** allows data transfer from Database to Visicalc.

Database

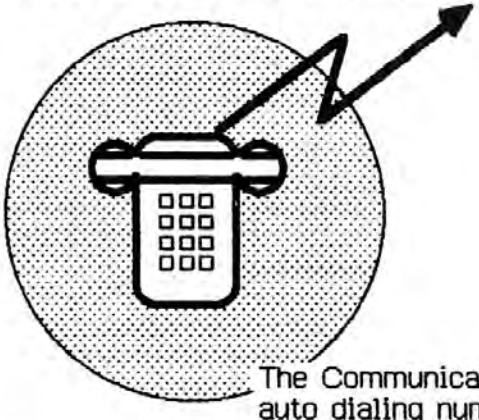


Remarkably universal and efficient data processing system.

Unlimited number of files, true multikey access, **no sorting** required.

Ideal for personal use and business systems.

Communications

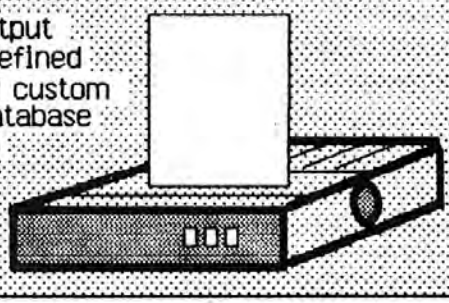


The Communication facility can be used for text and data transfer, auto dialing numbers from the data base, customized protocols, etc.

(Communications available 1984;priced separately)

Forms & Reports

Information can be output in a variety of user defined formats. Standard and custom letters, merge with database invoices, inquiries and reports can easily be specified and printed



158

The System is compatible with TRS-80 mod I/III (min 2 drive 48k) LDOS, NEWDOS/80, DOSPLUS and MULTIDOS

price: $450.00

XYZT Computer Dimensions
2 Penn Plaza, Suite 1500
NY NY 10121 (212)244-3100

See List of Advertisers on Page 211

✓ See List of Advertisers on Page 211

Is Software for Everyone or Just the Moneyed Elite?

A reader called me one day because he was having a problem loading a cassette. I told him to try other volume settings, explained how to align the head on his recorder, and finally said, "If it still won't load, perhaps it's a bad tape. Return it and let me send you another."

"Oh no," the caller replied. "I know the tape is fine. Got it from my friend and it works great for him."

Maybe I'm uncharitable. I can see how the friend who got this neat tape would want to share it with a buddy. And yet, I had been talking to a software pirate who had conned me into giving him something else to which he had no right—my time.

Intuitively, lending software seems different to me than lending a record or a book because of the support involved. When you play a record, you don't call Capitol if your stereo isn't working. If you don't understand a word, you don't call the author. But when software messes up, you expect your dealer to straighten it out. And that's how it should be.

Good software comes with support. Load 80, for example, has a team of people providing service 40 hours a week. We love doing it, love talking to you, helping you straighten things out. But we can afford to support only Load 80 purchasers, not the general public. You know the old saw about "only so many hours in a day." Quality support requires that our staff spend many hours identifying and resolving problems—hours for which people must be paid.

Freeware?

There's the rub. Looked at in terms of development and support time, a megabucks price tag on software seems reasonable. Looked at in terms of helping people and sharing useful information, software should be free.

Some friends of mine have developed a "freeware" concept. As they envision it, their programs would circulate on bulletin boards with the following message: "If this program is useful to you, send us some money."

They would provide no support for the product, thus minimizing their investment. The optimist in me thinks that they'd probably make some money, but not enough on which to raise a family.

Freeware completely defuses the question of piracy. You can't steal something that's given away. If you like the programmer's work, you contribute to him or her. But, judging from my mail, I'd say that malfunctioning software frustrates some users to the point that only talking to an informed person will help.

Perhaps the idea is not to sell the actual lines of code, but to sell the support. In selling support, we could take a lesson from the newspapers' personals columns which are filled with advertisements from women who will talk to you. The first question they ask is "What's your Visa or Mastercard number?" If the software is free, the support will have to be on a payment for time basis. Theoretically elegant, logistically difficult to administer.

Should the government step in and create a new agency: the Department of Software Social Services? Its charter could be to provide free support for computer software. A new kind of professional would emerge: the Computer Social Worker. This course would institutionalize the implicit doctrine of software pirates—that everyone has a right to software, documentation, and support.

Clearly, there's nothing easy about this issue. While companies wait for the courts to set a legal precedent, the individual has to work out the software piracy question for him or herself. What is a reasonable limit on

Article	Page	Cassette File Spec	Disk File Spec	Comments
Side A				
		A	TITLE/BAS	Basic
The Play's the Thing	52	PLAYBYTE	PLAYBYTE/CMD	System
The Play's the Thing	52	SAMPLE	SAMPLE	System
Stepping Through Basic	56	SINSTEP	SINSTEP/CMD	System
Ground Control to Major John	62	D	ADVENTR/BAS	Basic
Grand Opining	84	E	MICROTAB/BAS	Basic
Side B				
Letter Perfect	96	F	WORDCHKR/BAS	Basic
Borderline Case	158	MARQUEE	MARQUEE/CMD	System
Synthetically Speaking—Part II	126	H	VOTRAX/BAS	Basic
Synthetically Speaking—Part II	126	TTSPRG	TTSPRG/CMD	System
Synthetically Speaking—Part II	126	TABLE	TABLE/CMD	System
Project 80	200	PROJECT2	PROJECT2/CMD	System
Project 80	200	PROJECT4	PROJECT4/CMD	System

February Load 80 directory.

copying software? When does it become piracy? Are manufacturers of software responsible for supporting those pirated copies?

On the other hand, should software (or software support) be the exclusive domain of a moneyed elite? If the only way to get hold of exciting software is to purchase it (or write it yourself if you're talented enough), then the computer world of the future looks undemocratic.

Ah well, better minds than mine (yours, for instance) have been mulling over these questions for some time. I haven't reached a satisfactory conclusion, though I've developed some working parameters. Seeing the definition of piracy as an individual ethical decision, not an institutional one, I am committed to seeing commercial programmers adequately reimbursed for their efforts and computer users adequately supported in their efforts to get the most from their machines. Your comments on these questions are of great interest to me. Please write. ■

Continued from p. 22

point on, you load and use the compiled program.

Compiled programs, being in machine code, run much faster than their Basic equivalents. This is because Basic isn't involved in running the program and doesn't have to translate each line of the program before it can execute that line's instructions. String reorganization is a function of Basic, not machine-language programs, so time lost in this function in Basic has no equivalent time loss in machine code. On the other hand, all string operations in machine code must have the exact amount of required room preallocated for data storage. You can't just have the program save 2,000 bytes for string storage. Each machine code string save must have an explicit location and length to store the string. If it doesn't, your program will crash.

As yet, there aren't any Basic compilers for TRSDOS 6.0, but you can use several Basic tricks to speed up your Basic program up to 50 percent. First, get the IJG book *Basic Faster and Better* (see address and price at the end of this column). Most of these tricks work with any version of Microsoft Basic, and so should work with TRSDOS 6.0.

Next, Prosoft (Dept. G, Box 560, North Hollywood, CA 91603, 213-764-3131) has two programs which might help you out: Trashman ($39.95) and Faster ($29.95). Trashman is a machine-language program that takes control away from the built-in Basic string-reorganization routine, and makes it faster and more efficient, with an average improvement of about 95 percent. Faster is a machine-language program that analyzes your Basic program while it's running, and then tells you how to organize your program's variables in such a way as to allow your program to run faster and more efficiently. Call Prosoft and ask them about compatibility with TRSDOS 6.0.

Q: The June 1983 Feedback Loop contained a letter signed by D.S. of Fountain Valley, CA (p. 353), which I surmise to be the programmer of Disk Doctor, Dave Stambaugh. I tried to locate him after purchasing his program because I discovered a bug in it. When trying it out, I was unable to get any action from test "T".

After several calls to the local Alpha Store, I was referred to Apparat. After confirming the problem, which occurs in a Model I in double-density mode with the Percom doubler board, they promised to contact the author.

When I called back, they said Dave acknowledged the problem, but was very sorry because he couldn't do anything about it since he no longer has a Model I computer.

I'm very sorry too, since software rip-offs seem to continue to plague the industry. (Henry Ball, Burbank, CA)

A: A rip-off is when you spend money for an inferior product that doesn't do what is promised, and you can't get your money back. If you get your money back, how can you be ripped off? Did Alpha Stores refuse to let you return Disk Doctor for a refund?

If they did, the rip-off is their fault, not Dave's. Disk Doctor is a good product, one of the more useful ones that I own. It's too bad that there's a bug in it, but one bug doesn't make for a rip-off. I think you're being unreasonable. What do you expect him to do? Buy a Model I and expansion interface with disk drives and Percom doubler (especially when they aren't sold anymore) just to track down a minor bug? And what about all the other doublers, disk drives, and expansion interfaces for the Model I? Is he supposed to buy all of those too? If he did, any money he earned in royalties would all be spent on hardware.

In the several years that his product has been on the market, this is the first that I've heard of this problem (that's why I call it minor). However, I thank you for writing and telling everyone that the "T" test in Disk Doctor doesn't work with the Model I in double-density with a Percom doubler.

Q: I own a Model I and I recently bought the Radio Shack lowercase modification for it. Unfortunately, Radio Shack didn't include installation instructions. Can you either give me instructions on how to do it, or tell me where I can get the instructions? (J. Michael Hawthorne, Southern Pines, NC)

A: Either buy the Service Manual (part number MS-2601104) for $1.49 from National Parts (see the address at the end of this column), or take your kit and computer to any Radio Shack Computer Repair Center and have them install the kit for only $15. If Radio Shack does it, there won't be a guarantee because you bought the kit separately.

Q: Is there a simple way to interface a Facit 4554 printer to the parallel port of a Level II Model I? The printer was designed to operate with a Facit SP-1 interface for parallel data transfer, but I can't find the documentation on this hardware.

Also, would building a parallel-to-serial converter and using the printer's serial input port be an easier task? (John Maikisch, Morris Plains, NJ)

A: Since you can't find the documentation for building a Facit SP-1, your best choice would be to build or buy a parallel-to-serial converter. (Binary Devices, 11560 Timberlake Lane, Noblesville, IN 46060, 317-842-5020, sells one for $149.95: the UPI-3VB.)

Q: I'm thinking of buying an LNW-80, and I've been told different stories about how good it is. If anyone who owns or works with one would write about its good and bad points, I would really appreciate it. (Brian Blasjo, Riverside, CA)

A: All right you LNW-80 users, tell him about your experiences!

Q: I followed your instructions in the June Feedback Loop (p. 350) and installed another 16K of memory in my Model III. I now have 48K of memory, but I'd like to have 64K. However, there aren't any more empty sockets. Can you tell me how to install the next eight chips? (Paul Ferris, Pound, WI)

A: You can't do it easily. It requires rewiring most of the primary circuit board. You can, however, buy kits that will let you install a full 64K in your computer. The only

problem is that with full 64K RAM there's no room for the ROM (Basic and your operating system). Most of these 64K upgrades get around this by making the computer into a Z80 64K CP/M computer. For more details on three of these systems, see the December 1983 issue of *80 Micro* (p. 122).

Q: I thought you might be interested in the enclosed response to M.L. from Wichita, KS (July 1983, p. 366): Picto-Script, a user- and programmer-friendly word and graphic processor, uses dynamic run-time hyphenation. It's accurate about 75 percent of the time, and correction is simple if you catch the mistake while on the next line (or you can use INS-DEL editing commands). It doesn't have proportional spacing since the current print driver is written totally in Basic (graphics are slow).

The program costs $49 and is available from Dan Baright, 281 North Jackson #9, Lebanon, MO 65536. (D.B., Lebanon, MO)

A: Good luck with your program.

Q: In addition to the information you gave M.L. (July 1983, p. 366) with respect to the hyphenation problems he was having with Scripsit, Lazy Writer has fine semiautomatic hyphenation, as well as visible and adjustable page breaks, with none of the problems Scripsit apparently has.

Also, it interacts with the new Electric Webster to provide fully automatic hyphenation if so desired. (Sidney Bloom, Frederick, MD)

A: Thanks for the information. Since I don't use Lazy Writer, I didn't know it could do that.

Q: I just purchased a Model 4 computer, with SuperScripsit for the Model III mode. While running SuperScripsit, I note that it continues to display 40 characters instead of the 80 characters possible in the Model 4 mode. Is there a patch or anything else I can do to change it to 80 characters? (Edward Markle, New Orleans, LA)

A: Unfortunately, the design of the Model 4 uses an either/or setup; you're either in one mode or the other. SuperScripsit was designed to operate on a Model III, and requires certain parameters for that operation. The Model 4 mode uses slightly different parameters, different enough that SuperScripsit can't operate in Model 4 mode.

One of these parameters is the video display, which is memory mapped (that is, the video display actually occupies RAM locations, which you can directly access). The video map in the Model III mode (64 characters by 16 lines, 1,024 locations) is different from the Model 4 mode (80 characters by 24 lines, 1,920 locations). These maps are in different locations. Changing SuperScripsit, which directly accesses video memory, to use the Model 4 video map would require major surgery by an expert programmer. There isn't a simple patch.

It's possible that Radio Shack will have SuperScripsit rewritten for the Model 4 if enough people ask for it, but that could mean a wait of up to a year. In the meantime, you're stuck with 64-character lines (not 40). ■

Terry Kepner is a freelance writer and programmer, and the vice president of Interpro. He's been writing about microcomputers since 1979.

Frequently Needed Numbers

Radio Shack, National Parts Division, 900 East Northside Drive, Fort Worth, TX 76102, 817-870-5662. M/C and Visa accepted, each order has $1.50 handling charge.

IJG Inc., 1260 West Foothill Blvd., Upland, CA 91786, 714-946-5805. Publisher of *TRS-80 Disk and Other Mysteries* ($22.50), *Microsoft Basic Decoded and Other Mysteries* ($29.95), *The Custom TRS-80 and Other Mysteries* ($29.95), *Basic Faster and Better* ($29.95), *Machine-language Disk I/O and Other Mysteries* ($29.95), *TRSDOS 2.3 Decoded and Other Mysteries (Model I)* ($29.95), *How to do it on the TRS-80* ($29.95), and the Electric Pencil Word Processor ($89.95).

Continued from p. 48

If you have a Model I with a lowercase conversion, load the lowercase driver before loading The Word Machine. The program recognizes lowercase and uppercase commands, with a couple of exceptions.

The text entry, editing, and print formatting functions are those used for preparing normal correspondence. The Fill mode lets you enter text when you start preparing a document or append it to existing text.

The capacity of the text buffer in a 48K system is 290 lines. The program displays the number of lines available in the buffer in the screen's upper right corner. You can insert a line or multiple lines into the text, kill (delete) a line or block of lines, or search for a string located at the start of or within a line.

You can move text on the monitor one line at a time or scroll it up or down. The Up versions, single line or scrolling, display the lines in reverse order so you have to get used to that.

You can center text on a line with a two-key command. The Word Machine uses word wraparound. This program is line-oriented and what you see on the monitor is what you get in the printout.

The edit functions are simple and easy to learn. In fact, if you use the Basic editor at all, you'll feel right at home with this editing approach. The cursor is nondestructive and the up-arrow key acts as a control key for many edit functions.

You can insert or delete characters within a line without first specifying the number of characters. Hack and clear from cursor to the end of the line and Extend line both put you in the Insert character mode.

From edit you can go into the graphics mode. This lets you embed block graphics or printer codes in a line of text. You depress numeric keypad keys and record the values by pressing the period key.

The program calculates ASCII code values from the key values according to a code explained in the manual. The clear key lets you cancel any edit changes made and exit the edit mode, while enter updates the text line to the buffer and returns to the command mode. You edit on one line at a time.

"That $20 price is no misprint. The Word Machine is fast, effective, and easy to learn and use."

The program recognizes lines starting with a semicolon as command code lines. The commands can be printer control codes or commands to skip lines, eject a page, and so on.

In addition, a Pause comment command pauses text printing, displays the comment line on the screen, and waits for input from the keyboard. This lets you type names and addresses into a form letter.

If you have trouble remembering the commands, Help displays and briefly describes virtually all the commands and controls. The printer command works with the text currently in the buffer, not on a disk file.

Before printing begins the program asks how many copies (up to a maximum of 999) you want. As soon as it receives a response, the program starts printing at the current text line and continues to the end of text. If you want to print the complete text, move the line pointer to the top of the text.

The Form command displays and permits changes to the print format parameters. Set these prior to the initial entry of text for a given document. The format variables, all with default values, include margins, line length, page length, tab positions, and line spacing (single or double). Alphabetic input must be in uppercase.

Set the end-of-page pause option for single sheets, and set the option to print command lines if you want a record of the complete file including command lines.

The Word Machine doesn't forget Model I owners without lowercase modifications. With the Reverse Caps option, you can key in typewriter style (shift for capitals). At print time the program reverses this to print normally, although you still won't see lowercase on your monitor.

The last variable in the Form screen is "Retain line count after printing." This is necessary to chain files together for printing a continuous document. You must do chaining manually. When you've finished printing a file, load the next file and give the Print command. With the line count carried over from the previous printout, the new file's pagination is correct. This doesn't work when you specify multiple copies as a print option.

The Word Machine provides interesting options for the Save and Load disk file commands. Extensions must be uppercase. Normally, you save files with a /TXT extension and include the print parameters at the beginning of the file. If the file name already exists, the program asks if you want it replaced; if it doesn't exist the program asks if it should create the file.

If you use the extension /DOC, the program saves the file without the print parameters in an ASCII file. You can use these files for transmission to bulletin boards and videotext systems.

When you ask to load a disk file, The Word Machine first prompts you for the file name, then gives you the choice of clearing and loading the buffer, or of inserting the file into the buffer between the current line and that following it. This lets you save stock paragraphs or text segments in small files and insert them anywhere in your text as you need them.

This particular capability saves a great deal of time. Finally, before you actually load the file, the program asks whether to use the file format values. If the response is N, the program uses the values currently set with the text.

You finish creating and printing one document and are ready for another.

The Word Machine

★ ★ ★ ★

Pel-Tek
P.O. Box 1026
Southampton, PA 18966
Models I and III
32K, one disk drive
$20

Easy to use?	★★★★☆
Good docs?	★★★☆☆
Bug-free?	★★★½☆
Does the job?	★★★★½

CHEAP!

EPSON PRINTERS

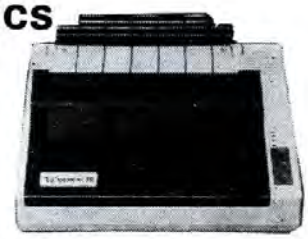
The Most Reliable Printers We Sell!

- RX-80
- RX-80 FT
- RX-100
- FX-80
- FX-100

Lowest Prices!
Call
800-331-3896
For More Information

Star Micronics

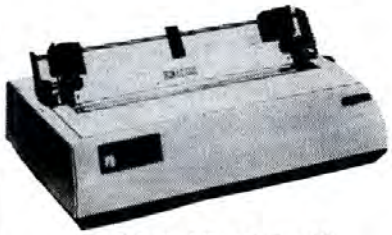
GEMINI 10x
GEMINI 15x



Call for Unbelievably Low Prices!

800-331-3896

OKIDATA PRINTERS



Microline 82A
Microline 83A
Microline 92
Microline 93

"Built Like A Tank"

Is Your Printer Too Slow?

The Microfazer is a Printer Buffer. A Printer Buffer won't make your printer print any faster, but it will free up your computer while your printer is printing. The computer can dump the information to be printed into the Printer Buffer almost instantly. The Printer Buffer then waits on the printer and passes on the information to the printer when the printer can accept it. The computer is free to do more important things (instead of waiting on the printer). So, instead of spending $500, $800, $1000 or more on a new printer, try our

128K Microfazer Printer Buffer! $299.95

DISK DRIVES!

TRACKS	SIDES	TANDON Bare	TANDON w/ Power Supply	TEAC Bare	TEAC w/ Power Supply
40	1	$179.95	$219.95	$179.95	$219.95
40	2	$269.95	$319.95	$219.95	$259.95
80	2	$319.95	$369.95	$289.95	$329.95

Special — Buy 2 TEAC's in a Single Power Supply for the Price of 1 Bare TEAC Plus the Price of 1 TEAC with Power Supply

DISKETTES—$19.95

Premium Quality CDC Brand, in Boxes of 10 Diskettes with hub rings. At this price, you can't go wrong! 5¼" Single Sided Double Density

THESE ARE OUR CASH DISCOUNTED PRICES C.O.D. AND CHARGE ORDERS ARE 3% HIGHER. ALL COMPUTERS ARE SHIPPED FREIGHT COLLECT. PRICES, SPECIFICATIONS, AND AVAILABILITY ARE SUBJECT TO CHANGE WITHOUT NOTICE. IBM & IBM PERSONAL COMPUTERS ARE TRADEMARKS OF INTERNATIONAL BUSINESS MACHINES INC. TRS-80 & RADIO SHACK ARE TRADEMARKS OF TANDY CORPORATION

918/825-4844

AMERICAN COMPUTERS

SMALL BUSINESS

118 SO. MILL ST
PRYOR, OK 74361

✓217

NEED INFO QUICK?

Then just call our automated TOLL FREE answering machines. Leave your name, address and phone number. Requests will be mailed 1st Class the same day. Only requests for product information will be processed. All others will be disregarded by our answering service.

Call 1-800-331-3896
In Oklahoma Call
1-918-825-4844

To get rid of all that text in the buffer and make room for the new, just type NEW on the command line (the top line on the monitor) and the buffer clears.

When you want to go on to other programs, type QUIT to get to DOS READY. Suppose you forgot to save that file before quitting or you had to reboot. When you reload The Word Machine, it senses an active buffer and offers you the option of a warm start. If you take that option, the program starts with your complete buffer intact.

Limitations

The Word Machine can't control right justification, proportional printing, and header or footer titles. It cannot support block moves or replace or change strings, but it wasn't intended to perform those functions.

So what are its real limitations? I haven't yet found a way to center lines of text entered from the keyboard during print time except by adding spaces in front of the text.

You must install a lowercase driver in the Model I before loading The Word Machine. DOSes such as DOSPLUS 3.4 have a built-in driver in low memory that is automatically installed when the DOS senses a lowercase modification. But operating systems such as DOSPLUS 3.5 and LDOS 5.1 use large keyboard drivers that do more than provide the lowercase driver.

The DOSPLUS 3.5 keyboard driver is larger than the space reserved by The Word Machine, so you can't use it on the Model I. The same situation might exist for the LDOS driver. As a result, I dusted off the Radio Shack lowercase driver and installed that—it works just fine.

Pel-Tek indicates that the next version, due out after the first of the year, will probably incorporate its own lowercase driver. This situation doesn't exist in the case of the Model III.

The manual details the operation of the Proof command, originally incorporated to call the inexpensive Aspen Company's Random House Proofreader (spelling checker) program. Unfortunately, that program is no longer available.

Spelling checkers such as Electric Webster should work on The Word Machine files but at considerably greater cost. Pel-Tek states that a current project is writing their own spelling checker program.

Support and Documentation

Pel-Tek's technical support is excellent. I ran into a couple of bugs while putting The Word Machine through its paces.

Epson printers with Graftrax Plus use the zero as a control code for several functions, including underlining and superscripting. Unfortunately, the Model I ROM Print routine sees this as a null and throws it away. The only way to get the control code to the printer is to POKE it there.

The embedded control code routine of The Word Machine sends these codes to the printer as LPRINT CHR$(code) so the zero codes don't get to the printer. I talked to Ed Levy of Pel-Tek on a Saturday morning and had a corrected version on disk by the following Wednesday. That's a fast response.

I was also told that the corrected version would be shipped on all new orders and to current owners who have the same problem. This is not a problem on the Model III.

Another bug I encountered concerns the Search function. The first time you use it in a session, it returns an error message, but if you repeat the command it works perfectly from then on.

The Word Machine's documentation consists of 22 pages of clear, concise text, illustrations, and tables in a 5½- by 8½-inch format. One appendix provides detailed instruction on embedding graphic/control codes in a line.

A table shows the numeric key combinations corresponding to ASCII values from zero to 255. Finally, the back cover of the manual has a summary of the commands, controls, and format variables. The manual is quite adequate.

Conclusions

Perhaps you've seen the ad for The Word Machine and decided that either

the price is a misprint or that this couldn't be an effective word processor. Don't believe it! That $20 price is no misprint and The Word Machine is a fast, effective, easy to learn and use word processing program.

It includes a surprising number of professional word processing features, especially considering the cost. A revised version that includes its own lowercase driver might be out as you read this. A spelling checker by Pel-Tek might also be available as an integrated option.

This is a best buy, not only for the person who has a limited need and budget, but as a second, quick job word processor for those who normally use a program like Newscript or SuperScripsit. I like this program! ■

Managing Data: Filemate II

by Alan Neibauer

Filemate II is a data-base manager that provides advanced features for a program in its price range. It offers form letter merging, transfer of fields between data bases, and arithmetic abilities.

You can revise or expand the file structure at any time without reentering data. As your needs change, you can add or increase fields, or change them from alphabetic to numeric format.

The program's major weaknesses derive from the fact that it's written in Basic and requires knowledge of the operator beyond simply booting up the system.

No.	Name	Type/Size
1.	Name	Alpha—20
2.	Address#1	Alpha—20
3.	Address#1	Alpha—20
4.	Telephone	Alpha—14
5.	Territory	Alpha—10
6.	#Sales	Numeric—8
7.	#Percentage	Numeric—8
8.	#Commissions	Numeric—8

Relational Keys: H, S, M

Figure 1. File structure for company sales records.

```
MENU
< A >—ADD A RECORD
< V >—VIEW, CHANGE, OR DELETE
< F >—FETCH A RECORD
< G >—GLOBAL SEARCH
< P >—PRINT-OUT RECORDS
< S >—SORT
< M >—MODIFY NAME OF A FIELD
< C >—CALCULATE
```

Figure 2. File module main menu.

Filemate II comes on a single-density data disk with instructions for transferring eight Basic program files to a minimum operating system disk. You need NEWDOS/80 2 or DOSPLUS 3.4 for the Model I, and TRSDOS 1.3 or LDOS for the Model III.

Because of syntax differences between the Models I and III systems, Datafile Systems provides two separate versions of the File and Sort modules.

Once you transfer or convert Filemate II, call up Basic and run the ORG module to initiate the file structure. The program then uses the File module to manipulate the data and perform other functions.

If you use DOSPLUS, you must reserve three file buffers each time you load Basic. You can avoid this manual loading of Basic and the program file by using a DOS Auto command.

As with all data bases, you must do some preliminary work before initializing the program. Decide what fields you need, their length in characters, and which numeric fields you'll need in calculations.

Filemate II provides selection of relational keys, user-defined codes that represent groups or sets to which the record belongs.

Using Filemate II

To use Filemate II, first load Basic and run the ORG module, which names the data base and designates a drive to hold the file. Then enter the names of up to 20 fields, preceding those on which you'll perform calculations with a # sign.

The first field name is the most critical since it generates the hash code used to place the record in the file.

See List of Advertisers on Page 211

Once you enter all field names, input the maximum character length with a maximum limit of 255 characters. Finally, input the relational keys.

The resulting file structure appears in Fig. 1. Each numeric field contains eight characters to handle double-precision variables.

When you've written the structure to disk, load the File module to display the main function menu (see Fig. 2). The system gives you the option of Model I or III versions of File.

To add a record, enter each field in response to the field name and row of dashes. After the last field, add the appropriate relational keys. After each complete record, Filemate II requests a final confirmation before hashing the first field and writing the data to disk.

Select the V option from the main menu to display the desired record number. Since hashing places records in various locations on the disk, the order in which you enter data does not correspond to its record number. After you display the requested record, you can change, delete, or print the data, or increment or decrement the displayed record.

The only problem arises if you change the first field. Since this field determines the record number, changes can result in a new hash code and you might not find the record with later Fetch commands. The manual suggests deleting the entire record, then reentering the data with the new first field.

The Fetch option retrieves records by first field name rather than by record number. The system computes the hash code of the string entered and searches for the appropriate record. You can also use the Global option for data retrieval but it searches entire records for the input string.

Printing

The Print and Sort options require more input. You define the report layout and have the option of saving the format to disk. You can save and easily retrieve any number of such formats.

A nice feature of the Print module is layout flexibility. You can generate single and multiline formats or mailing labels to correspond to any report style. The program displays fields as it prints them.

To sort records, you create an index file rather than physically sort the records on disk. This makes it possible to make several sorts for easy retrieval.

The Sort option first asks you for the data base desired, then the relational keys used to select records. Pressing the enter key alone includes all records in the sort.

By using the keys and Boolean operators, you can select specific records for a particular index file. While you can designate two fields for the sort, the program uses only the first five characters of each. You can use the completed index file immediately or save it on disk.

While entering the format number easily retrieves print formats, you must describe index files when retrieving them from disk. Although no actual sort takes place, you must describe the sort parameters in the same

way you initially created them. It would be more useful to save and recall index files by number.

Calculations

The remaining options on the main menu provide the means for additional data manipulation, from changing a field name to transferring data between related files.

One good feature of Filemate II is the Calculate mode, which performs arithmetic operations on fields. Calculations can involve addition, subtraction, multiplication, and division between fields; a designated constant; or square root, sine, and cosine operators. The system also assigns temporary stack fields in which you can store results until the program determines the final figures.

When initiating the Calculate mode, input the record you want used. The program provides the options of applying the algorithm to all records, saving it for future use, or reporting cumulative totals of all records in the file.

Algorithms aren't limited to manipulation of two fields. The program performs any number of passes until you achieve the desired result. The Filemate II manual gives an example of a 12-pass operation to compute selling price from a list of cost and overhead figures.

Modification

Filemate II lets you add or delete fields and change the length and nature of fields. The MODFIELD module permits such alterations without affecting the integrity of the current data. The Keys module permits the same changes to relational keys.

The XFRMERGE module lets you transfer fields between files. This module has two options: You can create new data bases by moving fields from an existing file to an empty one, and you can transfer data into an existing data file.

As you update records in one file, you can use the XFRMERGE module to update all related files. With these options available, you can create a series of related data files when total records would exceed the 255-byte limit the program imposes.

The final file manipulation module, XPNDFILE, expands the file to hold

Filemate II

★★★★

Datafile Systems
801 Welch Road, Suite 211
Palo Alto, CA 94304
Models I and III
$75 disk and manual
$10 manual only

Easy to use?	★★★☆☆
Good docs?	★★☆☆☆
Bug-free?	★★★★★
Does the job?	★★★★★

80 MICRO's REVIEW GUIDE

Please send me _____ Review Guides at $7.95 each plus $1.50 per book shipping and handling.

Please allow 6-8 weeks for delivery

☐ Check enclosed ☐ MC ☐ AE ☐ VISA

Card#_____ Exp. Date_____

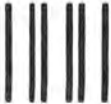
Signature_____

Name_____

Address_____

City_____ State_____ Zip Code_____

Wayne Green Inc. • 80 Pine St. • Peterborough, NH 03458 **80RG2B**



NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 73 PETERBOROUGH, NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE

Wayne Green Inc.
Attn: Marcia Stone
80 Pine St.
Peterborough, NH 03458

80micro's REVIEW GUIDE

The guide that will save you time and money.

You'll spend more time at your computer and less money for the right product.

80 Micro's Review Guide is the most comprehensive collection of TRS-80* reviews ever compiled in book form. You get 500 hardware, software and book reviews packed with information about:

- Modems
- Games
- Business programs & Word Processing
- Joysticks
- Books on programming
- Printers
- Utilities
- Editor/ Assemblers
- Educational software
- Monitors
- Databases
- And more

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

Don't waste valuable time chasing down the right product. We've done it for you with **80 Micro's Review Guide**, compiled and condensed from the pages of **80 Micro**, the magazine you've come to rely on.

A five-star rating system lets you decide what's best. And each review gives you the product manufacturer's information so you can order and receive your product *fast*.

All for $7.95

Order now. Use the attached order form, the coupon, or call toll free 1-800-258-5473.

(Outside U.S.A., and in N.H. call 603-924-9471)

yes, I want to save time and money.

Send me **80 MICRO's REVIEW GUIDE** today.

Please send me _____ Review Guides at $7.95 each and add $1.50 per book shipping and handling.

☐ Check enclosed ☐ MC ☐ VISA ☐ AE

Please allow 6-8 weeks for delivery

Card #_____ Exp. Date_____

Signature _____

Name _____

Address _____

City_____ State_____ Zip Code_____

80RG2

Wayne Green Inc. • 80 Pine St. • Peterborough, NH 03458 **Attn: Marcia Stone**

a greater number of records than it was originally designed to serve. Since the program actually rewrites the entire file onto a new disk, the hashing algorithm changes the number of each record. Of course, you have to re-create the sorted index files.

Filemate II provides two last functions that are its least useful segments. If the data base is structured along certain specific lines, the XFRMERGE module can merge the file with a standard form letter.

Actually, you perform merging by retrieving and printing the fields in the correct sequence, then calling on the DOS Print function to print the text message. Since you must create the text itself with a word processor or screen editor, a better merge function might be available already.

Finally, a Convert program transfers your existing sequential access data files, created by your own Basic program, into a random-access format compatible with Filemate II. Unfortunately, this section in the manual is a little confusing and the process requires knowledge of Basic programming. It would be easier, except for very large data bases, to create a new Filemate II file and reenter the data.

Evaluation

I changed my mind about Filemate II as I worked on this review. I was initially put off by placing the files on a system disk and remembering to initiate the correct number of buffers. It seemed that the state-of-the-art in data-base managers should go beyond what a Basic program can provide.

I was impressed by the speed at which I could retrieve records using the hash code technique, but was troubled that I had no control over the record number. The entire file has to be scanned or printed to determine how the records are numbered. I'd suggest that the record number appear on the screen as you initially save it or that the program create a small reference file.

As I used the program, I became quite pleased with its overall flexibility, especially the Print routine and data transfer modules. I also appreciate the Calculation mode and the general ease with which I can use most of the functions, even though Basic slows the overall performance.

I'm not completely pleased with the instruction manual. I find sections a little confusing and would prefer that it take the reader through every step using one sample data base.

I am impressed with the abilities and flexibility of Filemate II, and the support it receives from Datafile Systems and Jack Egbert. Jack quickly responded to some minor problems I encountered in the program by correcting them and actually improving the program's performance.

I would now rather use Filemate II than Profile or some more costly file management program on my Model I. I recommend it for TRS-80 users looking for an inexpensive data file manager. ■

A Learning Language That Slithers

by Mare-Anne Jarvela
***80 Micro* staff**

Snake is a computer language originally written to introduce children to Basic. Like Logo, it draws pictures on the screen when a child provides Basic commands. The more a child knows about the commands, the more complex a picture he can draw.

If you liked Logo, you'll like this program. The graphics are a bit less sophisticated than the ones on the Apple or Texas Instruments computers, but they're not bad. Snake is a good teaching tool.

Using Snake

When you start programming, type in AUTO for automatic line numbering. See Table 3 for the list of commands that move the cursor. A very simple program appears in the Program Listing.

After you enter the listing, type RUN. A cursor will appear on the screen. As the cursor moves about the screen, it draws a line. The user deter-

EXPAND and UPGRADE your TRS-80®

High SPEED - mass storage for the model 100

The PMD 100 - Fast storage and retrieval of programs and files using miniature high speed wafer tapes.

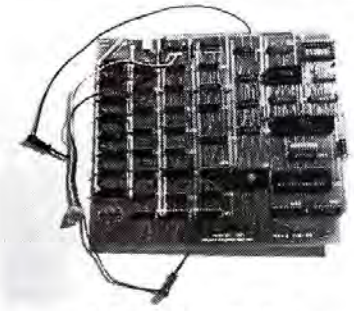
- Battery Operated (rechargeable)
- Fully portable
- Rom Based operating system

(Includes - wafer tapes, battery charger/ adapter, coiled cable, owners manual.)

$349.95

8k Memory Modules for the Model 100
HOLMES IM100™ - only $69.50 Up to 32k memory - direct plug in - no mod's req'd

FAST DELIVERY-QUANTITIES AVAILABLE



MODEL III - Add 64K CP/M 2.2 with 80 x 24 Video Display. . . only **$399.50**

112K extended memory option **$125.00**

HOLMES VID 80™

Ask about MOD IV software upgrades for MOD III

ORDER TODAY - STOCK DELIVERY

"DOUBLE YOUR SPEED"
CUT COMPUTER OPERATION TIME IN HALF

Sprinter II for Model I	$99.50
Sprinter III for Model III	$99.50

MEMORY EXPANSION
Internal Installation

Model I - IM-2 (W/32K RAM)	$139.50
Model 100 IM100 (8k RAM module)	$ 69.50

MODEL I SPECIAL!!

Add disk drive capability, an RS232C serial communications port, and an additional 32k RAM (48k total) to your Model I with the Holmes Expansion Main Frame Package only **$449.00** (a $508 value)
(Includes a DX-2D, RX232M, and MF-1)

OTHER PERIPHERALS:

Expansion Mainframe-MF-1 (For 4 boards)	$150.00
Color coded diskettes (box 10) 96 TPI SSDD $29.95 DSDD	$ 37.95
Epson Printer Driver for SUPERSCRIPSIT	$ 24.95
Drive O Package w/DX-3DC, drive, power supply	$545.00
RS232 (RX-232) (Requires MF-1)	$119.50
RS232 w/32K RAM (RX-232M-Requires MF-1)	$199.50

DISK DRIVE CONTROLLERS FOR TRS 80 MODELS I, III and IV Computers

Disk Controller w/clock DX-3DC (Model III & IV)	$157.50
Disk Controller w/o clock DX-3D (Model III & IV)	$137.50
Double Density Adaptor DX-1D (Model I)	$ 99.50
Disk Controller DX-2D (Model 1, requires MF-1)	$159.00

ORDERING INFORMATION:

- Add shipping and handling charges of: $9.00 for PMD100, and MF-1, pkg. USA, $15.00 Canada & 15% for Overseas
- Prices subject to change without notice.
- Payment by cashiers check, money order, cash, Visa or Mastercharge. Personal checks subject to 3 week clearance.

About HOLMES ENGINEERING, INC.

— All Holmes Products come complete with easy to follow installation and trouble shooting guides and our technical group is available for support as required. (Monday thru Friday 8 a.m. - 12 noon (MST)
— Installation can be performed with out system modifications, soldering and little or no technical background.
— MORE DETAILED PRODUCT INFORMATION IS AVAILABLE UPON REQUEST. Send a self-addressed stamped envelope or call today for a FREE full product brochure.
— QUALITY IS BUILT INTO HOLMES PRODUCTS FROM THE START - ALL PRODUCTS ARE FULLY TESTED AND GUARANTEED FOR 90 DAYS.

TRS80 is a registered trademark of Tandy Corp., CPM is a registered trademark of Digital Research Corp.

HOLMES ENGINEERING, INC.

Bus. Hours: 8-5:30 (MST) Mon. - Fri.
10-2:00 (MST) Sat. (Sales only)
Service Hours: 8-12 noon (MST) Mon. - Fri.

5175 Green Pine Drive Salt Lake City, Utah 84123
(801) 261-5652 or 24 Hr. BBS (801) 263-1103

✓ 153

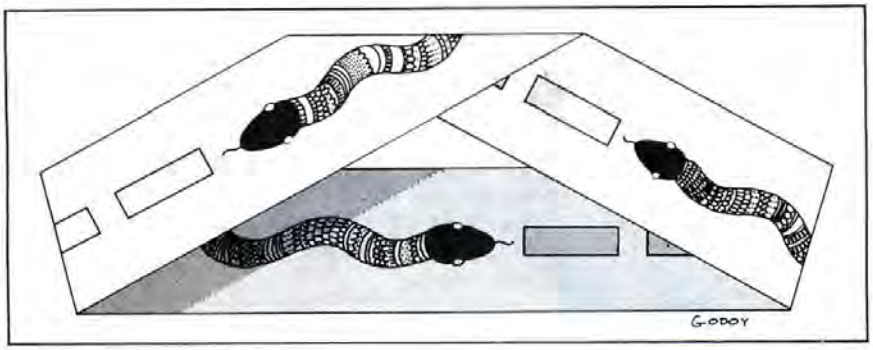
master charge

VISA

DISTRIBUTORS:
BI Tech, N.Y. (800) 645-1165 / Level IV Products, MI (800) 521-3305
Digital Distributing, TX (214) 330-1332
M & M Micro Mart, Quebec, Canada (514) 731-9486

PLEASE CALL OR WRITE FOR AUTHORIZED DEALER NEAREST YOU.
DEALER INQUIRIES INVITED.

✓ See List of Advertisers on Page 211



mines what kinds of moves the cursor makes when he writes the program.

The cursor starts out in the middle of the screen. A small symbol in the upper right corner of the screen constantly indicates the cursor's direction of movement.

You can use a repeat command (RPT) with the sample program above. This draws the same box, but the programming is simpler:

```
 5   RPT   4
10   FD    10
15   RT    90
20   ENDR
```

Other commands include Get (user input) and Text (print words on the screen). Snake also allows nested loops, variable use, subroutines, and If...Then statements, among other functions.

The operating system uses many simple Basic commands, including New, Load, Run, Delete, RENUM, LLIST, and DIR.

Error messages are clear and informative. They might include: Variable not initialized, Error in RPT loop, or Jump to missing line number.

The manual comes with sample programs and is easy to follow. All the commands are clearly explained.

Conclusion

Children find this language simple to learn: Its visual orientation makes Snake very exciting. Another plus is that the user gains experience with Basic.

I found Snake very entertaining and rewarding to work with. It's great that the Model III has a language comparable to Logo. ■

Snake

★★★★

James W. Burgmeier
P.O. Box 9241
So. Burlington, VT 05401
Model III
16K cassette
32K or 48K disk
$42.50

Easy to use?	★★★★☆
Good docs?	★★★☆☆
Bug-free?	★★★★☆
Does the job?	★★★★☆

From the Inside Out: Super Utility Plus

by Amee Eisenberg
***80 Micro* staff**

There's something sad about the guy who buys a Maserati and drives it like his old, slant-six Dodge; likewise the person who buys Super Utility Plus 3.0 and parks it on the shelf. Luckily Paul Wiener and Gary Camp, coauthors of *Inside Super Utility Plus: Series 3.0*, have written a book that can teach anyone to drive this super-charged utility.

First and foremost, *Inside Super Utility Plus* tutors you in using Kim Watt's newly revised, multipurpose disk utility (see the SU+ review in *80 Micro*, October 1983, p. 110). This volume is also a revision, essentially the same as the original *Inside Super Utility Plus*, but changed where the utility itself has changed.

The book begins with a "Technical Introduction" unique in computer literature for its clarity, informativeness, and friendliness. Even if you don't own Super Utility Plus, you might consider investing $19.95 just to read this chapter.

Its subject is your TRS-80's disk system. If you've ever wondered how the Z80, the floppy disk controller (FDC), and the disk drives communicate with each other, or how disk operating systems (DOSes) organize information on a disk, check out this book.

Because the utility works on almost every DOS available to the TRS-80 owner, the first step in using SU+ 3.0 is telling it what DOS is on your disk. In SU+'s first incarnation, this was no fun. Version 3.0 displays a configuration table that appears formidable at first glance, but with the help of Wiener and Camp's text it's easy to gain insight into the mysteries of DOS organization.

Since the business at hand is usually repairing a blown disk, the book provides a "problem recipe," a guide through the sections that deal with solving your current problem. In keeping with the book's overall style, these sections are clear, informative, step-by-step guides, telling you exactly which buttons to push and what SU+'s response will be.

Even better, these sections explain why something works or doesn't work, and what an error message really means. The voice used throughout the book is friendly: common sense advice about using your computer in general and SU+ specifically, offered

FD	Move forward
RT	Turn right
LT	Turn left
JP	Jump (draw no line)
POS	Position snake at coordinates

Table 3. Cursor movement commands.

```
 5   FD 10  (moves forward 10 steps)
10   RT 90  (turns snake 90 degrees right)
15   FD 10  (moves forward 10 steps)
20   RT 90  (turns snake 90 degrees right)
25   FD 10  (moves forward 10 steps)
30   RT 90  (turns snake 90 degrees right)
35   FD 10  (moves forward 10 steps)
```

Program Listing. Drawing a square.

↙ See List of Advertisers on Page 211

by someone who knows his stuff.

Of course, it's the kind of common sense that takes years of climbing around inside of computers to develop. Here, Wiener and Camp share their experience in a volume loaded with solutions for any number of disk problems.

In a section towards the book's end called "Undocumented Features in Super Utility," the authors describe some undocumented ways to configure SU+ 3.0 so that it automatically determines what DOS is on your disk (useful for getting into that unlabeled data disk your cousin in Australia just sent you).

You can use this book in two ways. The first is to read it from cover to cover while sipping a cold beer in the shade. Granted, some of the information won't make a lot of sense until you have a computer in front of you, but most of it will educate you in the workings of your disks and SU+.

The second is to use it like a cookbook; when your disk blows, find the problem recipe and cook according to directions. Both ways give you something of value.

My biggest complaint with this book (and its predecessor) is its lack of an index. You have to skim through the entire book to find anything. The table of contents is brief to the point of absurdity.

At this point, my copy is underlined in about 10 different colors. You'll have to find your own method of locating information. If there's an SU+ 4.0 and accompanying "inside" book, I hope it's indexed.

All in all, the difference between the guy who keeps SU+ 3.0 on the shelf and the guy who drives it around his disks is a few hours spent reading *Inside SU+: Series 3.0*. For my money, I'd rather drive. ■

Inside Super Utility Plus

★★★★

Paul Wiener and Gary Camp
Breeze/QSD Inc.
11500 Stemmons Freeway
Suite 125
Dallas, TX 75229
Softcover, 91 pp.
$19.95

Well-written?	★★★★★
Organized?	★★★☆☆
Thorough?	★★★★½
Readable?	★★★★½

BASIC Learning Programs

by Mary Gasiorowski

BASIC Learning Programs is another unexceptional package for teaching Basic programming.

A popular way to teach Basic is to provide sample programs that the student runs and modifies and then provide questions to encourage the learning process. *BASIC Learning Programs* follows this school of thought; the 82 programs are on disk and listed in the accompanying manual, with some explanations and exercises in the manual.

In terms of the number of commands covered, *BASIC Learning Programs* compares very favorably with other packages. It demonstrates most of the possible Basic commands, including Load, Run, List, Print, End, variables, math operations, REM, Input, Read-Data, GOTO, If-Then, For-Next-Step, arrays, GOSUB-Return, string variables, LEN, Right$, MID$, Left$, sorting, Clear, disk files, DIR, RND, INT, Tab and Print @, CLS, graphics, and CHR$.

The programs progress from easy to more difficult; all of them are short to key in on a single programming concept. The programs are grouped into the areas of LPs (introductory commands), Lists (one-dimensional arrays), Tables (multidimensional arrays), GOSUBs, Strings (and string functions), Files (sequential), Random (RND and INT), Tabs (Tab and Print @), and Graphics.

The manual presents a brief introduction to general computer concepts, then goes into the learning programs.

Each group of programs begins with some general information: the purpose of an array in programming or the importance of files, for instance.

The manual lists each program with its objective, a sample execution, some explanation of the new command, and some exercises designed to make the student modify or rewrite the program to further explore the commands' uses. The manual ends with a glossary of computer terms and an index.

One nice feature of this package is the authors' statement of permission (even encouragement) to make as many copies of the programs as you need. Finally someone recognizes the teacher's need to have enough materials for an entire class without spending hundreds of dollars buying legal copies.

The major problem with this package is that it isn't exciting—or motivating. The material is extremely sketchy and leaves a lot for the teacher to fill in. This is called *discovery learning*: With luck, the student discovers what the author is trying to accomplish. If the package provided more information, you'd need less luck.

The disks add nothing to the value of the package; you might as well type the programs in from the manual and save them on disk yourself. In addition, the more developed packages on the market today include answers to the exercises and suggestions to the teacher on use of the program.

As a set of supplementary materials for a knowledgeable and motivated teacher, *BASIC Learning Programs* might be useful. A computer-novice teacher who's looking for a Basic programming curriculum should keep looking. ■

BASIC Learning Programs

★ ★ ☆

Academic Computer Center
University of Wisconsin
River Falls, WI 54022
Manual $6
Disk (two for Model III only) $30

Well-written?	★★★☆☆
Organized?	★★★☆☆
Thorough?	★☆☆☆☆
Readable?	★★★☆☆

Write *Your First BASIC Program*

by Richard Ramella

Here is a clever, concise, entertaining beginner's approach to Basic programming.

Although the book earns its high rating with a scintillating presentation, its appeal might be too broad in this era of many versions of Basic. Mass appeal, even within the computer subculture, seems impossible.

Rodnay Zaks realizes his aim: to explain "a universal mini-Basic so that you can learn skills applicable to all versions of Basic." Now that's a good idea. The beginner can turn out pleasing but limited programs in a hurry, then go on to the manual with his machine and learn the fine points of his own Basic version.

Zaks gives you a precise tour of start-up Basic. Even if you've been programming a while, this book might inspire you to reconsider some ideas you've filed away without uncovering their full potential. *Your First BASIC Program* is a valuable cross-reference to your computer's manual.

Zaks covers the history of Basic, explains the keyboard, and gently eases the reader into an understanding of programming. The programming examples are limited by the available Basic commands.

The book's design and illustrations are superlative. The cartoon art, by a curiously uncredited artist, is a combination of high humor and calculated lunacy. It takes wild imagination to present a variable as a big-nosed little creature with antennae.

On the back cover is the statement: "If you're 8 years old or 88 and want to learn how to program a computer, this book is for you."

I happened to have an eight-year-old handy at that moment. "You understand this?" I asked. Then I opened the book randomly and read a paragraph.

"No," he said, "do it again, like, in English."

This book's lessons are for adults, but the kids will love the pictures. *Your First BASIC Program* is a well done introduction to programming. ■

Your First BASIC Program

★ ★ ★ ★ ½

Rodnay Zaks
Sybex
2344 Sixth St.
Berkeley, CA 94710
Softcover, 190 pp.
$9.95

Well-written?	★★★★★
Organized?	★★★★★
Thorough?	★★★★☆
Readable?	★★★★☆

↵ See List of Advertisers on Page 211

Portable Printer

Now you can complete your portable computer system with Sprinter, an 80-column, 160 characters-per-second dot-matrix printer. It features a high-speed space skip-over, five character sets including correspondence quality, built-in friction and tractor feed mechanisms, user-programmable character design, 4K buffer, and full graphics capability.

Options include data buffers of up to 68K, IEEE 488 interface, RS-232 interface, and the SoftSwitch keypad with battery backed-up RAM. The latter can control horizontal tabbing, selection of alternate character sets, form length, baud rate, horizontal print density, and much more. Its battery backed-up RAM retains your set functions even with the power off.

Measuring 18 by 15 by 7 inches and weighing 17 pounds, the Sprinter has a suggested list price of $795 from Micro Peripherals Inc., 4426 South Century Drive, Salt Lake City, UT 84107, 800-821-8848.

Reader Service ✓ 561

Infoscan: Now you can keep track of what's being published where in the computer industry.

Computer Magazine Index

A free, monthly computer guide called Infoscan helps you make sense of the multitude of computer magazines on the market. Published by Syncom, a manufacturer of 5¼-inch disks, the guide for personal computer users indexes over 400 articles each month from leading computing magazines like *80 Micro* (of course), *Softside, Compute*, and *PC World*.

If you want to keep up-to-date with the latest computer software and resources, contact Syncom at P.O. Box 130, Mitchell, SD 57301, 800-843-9862 and ask for your free subscription to Infoscan.

Reader Service ✓ 575

MPI's Sprinter, a lightweight portable printer and companion for your portable computer.

Building a legend: Clone, a disk copying utility.

List Maintenance

Microcomputer Services (639 Carroll St., Brooklyn, NY 11215, 212-857-9157), has written a dedicated List Maintenance software package that takes advantage of your Model II's sequential file access technique to store more than five times the data of other list software. The program can store approximately 8,000 names and addresses on a single 8-inch disk.

The system allows for updates, choice of label format, and printouts. All the programs are written in Basic and you can easily alter them to meet your individual requirements. It runs on the Model II, using TRSDOS and a dot-matrix printer, and costs $199.

Reader Service ✓ 582

Cloning

You've owned a great, but expensive, software program for months now, but you're worried that it may crash soon, forcing you to buy another copy. What do you do? Try Clone, an advanced disk-copying utility. It copies any disk regardless of density, password, or copy protection, and does so faster than your DOS's back-up utility.

It takes Clone just 1 minute and 25 seconds to format and copy a disk. It duplicates Model I, III, 4, and Color Computer disks. It handles all formatting chores automatically, and doesn't care how or if you previously formatted the disk.

As a bonus feature, Clone is excellent at reading damaged disks and unusual formats. When it finds a problem, you can command it to keep trying, take its best guess, or give up. Clone costs $79.97 from Gibberman Enterprises, 13000 San Fernando Road, Sylmar, CA 91342, 213-367-0887.

Reader Service ✓ 551

SAT Prep

We all know that the SAT plays a predominant role in determining who goes to which college. You might make certain your child gets to attend the college he or she deserves.

EXTRAORDINARY VALUES

PANTHER

5 MEG	$1995.00
5 MEG + 5 MEG REMOVABLE	2895.00
10 MEG	2295.00
10 MEG + 10 MEG	3495.00
10 MEG + 5 MEG REMOVABLE	3395.00
15 MEG	2395.00
15 MEG + 15 MEG	3795.00
15 MEG + 5 MEG REMOVABLE	3495.00
Underdesk mounting bracket	34.95
Boot Rom (Model III & 4)	34.95

PANTHER CUB

5 MEG FIXED *MASTER	$1795.00
5 MEG FIXED +SLAVE	1349.00
5 MEG REMOVABLE *MASTER	1995.00
5 MEG REMOVABLE +SLAVE	1495.00
Cartridge	99.95

* Master Drive includes power supply, disk controller, I/O adapter, enclosure and applicable cabling.
+ Slave Drive includes power supply, enclosure, connector cabling.

ADAPTER MODULES

IBM PC/XT • Apple • ComPAQ • LNW-80, II • Franklin • TRS-80 Model I, III, 4
Coming soon Xerox 820 — II adapter. • 180 day warranty • Information available upon request.
CALL FOR YOUR LOCAL PANTHER/PANTHER CUB DEALER.

Model 4 $1699.

128K 2 drive, VR-RS232 Sound generator, TRS-DOS & manual

Disk III kit

Internal floppy disk subsystem for Model III & 4 (includes controller, power supply, cabling, bracket and all associated mounting hardware).

DISK III KIT — Without Drive	$299.95
DISK III KIT — With 1 drive (ss,dd)	449.00
DISK III KIT — With 2 drives (ss,dd)	699.00
External Drives includes power supply, enclosure and extender cable	$99.00
2 drive cable	25.00
VR-RS232	80.00

SPECIAL BUYS ON COD & PREPAID ORDERS ONLY:

Epson RX-80	$399.00	Smith-Corona Printwheels	$4.98	Generic Diskettes	$1.75
Epson RX-80FT	425.00	Smith-Corona Ribbons	4.25	Green Screens	99.00
Epson FX-80	650.00	Spinwriter Multistrike Ribbons	4.95	Percom Doubler	169.00
Epson FX-100	850.00	C. Itoh & Diablo Ribbons	7.50	DC Hayes Smartmodem 300 baud	229.00
Epson MX-100	750.00	Verbatim Diskettes 525	26.90	DC Hays Smartmodem 1200 baud	569.00
Epson Ribbons	Call	Nashua 8" Diskettes	42.95	Paper 9½" x 11" fanfold	29.95
Epson Graftrax	75.00	8" Library Cases	4.00		
Amber Screens	99.00				

Call Toll Free • 800-345-8102

Published prices reflect cash discount. All prices are subject to change without notice. TRS-80 and TRSDOS are trademarks of Tandy Corp.

Telephone Hours: 8:30am—7pm MON-FRI., SAT 10-3 EST
CABLE "VRDATA" TELEX 845-124

MasterCard VISA AMERICAN EXPRESS

VR data

(215) 461-5300
777 Henderson Blvd.,
Folcroft, PA 19032

VR DATA — WEST
WATONGA, OK
1-405-623-8664

See List of Advertisers on Page 211

NEW PRODUCTS

Traveling Software's Business Manager Series: It's like taking your office with you.

There are several tutorial methods available, and one of the best is Krell Software's How to Beat the SATs.

A 42-program series that comes on seven disks, the system prepares your child for the SATs in English grammar, mathematics, word relationships, vocabulary, word problems, and more. Every section realistically simulates what the real McCoy looks like in both appearance and difficulty level. It gives your child a competitive edge.

The entire program costs $299.95 from Krell Software Corp., 1320 Stony Brook Road, Stony Brook, NY 11790, 516-751-5139. They offer a full money-back warranty if your child doesn't increase his/her score by a minimum of 70 points. 1470, here I come!

Reader Service ✓ 580

Portable Billing System

The Time Manager is one of eight business programs for the Model 100 offered by Traveling Software Inc. (11050 Fifth Ave. NE, Seattle, WA 98125, 206-367-8090). It provides you with a complete time billing system. Time expenditures are recorded by project, client, and/or by individual work activity. You can specify hours as nonbillable or billable.

The program allows for flexible reporting of summary information on the LCD display, or schedules printed on a printer. Printed reports are designed for both an 80- and 40-column format printer.

You can use the Time Manager in a variety of ways, including managing hourly equipment and machinery use, keeping track of student homework assignments and test scores, and even recording the number of miles an athlete runs. The system costs $59.95. Contact Traveling Software Inc. for information on the other seven programs included in this business software series.

Reader Service ✓ 553

Bulls or Bears?

If you're into Stock Market investments but can't decide which stock to invest in, consult Tele-Stock before putting your money down. It's a stock market inquiry, analysis, and portfolio management system developed for the Model 100, and it's designed to

✓ See List of Advertisers on Page 211

The DTC Style Writer comes standard with 35K buffer memory for speedy output of 20 pages of text.

help you make good management decisions.

Tele-Stock uses the autodialing modem and real-time clock features of the computer to automatically call the Dow Jones News Service any time during the day and retrieve market activity on the stock of interest. The retrieved data automatically updates history files for each stock, which are then used to display and print the latest market activity. It also graphically presents the high, lows, close, and volume, as well as the return-on-investment calculations.

The program requires a 24K Model 100 with either direct-connect modem or acoustic coupler, and costs $59.95 from Telesoft, 939 Deerspring Place, Newbury Park, CA 91320, 805-499-6271.

Reader Service ✓ 562

High-Quality Printer

The DTC Style Writer is an inexpensive answer to your printing needs. A letter-quality printer, the Style Writer features a 35K buffer that stores up to 20 pages of text and a printing speed of 11 characters per second. Other notable features include an interchangeable print wheel cassette, bidirectional printing, automatic proportional printing, multicopy feature, automatic underscore, sub- and superscript, graphics plotting capability, printing pause switch, and two-color printing.

A special bonus is a self-test diagnostic routine that completely evaluates the printer's internal electronic circuits and print mechanism. All error conditions are indicated by labelled LED lights. Options for the Style Writer include a forms tractor for continuous paper feed, a bidirectional cut-sheet feeder, and 17 different type fonts.

The Style Writer measures 18 by 13 by 7 inches, weighs about 20 pounds, and costs $899 from Data Terminals and Communications, 590 Division St., Campbell, CA 95008, 408-378-1112. As an option, a 67K expanded buffer is available at $49.

Reader Service ✓ 560

Complete Disk Security

Want to make certain that no one accesses sensitive data disks or tapes? Store them in the Computer Software Safe and rest easy tonight. The safe protects up to 40 disks against fire and theft. In fire tests during which the safe was exposed to temperatures of up to 1700 degrees, the safe's interior remained below 125 degrees, and the relative humidity stayed below 85 percent. No data was lost.

The safe, Model 5750, features a 4-inch solid-steel locking bolt, two steel bolts for added burglary resistance, three-number combination lock, a three-year warranty, and additional space for software documentation. It has a list price of $600, or $549 directly from the manufacturer. For additional information, contact Value-tique Inc., Dept. EDP-36, P.O. Box B, Leonia, NJ 07605, 201-461-6500.

Reader Service ✓ 555

A Better Disk?

Quality is extremely important in a floppy disk;

Continued on p. 258

The Computer Software Safe keeps your valuable data disks safe from burglary and fire.

✓ See List of Advertisers on Page 211

MODEL 4 DRIVE KITS $299

MODEL 4 STUFF
MODEL 4 COMPUTERS

all include an RS232

No charge for shipping on any Compukit Computers
64K one single headed drive . . . . . *$1495.*
64K two single headed drives . . . . . *$1695.*
64K two double headed drives . . . . . *$1895.*
128K systems add only . . . . . *$99.*

MODEL 4 64K UPGRADE $62.95
64K TO 128K UPGRADE $82.95

prime grade pre-tested RAM with instructions

MODEL 4/III RS232 KIT $69.95

completed and tested ready for installation

Model I, III, 4 green phosphor antiglare CRT Kit $89.

a complete new CRT, not a filter

MODEL 4 DISK DRIVE UPGRADE KITS

All of the Compukit Model 4 Disk Drive Upgrade Kits contain these features that other companies usually do not provide. Switching power supplies * Tandon disk drives * 64K of Model 4 RAM * Sound * and an Easy to Use installation Manual. Requires only a screwdriver (no soldering) The no drive upgrade kit . . . . . $299.
One drive upgrade kit . . . . . $499.
Two drive upgrade kit . . . . . $699.
Two double headed drive kit . . . . . $899.

MODEL III to MODEL 4 UPGRADE $750.

Converts your Model III into a Model 4 (except for Cabinet and disk drives). Includes new keyboard, 64K RAM, Sound, and free installation (required). Ship us any working Model III, even if it's not all factory equipment, and get back a Model 4, in your case.

MODEL 4 SOUND UPGRADE $29.95

No soldering required, Includes instructions

COMPUKIT DOCTOR $29.95
Disk based diagnostic software package for the Model I, III, and 4

MODEL III DISK DRIVE UPGRADE KITS

All of the Compukit Model 3 Disk Drive Upgrade Kits contain these features that other companies usually do not provide. Switching power supplies * Tandon disk drives * 32K of Model III RAM * Compukit Doctor * and an Easy to Use installation Manual. Requires only a screwdriver (no soldering).
The no drive upgrade . . . . . $279.
One drive upgrade kit . . . . . $479.
Two drive upgrade kit . . . . . $679.
Two double headed drive kit . . . . . $879.

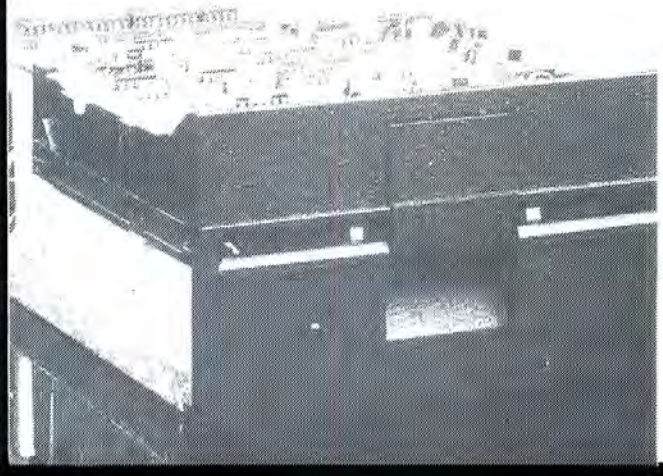
TANDON DISK DRIVES

Perfect for replacement or add on drive for Any 5" drive system.

TM100-1 40 Track Single Sided . . . . . $199.
TM100-2 40 Track Double Sided . . . . . $299.
TM50-1 20ms trk-trk Single Sided Thinline . . . . . $149.
TM50-1 6ms trk-trk Single Sided Thinline . . . . . $169.
TM50-2 6ms trk-trk Double Sided Thinline . . . . . $219.
TM55-2 microprocessor Double Sided Thinline . . . . $249.
Single case with extenders with drive above $49.95
Dual case with extenders for drives above $79.95
Two Drive Cable $24.95

64K COCO KIT $62.95

COCO DRIVE O
$379.
with a Tandon Thinline
SPECIAL INTRODUCTORY PRICE
DUAL DRIVES $529.



COMPUKIT CORPORATION

ordering Information
We accept Visa, Mastercard, Wire Transfers, and Certified Checks for quickest shipping. Orders received on personal checks are held for clearance.

1-713-480-6000 order line 1-800-231-6671
16206D Hickory Knoll, Houston, Texas 77565

COMPLETE MODEL III/IV

HARD DRIVE $1095.

COMPLETE PRIMARY DRIVES
MODEL III & IV

5 MEG $1095.
10 MEG $1395.
15 MEG $1695.

MODEL I & LNW

5 MEG $1145.
10 MEG $1445.
15 MEG $1745.

COMPLETE SYSTEM FEATURES
TANDON DISK DRIVES
WESTERN DIGITAL 1002 DRIVE CONTROLLER
ONE YEAR WARRANTY
SWITCH SELECTABLE HOST
POWER ONE HD POWER SUPPLY
HD COOLING FAN
SIZE 8-1/4"X6-1/2"X13"
ALL DRIVES RATED AFTER FORMAT
COMPLETE READY TO PLUG IN (JUST ADD YOUR DOS)
SUPPORTS NEWDOS 80 V2.5, DOSPLUS 3.4f, 4.0, 3.5, IV, LDOS, and soon TRSDOS 6.0

SECONDARY DRIVES

5 MEG $695.
10 MEG $995.
15 MEG $1295.
(READY TO PLUG INTO YOUR PRIMARY DRIVE)

NEW!

33 MEG MODEL PRIMARY HARD DRIVE (BY QUANTUM) $2395.
MODEL I & LNW $2445.

NEW PRODUCTS

READY FALL 83 *CALL ON AVAILABILITY*
21 MEG TAPE ***$649.***
Will Work on any Hard Drive System That uses DOSPLUS or NEWDOS

MODEL 3/4 BOOT ROM ***$39.95***
ALLOWS YOU TO BOOT DIRECTLY FROM YOUR HARD DRIVE FOR DOSPLUS and NEWDOS SYSTEMS ONLY

MULTIPLEXER $995.

ALLOWS UP TO 4 COMPUTERS TO ACCESS A HDS HARD DRIVE
Includes Master Control Unit and Cable / Host Adapters for externally connecting 2 Computers

HARD DRIVE SPECIALIST *a division of Compukit Corporation*

1-800-231-6671 ***1-713-480-6000*** ***16206D Hickory Knoll Houston, Texas 77059*** ↙455

↙ See List of Advertisers on Page 211

NEW PRODUCTS

Continued from p. 255

hence, Mensa Media was born. Introduced by Adams Magnetic Products Inc. (194 Passaic St., Hackensack, NJ 07601, 201-488-3993), the new 5¼-inch disk line is produced with high quality in mind: They've passed the testings of major industrial users, and were found to be exceptional in performance. In fact, the Mensa Disks even passed a severe test from Memorex.

The disks have either 48 or 96 tracks and come single-sided, single-density, or single-sided, double-density, or double-sided, double-density. Boxes of 10 disks cost $39.90, $42.90, and $52.90 respectively. ADM anticipates selling the disks individually in lots of two or three in the near future.

Reader Service ✓ 568

Feed Your Printer

Nothing grates on your nerves more than hand-feeding cut sheet paper to your printer. Personalfeeder, from Ziyad (100 Ford Road, Denville, NJ 07834, 201-627-7600), does the job for you. It works automatically in an unattended mode to insert, remove, and collate cut sheets of paper.

The paper bin holds up to 200 sheets of paper, ranging in length from 6 to 14 inches. The printer automatically collates the paper in print sequence. The feeder's design allows printing of superscripts, subscripts, and reverse lines. A feed slot also lets you manually insert forms, envelopes, and card stock without removing the feeder.

The Personalfeeder is presently available for the Diablo 620, 630, and 630 ECS printers; the Qume Sprint 8, 10, and 11 printers; the Ricoh 1300 and 1600 printers; the NEC 3500 and 7700 series printers; and TEC F10 printers. It has a suggested price of $495.

Reader Service ✓ 573

Totable Color Graphics

The MikroKolor Color Graphics Interface board gives your Model 100 high-resolution graphics and text capability using a standard color television or monitor. It provides you with four operation modes: a Text mode that displays 24 lines of 40 characters each, and gives you 256 definable characters; a Multicolor mode that provides 64- by 48-pixel color graphics; a Graphics 1 mode that provides 256- by 192-pixel color

The Personalfeeder is a single-bin paper feeder for letter-quality impact printers.

✓ See List of Advertisers on Page 211

graphics, 24 lines of 32 characters each, using an 8-by 8-dot matrix with two colors per character; and a Graphics 2 mode with the same features as the Graphics 1 mode except that it allows 16 colors per character.

The board requires no hardware modifications—it plugs into your Model 100's existing expansion socket, and comes with 16K of RAM. The MikroKolor Model 100 costs $335 coming fully assembled and tested; $295 for the unassembled kit with documentation. For further information, contact Andreasen's Electronics Research & Development Inc., 1548 Monterey St., San Luis Obispo, CA 93401, 805-541-6398.

Reader Service ✓ 576

Bringing the TRS-80 to Kids

If you have a child, or know of someone who does, why not introduce them to the TRS-80? To help make them computer literate, you could give them Michael Zabinski's book entitled *TRS-80 for Kids 8 to 80, Volume 1.*

It's a self-paced guide for beginners, offering instant exposure to programming concepts, commands, logic, disk storage, printing, and more. The author's made it fun and easy to use, with a generous number of illustrations. Games, quizzes, experiments, and other participation activities offer the child practice as they check progress.

Introduce your child to the TRS-80 with this introductory and fun book.

A paperback, *TRS-80 for Kids 8 to 80* costs $9.95 from Howard W. Sams & Co. Inc., 4300 West 62nd St., Indianapolis, IN 46268, 317-298-5400.

Reader Service ✓ 565

New Diskette Family

If you're not satisfied with your present disks, perhaps you should take a look at the new diskettes being produced by Comrex Internation Inc. (3701 Skypark Drive, Torrance, CA 90505, 213-373-0280). The four new models (CR-10005, CR-10010, CR-10015, and CR-10020) are all configured with a soft-sectored format, a single index hole, 48-track-per-inch density, and 40 tracks for each recording surface.

The Comrex disk is a flexible disk permanently enclosed in a semi-rigid, protective jacket. The jacket contains a liner that cleans the disk as it turns. The disks are certified to be

The new family disk line from Comrex has a protective jacket that cleans the mylar.

See List of Advertisers on Page 211

The Holmes PMD-100: a fast and efficient storage system for the Model 100.

100 percent error-free. They are tested for signal amplitude, resolution, low-pass modulation, overwrite, and more. They meet or exceed all standards set by IBM, ANSI, ECMA, Shugart, ISO, and JIS.

Ten-pack prices for these disks are $29.95 for the single-sided, single-density disk (CR-10005); $44.95 for the single-sided, double-density disk (CR-10010); $49.95 for the double-sided, double-density disk (CR-10015); and $59.95 for the premium double-sided, double-density disks (CR-10020). The latter comes with a free hard plastic storage box that normally sells for $7.50.

Reader Service ✓ 558

Portable Micro Drive

One of the Model 100's drawbacks is its restricted data storage capability. Well, the PMD-100 Portable Micro Drive from Holmes Engineering (5175 Green Pine Drive, Salt Lake City, UT 84107, 801-261-5652) changes that. It's a completely portable, battery-operated system that lets you save and load programs and files at high speed using miniature continuous-loop tape cartridges.

The drive attaches to your Model 100's RS-232C connector via a coiled telephone cord. The operating system is then downloaded to the 100 using the built-in Telcom software built into the computer. To use the drive, insert a tape cartridge and save or load from there. You can delete files from anywhere on the tape, and save another file in its place.

The PMD-100 operates at 9,600 baud, and a 16K RAM buffer in the PMD-100 makes file transfers fast and efficient. The drive fits with the Model 100 into a standard briefcase, and its rechargeable batteries give you hours of portable operation.

The Portable Micro Drive retails for $349.95 including five wafer tapes, a battery charger, a cable to connect the drive to the computer, and complete operating instructions. A cable that lets the Model 100 operate from the battery in the PMD-100 is also included.

Reader Service ✓ 557

Double Up

Double your present media storage without having to buy another disk. The Disk Doubler kit lets you use the back sides of your disks to store additional

programs and data. It comes with a template, hole punch, and other equipment necessary to alter the floppy disk jacket, allowing insertion of either side of the disk into a drive.

The four-step alteration process is safe and fast. The Disk Doubler works with any single-sided, soft-sectored 5¼-inch floppy disk. It has a suggested retail price of $19.95 from Evolving Technology Co., 3725 Talbot St., Suite F, San Diego, CA 92106, 619-224-3788.

Reader Service ✓ 556

Computer Lighting

A common problem for computer owners is trying to find a light that's soft enough to reduce glare on paper, but bright enough so that you can easily read a program listing. The IPL-600 meets both of these requirements. It's a cool, color-corrected, minifluorescent bulb that draws only 13 watts while yielding as much illumination as a conventional 75-watt bulb.

The light has a custom-designed reflector that pro-

Shed a little light on your programs and text with this award-winning light.

✓ See List of Advertisers on Page 211

NEW PRODUCTS

duces a solid, even illumination with no hot spots. The continuously curved surface directs the light down in a sharply defined pear-shaped pattern. It has an arm reach of 28 inches, and is easily moved into any position.

Made of rugged ABS plastic, the IPL-600 is UL listed and features a grounded three-wire cord and plug for safety. It costs $99.95 from Ledu Corp., 25 Lindeman Drive, Trumbull, CT 06611, 203-371-5500. It's available in almond, brown, or black.

Reader Service ✓ 577

Scientific Graphs

PlotPro is a set of three MicroSoft Basic programs that make scientific graphs on any 80- or 132-column printer. It creates linear, semi-logarithmic, and full logarithmic plots and can plot multiple functions on the same graph. The programs also support forced scaling and auto-scaling, as well as optional grid lines to aid in graph interpretation.

One of the programs, ProTemp, creates templates of the physical appearance of the graph. With these templates you can specify the type of scaling, axis labeling, ranges for each axis, user-specified control characters, and other information. The ProQuick module controls plotting and printing of infinite length graphs limited only by paper length.

Besides producing graphs, you can also use PlotPro to generate vertical formats suitable for reports, viewgraphs, and so on. PlotPro requires no special plotters or complicated data interfaces. It generates data files in Basic, Fortran, Pascal, or Assembly language. Available for the Model I/III/4 on TRSDOS, PlotPro costs $49.95 including a manual, from BV Engineering, P.O. Box 3351, Riverside, CA 92519, 714-781-0252.

Reader Service ✓ 552

Learn a Language

Only your knowledge of the language will save a Spanish bullfighter from the charging bull or an endangered French aristocrat from the guillotine. Two new education games, La Corrida de Toros and La Guillotine, help teach you Spanish and French. Based on the popular 'Hangman' word game, you're given the number of letters of a word and must guess that word by selecting the letters to fill in. If you win, the game figure survives. If you lose...well, the computer shows you what happens.

You can select vocabulary words from five different categories: days, months, weather; restaurant; school; the house; and professions. Once you've decided on a category, you can review and study the list of words and their translations, do a matching exercise to test your memory, or play the game. You're allowed four incorrect letter choices before the guillotine drops or the bull charges.

Available for the Model I/III/4, both games have a suggested retail price of $27.95 from Gessler Educational Software, 900 Broadway, New York, NY 10003, 212-673-3113.

Reader Service ✓ 579

Graphics Editor

Never quite satisfied with your graphics? Why not use ZGraph, a powerful machine-language graphics editor package. It gives you the tools to construct screen images using your computer's block graphics capabilities. You can save these images to disk and convert

70 INCOME TAX PROGRAMS

(For Filing by April 15, 1984)
TRS-80* Models I, II, III, 4, 12 & 16
(Also for APPLE 3.3 DOS)

FEATURES: —

1. *Menu Driven.*
2. *"Save on Disk"*
3. *BASIC, Unlocked, Listable*
4. *Name/SS No./F-S carried over*
5. *Inputs on screen before printing*
6. *I.R.S. approved REVPROC format printing*
7. *Prints entire Form/Schedule*
8. *Calculates Tax etc.*
9. *For Mod. III/4, CONVERT*
10. *For Mod. 12/16, use 2.0b*
11. *Use GREENBAR in Triplicate — don't change paper all season!*
12. *Our 5th Year in Tax Programming*
13. *We BACK-UP our programs!*

For the Tax Preparer, C.P.A., Lawyer and Individual. 70 Tax Programs on 13-5 1/4", Format disks, or on 3-8" Format disks. Order only the disks you'll use.

Programmed for easy-use. Follow the Form or Schedule closely. Check-points along the way. Results on screen before printing.

70 TAX PROGRAMS include: Forms 1040, 1040A, 1040EZ, 1120, 1120S, 1065 and 1041. Schedules A, B, C, D, E, F, G, R, RP, SE and W. Forms 1116, 2555, 2106, 2119, 2210, 2440, 2441, 3468, 3903, 4136, 4137, 4255, 4562, 4684, 4972, 4797, 5695, 5884, 6251, 6252 and 6765. Order only the disks you need.

Also we have TAX PREPARER HELPER: includes 12 PROGRAMS such as INCOME STATEMENT, RENTAL STATEMENT, SUPPORTING STATEMENT, IRA, ACRS, 1040/ES, ADD W-2's, PRINT W-2's and others.

MasterCard VISA

TRY ONE DISK AND SEE FOR YOURSELF. 5 1/4" DISK IS $24.75; THE 8" DISK IS $29.75, BOTH POSTPAID.

Write:—

GOOTH TAX PROGRAMS

931 So. Bemiston • St. Louis, Mo. 63105

*T.M.Reg. by Tandy Corp. Ft. Worth, Tx.

✓ 185

MAKE IT EASY TO SAVE your copies of 80micro®

Your magazine library is your prime reference source—keep it handy and keep it neat with these strong library shelf boxes. They are made of white corrugated cardboard and are dust resistant. Use them to keep *all* your magazines orderly yet available for constant reference.

Self-sticking labels are available for the following:

80 Micro	*73 Magazine*	*Radio Electronics*
Microcomputing	*QST*	*Personal Computing*
inCider	*CQ*	*Byte*
Desktop Computing	*Ham Radio*	*Interface Age*

One box (BX1000) is $2.00, 2-7 boxes (BX1001) are $1.50 each, and 8 or more boxes (BX1002) are $1.25 each. Be sure to specify which labels we should send.

Call **TOLL-FREE** for credit card orders:

1-800-258-5473

Or use the order form in this magazine and mail to:

80micro®

Attn: Book Sales, Peterborough, NH 03458

☐SHIPPING AND HANDLING CHARGES $2.00 per order up to and including a quantity of eight 25¢ for each additional box ordered.☐

NOW ONLY $99.00

CopyArt II

WILL CHANGE THE WAY YOU THINK ABOUT YOUR TRS-80!

26. SIMPLE CURSOR commands. Simply use the arrow keys to move your cursor around the text. The screen will scroll both vertically and horizontally. Shift arrows take you to the beginning or end instantly.
27. Hi-Resolution graphics supported.
28. COMPLETE MARGINS CONTROL. You tell CopyArt II what margins you desire. You can even change margins within the same text. You may also have parts of your text with 2 columns, some with one etc. It's super easy to use.
29. BASIC PROGRAMS can be edited easily. CopyArt is really useful for inserting graphics within quoted strings to give your programs super animation without the hastle of calculating the CHR$ of the graphics!
30. VISICALC files can be loaded into CopyArt II to be manipulated easily. Great when you want to accompany your Visicalc reports with written reports, GRAPHS and BOLDFACING etc. Visicalc reports up to 255 wide can be loaded.
31. SPECIAL SCRIPSIT FILE LOADER. Allows you to load your old Scripsit files without having to save them in ASCII. Copyart will also load Pencil files and other normal ASCII files.
32. Similar to Scripsit. If you have used Scripsit, you can use Copy Art in minutes.
33. CONTROL CODES. Lets you insert special printer control codes in your text. CODES between 0 and 255.
34. BLOCK MOVE. Simple and powerful block move. Lets you move paragraphs or lines of text around easily. No complicated marker settings required.
35. FIND/REPLACE/REPEAT. Lets you find a string of characters and replace them with any other string of characters up to 20,000 times! WILDCARD search also supported.
36. Professional Manual in easy to understand English.
Copyart II requires a TRS-80 Model I or III, (or PMC-80 or LNW), 48k and 2 disk drives with Newdos-80, Ldos, Multidos, Dosplus or TRSDOS. Double density disk drives recommended for the Model I.

CopyArt II THIS PROGRAM DOES IT ALL!

PLEASE SPECIFY which COMPUTER and PRINTER you have when ordering.

Copyart 11 with one printer driver 99.00
Additional printer drivers ..... 19.95 each

Copyart I owners. Updates available. To registered owners for $15.00.

COD and Credit Cards CALL TOLL FREE to order: 1-800-528-1149
Or send check or money order to:

SIMUTEK

✓ 163 Computer Products Inc.

SIMUTEK COMPUTER PRODUCTS INC., 4897 E. SPEEDWAY BLVD., TUCSON, AZ 85712, (602) 323-9391
DEALER, DISTRIBUTOR, & PRINTER/MANUFACTURER INQUIRIES INVITED
TRS-80 and Scripsit are TM of Radio Shack a Tandy corp.
***Indicates printer must have capability to do function.**

✓ See List of Advertisers on Page 211

NEW PRODUCTS

The RM 1000 Radio Modem lets you transfer news and wire services to your computer.

them into a format Basic and machine-language programs can use.

The ZGraph editor provides automatic line, circle, and rectangle generation. You can duplicate, magnify, or reduce sub-images. You can also invoke windowing, reverse-imaging, or flipping the X or Y axis on the entire screen. Besides the editor, the package includes five utility programs with which you can create, display, and manage ZGraph screens.

ZGraph is for the Model I/III with LDOS, while Pro-ZGraph is for the Model 4 operating under TRSDOS 6.0. Each costs $50 plus $2 shipping from Misosys, P.O. Box 4848, Alexandria, VA 22303, 703-960-2998.

Reader Service ✓ 563

Radio Modem

The RM 1000 is a modem that lets your computer send and receive Morse code and radioteletype over a radio. It lets ham operators and shortwave radio listeners copy news and wire services. It features commercial-quality demodulators, dual bar graph tuning, and extensive software capabilities.

This radio modem offers three RTTY shifts selectable from your computer keyboard. A hardware clock continuously displays time and you can insert the time into text in any format.

The modem's multilevel split-screen display lets you see transmitted and received text in chronological order. A review window lets you edit text that has scrolled off the screen. A buffered ASCII parallel printer output lets you print current text or text in the review window.

A wall plug transformer supplies power for either 110- or 220-volt operation. The RM 1000 costs $239 from Macrotronics Inc., 1125 N. Golden State Blvd., Turlock, CA 95380, 209-667-2888. Software and interface card/cables for the Models I, III, and 4 cost an additional $99.

Reader Service ✓ 572

Model 100 Spreadsheet

Now you can play around with budgets and financial proposals in the comfort of your home, thanks to PortaCalc, an electronic spreadsheet program for the Model 100 from Skyline Marketing Corp. (4510 West Irving Park Road, Chicago, IL 60641, 312-286-0762). The system features a 14-column by 26-row workspace and uses the built-in function keys to save, load, screen print, report print, or to look behind the data at the formulas in use.

✓ See List of Advertisers on Page 211

You can save, load, or merge worksheets using the computer's internal memory or cassette storage. You can also save often-used templates in memory for instant recall. Two utility programs are included with PortaCalc: PortaPrint and PortaDex.

PortaPrint is an enhancement to the Model 100's text editor, adding the ability to adjust left, right, and top margins, as well as page length. It also lets you control page numbers, headers, centered lines, pagination, and more.

PortaDex is a data exchange program that reformats PortaCalc files into the DIF format used by VisiCalc.

The system is supplied on tape, and requires 24K RAM. It costs $69.95 plus $2 handling, and includes illustrated instructions and a quick-reference card.

Reader Service ✓ 554

Your Own Mouse

The Joy-Mouse is a unique hardware add-on that interfaces any device designed to be plugged into the Color Computer joystick port with your Model III/4. The hardware provides instantaneous high-resolution X and Y position values ranging from 0–255. It also provides sound and music. The cassette cable plugs into the built-in audio amplifier that features proportional volume control with an on/off switch.

Four analog-to-digital ports are available for monitoring any analog signals, such as temperature, wind speed, light intensity, voltages, and so on. You can use the Joy-Mouse with either Basic or Assembly-language programs. It also works with all game programs written for joysticks.

The Joy-Mouse comes in a black plastic case with its own power supply, and connects to the I/O expansion port. However, the I/O bus is extended so you can connect other peripherals at the same time. The special introductory price for the Joy-Mouse is $99.95 from Micro-Labs Inc., 902 Pinecrest, Richardson, TX 75080, 214-235-0915.

Reader Service ✓ 578

Printer PJ's

Protect your printer from dirt and abusive handling with a tailored cloth cover from Discovery Designs Center (P.O. Box 72289, Roselle, IL 60172, 312-893-5468). The cover's professional design avoids the "kitchen-appliance" look by using rich colors and fabrics that match any decor. You can select from tan duck with navy blue trim, blue denim with white trim, or luxurious chestnut suede cloth trimmed in chocolate brown.

These protective printer PJ's are designed to fit the Epson, NEC, Smith-Corona, Apple, IBM, Commodore, Atari, and similar-sized printers. Also, the covers don't contribute to static-electricity problems. Each printer cover costs $12.50 which includes first-class mail delivery.

Reader Service ✓ 574

What Now?

What Do You Do After You Plug It In? That's a

✓ See List of Advertisers on Page 211

common question asked by first-time computer owners. It's also the title of William Barden's new book published by Howard W. Sams & Co. Inc. (4300 West 62nd St., Indianapolis, IN 46268, 317-298-5400).

This 198-page text presents a complete tutorial on the use of microcomputer hardware, software, languages, operating systems, and data communications, followed by a second tutorial on workable solutions to the practical problems that occur during their use. Barden also covers packaged applications software, bubble memory, high-resolution graphics, disk drives, monitors, print heads, and so on.

What Do You Do After You Plug It In? is written in a clear and humorous easy-going style with illustrative figures and numerous examples demonstrating key ideas. It costs $10.95 from participating Sam's dealers and bookstores nationwide.

Reader Service ✓ 564

Take Your Marks...Set...Go!

Now you can instantly see any portion of a printout on a monitor instead of having to wait for a printout. The Sprinter is a 62K printer buffer with a video monitor output and a keypad to let you scroll through its contents. The 12-button keypad features bidirectional scrolling as well as the ability to jump between any tab points you set.

The Sprinter is thin—it measures just 12 by 9 by 1 inches—and it's made of heavy gauge steel on which you can place a monitor. The display is switch-selectable for either 80- by 24-pixel or 64- by 14-pixel resolution. The baud rates are also switch-selectable, ranging from 300–19,200 baud.

MPI's Sprinter, a lightweight portable printer and companion for your portable computer.

No modification is necessary to your computer or printer, and the Sprinter is RS-232C compatible. The serial interface version costs $680; parallel interface $580. For further information, contact The Alien Group, 27 West 23rd St., New York, NY 10010, 212-741-1770.

Reader Service ✓ 571

A Relational Manager

Map-Mate is a relational data-base management system that applies the signature screening method to numerical data and structured text, resulting in faster searches. Menu-driven, it handles primary and secondary records (up to 50 fields in each), allows quick multiple-field searches, and has a powerful report generator.

The system runs on either

✓ See List of Advertisers on Page 211

NEW PRODUCTS

No, it's not the lunar rover. It's the Scorpion, a robot toy for the serious programmer.

the 64K Model II/12/16 or the 48K Model III/4. MapMate costs $295 from Softshell Corp., P.O. Box 18522, Baltimore, MD 21237, 301-686-1213. An introductory version is available for $35.

Reader Service ✓ 581

A Serious Toy

Everyone still has the child inside them, but most adults are afraid to admit it. Now you can with the Scorpion, a small and highly sophisticated robot. Resembling a miniature lunar explorer, it is completely software programmable with two 6522 interface chips providing 32 lines of I/O and four programmable timers.

The Scorpion's main feature is its versatility. It can see, make noises, and move along the ground in any direction at 99 different speeds. With its eight microswitches, it can learn the position of obstacles in its environment and retrace a preset path, avoiding these obstacles in the future.

The Scorpion follows instructions and answers questions from any host computer with an RS-232C serial interface. Its instructions are translated into action by the computer on board that contains a 6502 microprocessor, 8K of EPROM, and 2K of RAM (expandable to 64K).

This toy for the sophisticated hobbyist costs $660 from Rhino Robots Inc., P.O. Box 4010, Champaign, IL 61820, 217-352-8485.

Reader Service ✓ 567

File Recovery

Every Ryan-McFarland Cobol and Compiled Basic user has probably had his ISAM file structures damaged by power failures, operator errors (removed the disk too soon), or program errors (failure to close the files). Well, if it's happened to you, you don't have to pull your hair and wail anymore. BTB Inc. has developed a software program that repairs damaged ISAM file structures fast and reliably.

The program prompts you for the file name and drive location of the damaged file you want repaired. Once entered, the repair utility informs you of each adjustment made to the file, which should also help you in reconstructing any record

that could not be recovered.

This software program costs $100 from BTB Inc., 1447 Shady Birch Road, Memphis, TN 38116, 901-346-6989. Why rely on backup files when you can recover damaged ISAM files with this invaluable utility?

Reader Service ✓ 583

A Sweet Treat

Share your love for computers with your beau this Valentine's Day with this ideal gift: Eat-Only Diskettes. They are edible versions of the floppy disks lying around your computer. Two disks, made of a half-pound of gourmet milk chocolate and measuring the same dimensions as a 5¼-inch floppy, come with each order. They are great gifts or premiums for anyone in the computer industry.

This user-delicious sweetware is carefully packaged in a corrugated box to prevent the disks from becoming "too floppy." The shelf life of these edible disks is nine months. Priced at $12.95 for two disks, you can buy them at local department stores and computer retailers, or order them directly from Sweetware Inc., 516 Shelburne Road, South Burlington, VT 05401, 802-862-6939.

Reader Service ✓ 559

Disk Reconstruction

DataCure is a utility program for CP/M microcomputer systems that accurately and quickly reconstructs the original contents of disk sectors that have gone bad. DataCure regenerates and

✓ See List of Advertisers on Page 211

refiles all the information on a complete track or individual sectors that are damaged. Tests by the manufacturer show that DataCure even rebuilds files on disks that have had holes punched in them.

The product is self-configuring, and requires no special knowledge or skills on your part. It supports sector sizes in the range of 128–1,024 bytes, both interleaved and non-interleaved. It also supports single- and double-sided media, both 8- and 5¼-inch disks.

DataCure is distributed on an 8-inch SSSD format for the CP/M 2.2 and Pickles & Trout V2.2E operating systems and costs $99. A demonstration version with restricted error correction is available for $19. For further information, contact Colorado Online Systems Inc., 40 Balfour Lane, Ramsey, NJ 07446, 800 225-0103.

Reader Service ✓ 569

The K-Cover provides complete keyboard protection.

Protect Those Keys!

If you've ever moved your TRS-80 around, either from the office to your home, or just around the house, you know how vulnerable your keyboard is to scratches, bumps, and children's curiosities. With the K-Cover from Penguin Products (P.O. Box 7008, Roseville, MI 48305, 800-732-0614), you can finally give your keyboard the protection it deserves.

Made of unbreakable and anti-static plastic, the K-Cover fits over your computer keyboard to provide protection from dust, shock, and prying fingers. Measuring approximately 6 by 14 by 2 inches, it fits the keyboards for all the TRS-80's except the Model II. The K-Cover costs just $9.95.

Reader Service ✓ 550

New Products listings are based on information supplied in manufacturers' press releases. 80 Micro *has not tested or reviewed these products and cannot guarantee any claims.*

Introducing the Most Powerful Business Software Ever!

TRS-80™ (Model I, II, III, or 16) • APPLE™ • IBM™ • OSBORNE™ • CP/M™ • KAYPRO™

FREE--TRS-80 Mod I,III & 4 programs supplied on DosPlus (minimum system). Complete DosPlus also available.

The VERSABUSINESS™ Series

Each VERSABUSINESS module can be purchased and used independently, or can be linked in any combination to form a complete, coordinated business system.

VERSARECEIVABLES™ $99.95

VERSARECEIVABLES™ is a complete menu-driven accounts receivable, invoicing, and monthly statement-generating system. It keeps track of all information related to who owes you or your company money, and can provide automatic billing for past due accounts. VERSARECEIVABLES™ prints all necessary statements, invoices, and summary reports and can be linked with VERSALEDGER II™ and VERSAINVENTORY™.

VERSAPAYABLES™ $99.95

VERSAPAYABLES™ is designed to keep track of current and aged payables, keeping you in touch with all information regarding how much money your company owes, and to whom. VERSAPAYABLES™ maintains a complete record on each vendor, prints checks, check registers, vouchers, transaction reports, aged payables reports, vendor reports, and more. With VERSAPAYABLES™, you can even let your computer automatically select which vouchers are to be paid.

VERSAPAYROLL™ $99.95

VERSAPAYROLL™ is a powerful and sophisticated, but easy to use payroll system that keeps track of all government-required payroll information. Complete employee records are maintained, and all necessary payroll calculations are performed automatically, with totals displayed on screen for operator approval. A payroll can be run totally, automatically, or the operator can intervene to prevent a check from being printed, or to alter information on it. If desired, totals may be posted to the VERSALEDGER II™ system.

VERSAINVENTORY™ $99.95

VERSAINVENTORY™ is a complete inventory control system that gives you instant access to data on any item. VERSAINVENTORY™ keeps track of all information related to what items are in stock, out of stock, on backorder, etc., stores sales and pricing data, alerts you when an item falls below a preset reorder point, and allows you to enter and print invoices directly or to link with the VERSARECEIVABLES™ system. VERSAINVENTORY™ prints all needed inventory listings, reports of items below reorder point, inventory value reports, period and year-to-date sales reports, price lists, inventory checklists, etc.

VERSALEDGER II™ $149.95

VERSALEDGER II™ is a complete accounting system that grows as your business grows. VERSALEDGER II™ can be used as a simple personal checkbook register, expanded to a small business bookkeeping system or developed into a large corporate general ledger system **without any additional software.**

- VERSALEDGER II™ gives you almost unlimited storage capacity (300 to 10,000 entries per month, depending on the system),
- stores all check and general ledger information forever,
- prints tractor-feed checks,
- handles multiple checkbooks and general ledgers,
- prints 17 customized accounting reports including check registers, balance sheets, income statements, transaction reports, account listings, etc.

VERSALEDGER II™ comes with a professionally-written 160 page manual designed for first-time users. The VERSALEDGER II™ manual will help you become quickly familiar with VERSALEDGER II™, using complete sample data files supplied on diskette and more than 50 pages of sample printouts.

SATISFACTION GUARANTEED!

Every **VERSABUSINESS™** module is guaranteed to outperform all other competitive systems, and at a fraction of their cost. If you are not satisfied with any **VERSABUSINESS™** module, you may return it within 30 days for a refund. Manuals for any **VERSABUSINESS™** module may be purchased for $25 each, credited toward a later purchase of that module.
All CP/M-based Computers must be equipped with Microsoft BASIC (MBASIC or BASIC-80)

To Order:
Write or call Toll-free (800) 431-2818
(N.Y.S. residents call 914-425-1535)

* add $3 for shipping in UPS areas
* add $4 for C.O.D. or non-UPS areas
* add $5 to CANADA or MEXICO
* add proper postage elsewhere

DEALER INQUIRIES WELCOME
All prices and specifications subject to change / Delivery subject to availability.

COMPUTRONICS INC

50 N. PASCACK ROAD, SPRING VALLEY, N.Y. 10977

TRS-80 trademark Tandy Corp. - APPLE trademark Apple Corp. - IBM PC trademark IBM Corp. - OSBORNE trademark Osborne Corp. - XEROX trademark Xerox Corp. - KAYPRO trademark Non-Linear Systems, Inc. - TELEVIDEO trademark Televideo Systems, Inc. - SANYO trademark Sanyo Corp. - NEC trademark NEC Corp. - DEC trademark Digital Equipment Corp. - ZENITH trademark Zenith Corp. TI PROFESSIONAL COMPUTER trademark Texas Instruments, Inc. - SUPERBRAIN trademark Intertec Corp. - CP/M trademark Digital Research - EPSTON trademark Epson Corp.

That's what you get with the LNW80 Model 2—undoubtedly the most versatile, powerful and fully equipped microcomputer in its class today. A machine so superior in concept and design, that it will define the standards of microcomputer performance for years to come.

VERSATILITY

The LNW80 2 performs wonders with the most complete library of software available to any microcomputer on the market today. Every LNW80 2 comes complete with this outstanding library of Business Software. **LNW SMALL BUSINESS AND PROFESSIONAL ACCOUNTING SERIES™** General Ledger, Accounts Receivable, Accounts Payable, Payroll; **ELECTRIC SPREADSHEET™; ELECTRIC PENCIL™ WORDPROCESSOR; MICROTERM™ MODEM PROGRAM; CHART EX™ HIGH RESOLUTION BUSINESS GRAPHICS CHARTING PROGRAM; CP/M 2.2®; DOSPLUS™; LNW-BASIC™;MICROSOFT BASIC.™** In addition to a comprehensive line of LNW80 2 Software, it is also fully compatible with software from TRS80® (Models 1, 3, 4), CP/M® and Cromemco® worlds—a capability which gives you access to the most extensive and mature libraries of business, scientific, engineering and entertainment software applications. So no matter how far you expand into user applications, the LNW80 2 will expand right along with you.

POWER

The LNW80 2 performs miracles with the computing power of 96K RAM (standard) of user memory matched with a mass storage capability which handles 5¼" floppy disks and 5¼" hard disk drives. And while the unit comes with built-in controllers for 5¼" and 8" floppy disks (single/double sided, single/double density, up to 4.5 Megabytes capacity), the LNW80 2 also gives you the unique ability to read and write diskettes from a greater variety of other popular computers than does any other microcomputer. So regardless of how big you grow, you will never end up with thumb-twiddling down time while you expand to a more powerful system. The LNW80 2 will always have enough muscle to handle your biggest and toughest jobs.

FULLY EQUIPPED

The LNW80 2 was developed to anticipate the needs of both expansion and compatibility. So the computer was designed with enough built-in features to keep you from having to spend a small fortune as you move down the road to higher levels of user sophistication. Standard features include high and low resolution graphics in both color and black-and-white, an asynchronous serial communication channel, and a wide variety of tape, printer, monitor and hardware expansion ports. In addition, the LNW80 2 contains an array of quality construction features that fully justify its remarkable one-year limited warranty.

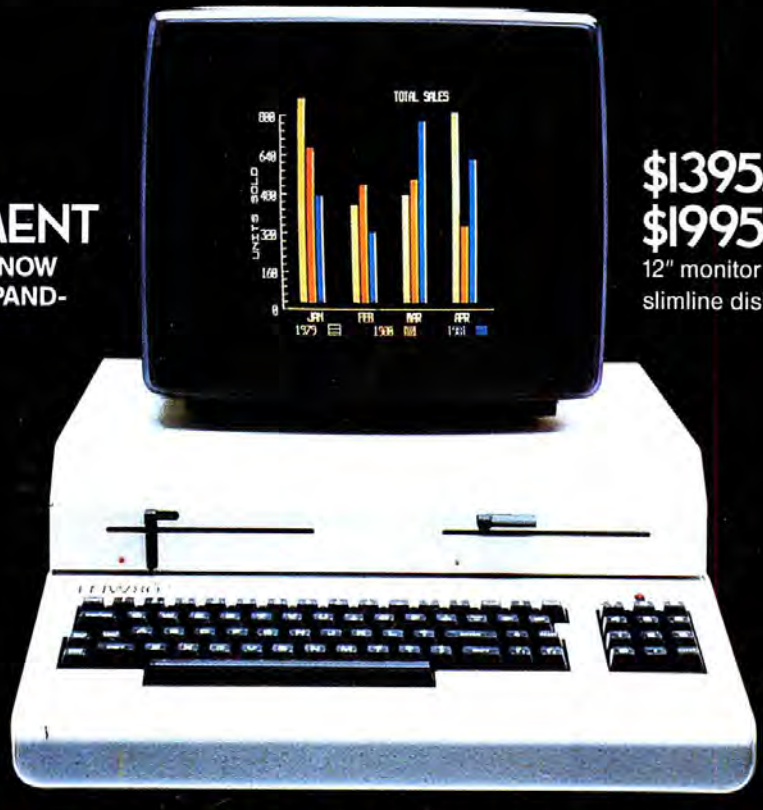
So if you're looking for a microcomputer that will satisfy your performance needs as you grow and develop, take a long, hard look at the LNW80 2. It's the one microcomputer built to meet the challenges of tomorrow—for a long time to come.

For more information and the name of the dealer nearest you, write or telephone:

LNW Computers
2620 Walnut, Tustin, California 92680
Telephone: 714/544-5744 ↙31

SPECIAL ANNOUNCEMENT

80x24 HARDWARE VIDEO NOW STANDARD. MEMORY EXPANDABLE TO 160K.



$1395. computer keyboard only.
$1995. computer keyboard, 12" monitor and dual 5¼" single-sided, slimline disk drives.

STATE-OF-THE-ART ENGINEERING
STATE OF TOMORROW PERFORMANCE.

LNW COMPUTERS