March 1981
US $2.50/DM 9

80 microcomputing™

the magazine for TRS-80 users*

Which Disk Operating System Is Best for You?

80 examines competing Disk Operating Systems
Are they worth the price?

Plus: ***80* interviews:**

John Roach takes charge at Tandy Corporation

Lew Kornfeld reflects on 30 years at Tandy

*Trademark of Tandy Corp.

0 74470 65947 03

409 ★★★ **A PERCOM BULLETIN** ★★★

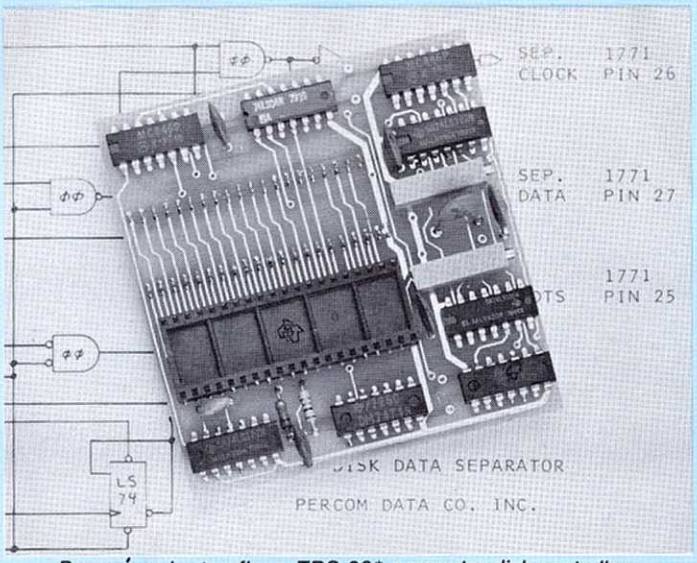
Adapter for TRS-80* computer eliminates disk read errors

Garland, Texas — Harold Mauch, president of Percom Data Company, announced that the company is marketing a simple plug-in adapter for TRS-80* computers that corrects a design deficiency in the disk controller circuit.

The problem, which causes disk read errors, has been traced to Tandy's reliance on a circuit internal to the FD1771 controller IC to perform the function of separating clock and data pulses.

As explained in the *Backgrounder,* use of the internal chip circuit for reliable data-clock separation is a design shortcut which the manufacturer of the controller IC warns against.

The Percom solution, a PC card adapter called the SEPARATOR™, eliminates the problem by substituting an explicit data separator circuit — one which has been used reliably in Percom disk controllers since 1977 — for the internal IC separator circuit.



Percom adapter fixes TRS-80 computer disk controller.*

The SEPARATOR™ is installed without modifying the host system. The user merely removes the FD1771 IC from the host controller, installs the IC in the DIP socket on the SEPARATOR™ card, and plugs the adapter into the vacated socket of the host controller.

Percom cautions that opening the Expansion Interface of the TRS-80* computer, which is required to install the SEPARATOR™, may void the computer's limited 90-day warranty.

The SEPARATOR™, which sells for $29.95, may be purchased from Percom dealers or ordered direct from the factory. The Percom toll-free order number is 1-800-527-1592.

Payment for mail orders may be made by certified check, cashier's check or money order, or charged to a Master Card or VISA account. Texas residents must add 5% sales tax.

408 ## Percom Mini-Disk Drives Store More, Cost Less.

Percom mini-disk drives store more data, are more reliable, yet a 40-track Percom drive costs **$100.00 less** than a 35-track Tandy drive.

You can store over 102 Kbytes per disk on Percom TFD-100™ 40-track drives, over 197 Kbytes per disk on TFD-200™ 77-track drives. A patch — supplied free on minidiskette — upgrades TRSDOS* for operation with the newer 40- and 77-track drives.

Both TFD-100™ and TFD-200™ models are available in one-, two- and three-drive configurations.

Prices start at $399 for a single-drive TFD-100™, $675 for a single-drive TFD-200™. Drives are supplied with heavy-duty power supplies. Metal enclosure is finished in compatible silver enamel.

See your nearby Percom dealer or order direct by calling toll-free 1-800-527-1592.

1 ## Five-Inch Disks Store More Than Eight-Inch Disks!

Garland, Texas — June 25, 1980 — Percom Data Company has begun production of a double-density disk controller adapter for TRS-80* Model I computers.

Harold Mauch, president of Percom, made that announcement here today, saying that data storage capacity using the adapter and double-density disk operating system — which is included — can be increased to as much as 354 Kbytes per minidiskette.

By comparison, the maximum storage for larger eight-inch disk systems used with the TRS-80* Model I computer is about 290 Kbytes.

Mauch said the PC card adapter, which plugs into the controller chip socket of the computer Expansion Interface, works equally well for either single-density or double-density storage, and users may continue to run programs under TRSDOS*, OS-80™ and other single-density operating systems with the adapter installed.

Price, for the plug-in adapter, the TRSDOS*-like double-density DOS and a utility for converting files and programs from single- to double-density format is $219.95.

410 **BACKGROUNDER**

CRC ERROR! TRACK LOCKED OUT!

by the Technical Staff
Percom Data Company

This problem started while we were studying an annoying problem with the TRS-80* computer. Disk drives sold by Percom are realigned and tested before shipment. We noticed, however, that some disk drives would pass the Percom inspection but just would not work reliably on the inner tracks with a TRS-80* computer. These drives were within the manufacturer's specifications, and would function perfectly on other disk systems Percom manufactures — "perfectly" here meaning more than 50 million bytes read without error!

The disk read data separation arrangement in the TRS-80* computer Expansion Interface uses an internal data separator of the FD1771 disk formatter/controller IC. Use of the FD1771 internal data separator is not recommended by Western Digital, the IC manufacturer. The following note appears on page 17 of the FD1771 data sheet:

> Internal data separation may work for some applications. However, for applications requiring high data recovery reliability, WDC recommends external data separation be used.

We suspected the data separator because the problem was most severe on disk inner tracks where storage density is highest and data separation is most critical.

To prove our point, a technician breadboarded a standard Percom data separator circuit, and configured it to plug directly into the FD1771 IC socket of the TRS-80* computer controller.

When connected to the TRS-80* computer, a troublesome drive functioned perfectly! We ran a BACKUP utility many times and never got a track lockout. Before we added the external data separator circuit to the computer, this same drive would always lock out tracks, and would have difficulty reading from the inner (higher number) tracks.

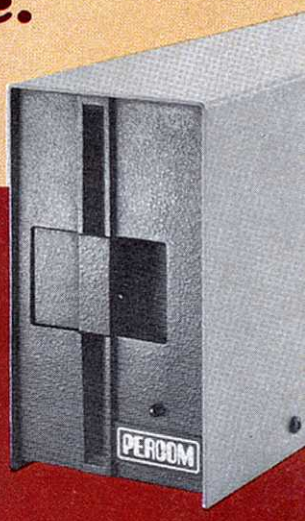
The Percom data separator circuit fixes the mini-disk controller of the TRS-80* computer. The type of drives being used is irrelevant; the circuit eliminates disk read errors resulting from the inability of the Tandy controller design to reliably separate clock and data signals when reading high density inner tracks.

PRICES AND SPECIFICATIONS SUBJECT TO CHANGE WITHOUT NOTICE.

PERCOM DATA COMPANY, INC. 211 N. Kirby Street Garland, Texas 75042 (214) 272-3421

TRS-80* Model I Computer Owners . . .

Double-density storage. It's really here!



Here at Percom. And your authorized Percom dealers.

And double-density storage is here in a big way. Because now you can choose from *three different levels* of mini-disk systems — all *double-density rated.*

And get the storage that precisely meets your application needs.

Not to mention the service and quality that's made Percom the industry leader.

Although rated for double-density operation, all levels of Percom drives *work equally well in single-density* applications.

You can operate these drives in ordinary single-density format using TRSDOS*, Percom OS-80™ or any other single-density operating system.

Or, you can add a Percom DOUBLER™ to your Tandy Expansion Interface and store data and programs in *either* single- or double-density format.

Under double-density operation, you can store as much as *350 Kbytes* of formatted data — depending on the drive model — on one side of a five-inch minidiskette. That's *four times* the capacity of standard 35-track Model I minidisks, almost 100 Kbytes *more than* the capacity of the *eight-inch* IBM 3740 format!

Available in 1-, 2- and 3-drive configurations in all three model lines, Percom *burned-in, fully-tested* drives start at only $399.

TFD-40™ Drives

TFD-40 Drives store 180 Kbytes (double-density) or 102 Kbytes (single-density) of **formatted** data on one side of a 40-track minidiskette. Although economically priced, TFD-40 drives receive the same full Percom quality control measures as TFD-100 and TFD-200 drives.

TFD-100™ Drives

TFD-100 drives are "flippy" drives. You store twice the data per minidiskette by using both sides of the disk. TFD-100 drives store 180 Kbytes (double-density) or 102 Kbytes (single-density) **per side.** Under double-density operation, you can store a 70-page document on one minidiskette.

TFD-200™ Drives

TFD-200 drives store 350 Kbytes (double-density) or 197 Kbytes (single-density) on one side of a minidiskette. By comparison, 3740-formatted eight-inch disks store only 256 Kbytes. Enormous on-line storage capacity in a 5" drive, plus proven Percom reliability. That's what you get in a TFD-200.

The DOUBLER™ — This proprietary adapter for the TRS-80* Model I computer packs approximately twice the data on a disk track.

Depending on the type of drive, you can store up to four times as much data — 350 Kbytes — on one side of a minidiskette as you can store using a Tandy standard Model I computer drive.

Easy to install, the DOUBLER merely plugs into the disk controller chip socket of your Expansion Interface. No rewiring. No trace cutting.

And because the DOUBLER reads, writes and formats *either* single- or double-density disks, you can continue to run all of your single-density software, then switch to double-density operation at any convenient time.

Included with the PC card adapter is a TRSDOS*-compatible double-density disk operating system, called DBLDOS™, plus a CONVERT utility that converts files and programs from single- to double-density or double- to single-density format.

Each DOUBLER also includes an on-card high-performance *data separator circuit* which ensures reliable disk read operation.

The DOUBLER works with standard 35-, 40-, 77- and 80-track drives rated for double-density operation.

Note. Opening the Expansion Interface to install the DOUBLER may void Tandy's limited 90-day warranty.

Free software patch with drive purchase. This software patch, called PATCH PAK,™ upgrades TRSDOS* for single-density operation with improved 40- and 77-track drives.

Quality Percom products are available at authorized dealers. Call *toll free 1-800-527-1592* for the address of your nearest dealer or to order directly from Percom. In Canada call 519-824-7041.

PERCOM

Prices and specifications subject to change without notice.

™ trademark of Percom Data Company, Inc.

* mark of Tandy Radio Shack Corporation which has no relationship to Percom Data Company.

✓408 PERCOM DATA COMPANY, INC.
211 N. KIRBY • GARLAND TX • 75042
(214) 272-3421

80 Contents

PUBLISHER
Wayne Green

EXECUTIVE VICE PRESIDENT
Sherry Smythe

CORPORATE CONTROLLER
Charles Garniss, Jr.

ASSOCIATE PUBLISHER
Edward Ferman

ASSISTANT PUBLISHER
Jeff DeTray

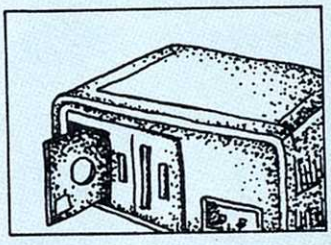
ADVERTISING MANAGER
Kevin Rushalko

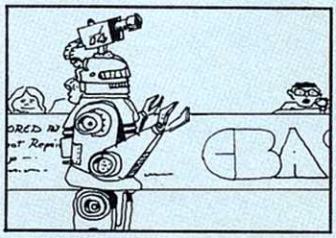
CIRCULATION MANAGER
(603) 924-7296
Debra Boudrieau

BULK SALES MANAGER
Ginny Boudrieau

ADVERTISING SALES
(603) 924-7138
Penny Brooks
John Gancarz

80 Microcomputing (ISSN -0199-6789) is published monthly by 1001001 Inc., 80 Pine St., Peterborough NH 03458. Phone: 603-924-3873. Subscription rates in U.S. are $18 for one year and $45 for three years. In Canada, $20—one year only, U.S. funds. Canadian distributor: Micro Distributing, 409 Queen St. West, Toronto, Ontario, Canada M5V 2A5. Foreign subscriptions (surface mail), $28—one year only, U.S. funds. Foreign subscriptions (air mail), $60—one year only, U.S. funds. In Europe contact Monika Nedela, Markstr. 3, D-7778 Markdorf, W. Germany. In South Africa contact *80 Microcomputing*, P.O. Box 782815, Sandton, South Africa 2146. Australian Distributor: Electronic Concepts, Attention: Rudi Hoess, 55 Clarence Street, Sidney 2000, Australia. All U.S. subscription correspondence should be addressed to *80 Microcomputing*, Subscription Department, P.O. Box 981, Farmingdale, NY 11737. Please include your address label with any correspondence. Postmaster: Send form -3579 to *80 Microcomputing*, Subscription Services, P.O. Box 981, Farmingdale, NY 11737.





REGULAR

COMING NEXT MONTH

The April issue of *80* will be packed with graphics. Bob Boothe describes how you can create a surprising display of high resolution math graphics with some very brief programs. In that same issue, he'll be starting an extraordinary series for *80* on straight line graphics.

80's second high resolution solution is from authors Dennis Murray and Paul Fowler. In the first of their two-part series, you'll learn how to assemble a high-density high resolution graphics board.

PUBLISHER/EDITOR
Wayne Green

MANAGING EDITOR
Michael Comendul

TECHNICAL CONSULTANT
Jake Commander

PRODUCTION EDITOR
Clare McCarthy

NEWS EDITOR
Nancy Robertson

REVIEW EDITOR
Pamela Petrakos

ASST. TECHNICAL EDITOR
Chris Brown

ASSISTANT EDITORS
Chris Crocker
Debra Marshall

TECHNICAL CONTRIBUTING EDITOR
Dennis Kitsz

EDITORIAL ADMINISTRATION
Cresca Clyne
Pat Graham
Nancy Noyd

DESIGN ASSOCIATE
Diana Shonk

MANUFACTURING MANAGER
Noel Ray Self

PRODUCTION MANAGER
Nancy Salmon

ASST. PRODUCTION MANAGER
Michael Murphy

AD GRAPHICS MANAGER
Robert Drew

AD COORDINATOR
Sue Symonds

ADVERTISING PRODUCTION
Steve Baldwin, Bruce Hedin, Maryann Metivier, Dion Owens

PRODUCTION DEPT.
Joan Ahern, William Anderson Jr., Linda Drew, Bob Dukette, Kenneth Jackson, Ross Kenyon, Theresa Ostebo, Jane Preston, Patrice Scribner, Thomas Villeneuve

PHOTOGRAPHY
William Heydolph, Terrie Anderson, Bill Suttenfield

TYPESETTING
Barbara Latti, Sara Bedell, Michele Desrocher, Luann Keddy, Mary Kinzel, Linda Locke, Karen Podzycki

---

Cover by Don Clineff & Associates

MODEL III FOR $320.00

Yes, double density is here! Two 40 track drives give you 368K of storage. And if thats not enough, four 80 track drives will give you 1.4 megabytes of on line storage. All this with the added speed of double density operation, not to mention 80% more storage per diskette.

No loss of your software library or conversion of your single density diskettes is necessary. DOSPLUS 3.1D reacts to the diskette. It will read single or double density with equal ease. It is not JUST a double density operating system, it is a double AND single density system. It is the first of it's kind. No one can offer you what we can! To change a disk to double density you need only format a double density disk, and then ONE command (TRANSFER), will bring over all your files. Very simple to use, but not necessary, because DOSPLUS will read them as they are now. Just insert single density diskettes and run (with our operating system). The machine will know what you are doing without having to be told ANYTHING!

If you want your computer to **chain functions,** then you need DOSPLUS 3.1D (for example, from powerup you could have your machine call another computer, scan a data base for appointments, recording any that appear on your printer; and then load in our new BASIC, setting files and mem/size, and start your program so your secretary can go right to work). With our BUILD, DO and AUTO, all of the preceding would be child's play. By the way, you could have also set the time and date, looked at the directory of all your drives as well as checked the free space map (which tells you not only how much space you have on a diskette, but also where it is and what it is used for) DOSPLUS is a truly intellegent, easy to use operating system that gives you all your computer is capable of delivering.

You probably guessed that for $320.00 what we are doing is turning your Model I into a Model III. It's so easy **anyone can do it in 10 minutes!** Why spend $2,500.00 for a Model III when you can get the same computing power, with our kit, and NO SOFTWARE CONVERSION, for only $320.00. Don't throw away your Model I, let us expand it! Move up to the world of double density.

You will reap the benefits of our **error-free software.** No miracles, just plain hard work and a lot of testing by experts and novices alike. Test us and judge yourself. We guarantee you will be 100% satisfied. If you are the first to find a legitimate "BUG" in our software you will be rewarded with a brand new $100 dollar bill.

Your systems disk will come complete with an all new single/double density disk editor called DISKZAP, and a BASIC program compressor called CRUNCH. Also included is PURGE, a utility to make the mass removal of unwanted files from a diskette easy, and RESTORE, a program that makes recovering a dead file as easy as typing in a command line. TRANSFER is just as it sounds, a program that moves all files (except systems) from one diskette to another. Single density to double or vice versa. CLEARFILE is used to zero data files on a diskette for a "clean slate". DISKDUMP is a new machine language sector display/modify program that works with filespecs instead of tracks and sectors. Used in conjunction with DISKZAP, you will have more disk editing power than ever before, with less frustration than was ever thought possible.

You will now be able to use all your **DOS commands from our new BASIC** with the CMD feature. And how about variable length records that really work, first time, EVERY time! This will allow you to use the ISAM programming technique for vastly improved handling of large data bases and lightning speed unheard of in BASIC. (ISAM stands for Indexed Sequential Access Method).

If inflated computing power without an inflated price tag is what you're after, contact us at the address below.

Micro Systems Software, Inc. — 384

Specializing in the Tandy Line

MC Inc.

(305) 983-3390

5846 Funston Street
Hollywood, FL 33023

CALL TOLL FREE FOR FAST SERVICE
(800) 824-7888, OPERATOR 193
FOR VISA/MASTERCHARGE/C.O.D. ORDERS
California dial (800) 852-7777, Operator 193
Alaska and Hawaii dial (800) 824-7919, Operator 193
TOLL FREE LINES WILL ACCEPT ORDERS ONLY!
For Applications and Technical information, call (305) 983-3390 or drop us a card.

master charge

VISA

Dealers inquiries invited

80 REMARKS

by Wayne Green

"Despite the claims by some other manufacturers, market surveys still show Radio Shack with more than 50 percent of the total domestic microcomputer market."

Greener Pastures

In my travels I make it a point to drop in and talk with as many computer store managers as I can. One of the things which I've learned from these visits is that a great many store owners have not yet adapted to the TRS-80. There is still a level of anger and frustration over the system, obviously borne of its being the almost exclusive territory of Radio Shack. It is in these environments that the term "Trash 80" is most heard.

Even a short visit to a computer store is likely to convince the newcomer to computing that he is indeed facing experts, and, thus, there is a tendency to believe this Trash 80 nonsense. It's sour grapes and nothing else. The TRS-80 system has no more faults than any of the others on the market, and it has a lot of benefits the others don't have.

Despite the claims by some other manufacturers, market surveys still show Radio Shack with more than 50 percent of the total domestic microcomputer market. Yes, there are more TRS-80 systems than all others combined! This has resulted in some powerful benefits to TRS-80 owners. For instance, as I've written before, there are perhaps twenty times as many programs available for the TRS-80 and, sadly for other systems, they are far better.

Add to that the growing wealth of supporting accessories and, even more important, the enormous amount of available information about the system, and you have a package that is not matchable by any other manufacturer.

For all of its faults, Radio Shack seems to be providing far greater service support for their system than any other manufacturer has been able to organize. Service is of critical importance for business applications. We may be able to make do without our system for home use and even for school use, but when the office system stops, so does the office. It's there that we need service measured in hours rather than days, and Radio Shack is gearing up for this.

While Radio Shack's efforts to supply its own software have been somewhat bungled, their approach is still more enlightened than most of the other firms. The general approach of most manufacturers is to make the hardware and let the dealers and users worry about the programs. Harking back to the recent dark ages of maxi and minicomputers, manufacturers are apparently unaware that micros are different and that it is not going to be practical for a million owners of a system to join user groups and swap programs.

I have personally talked with the software acquisition managers of *every* major American microcomputer firm, and, with the exception of Texas Instruments, they are all of the same mind: User groups worked for DEC (Digital Equipment Corp.),

What I'm saying is this: When you find your mind wandering at a computer store and you're thinking of swapping your TRS-80 for some "better" system, think again.

In Peterborough, we are using TRS systems for much of our business work. We use them for tracking Instant Software programs. We use them for mailing lists. We use them for some bookkeeping functions. We use them for word processing and article indexing.

Computer stores have their good aspects, so don't let their anger at the TRS-80 get you down. ■

Invest in Yourself

The need of this magazine for articles and the microcomputer industry's need for software are needs that you can exploit.

TRS-80 owners (and prospective owners) want to know everything possible about their systems. Any programs you've developed which might benefit them—improvements on existing programs, operating systems, utilities, diagnostics—will let our readers get more out of their computer.

Readers want to know when you've found a particularly good program. They want to know about any hardware accessories which are good. They need ideas on using their systems. They want to know about each new item Radio Shack releases.

Businessmen want to know if you've put a TRS-80 into a business application with success. They want to read, in English, what you used, where you got it, how it worked, what problems you had, how you surmounted them, and what it costs you.

Educators want to know about microcomputers in school, either for administrative applications or teaching. They want to know everything you can tell them about how to use computers best with the students. They want to know what software you used, what you developed.

The recently released Radio Shack color system and the pocket computer both need to be supported with information. Can the TRS-80-PC actually be used by a businessman? Have you modified any of the regular TRS-80 programs to run on this tiny computer? Have you written some application programs?

Matsushita, through their Panasonic and Quasar distributors, have been showing a computer very similar to the TRS-80-PC, but with expansion memory, a modem and even a printer attachment. How about you developing such a system around the PC and writing it up for us? That Sharp Memowriter should interface to the PC to provide a printer for the computer. Then, perhaps we could use one of the midget television sets as a better screen for the system so we can write several lines, provided you find a character generator which will do the job. Will the new Panasonic liquid crystal television make a usable screen for the pocket computer?

Programming

The other aspect of the writing field, which is developing rapidly, is program writing. The need for good programs is endless, and you have a decided advantage over the hardware folk in that most programs, once written, are good for a long time. Hardware may come and go, but good programs continue on.

A good program can be converted for use on just about any microcomputer and can be translated for use in any country. The main trick is to develop the expertise

in this aspect of computing. Then, practice your skills and establish your name.

There are plenty of books, supplemented by fine articles, which will teach you just about everything there is to know about programming. Your next step is to sit down and use the tools you have been given to produce programs. This is more of an art than a science, with some people taking to it naturally, turning out programs of genius, while others turn into plodders, who turn out useful, if uninspired, programs.

What kind of programs are needed? This list is endless. Every type of business has a need for programs to tackle the problems in that particular industry. It is a waste of time to start plugging at a bookkeeping system at this late date unless your expertise gives you some special insight into such a program.

Hundreds, perhaps thousands, of programmers have written specific applications, often in conjunction with a local computer store, without it ever occurring to them that there might be several thousand other businessmen who could use the program if it were brushed up a bit, made easier to use and published.

These are the very programs which computer stores all over the country desperately need to attract business clientele.

What do you have to do to make a program commercially salable? Well, once you have a program which fills a business need, the next step is to make that program as easy to use as possible. Put in defaults so the program doesn't dump when a wrong key is pushed. Locate it in memory so it will work with the various operating systems, disks, with cassettes and fast loaders. Write up as complete a manual of instructions as you can. Make it self-prompting so the instructions are only a support. Remember, some people have to be retrained after every coffee break. ■

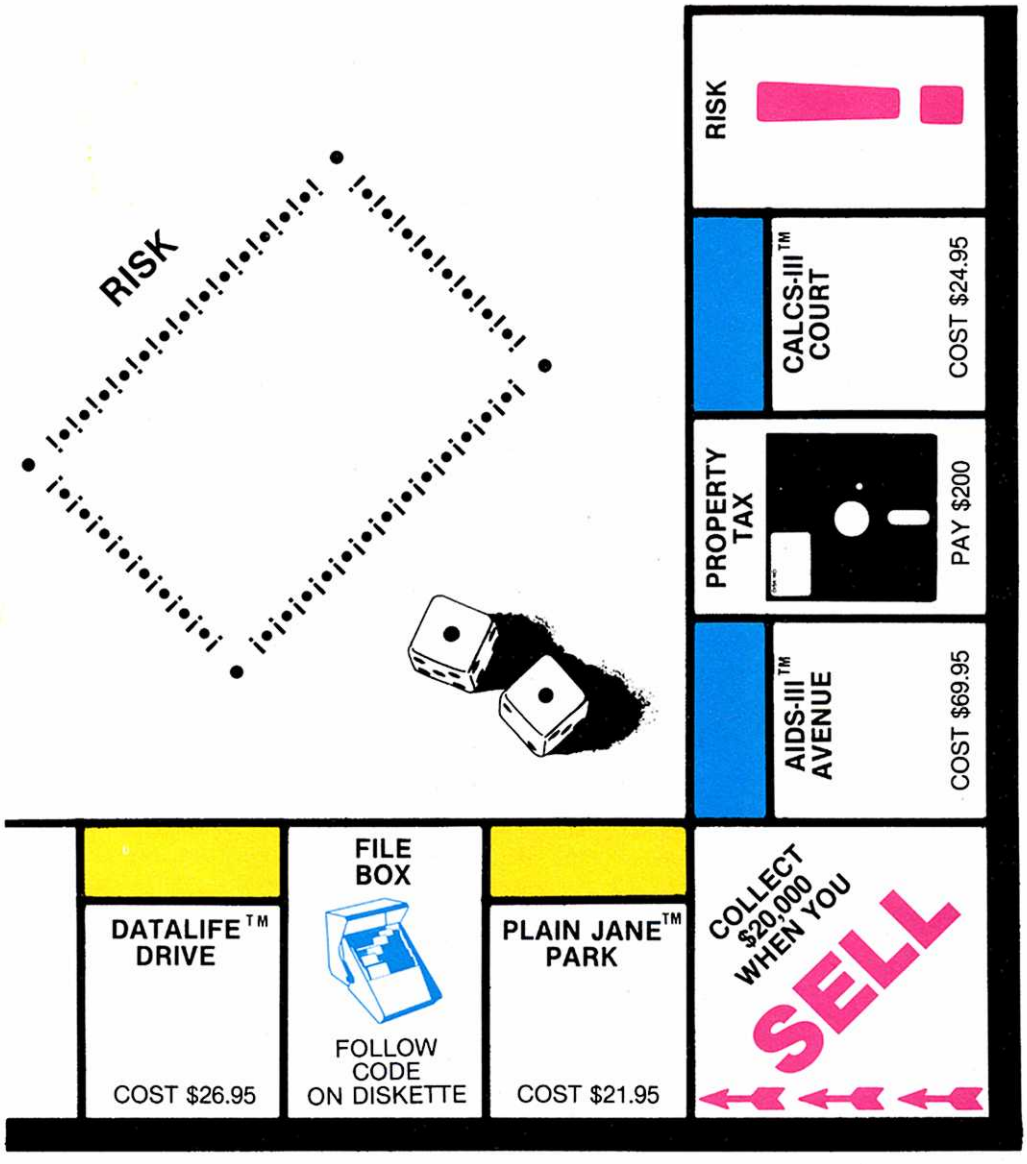
Tandy Overseas

In the U.S. the people at Instant Software find that the primary demand is for TRS-80 programs, even in stores which do not sell the system. The Apple and Commodore programs just do not sell as well, by a wide margin. Overseas, it is a different story. There, Instant Software not only translates programs into the local language, but also converts them to run on the Commodore or Apple systems, which sell far better than the Tandy.

Recently a dealer in Iceland pointed out that he had been unable to get the TRS-80 and so was settling for the PMC (Personal Micro Computer, Inc.) instead. This is fine with us since TRS-80 software runs fine on the PMC. Iceland is a small country, so perhaps these sales will make little difference to Tandy. Yet, this seems to be a pattern rather than an isolated situation, and one wonders about the foreign sales management at Tandy.

In Italy the first systems being pushed were the Southwest Tech. A lack of support sidetracked these sales and Commodore stepped in, with Apple right behind. Radio Shack? Negligible. Indeed, the reports from the publishers of most of the European microcomputer magazines are very critical of Tandy and their European management. The result has been that the top selling systems in Europe are the Commodore and then the Apple. ■

WHY TAKE A CHANCE WITH YOUR ADVERTISING?



YOU DON'T TREAT YOUR BUSINESS LIKE A GAME . . .

And neither do we. META MEDIA can help you reach your market more effectively and for less money than you may think you have to spend. Business is risky enough. Why not deal with people who understand products for small computers and how to sell them. You have a lot at stake . . . time, money, your future. Why roll the dice? ⌐219

Meta Media Productions, Inc.

26111 Brush Avenue Euclid, Ohio 44132 (216) 289-1100

We Accept: VISA/MasterCard

META MEDIA PRODUCTIONS, INC. is an affiliate of META TECHNOLOGIES CORPORATION, INC.

INSIDE 80

by Ed Juge, director of
computer merchandising, Tandy Radio Shack

"Our products, however, must adhere to the same standards whether the vendor is an outside company, or one of our facilities."

I've been waiting for some time to tell you a little about what goes into a Radio Shack product. The procedures I'll describe are about the same for all Radio Shack products, not just computers. (Obviously, each product line has its own unique set of rules.)

Radio Shack's 26 (as of 1980) plants build about 42 percent of the products we sell. Our products, however, must adhere to the same standards whether the vendor is an outside company, or one of our facilities.

Marketing a Product

Let's assume you're a vendor with a computer product you would like to sell to Radio Shack. First, write our merchandising department and tell us about it. If it sounds good, we'll set up an appointment for you to meet with the appropriate product line manager. We'll let you demonstrate the product on another brand computer, but we'll want to see an operating TRS-80 version before any commitments are made.

You will receive a copy of a multi-page addendum to our purchase order, explaining the conditions you and your product must meet. For instance, there are several "stages" of samples required:

- Evaluation sample: Fully operational, for initial testing.
- Working sample: Contains all the specified features of the final product. It is kept as a standard and used as a comparison of production units and their operation.
- Appearance sample: Is a mockup or prototype working unit. This is used for ad photos. It must be submitted three to four months prior to actual advertising time.
- Packing sample: This is the complete product in final packaging. It is subjected to shipping and drop testing by our quality assurance people.
- Preproduction sample: Is constructed using the final tooling, boards, etc. and submitted for approval.
- Production samples: Ten finished products, built by your assembly line people, not engineers.

You can also expect an unannounced visit to your factory by our quality assurance people to see how you're set up, and what kind of QA practices you employ. And you'll probably see them again any time there's a QA problem.

Once the product is "nailed down," changes in design, features, or components are not allowed without prior approval and retesting, if appropriate.

Heat 'em, Freeze 'em, Drop 'em

We drop-test your product (one drop on each of six surfaces) from heights dependent on its weight. A drop can vary from 36 inches for a ten-pound box, to 18 inches for a 50 to 70-pound item. We also drop it on each of the four box corners, and on two edges. The drops are made onto a steel or concrete surface.

We heat 'em, freeze 'em and operate them to the limits of temperature and voltage. They run for hundreds of non-stop hours under grueling conditions to see if and how well they stand up. Every function is tested, and re-tested.

Specifically, we test an electrical item at 120 V ac, and test its operation from 105 to 135 volts. Your product will be subjected to storage/shipping temperatures from −40 to +160 degress Fahrenheit. It must operate between 32 and 110 degrees Fahrenheit and operation must stay strictly within all specifications from 65 to 95 degrees. We've seen as many as eight out of ten popular, name-brand product samples fail under this punishment.

When our engineers, quality assurance group and product line manager are all satisfied that the product is good, you'll be given the authority to make your first shipment to us.

Uh... by the way, we won't accept delivery until we've received two very important things: our initial supply of repair parts (to our order), and a supply of Service Manuals.

At first, every piece is subjected to testing by our incoming quality assurance team. If fewer than three percent of the shipment fails, we'll fix them, or pay the freight to return the defectives to you for repair.

If greater than three percent of the shipment is "rejected," you'll have to send your folks to our warehouse(s) to service them, or pay freight to return the goods to your plant. When we see evidence that your shipments are below the three percent failure rate, we'll run only samples through QA testing... unless the failure rate resumes.

In the process of dealing with you, you'll find that our product line managers will insist on a top quality product, styled and designed to meet what we feel the product should be. It is rarely a matter of "You have a good product, so private label it, and we'll buy." It's more like "We want a widget for our line. We know what we want it to be and to sell for. Your expertise in widgets makes us think you could build ours for us. Are you interested?"

You'll also have to write both owner and service manuals in our format and style, send rough drafts to us for approval and then print the final versions.

We'll specify the packaging, printing on the box, manual design, product appearance, color and labels. The product must arrive at our warehouse complete, ready to hand to the customer.

The requirements are more than some outside vendors want to tackle.

Getting it Right

Radio Shack takes great pride in its ability to design and build high-technology products of its own, like the TRS-80. Except for the Pocket Computer, we did them all ourselves. No, we didn't do all of it right. Both Radio Shack and you, the customer, suffered in the early Model I days. But we were learning.

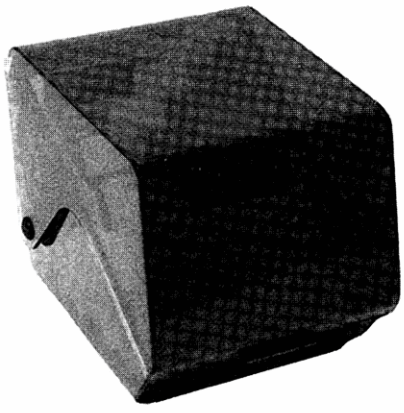
While I don't expect it to soothe any ruffled feathers, I hope this discussion has reinforced your pride on wanting a TRS-80. One fellow, who called me a couple of weeks ago, will understand: He has owned two 48K, two-disk systems for two years. His monthly repair costs per system have averaged only $2.35! We can't promise such outstanding figures for everyone, but we're trying.

I hope you'll rest easier knowing that nobody could have put more effort into delivering to you a more reliable, quality product.■

META TECHNOLOGIES
FOR YOUR DISK SYSTEM

FILE BOX
DISKETTE STORAGE SYSTEM

$24.95 for 5¼" disks
for 8" disks . . . $29.95



MTC brings you the ULTIMATE diskette storage system, at an affordable price. Storing 50 to 60 diskettes, this durable, smoke-colored acrylic unit provides easy access through the use of index dividers and adjustable tabs. Unique lid design provides dust-free protection and doubles as a carrying handle.

PLASTIC LIBRARY CASES
(not shown)

An economical form of storage for 10 to 15 diskettes, and is suitable for your bookshelf! Case opens into a vertical holder for easy access.

5¼-inch diskette case $3.25
8-inch diskette case $3.50

Single Sided, Single Density, Soft-Sectored 5¼-inch, (for TRS-80™) Mini-floppy

DISKETTES

$21.95 box of 10

These are factory fresh, absolutely first quality (no seconds!) mini-floppies. They are complete with envelopes, labels and write-protect tabs in a shrink-wrapped box.

PLAIN JANE™ DISKETTES
The Beautiful Floppy with the Magnetic Personality™

Thousands of people have switched to this low-cost alternative. These quality diskettes are packaged in a plain white box . . . no fancy printing, fancy names or fancy labels, not even our own (labels cost money). Trust us.

PLAIN JANE™ Diskettes $21.95
10 boxes of 10 . . . . (each box)$21.50

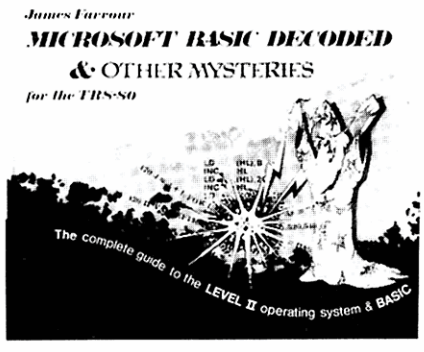
VERBATIM'S PREMIUM DISKETTES AT AFFORDABLE PRICES

DATALIFE™

Seven data-shielding improvements mean greater durability and longer data life. These individually certified diskettes feature thicker oxide coating, longer-lasting lubricant, improved liner, superior polishing and more! Meets or exceeds IBM, Shugart, ANSI, ECMA and ISO standards. Buy the best . . . buy DATALIFE™.

VERBATIM DATALIFE™ DISKETTES
5¼-inch (box of 10)
MD525-01 . . . . . $26.95
10 boxes of 10 . . . . (each box)$25.95
8-inch FLOPPIES
Single-Density, FD34-1000 . . . $29.95
Double-Density, FD34-8000 . . $39.95

TRS-80™ PRODUCTS



NEWDOS/80 by Apparat $149.95
NEWDOS+ with ALL UTILITIES
35-track $69.95
40-track $79.95
TRS-80™ DISK AND OTHER MYSTERIES $19.95
MICROSOFT ™ BASIC DECODED & OTHER MYSTERIES for the TRS-80™. $29.95

'RINGS' & THINGS

Help prevent data loss and media damage due to improper diskette centering and rotation with the FLOPPY SAVER™ reinforcing hub ring kit. 7-mil mylar rings install in seconds. Kit is complete with centering tool, pressure ring, 25 adhesive backed hub rings and instructions.

HUB RING KIT for 5¼" diskettes . . $9.95
REFILLS (50 Hub Rings) $4.95

Protect your expensive disk drives and your valuable diskettes with our diskette drive head cleaning kit. The kit, consisting of a pair of special "diskettes", cleaning solution and instructions, can be used for 52 cleanings. Removes contamination from recording surfaces in seconds without harming drives.

CLEANING KIT for 5¼" drives . . . $24.95

MORE PRODUCTS →

Products damaged in transit will be exchanged. Prices, Specifications, and Offerings subject to change without notice.

MOST ORDERS SHIPPED WITHIN ONE BUSINESS DAY

PRICES IN EFFECT
March 1, 1981 THRU
March 31, 1981

WE ACCEPT
- VISA
- MASTER CHARGE
- CHECKS
- MONEY ORDERS
- C.O.D.

- Add $2.50 for standard UPS shipping & handling
- $2.00 EXTRA for C.O.D.
- Ohio residents add 6½% sales tax.

FOR PRODUCT INFO 1-800-321-3640
CALL TOLL FREE
TO PLACE ORDER 1-800-321-3552
IN OHIO call (216)289-7500 (COLLECT)

META TECHNOLOGIES CORPORATION
26111 Brush Avenue. Euclid. Ohio 44132

master charge
VISA

✓20 810115

TRS-80 is a TM of Tandy Corp.
PLAIN JANE is a TM of MTC.

✓Reader Service—see page 290

80 INPUT

"We agree, the Editor is slow, but in the BASIC language and with sequential files, there is a . . . time overhead you just cannot beat."

Cross-Reference

In the January issue of *80 Microcomputing* you have an article by D.N. Ewart on how to have a cross-reference program for a TRS-80. It is unfortunate for Mr. Ewart that he does not own a TRS-80 Model II so that he could run it with Oasis by Phase One Systems.

Oasis provides a cross-reference listing which I find to be very helpful. In writing business programs, I find that I can control my use of variables and line labels much more easily with the use of a cross-reference. It also helps trace errors in programming logic.

Donald M. Dealy
EDP Director
Fuller Memorial Hospital
So. Attleboro, MA 02703

Baker Attacks

I must call your attention to a deceptive advertisement which appeared in your November issue (pg. 214)—Simutek's ad for their ZBASIC compiler. The main thrust of this ad is to play off their compiler against Microsoft's BASIC Compiler. They do this by presenting a BASIC program that involves a great number of floating point operations.

The Microsoft Compiler (which handles floating point) compiles this into machine code that runs at about the same speed as the original BASIC program. This will always be the case for a program heavy in floating point math. Simutek's compiler (being integer-only) produces code that, although incorrect, appears to execute much faster.

Simutek should have defined all variables as integers.

As a matter of interest, I added: 5 DEFINT A-Z to Simutek's test program, and compiled it with Microsoft's compiler. Surprise! Compile time was 3-1/2 minutes (as opposed to 14 minutes as stated by Simutek). This was for a completed, runnable CMD file on disk. Run time for the Microsoft code was 2-1/2 minutes (as opposed to the 17 minutes stated in the ad).

I'm not picking on ZBASIC itself. It looks like a quality piece of software that has definite uses—even if it is priced a bit high for an integer-only compiler. It has many nice features, appears easy to use, and has no royalty strings attached (Hurrah!). However, the array dimensions it will handle are not specified, and not all of the TRS-80's BASIC statements are implemented. And, it only handles integers.

Roxton Baker
San Francisco, CA

Simutek Responds

After receiving a letter from William H. Gates, President of Microsoft, we found a very misleading statement in our ad for ZBASIC.

In our ad we state that the example program, after compiling with Microsoft's compiler, took 17 minutes to run. This is true, but very misleading because Microsoft's compiler runs single precision, as does BASIC. If the statement, 5 DEFINT A-Z, had been added to the program, the run time would have been decreased to under two and a half minutes!! We publicly apologize to Microsoft and have discontinued the advertisement mentioned.

Be assured advertisements will be screened much more thoroughly in the future.

Michael A. Gariepy
President-Simutek
Tucson, AZ 85712

ASA Review

In Robert P. Johnson's recent review of the Radio Shack Advanced Statistical Analysis Package (December 1980), Mr. Johnson failed to mention one of the most important features of the system. In addition to supporting data files on cassette tape, the package contains a disk file management program which is used to prepare, update, and list disk data files which can then be used in any of the analysis programs. If the programs in the package are transferred from the cassette tapes on which they are supplied by Radio Shack to disk, the entire package can be run efficiently in a disk-only environment.

A new statistical analysis package is now available from Radio Shack for use on the Model II. In addition to the features contained in the Model I package, the new system includes a common random data file structure which can be used by any of the analysis programs, "select if" and "missing data" options, standard error of estimate for predictions, error correction during input, two-way analysis of variance, additional statistical programs, and a program for scoring tests/surveys and performing item analyses. Information on this package should be available soon.

Stephen W. Hebbler, Author
Statistical Analysis Packages
Dallas, TX

Pensadyne Rebut

In the January 1981 issue of *80 Microcomputing*, you published a review of our "Pensawrite 1" word processing system, written by a Mr. Louis Zeppa. To say the least, the review is not what we would like to have seen, but criticism is something that you have to accept and learn to deal with. I feel to this end, we have accepted the points we feel are valid, and have begun to adopt these into our system. But the specific points of the review are not my principal concern: It was the way in which the review was published.

In was indeed unfortunate that the *80 Microcomputing* staff chose to publish the review in the issue that it did, as this was the issue following the release of our newest product, the Pensa-write 2 Word Processing System.

As far as the review itself, there are several points that the reviewer has misrepresented. At no point in the article does the author of the review relate the features offered in the system to the price of the product. We agree, the editor is slow, but in the BASIC language and with sequential access files, there is a natural time overhead that you just cannot beat. And as for the point about random access files; random access would also have been very nice, but when we set out on the project, we had two objectives: to keep the price low, and to keep the system within

the capabilities of a 16K system. If we had re-written the product with random access capability, the system would be a great deal more complex, and, you guessed it, a higher price. More complexity would also have lead to more code, which would have put it out of the range of the 16K. All this was rationalized by the premise that the user is interested in a 16K program with one disk drive is on a very tight budget, and hence the low price. To even imply that the Pensa-write is inadequate because it won't handle special needs is at best thoughtless, and at worst, foolish. There are many improvements that could be made to the system, but at that point, the price would have gone up, and we would have lost sight of the original idea. If Mr. Zeppa wants Rolls-Royce features in his software, he should straighten his thinking around to pay more than $30 for a program. The Pensa-write 1 is a good piece of software, but at $7.50 he shouldn't expect competition for the $100 masterpieces.

We are an honest organization that is trying to do a good job for the microcomputing industry, and all we hope is that all the consumers who have read this so-called review will take Mr. Zeppa with a grain of salt, and give us the chance that we deserve.

Terence C. Gannon, President
Pensadyne Computer Services

UCSD Pascal

I would like to hear from anyone who has been using the UCSD Pascal system from FMG Corporation. In particular, anyone who has had any problems doing direct access file operations on the system, and has found a way to solve them. The symptom is to write a block onto the diskette and then, attempting to read it back, get either a CRC error or a sector not found error message.

Richard J. Bonneau
6 Tanglewood Drive
Shrewsbury, MA 01545

We called FMG. They say direct-access file operations are very critical and should be used with care as the normal operating system is being bypassed. The problem could be a hard I/O error (such as a flaky disk), or an attempt to perform I/O to a non-existent track or sector.—Eds.

Another Ad View

In contrast to Mr. Hall (80 Input, December, 1980), I am very interested in the advertisements your magazine carries, and in fact would probably not renew my subscription if you did not have them. This is not to take away anything from other content of the magazine which I enjoy and use, however I find a great deal of information in those ads. Prior to those ads I had never heard of Apparat, Inc., speed up boards, Harv Pennington, the alternate source, BSR interfaces or any of those people and things that have made my TRS-80 much more useful and enjoyable.

My only negative feeling concerning the advertisements is indirect. You carry a great many software ads but review so very little of it. The possibility of buying garbage or not "as promised" software is so great that I would like to see you greatly expand your reviews.

Lance B. Golden
Glencoe, IL

Basic Assembly

I am a 13 year old boy, and I enjoy my family's TRS-80 very much. I know everything from simple PRINT statements, to making my own music.

When I saw an assembly language program, I was impressed by the speed. I decided to try and learn assembly language. I read a few books, but the complexity of it all was too much. What I want is simple, step-by-step assembly programming. I am sure I am not the only person with this problem. I hope to see more information on the subject in future issues of *80 Microcomputing.*

Greg Hakim
Sea Girt N.J. 08750

You will be seeing several articles on simple assembly language in future issues.—Eds.

Reader Challenge

I challenge all readers to modify Eliza so that the entire conversation can be saved on disk and reloaded at a later date, so the player does not have to start from the beginning each time.

Wayne Beard
Scottsdale AZ 85257

Computer Control

Ref: *80 Microcomputing*, November 1980, page 206, "Resurrect it!"

Thomas L. Quindry's program is probably doing the trick but I want to pass along another procedure to regain control of your Level II BASIC program. I found this program in the Dutch magazine *Remarks*; it didn't mention an author.

You typed : CLOAD"A" (enter). What you really intended was CSAVE"A", but too late. Your cassette recorder didn't enter data into the computer.

Or, no printer is available. You typed LIST, but kkey-bbounce made LLIST out of it. The TRS-80 froze up.

Regain control with the following procedure:

1) Reset the computer.
2) MEMORY SIZE? (ENTER)
3) POKE 17130,1 (ENTER)
4) SYSTEM (ENTER)
5) /11395 (ENTER)
6) LIST (ENTER)
(Check if program is available again.)

This procedure is so simple, and doesn't need any tape loading to be passed along.

Daniel Lesenne
Dwarsstraat 10
B9690 Kluisbergen, Belgium

Faster Slower

Did you ever speed a program up by using less efficient code? Well, I did.

My application was for disk. Each disk record (buffer) contained two logical records of 124 bytes each. The process was skip sequential, which means I started at a certain place in the file, and processed sequentially until the last requested record was processed.

To minimize disk I/O, my program detected that, when processing the second logical record, no I/O was necessary, since it was still in the buffer from the previous GET. Unfortunately, by the time the second record had been processed, and the next logical record was required, the disk drive motor had stopped. This meant that when I requested the record, my program had to wait until the startup delay (about one second) had expired.

No matter how hard I tried, I couldn't tune the BASIC code to complete the second logical record before the disk motor went off.

So I slowed the program down. Even knowing the data I needed for the second logical record was already in the buffer, I initiated disk I/O to re-fetch it. This little trick kept the disk motor running, and the overhead was minimal since I knew the head was already positioned to the right track. The performance improvement was 30-40 percent.

Marc J. Staley
Cleveland, OH

80 DEBUg

Reference Handbook Bugs

The following is a list of errors associated with the "TRS-80 Micro Computing Technical Reference Handbook." Other owners might be interested in the mistakes I found which, most likely, are not exhaustive.

Peter Gibbs
University of the West Indies
Bridgetown, Barbados

1. Page 14, 10th line from end of right half of page: change both references to capacitor "C4" to "C42".
2. Page 19, 2nd line from end of left half of page: insert "," after "six bits".
3. Page 20; Z21, pin 3: change labelling from "D" to "B".
4. (a) Page 22; 13th line from end of right half of page: change "$\overline{CS}$" to "$\overline{CE}$".
   (b) Page 22; 12th line from end of right half of page: change "x72" to "x71".
   (c) Page 22; 11th line from end of right half of page: change "sheet 2" to "sheet 1".
5. Page 26; 12th line from end of left half of page: change "($\overline{CS}$)" to "($\overline{CE}$)" and hence "select" to "enable".
6. Page 45; 17th line from end of left half of page: change "pin 18" to "pin 13".
7. Page 62; 18th line from end of right half of page: change "Z71" to "Z51". N.B.: References to X71 are correct.
8. Page 63; block 8: change "Z71" to "Z51".
9. (a) Page 67; 19th line from end of left half of page: change "Z3" to "X3".
   (b) Page 67; 6th line from top of right half of page: change "Z3" to "X3".
10. (a) Page 69; signal name column: change "UDRV" to "VDRV".
    (b) Page 69; signal frequency column: change "11.835 kHz" to "15.835 kHz".
11. (a) Page 72; 4th line from end of left half of page: change "Z71" to "X71"
    (b) Page 72; 3rd line from end of left half of page: change "Z71" to "X71
    (c) Page 72; 1st line from end of left half of page: change "Z3" to "X3".
    (d) Page 72; 5th line from end of right half of page: change "Z3" to "X3".
    (e) Page 72; 4th line from end of right half of page: change "Z3" to "X3".
12. Page 76; line 60 in sample program: change "SET(27,X)" to "SET(127,X)".
13. Page 86; line 800 in sample program: change "A-PEEK-4096" to "A = PEEK(4096)".
14. (a) Page 89; tristate 74LS367: change to "Z3" *OR*
    (b) Page 89; inverter 74LS04: change to "Z3".
15. (a) Page 107; Schematic (sheet 1), Z13, pin 8: should be labelled " + 12V".
    (b) Page 107; Schematic (sheet 1), Z13, pin 13: change "CE/A6" to "$\overline{CE}$/A6".
    (c) Page 107; Schematic (sheet 1), Z40, pin 11: change "5V" to " + 5V".
    (d) Page 107; Schematic (sheet 1), R47, R64: change "5V" to " + 5V".
    (e) Page 107; Schematic (sheet 1), R48: add " + 5V" at open side.
16. (a) Page 108; Schematic (sheet 2), Z32, pin 11: trace this signal and change the labelling "VDPV" to "VDRV", and after passing through Z57, "VDEV*" to "VDRV*". Also label Z57 input pin, "1".
    (b) Page 108; Schematic (sheet 2), Z50, pin 11: trace this signal and change the labelling "HORV" to "HDRV".
    (c) Page 108, Schematic (sheet 2), Z63, pin 14: change "AO" to "A9".
    (d) Page 108, Schematic (sheet 2), Z63, Z62, Z61, Z45, Z46, Z47, Z48: change all references on pin 13 from "CE" to "$\overline{CE}$".
    (e) Page 108, Schematic (sheet 2), label CR1 as a Zener diode (i.e.  to ).

Level II to Model 33

In the January 1980 issue of *80 Microcomputing*, you ran an article which I wrote, describing an interface between the TRS-80 and a Model 33 Teletype (Level II to Model 33, pg. 70).

Recently, I have received several letters indicating that the interface won't work unless pin 16 of each UART is grounded. These letters puzzled me, because I don't remember grounding this pin on my prototype, and everything worked. Also, the problem was never mentioned by earlier correspondents.

I have looked into the manufacturer's literature, and it appears that there are some subtle differences between UARTs. In some versions, in 16 disables all of the status flags if it goes high; in other versions, pin 16 disables only the error flags (framing error, overrun, etc.) if it goes high. The interface will work with the error flags disabled because it doesn't use them. However, it won't work if all the status flags are disabled.

To avoid trouble, pin 16 of both UARTs should be grounded.

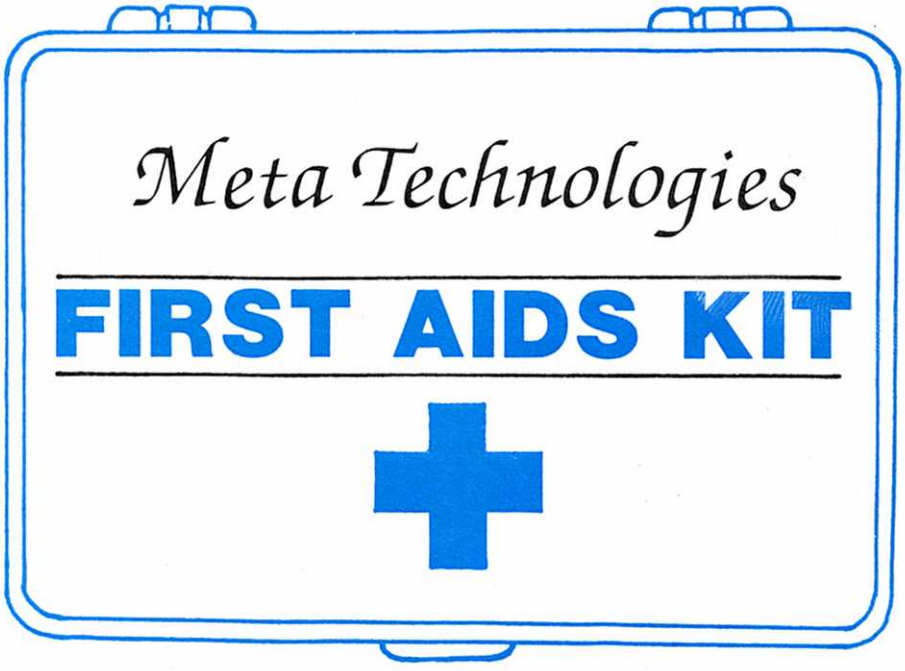
Lauren A. Colby
532 Pearl St.
Frederick, MD 21701

Printer Symbols

Valerie Vann, author of the "Seasons Greetings" program (December, 1980), has pointed out that our readers should again be reminded that the printer used by *80 Micro* prints a left square bracket ([) instead of the up arrow (↑) used as the exponentiation symbol in TRS-80 BASIC.

—Eds.

AILING INFORMATION ?



DOCTOR IT UP WITH MTC AIDS-I™

WORKS WITH CASSETTE AND DISK!

- Up to 10 user-defined fields, records up to 254 characters long.
- Machine-code assisted sorting: 200 records in about 5 seconds, any combination of fields.
- Full editing capabilities:
  Backspace, Right-justify, Delete field, Restore field, Skip field, Enter field, Skip record, Delete record.
- Full selection capabilities:
  Choose records to be worked on using any one of 7 comparisons.
  Examples: NAME greater than L or STATE equal OH or PRICE less than 99.00
- Selections effective for the following main functions:
  - LOAD records from cassette or disk
  - SAVE records to cassette or disk
  - SORT records
  - UPDATE/ADD records
  - DELETE records
  - PRINT/DISPLAY records
- Print/display any combination of fields in any order, in any position on a page-use for mailing labels, lists, etc.

MTC AIDS-I (Model I) $34.95

Write for our complete catalog, or

FOR PRODUCT INFO
1-800-321-3640

CALL TOLL FREE

TO PLACE ORDER
1-800-321-3552

IN OHIO call (216)289-7500 (COLLECT)

✓20

MTC

META TECHNOLOGIES CORPORATION
26111 Brush Avenue. Euclid. Ohio 44132

master charge
VISA

810115

TRS-80 is a TM of Tandy Corp.

✓ Reader Service—see page 290

80 AID

The New Character IC

In response to Patrick Kelley's letter on page 20 of the November issue, I can provide the following information.

The new character generator IC that Radio Shack is providing in new Model I TRS-80s, and with their lowercase modification, is part number AXX-3027. The catalog number is 26-1104 and they refer to it as a Character Word Processor. I obtained this part from the Radio Shack Customer Service Department 0821, 1803 South Beach Street, Ft. Worth, TX 76105 for $12.93 including postage.

I hope this information will help Patrick and others obtain this IC. From my point of view, it is really nice to have normal looking lowercase displayed. This includes descending tails for g, j, p, q and y plus a descending tail for a comma and semi-colon.

Eric Keener WB6EST
9163 Cody St.
Broomfield, CO 80020

Sound Problems

I enjoy your programs on sound because I feel it perks up a dull program. I have a problem though: I have tried all the sound programs you have published and they work well *until* I put them in a program. If there are any data statements in the program, I get an SN error. These are all decent data statements that worked before I put the sound subroutine in them. I don't know if you can answer my question, but why don't these sound subroutines work?

Dan Godfrey
10 Garden St.
Brewster, NY 10509

IRV and Interface

Help! Can someone tell me how to use "IRV" with my Disk BASIC? I recently added a single disk drive with TRSDOS 2.3 and Disk BASIC to my TRS-80 hardware, and I discover that IRV seems not to work.

First, I put IRV on disk using TAPE DISK (the 48K version). That seemed to work! Then, while in TRSDOS, I type IRV <enter>, and all Hell breaks loose!!! I get a screen full of garbage, with a giant READY somewhere in the middle, and no IRV!

I tried entering BASIC first... setting memory size at 64000 (per IRV's manual). Then back to TRSDOS with "CMD 'S'" and load IRV. OK again, the cursor is flashing, but a return to BASIC causes the same mess. I have also tried going to "BASIC *" and "BASIC2". BASIC * causes the same problem, and BASIC2 simply causes me to lose IRV completely.

While I'm sharing problems, I have one more which is also driving me nutty! Apparently the connection between the computer and the expansion interface keeps getting fouled somehow, and all kinds of interesting things happen! Most commonly I get totally illogical error messages and the program stops, although sometimes the whole system simply re-boots to DOS ready.

I can usually temporarily solve the problem by wiggling the connector between the buffer and the interface. I have tried the pencil eraser trick on the edge connector, and also used a tad of WD-40 on the contacts. I hope that someone has discovered a more permanent cure for this plague!

Rev. Richard W. Beebe
129 Main Street
Fryeburg, ME 04037

TAB in LPRINT Aid

This is in response to one of your readers requesting help with TAB statements greater than TAB(64) in LPRINT statements.

Many of the suggestions made by other readers help to solve the problem, but I have found a technique that makes formatting very simple and is not mentioned in the Level II manual.

This involves the use of the LPRINT USING statements with multiple fields and is implemented as follows: If you wish to LPRINT a numeric amount, a string, and then another numeric amount, first set up your LPRINT using string as F$ = ###### % % ####". Then print with a statement like LPRINT using F$;A,B$,C%. The computer will then fill in each variable in the next available field, as long as they are of the same type.

I used a short PRINTUSING string here for simplicity but it may be as long as you like. In addition, you may use literals and the computer will print them out in the position they occupy in the string, but will skip over them when positioning variables.

The following illustrates this:

```
10 A = 10:B$ = "JIM SMITH":C = 15.26
20 F$ = "### %    % DUES $###.##"
30 LPRINT USING F$;A,B$,C
```

Will print as:

```
10 Jim Smith Dues $15.26
```

LPRINT USING and PRINTUSING can be very powerful formatting tools if used creatively.

Jon Winchester
5922 N.W. 27th Street
Gainesville, FL 32601

Line Printer Squeal II

I am having a problem with my new Radio Shack Line Printer VI, and since Radio Shack has not been able to help, perhaps you or one of your readers could.

When the Line Printer VI is on, it emits a loud high-frequency whistle. Not only is this annoying, but after a half hour or so it causes almost everyone near my machine to get a headache. Incidentally, I have been advised by Radio Shack that all Line Printer VIs emit this sound. I have also been advised by Radio Shack's computer service hot line that they do not have a fix and one is not likely.

They are aware that the problem is being caused by the power transistors.

Perhaps one of your readers has the solution?

Roger Schechter
54 Park Ave.
Verona, NJ 07044

Eds. Note: This letter ran in the January Aid Section as Line Printer IV, an error.

META TECHNOLOGIES

CALL TOLL FREE 1-800-321-3552 TO ORDER

MTC AIDS-III™

✓20

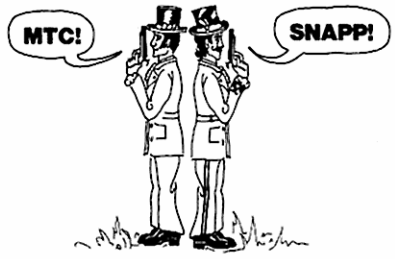
MODEL I . . .$69.95 MODEL II . . .$99.95

Introducing the latest addition to MTC's family of data management systems, AIDS-III. NO PROGRAMMING, easy to use. COMPLETE PACKAGE including demonstration application, documentation and MAPS-III (see below).

- Up to 20 USER-DEFINED FIELDS of either numeric- or character-type.
- CHARACTER-type fields may be any length (total: up to 254 characters).
- NUMERIC-type fields feature automatic formatting, rounding, decimal alignment and validation.
- Full feature EDITING when adding or changing records:
  - ENTER FIELD (can't type-in more characters than specified).
  - BACKSPACE (delete last character typed).
  - DELETE FIELD contents.
  - RESTORE FIELD contents.
  - RIGHT-JUSTIFY FIELD contents.
  - SKIP FIELD (to next or previous field).
  - SKIP RECORD (to next or previous record).
- SORTING of records is MACHINE CODE assisted.
  - 200 RECORDS (40 characters) in about 5 SECONDS.
  - ANY COMBINATION of fields (including numerics) with each field in ascending or descending order.
- SELECTION of records for Loading, Updating, Deleting, Printing and Saving is MACHINE CODE assisted.
  - Specify up to 4 CRITERIA, each using one of 6 RELATIONAL COMPARISONS.
  - LOAD or SAVE selected records using MULTIPLE FILES.
  - Example: Select records representing those people who live in the state of Colorado, but not in the city of Denver, whose last names begin with "F" and whose incomes exceed $9000.00.
  - Example: Select records representing those sales made to XYZ COMPANY that exceed $25.00, between the dates 03/15 and 04/10.

MAPS-III (MTC AIDS PRINT SUBSYSTEM), included at no charge, has the following features:

- Full AIDS-III SELECTION capabilities.
- Prints user-specified fields DOWN THE PAGE.
- Prints user-specified fields in titled, columnar REPORT FORMAT, automatically generating column headings, paging and (optionally) indentation.
- Can create a single report from MULTIPLE FILES.
- Prints user-defined formats for CUSTOM LABELS, custom forms, etc.



MTC is proud to announce MTC EXTENDED BASIC for the Model II, by R. Ryen. Features include "fixes" to existing BASIC, multi-line functions, extending an existing sequential file, PEEK, POKE, greatly enhanced screen control and expanded editing capabilities. The contents of variables are NOT CHANGED when editing, deleting, inserting or merging lines, allowing continued program execution! All this and much more. Compatible with SNAPP BASIC, below.

MTC EXTENDED BASIC . . . . . $ 99.95

MTC brings you the best of SNAPP, Inc.'s Model II BASIC interpreter at a very special introductory price. Written entirely in machine language, the enhancements are fully integrated into BASIC and require no user memory or disk space. Utilizes APPARAT's NEWDOS modifications to BASIC on the Model II. Features include 16 single keystroke commands for editing, listing, and other operations. An enhanced program line renumbering facility supports relocation and duplication of blocks of code. Includes a powerful cross-reference capability for producing a list identifying program line locations of user-specified variables and line numbers. Output may be displayed or printed. Compatible with MTC EXTENDED BASIC, above.

SNAPP BASIC for Model II . . . . . $ 99.95

MTC AIDS CALCULATION SUBSYSTEM-III™

MODEL I . . .$24.95 MODEL II . . .$39.95

MTC's most popular AIDS subsystem. Use for report generation involving basic manipulation of numeric data. Features are:

- User-specified page title
- Columnar Headings
- Optional Indentation
- Use for accounting, inventory, financial and other numeric-based information systems.
- Columnar subtotals generated when there is a change in a user-specified column.
- User-specified Columnar Totals
- Columnar values computed using constants and/or column values
- Balance forward calculations (Ex: Gross sales equals previous gross sales + sale amount + sales tax).

Compare AIDS-III™/CALCS-III™ with any other data management package under $100!

Others make claims, CALCS-III™ delivers!

CALCS-III™ REQUIRES THE PURCHASE OF AIDS-III™ OR AIDS-I™

MTC AIDS MERGE-III™

This subsystem will combine up to 14 AIDS-created data files into a single, large file. An optional purge capability removes duplicate entries while performing the merge operation (can even be used to eliminate duplicates in a single file). Machine-code assisted for high-speed performance, MERGE-III™ properly handles files sorted by any combination of fields, including numerics, with each field in ascending or descending order.

MTC AIDS MERGE-III™ $19.95
For Model II $29.95

MORE PRODUCTS

Let your TRS-80™ Teach You

ASSEMBLY LANGUAGE

REMSOFT's unique package, "INTRODUCTION TO TRS-80® ASSEMBLY PROGRAMMING" includes ten 45-minute lessons on audio cassettes, a display program for each lesson providing illustration & reinforcement, and a text book on TRS-80® Assembly Language Programming. Includes useful routines to access keyboard, video, printer and ROM. Requires 16K - Level II, Model I.

REMASSEM-1 . . . . . $69.95
FOR DISK SYSTEMS $74.95

Let Your TRS-80™ Teach You

ASSEMBLY LANGUAGE DISK I/O TECHNIQUES

REMSOFT does it again! REMDISK-1 is a concise, capsulated supplement to REMASSEM-1. Package consists of two 45-minute lessons on audio cassettes, and display programs providing illustration and reinforcement. Provides specific track and sector I/O techniques, and sequential and random file access methods and routines.

REMDISK-1 . . . . . $29.95

Let Your TRS-80™ Test Itself With

THE FLOPPY DOCTOR & MEMORY DIAGNOSTIC

by THE MICRO CLINIC

A complete checkup for your Model I. THE FLOPPY DOCTOR completely checks every sector of 35- or 40-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.

SYSTEM DIAGNOSTICS $19.95

✓ Reader Service—see page 290

80 REVIEWS

Edited by Pamela Petrakos

"As a matter of fact, CP/M is in such wide usage that it is considered the standard for business software."

The CP/M Handbook with MP/M
Rodney Zaks
Sybex, Inc.
Softcover, 321 pp.
$14.95

by Dennis Thurlow

There have been a number of personal computers for sale since the advent of the CPU chip, and it really makes little difference which one you have if all you plan to do is play games. When you're ready to progress to more serious applications, it's time to start looking for information on the chip in your board. One of the largest sources of information on computers is Sybex.

Sybex has published books on how to build systems from chips, how to program in high level languages, and how a beginner can learn machine language. Their manual on the Z-80 chip is the Bible for Z-80 users. This manual and many other Sybex titles were written by Rodney Zaks.

Knowing this, I expected an in-depth machine language study of the internal workings of CP/M. However, what we get is a simple instruction book for any level user.

The book is set up so that explanations of each item advance and the material goes into more depth as the book progresses. For example, pictures of hardware and simple explanations of their function are presented in the first chapter. It even explains how to boot up a system and load a program.

Reading further, you'll find that the book could also be a manual for the businessman. It explains how to determine if a system is adequate for an application, and how to recover from errors a typist might make. The difference in file protection between a multi user MP/M and single user CP/M is discussed, as well as how much space it takes to save a certain amount of data on disk. In the section explaining the uses of each command, the author anticipates any error messages a command might generate, explains how to recover, and then helps the reader understand why the error occurred and why the recovery works.

Examples are taken from commercially available software such as Micropro International's Wordstar text editor, and Structured Systems Group's NAD mailing list. This means the chances are good that the examples will be relevant to the system's actual use, which is a real plus!

The book is also ideal for the serious business programmer since most of the good business software being written today runs under CP/M. As a matter of fact, CP/M is in such wide usage that it is considered the standard for business software.

The CP/M overview chapter does an excellent job of describing how the console command processor (CCP), the BASIC I/O System (BIOS), and the BASIC disk Operating System (BDOS) work together, where they reside, and what each does. Internal file structure and installing and altering CP/M are all covered.

The standard features of CP/M up to revision 1.4 are given first, and then the changes in other revisions such as CP/M 2.2, MP/M 2.1, and Cromemco's CDOS are shown. Use of BASIC and CBASIC are discussed, and the book even discusses how the running of both of these may differ from one machine to another.

You won't be riding the crest of a new wave by reading this, and the author won't make you chuckle, but if your future in computing goes beyond playing games you'd better have a copy of this book on your shelf. ■

Microcomputing Techniques
With Sample Programs
Stanley J. Evans
Resten Publishing Co.
Engle Wood Cliff, NJ
229 pp.
$14.95

by Douglas Stewart

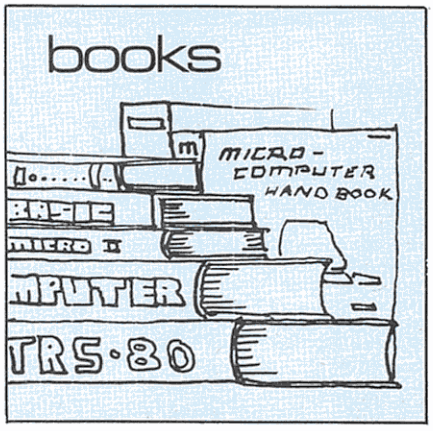
Everyone has preconceptions about a book before reading it, but any I had were quickly changed by the time I finished page 5.

The bulk of the book is about computer languages. BASIC gets 30 pages, PL/M which, according to the author, is a language specifically designed for the MSC-8 microcomputer (whatever that is) gets 77 pages and APL is covered in nine pages. The BASIC section has so many errors that anyone who was trying to learn BASIC would probably give up on computers altogether.

Part II consists of reference tables and "Useful Programs." The reference tables consist of hex to decimal conversions and hex arithmetic. The "Useful Programs" in BASIC consist of two programs. One is a number base conversion program that has no documentation and the other is a Dow Jones industrial average forecaster program written in Radio Shack's Level I shorthand. The latter is from the TRS-80 Level I Manual and has a couple of errors in it—one from the manual, the other, compliments of Dr. Evans.

Computer arithmetic is well described using binary, octal and hexadecimal based number systems. Mnemonic coding and ASCII code are given cursory mention.

The author covers the purpose of a program from its beginning. A problem is defined and a solution is found by means of a flow chart. From the flow chart a basic program evolves. The only problem is that there are errors in the program and what

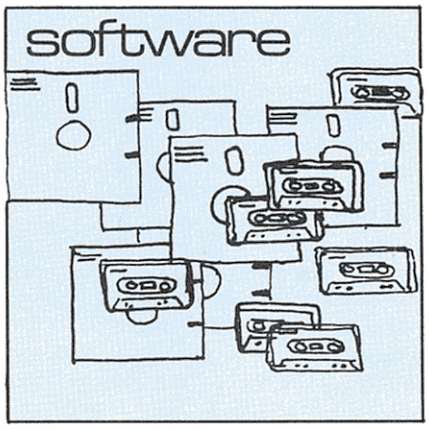


the program actually does is not what it is supposed to do according to the text. Anyone trying to learn from this book is going to be confused by all the errors. Anyone who catches the errors is experienced enough not to require this book.

The publisher describes the book as follows:

"Here's an all-in-one microprogramming guidebook that makes you an expert on the new generation of microcomputers. Even if you have little or no programming experience, this manual will show you how to turn out dependable, accurate programs with a minimum of time and effort."

I think the above description is a little less than accurate. Dr. Evan's book is not only a waste of money, it is also a waste of time. ■



File Management System (FMS)
Data-Base for the Exatron Stringy Floppy.
(Level II, 16K and up)
Exatron Corp.
Sunnyvale, CA
$19.95

by Fred Blechman

What is a data-base? It's an assorted file of specific individual records, where the file would be like a folder full of pages, with each page being a record. A record consists of several fields, each of which would consist of specific pieces of information on a page. For example, a name might be one field, an address another field, a phone number a third, and so on.

The purpose of a data-base program is to allow you to enter, change, delete, sort, total and retrieve, on printer or screen, the various records in a file. This is done by assembling, formatting and manipulating the fields to form records, and then sorting the records and saving them on tape or disk.

If you have a cassette-based TRS-80, you know how impractically slow any data-base program saves and loads data on cassette. A floppy disk solves this problem, but you need an expansion interface, more memory and at least one disk drive—about $900.

The Exatron Stringy Floppy (ESF) for about $250, saves and loads programs at 7200 baud, 14 times faster than a cassette. A special I/O (input/output) program allows you to write and read data to the ESF tape wafer at speeds that make data-base programs practical.

The FMS (File Management System), written by M. N. Kidder for Exatron, is a BASIC language data-base program that offers far more flexibility than any cassette-based data-base program I've seen. Because of its rapid I/O capability, it's practical to use. This program will allow you to use up to nine separate fields for each record, and each field can be either string, integer or single-precision, with prompting labels for each field. Records can be added, changed or deleted. A sorting subroutine allows you to sort the entire record alphanumerically on any field! You can find selected records in the file equal-to, less-than, or equal-to or greater-than any one of the fields. You can right justify numeric fields (single precision can have dollar signs and two decimal places, or zero to seven decimal places), and numeric fields can be totalled for all or a specified part of the file. All or selected parts of the file may be listed on the screen or printer, and a map function lets you establish the format. You can also hide selected fields from the display or printer. You can even leave the program to BASIC and return without losing the file data (unless you edit the program itself).

Records are entered from the keyboard initially, then saved on an ESF wafer. Next time you use the program, load the records in from the wafer. You can also append existing records with records from a wafer.

The 30-page manual included with the FMS wafer is very well written and specific. Like any data-base program, it takes some reading and practice—but then it's a breeze! The program is menu driven; that is, you almost always have a list of choices on the screen. When the menu is not displayed, various prompts make the program easy to use. The BREAK or ENTER keys almost always return to the menu if you get confused, and I found the program virtually crashproof!

Fig. 1 shows the screen display after entering the I/O program (1 second) and FMS program (12 seconds) from the Stringy Floppy wafer, and then typing RUN. Typing C allows you to CREATE a new file at the keyboard. Type L to LOAD data from a previously made wafer. Q (QUIT) returns you to a READY prompt in BASIC.

"The FMS . . . offers far more flexibility than any cassette-based data-base program I've seen."

To CREATE a file you must answer various screen prompts to define the type and name for each field. You must decide in advance exactly what you want contained within each field of your records. The manual uses an address list as an example.

```
—FILE MANAGEMENT SYSTEM—
C —CREATE A NEW FILE
L —LOAD FILE FROM WAFER
Q —QUIT
SELECT COMMAND→
```

Fig. 1. Initialization Menu

Entering the file from a previously made wafer is very fast compared to cassette load. I tested this with a file of 45 records, each containing first and last names, street address, city, state and zip code. It loaded almost three complete name and address records per second! (The Stringy-Floppy dumps the data in 256 byte blocks, with stop-start gaps between blocks at about a 50% duty cycle, or an average baud rate of about 3600). A typical file of 50 names and addresses would load in about 17 seconds. Pressing ENTER displays the menu. The file can then be added to, or changed, at the keyboard.

Fig. 2 shows the menu. A brief explanation of each command will help you see the tremendous versatility of the FMS.

Command A displays the last entered record and lets you ADD to the file, using the existing field definitions. The number of the record, and the name (label) of each field is shown on the screen, so you always know where you are within the record and the file. The documentation, with this and all commands, is very specific.

Command C displays and allows you to CHANGE any field of any record within the entire file. Command D is used to DELETE any complete record from the file, by record number. Command E allows you to EXTEND the file in memory by adding additional records from a wafer.

Command F (FIND) displays only specified records from the file. You define the field to be searched within each record, and you specify what the search value (alphabetic or numeric) should be. You also specify less than, exactly equal to, or greater than or equal to, this value. A reminder is placed on the screen of LIST to indicate the object of FIND, if active.

Command H allows you to prevent displaying or printing of any field. In a name and address list, for example, you would probably HIDE the record number from the printer.

Command J is used to JUSTIFY, or line up, integer or single-precision fields, with dollar sign formatting available for single-precision fields.

Command L (LIST) will display each record, in accordance with the instructions (FIND, HIDE, JUSTIFY, MAP, RANGE, SORT, TOTAL and VALUE) that are active at that time. This gives you incredible flexibility. Various on-screen indicators (such as FIND or RANGE) tell you whether these functions are active.

Command M allows you to format (MAP) the display or printer output. It works along with HIDE and JUSTIFY. A column guide is displayed at the top of the screen with numbers from 0 to 63. You select the position of each field, and can even have them on separate lines! For printer formatting, you can use columns 1 to 120.

Command N lets you start a NEW file by deleting the existing file and returning to the initialization menu (Fig. 1).

Command P is like LIST, but directs the output to the printer. PRINT is affected by the same commands as LIST.

Command Q allows you to QUIT the program without deleting the data. You can perform calculations or examine variables or @NEW an ESF wafer and then return to the FMS, with data intact, by typing CONT, unless you actually edit the program. Editing returns all variables to zero. Also, Q enables the BREAK key, resets the error recovery routine and establishes the re-entry point to continue the program.

Command R lets you set the RANGE of record numbers to LIST, FIND and PRINT. You set the high and low limit by record number. A reminder is placed on the screen of LIST to define the RANGE, if active.

Command S is used to SORT, on any field, in ascending sequence. While this BASIC sort is not as fast as machine language, it's quite adequate. I found the sort time to completely resort a file of 45 names and addresses took from 30 to 34 seconds. A very rapid automatic SORT is performed whenever LIST, PRINT, FIND or WRITE is entered after ADD or CHANGE.

Command T will TOTAL any one of the numeric fields for all or part of the file, using RANGE and FIND with LIST or PRINT.

Command V is used as a short-cut to change the VALUE after a FIND function, without having to again define the field being searched.

```
—FILE MANAGEMENT SYSTEM—COMMANDS
A —ADD        J —JUSTIFY    R —RANGE
C —CHANGE     L —LIST       S —SORT
D —DELETE     M—MAP         T —TOTAL
E —EXTEND     N—NEW         V —VALUE
F —FIND       P —PRINT      W—WRITE
H —HIDE       Q—QUIT        X —

SELECT COMMAND
```

Fig. 2. Command Menu

Command W will WRITE all the records in memory to a Stringy Floppy wafer. Any one wafer, if it's long enough, can hold up to nine files, numbered one to nine. Along with the records, WRITE also starts with a descriptive header on the wafer for that file, including MAP, SORT and JUSTIFY information, so when LOADed back into memory, you don't need to re-establish any parameters.

Command X displays a status message: number of records in existing file, maximum number of records, memory remaining and string space remaining.

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

Another FMS data-base I tried was one to keep track of distributors in my Amway business. Each record contained only the last name of the distributor, his sponsor's last name, the group leader's last name, and a zero or one to indicate renewal status. My 16K TRS-80 would only hold 145 of these records, and I needed over 300! Obviously, a 32K or 48K memory would immensely increase the capacity of FMS. Of course, separate files (A-K, L-S, T-Z or some such system) could be used, but that's a poor solution.

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

Superscript
Acorn Software Products, Inc.
Washington, DC
$29.95

by Terry Kepner

Scripsit, Radio Shack's word-processor for the TRS-80 microcomputer, has been on the market for over a year now, and many people have complained to Radio Shack about their problems when interfacing Scripsit to a non-Tandy printer. Computerists said Radio Shack did not sell a printer that could be effectively used as a word-processor. In fact, many were upset that Radio Shack did not assist them in writing driver routines for letter quality non-Tandy printers.

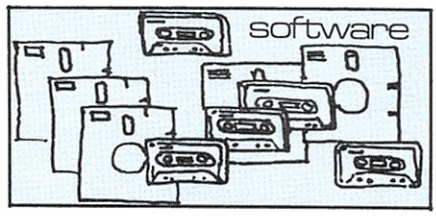
Patch It Up

Well, now is the time for all of these non-standard-printer owners to rejoice! A patch to Scripsit is now available that takes care of most of their problems.

Acorn Software Products has released a short program that patches Scripsit so that it can be used with a variety of serial and parallel printers not normally supported by Scripsit. And it fixes some rather troublesome bugs in it.

According to their users pamphlet, you can:

- get a disk directory that, in addition to listing all of the files on a disk, tells you how many free grans are left on that disk without losing your text file;
- kill a file while in Scripsit;
- create files with the patched Scripsit



✓ Reader Service—see page 290

that can be read by TRSDOS or NEWDOS;
- insert text into unjustified lines while printing, a necessity for generating form letters that require names (or other information) to be inserted into the letter;
- protect drivers in high memory from being written over by Scripsit;
- specify that Scripsit not put additional spaces between two words if you are justifying.
- if your printer has them, print special characters: left and right brackets ([ ]), left and right braces ({}), and the caret symbol (^);
- if your printer has half-linefeed and reverse half-linefeed, you can superscript and subscript;
- underline, print in boldface and slash zeros (Ø), if your printer acknowledges the backspace character;
- set the printer to 10-pitch (10 characters per inch) or 12-pitch (12 characters per inch), if your printer has this capability.

Even More

In addition to these capabilities, Acorn has:
- fixed Scripsit, so that you need a file-specification typed with the L command, thus eliminating the problem of text wipe-out. This happens when you try to get the length of your text without pressing the ? key before typing the L key a hitting ENTER;
- shortened status messages printed at the bottom of the screen by Scripsit;
- changed the keyboard driver, making it faster without disrupting the video display;
- supplied custom drivers for the Diablo and NEC 5530 printers, both serial and parallel versions;
- supplied a custom driver to let you design your own—serial or parallel.

This is what Acorn claims their patch will do, and that's exactly what it does. The directory command works great, except that you cannot get a list of the invisible or system files. You also can't get the file allocation specifications.

Other than those I have found little to complain about in Acorn's Superscript.

The drivers supplied on the disk are easy to use and are well explained in the pamphlet. The pamphlet also provides instructions on how to set the protocol switches provided in the NEC 5530 and the Diablo printers, and how to set the RS232-C switches in the Radio Shack expansion interface.

I use the recently introduced Radio Shack Daisy Wheel Printer II for word-processing. I was upset that this printer includes many features not supported by Scripsit, such as some twenty odd inaccessible characters.

I was however, very pleased to discover that my printer can use every Superscript option, though I did have to cheat a little on the boldface feature.

Unfortunately, I still cannot use most of the special characters on the Daisy printwheel, but at least I can use the additional brackets, braces, and the caret symbols, thanks to Acorn.

Because the Daisy Wheel II was just introduced, there isn't a special driver on the Acorn disk for it. I had to use the custom parallel driver to design a routine to patch my printer. When I finished, everything worked fine except the boldface routine. My printer just double-typed all the letters, using the backspace instead of typing the second letter on top of the first as it was supposed to do.

I finally came up with the proper sequence of commands to the custom driver to get it to approximate the commands needed by the patch.

Quite simply, the sequence is:

1B,01,08

This tells my printer to space forward 1/60 of an inch, then backspace one full character width. The patch then repeats the keystroke just typed.

Besides this little difficulty, I have found Superscript to be useful and definitely worth the price.■

Doubler
Percom Data Co.
Garland, TX
$219.95

DOSPLUS
Micro Systems Software Inc.
Hollywood, FL
$99.95

by Stewart E. Fason

One may grow very rich in a number of ways, two of which are: develop a dramatic innovation on a useful product or simply build a better mousetrap.

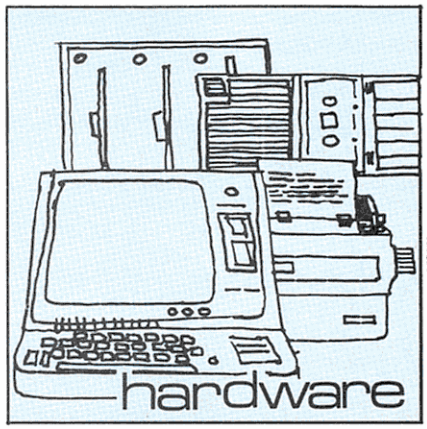
The Doubler

Percom's new double density system will revolutionize the world of disk operating. How can I be so sure? Well, there are some indications which cannot be overlooked. Three of my friends just left my home. All of them have huge software libraries (mostly bootleg I reckon) and are disk equipped. They had called to ask if it would be okay to stop by for a few minutes and see my new double density system. Few minutes? Ha! Four hours later I threw them out and the verdict was rendered... the three of them decided that there was no way they could live any longer without the Doubler. One look at Fig. 1 tells the double density story.

Do a FREE on your single density system and compare. A TRSDOS disk leaves one with about 49K free space compared to over 118K left on a 40 track DOUBLEDOS.

Percom's double density system has two parts. The hardware is a printed circuit board which plugs into the disk controller chip socket in the expansion interface. The installation is simple and takes about five minutes. The hardware is totally invisible to your single density system except that it contains Percom's data separator which will provide error free reading of your disks. The magic begins when you boot up the double density operating system.

The doubler is supplied with Percom's DBLDOS. TRSDOS users will have no problems. All TRSDOS commands are available in DBLDOS plus the following enhancement:
- The copy command will work with one drive and copies single to double, double to single, as well as like to like. It will not format or backup anything but double density. This is a source of aggravation. Should you want to supply a single density disk to a friend, you would have to format the disk with some other DOS. Further, you would have to copy each file separately with the DBLDOS copy command, which, if you had a large disk to copy, would be an onerous task. Prior to our session I spent many hours copying my single density disks to double, one file at a time.



META TECHNOLOGIES

FOR YOUR TRS-80™ DISK SYSTEM

PROGAMMING TOOLS
NEW LOW PRICE!

TDAM $14.95
For Model II $24.95
Includes MTC QUE Card!

Having trouble with RANDOM FILES? With MTC's Table-Driven Access Method (TDAM) you'll never fret over FIELDing again. No knowledge of random access files is required. Insert the TDAM "interpreter" into any BASIC program and type in a few DATA statements describing the information in your files. TDAM does the rest! Reads and writes fields and records of any type (even compresses a DATE field into 3 bytes!). Features automatic file buffer allocation/deallocation, memory buffering, sub-record blocking/deblocking, and handles up to 255 fields per record. Super fast and super simple! Complete with TDAM interpreter, instructions and demo program. Requires programming experience.

SIFTER..... $14.95
For Model II..... $24.95

Twelve in-memory high-speed sorts for use in any BASIC program: stable, non-stable, with/without tags, for numeric or string data. Random File Sort included. Some sorts written in machine code. Includes sort subroutines, demo programs and instructions. Relocate as needed with REBUILD. Requires programming experience.

SHRINK.....$14.95
For Model II.....$24.95

Makes Every Byte Count! Make programs smaller and faster! Combines lines & removes unnecessary code including remarks, without altering program operation. Typically reduces program size 25% to 40%.

SUPERSEDE $14.95
For Model II..... $24.95

A "must have" for the professional programmer or the serious amateur. Probably one of the greatest time-savers available. Write programs in shorthand - change variable names - generate program documentation - use with REBUILD and MINGLE to build new programs from old ones.

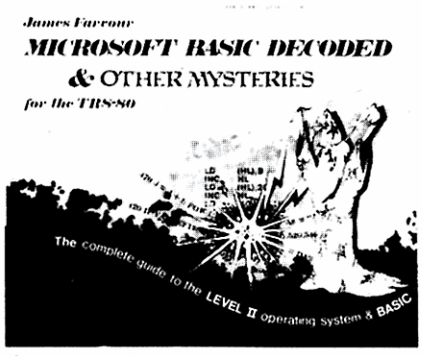
MINGLE-II..... $14.95
For Model II..... $24.95

Merge up to 14 files (Program or Data) into a single file. Data files may be merged in ascending or descending sequence with the ordering based on a user-specified comparison field. A very handy utility for consolidating data files.

"OTHER MYSTERIES" VOLUME II

foreword by

H.C. PENNINGTON



Call now and place your order for this new book, "MICROSOFT™ BASIC DECODED & OTHER MYSTERIES for the TRS-80™", from IJG, Inc. A primer for cassette and disk BASIC on the TRS-80™, the information provided applies to similar MICROSOFT™ BASIC interpreters. Features include definition of terms, an overview of BASIC and DOS, explanation of exits, error codes, verb actions, "cold" and "warm" restart procedures and examination of system utilities, arithmetic support and I/O driver routines, and the communications region in RAM. Individual routines are explained in detail, with an index provided for easy access. Appendixes include tables for BASIC and DOS vectors, stacks and interrupt locations, PLUS thousands of comment lines for the complete MICROSOFT™ BASIC.

MICROSOFT™ BASIC DECODED .. $29.95

The perfect supplement for your NEWDOS, from IJG, Inc.

"TRS-80™ DISK AND OTHER MYSTERIES"

by Harvard C. Pennington

132 pages written in PLAIN ENGLISH packed with HOW TO information with details, examples and in-depth explanations. Recover lost files and directories, remove file protection, make BASIC programs unlistable. How to use SUPERZAP, recover from DOS errors and MORE!

TRS-80™ DISK..... $19.95

NEWDOS/80

by Apparat

Apparat's long-awaited successor to NEWDOS+ is here! This is not an enhanced version of NEWDOS, but a completely new product. Simplified DOS commands can be instantly executed from BASIC, even within a program, without disturbing the resident code. System options, such as password protection, number and type of disk drives, BREAK key enable/disable and lowercase modification recognition, can be quickly and easily changed. Five new random-access file types allow record lengths of up to 4096 bytes, and no FIELDing! A powerful CHAIN facility allows keyboard INPUTs to be read from a disk file. An improved RENUMBER facility permits groups of statements to be relocated within program code. Diskettes may even be designated as RUN-ONLY! Features all NEWDOS+ utilities (SUPERZAP 3.0, etc.) and much more! One MTC technical staff member said having NEWDOS/80 is "better than sex" (you'll have to judge for yourself!). Includes 180-page instruction manual and MTC QUE card.

NEWDOS/80..... $ 149.95
MTC QUE Card only..... $7.50

CALL REGARDING OUR NEWDOS + UPGRADE PRICING

Complete for Model I with all utilities
Plus exclusive MTC QUE card!

NEWDOS +

$69.95 by Apparat

includes REF, RENUM, SUPERZAP, EDITOR/ASSEM., DISASSEM., DIRCHECK, and more! This is the original NEWDOS with all of Apparat's utility programs. Includes exclusive MTC QUE (Quick User Education) card.

40-Track Version..... $79.95
MTC QUE Card only..... $1.50

MORE PRODUCTS

Products damaged in transit will be exchanged.
Prices, Specifications, and Offerings subject to change without notice.

MOST ORDERS SHIPPED WITHIN ONE BUSINESS DAY

DEALER INQUIRIES INVITED

WE ACCEPT
- VISA
- MASTER CHARGE
- CHECKS
- MONEY ORDERS
- C.O.D.

- Add $2.50 for standard UPS shipping & handling
- $2.00 EXTRA for C.O.D.
- Ohio residents add 6½% sales tax.

FOR PRODUCT INFO 1-800-321-3640

CALL TOLL FREE ✓20

TO PLACE ORDER 1-800-321-3552

IN OHIO call (216)289-7500 (COLLECT)

master charge VISA

810115

META TECHNOLOGIES CORPORATION

26111 Brush Avenue, Euclid, Ohio 44132

✓ Reader Service—see page 290

● The device command from TRSDOS is deleted from DBLDOS (big deal!).

First Class Hardware

Percom could safely offer a money back guarantee for life because once you use double density, the old way seems like the stone age. However, let's examine this system more closely.

We all agreed that the hardware was first class. The software, DBLDOS, was okay. It worked just as the manual indicated. Only seven pages are devoted to the operating system—which is sufficient because there is little to add to the well-written TRSDOS manual. One of my friends even figured out how to install the hardware without taking the bottom off the interface (not recommended).

The Better Mousetrap

DBLDOS, like TRSDOS, is satisfactory until one finds something that is far superior: enter the better mousetrap.

Larry Stoddard, the number one man at Micro Systems Software, knew that I was reviewing the Doubler and wanted me to evaluate their newly developed double density operating system. He bragged about all the wonderful things that could be done with the DOS+ double density system. "OK, I'll look at it but if it takes a Ph.d. in computer science to understand the manual or use the system, I don't have the expertise to properly evaluate it," I said. "Furthermore, for every machine language addict, there are a hundred folks like me; pretty good at BASIC but not really into the F6A0 stuff." "Stewart, I only ask that you try it," he said.

Larry was accompanied by Mark, a wunderkind computer mastermind who would demonstrate the system. Mark seated himself at the keyboard and began typing at about 5000 words per minute and talking computerese at the same speed. How do they do it? I have trouble trying to walk and chew gum at the same time. He demonstrated the entire system in 12 seconds!

The manual is 28 pages long and easy for a BASIC programmer like me to understand. It is clear, concise, and mostly complete. There are several functions which are not explained in enough detail. I would grade it a B–. I give the operating system mostly straight As. It costs $99 which isn't exactly cheap. Is it really worth it, considering you get DBLDOS free with the hardware? Here are some of the features. You can decide if they are worth the money.

● DOSPLUS double density is fully compatible with TRSDOS. It also allows single drive backup, copy, etc.

● If you have lowercase hardware installed, upon booting up, your keyboard recognizes it and works like always. When you shift a lowercase letter appears. If you want typewriter format, shift-0 does the job.

● A repeating keyboard without keybounce. Hold down the R and, in about a half a second, R's begin to dance across the screen. Nice when one is editing.

● The transfer mode allows one to copy all the visible files from one disk to another, which can save a tremendous amount of time. Also, the system doesn't care about the density of the drives. It automatically recognizes disk density. I found this particular function invaluable when I wanted to transfer 76 data files from one disk which also contained the operating system to another pure data disk which already had 21 data files and no room (or need) for the operating system.

● It supports over 120 files. This is a big advantage because with an 80 track dou-

```
Drive      0      — DBLDOS  09/13/80  40        FILES,  95GRANS
Drive 1 — DBLDOS            09/13/80  40 FILES,         257 GRANS
Drive 2 — FORMAT40          11/13/80  48 FILES,         138 GRANS
```

THE VISUAL IMPACT OF DOUBLE DENSITY.
DRIVES 0 & 1 CONTAIN DBLDOS. 0 = 40 TRACK, 1 = 80 TRACK.
DRIVE 2 = 40 TRACK FORMATTED.

Figure 1.

DOSPLUS
Directory Drive: 2 REVDOS -11/11/11 - 3.1 DDEN

Filename		ATTRB	LRL	#LOG	#PHY	#GRN	#SEG	EOF
Nova	CMD	N*T0	256	34	34	6	1	0
Purge	CMD	I*U6	256	3	3	1	1	78
Bridge		N*X0	256	36	36	6	3	245
Dogstar		N*X0	256	58	58	10	1	72
Vaders	CMD	N*X0	256	12	12	2	1	189
Inotelo		N*X0	256	9	9	2	1	77
Airraid	CMD	N*X0	256	0	0	2	1	0
Format	CMD	I*U6	256	10	10	2	1	209
Backup	CMD	I*U6	256	9	9	2	1	144
Starwars		N*X0	256	49	49	9	1	67
Restore	CMD	I*U6	256	4	4	1	1	62
L	CMD	N*X0	256	20	20	4	1	0
Snakeggs		N*X0	256	55	55	10	1	225
Demon		N*X0	256	57	57	10	1	171
Transfer	CMD	I*U6	256	4	4	1	1	9
Sant		N*X0	256	47	47	8	2	207
DDT	BAS	N*X0	256	19	19	4	1	0
Yhatzee		N*X0	256	44	44	8	1	124
Bomber		N*X0	256	10	10	2	1	48
Bakgammon	CMD	N*X0	256	19	19	4	1	0
Opera		N*X0	256	28	28	5	1	107
Galaxy	CMD	I*U6	256	38	38	7	1	0
Wildwest		N*X0	256	53	53	9	1	41
Copy 1	CMD	U*U6	256	4	4	1	1	142
CLRFile	CMD	I*U6	256	2	2	1	1	153
Robot		N*X0	256	12	12	2	1	153
Sailpln		N*X0	256	55	55	10	1	28
Timetrek	CMD	N*X0	256	19	19	4	2	0
CLRfile	CMD	I*U6	256	2	2	1	1	153
Robot		N*X0	256	12	12	2	1	153
Sailpln		N*X0	256	55	55	10	1	28
Timetrek	CMD	N*X0	256	19	19	4	2	0
Diskzap	CMD	I*U6	256	11	11	2	1	144
Crunch	CMD	I*U6	256	3	3	1	1	68
Amazeing		N*X0	256	34	34	6	1	110
Othello		N*X0	256	13	13	3	1	116
H	CMD	N*X0	256	20	20	4	3	0
Diskdump	CMD	I*U6	256	4	4	1	1	84
Sargon2	CMD	N*X0	256	46	46	8	1	0
PSY		N*X0	256	56	56	10	2	85

*** 53 Free Granules ***

Figure 2.

ble density disk, one will probably run out of file space before disk space.

- A built-in glitch free screen printer, which is operated by pressing shift-clear.
- The break mode allows disable-enable of break key.
- BUILD, AUTO, and DO may be coupled to do some remarkable things. For example: My secretary arrives at 8 a.m. and turns on the computer. With no further operator input the system boots up, looks into my data bank and outputs to the printer my appointment schedule for the day and special dates I should remember. Next, it phones our company's main computer and checks for any messages keyed for me and prints them out. The possibilities of these utilities are endless thanks to a unique HBASIC in the DOSPLUS system. It allows you to go from BASIC to a command operation and, when the command operation is completed, return to exactly where you left off in the BASIC program. You may utilize this function with a statement in your BASIC program or even hit break, enter CMD"I", "DOS command with any needed parameters" and, when the command is finished, you will be back to your BASIC program. I first tested the function by hitting break in the middle of a for-next loop that was outputting data to the printer. I typed CMD"I", DIR:2 and the directory from Drive 2 was displayed and I was faced with READY. I typed CONT and pressed ENTER and the printer resumed operation exactly where it left off.
- CONFIG allows the option of adding the maximum number of tracks available to your drive to a disk that was formatted for a lesser number, set drive head step rate to maximum allowed by your drive, and automatically enable your clock speedup kit at boot up.
- The new printer driver allows forms and paging perimeters to be set and changed at will.
- RESTORE brings back files that have been accidently killed as long as you haven't subsequently written over those sectors.
- PURGE allows you to delete selected files by listing them one at a time and asking Y/N?
- DISKDUMP is a display sector/modify program and works with filespecs.
- DISKZAP is their new single/double disk editor. It's easy to use and works well.
- You may display current date/time from DOS.
- FREE displays the disk map and allocation of file space.
- CRUNCH deletes all spaces (not in quotes) and REM statements. If a BASIC program won't quite fit in available memory, crunch will save space.
- DOSPLUS figures GRANS differently. For example, their gran equals 1500 bytes, others about 1200. Somehow they provide a bit more space on a given disk.
- The Spooler works well. I tried it with the following short program; For K=1 to10: For I=50 TO 191: LPRINTI;: NEXT I, K. I hit ENTER and after a few seconds, the printer began printing and I was faced with a READY? I returned to DOS and backed up a disk while the printer/spooler was doing its thing. The data is held in high memory, waiting to be printed. This avoids the problems of some spoolers which store the data on disk and must access the disk occasionally.
- One is not restricted to an LRL of 256 (logical record length). Chose from 1 to 256. This allows maximum utilization of disk space.

Love at First Sight

The full impact of double density is obvious when one examines Fig. 2. All of my favorite games on one 80 track disk with 53 DOSPLUS grans or 79.5K of space left! A formatted 40 track wil get you over 170K usable space and over 350K usable on a formatted 80 track. While Percom will make a bundle from their Doubler, I predict that sales of drives will drop. A single 80 track drive system will now do the work of four 40 track single density drives (with some limitation). Four 80 track drives provide over 1.4 kilobytes on line!

The advantages of the Doubler are immediate and obvious. It's love at first sight. To fully appreciate the value of the DOSPLUS system, one must sit at the keyboard and use it.

The fact that I bought both systems (no freebees) means nothing...I buy darn near everything out of curiosity, most of which I never use, let alone review. However, when one of my friends, a miserly student-engineer, and the other, a man who works hard for his money, shell out that much cash—that should tell you something.

In case you are wondering, I have no connection with Percom or Microsystems Software. I'm just a satisfied customer.

I understand that double density zaps are now available for NEWDOS-80 and VTOS 4.0. I have not seen them in operation but will review their performance another time.■

CG-16 Symbol Generator
G. P. Associates
Sacramento, CA
$94.50

by Al Domuret

A relatively new TRS-80 hardware firm by the name of G. P. Associates has a useful hardware item for the Model I, an upper-lowercase and graphics generator board with a new collection of graphics symbols. The kit consists of a pre-assembled circuit board that is easily installed by the user.

The CG-16

The CG-16 Symbol Generator not only displays lowercase with descenders, but it significantly expands the TRS-80's graphics. In addition to the standard 64 TRS-80 graphics characters, it generates the complete ASCII character set and about 100 additional graphics symbols.

The circuit board can be installed in any Model I TRS-80 configuration. Additionally, it is compatible with both the Electric Pencil and Scripsit.

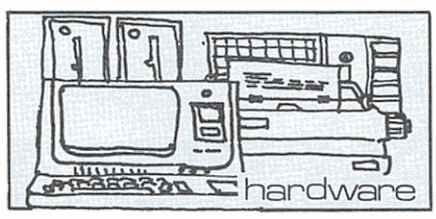
There are a good many TRS-80 programs around that utilize the traditional TRS-80 coarse graphics, and so the question of compatibility arises. With CG-16 all of the old TRS-80 graphics characters are retained and, as a bonus, the appearance of many existing graphics displays is improved.

Features

As for the lowercase video display with descenders—you have to see it to appreciate it. Reading improved text display with correct lowercase descenders is definitely easier on the eyes.

Another important feature of CG-16 hardware installation is that it has a selector switch. This allows the user to switch off the lowercase hardware if it is incompatible with the software he is running. Radio Shack's mod is permanently hardwired with no selector switch.

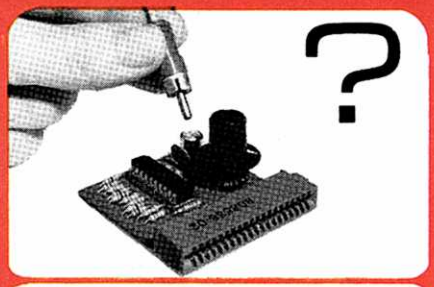
To install CG-16, the original TRS-80 graphics generator must be removed and retained for use in the new board. Complete installation instructions are provided. It takes about 45 minutes—or less, if you are an experienced hardware hacker.■



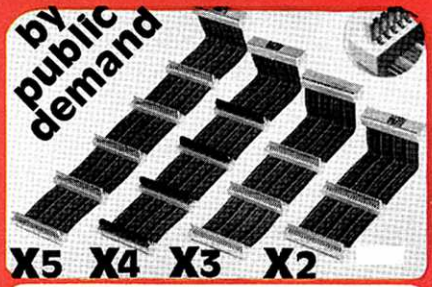
NEW

Joystick versions of the Fantastic Games by BIG 5 (see page 109) and Software Innovation (see page 77). Available on tape or disk, same price as plain version. One "Stick 80" works with all. Money back guarantee. In stock now.

STICK-80 MAKES KEYBOARD OBSOLETE.
Features the famous ATARI Joystick. 8 directions + fire control. Simple instructions to make joystick versions of most action games. Plugs into keyboard or expansion int... Price includes ATARI joystick with ALPHA interface and instructions. FREE "MAGIC ARTIST" program. . . . 1S39.95
Super Real Time Action Graphic Sound games for Stick-80 by Software Innovations: ALIEN INVASION.
COSMIC INTRUDERS. BREAKOUT Each $9.95
STELLAR ADVENTURE: super action with sound. . . . $14.95
Software authors and distributors: contact us for joystick conversion package for your existing games.

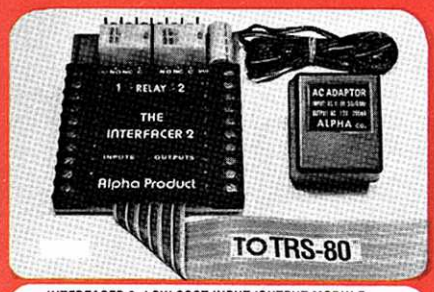


MUSIC-80 MUSIC-80 MUSIC-80 MUSIC-80 MUSIC-80
Use existing software
or write your own. With this low cost 8 bit digital to analog converter you can synthesize up to 5 music voices. Built-in volume control handy when stereo not near TRS-80. Simply plug the "MUSIC-80" into the keyboard or the E/I screen printer port and connect the output (RCA jack) to any amplifier. The Radio-Shack $12 speaker/amplifier works fine.
Fully assembled and tested. 90 day warranty $39.95



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.

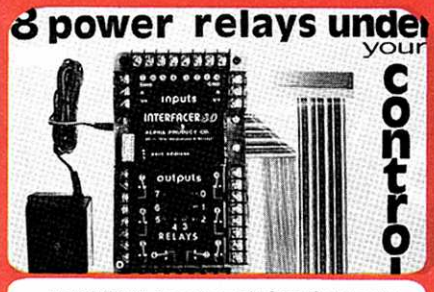


INTERFACER 2: LOW COST INPUT/OUTPUT MODULE.
Still the best value in sense/control devices. Use it for energy control, burglar alarm, darkroom, selectric drive, model trains, robots, Skinner box...
—8 latched TTL outputs. 2 relays SPDT 2A. 125V. contacts
—8 TTL/CMOS inputs. Input 0 and 1 are optically isolated.
—Neat and compact design, very easy to use:
10 A = INP(0) 'Reads the 8 inputs (if A = 0: all inputs are low) 20 OUT 0,X 'Controls the outputs and the relays
Assembled & tested, 90 day warranty. Price includes power supply, cable to KB or E/I, superb user's manual, free phone dialer program: $95. Manual only: $5.

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

LET THE "CHAIN BREAKER" 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 "CHAIN BREAKER" and replace the cover. Voila!!!
Now you can change and move your drives around without disassembly. Keep the cover on and keep the dust out. High reliability gold plated contacts, computer grade 34 conductor cable. Tested and guaranteed.
Get one for each drive.....only. $13.95



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 Product Co.

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.

85-71 79 St. WOODHAVEN N.Y. 11421 ☎ Info and order: (212) 296•5916

GAME BREAKTHROUGH:

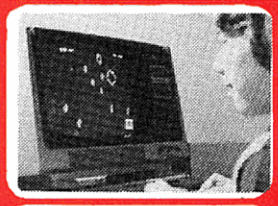
THE STICK 80

- ☑ FEATURES THE FAMOUS **ATARI JOYSTICK.**
- ☑ WORKS WITH ANY LEVEL 2 AND DISK SYSTEMS.
- ☑ PLUGS DIRECTLY INTO KB. OR E/I (SCREEN PRINTER PORT).
- ☑ INCLUDES SIMPLE, DETAILED INSTRUCTIONS TO MAKE JOYSTICK VERSIONS OF MOST ACTION GAMES.
- ☑ COMPATIBLE WITH **ANY** OTHER TRS-80 ACCESSORIES.
- ☑ FULLY ASSEMBLED AND TESTED. READY TO PLUG IN AND ENJOY.
- ☑ FUN TO EXPERIMENT WITH IN **BASIC:** SIMPLY USE INP(0).
- ☑ 90 DAY PART AND LABOR WARRANTY.

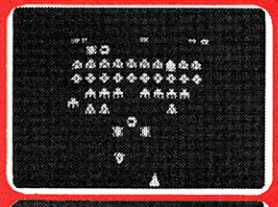
UNCONDITIONAL **MONEY BACK GUARANTY:** Try your STICK-80 for 14 days. If for any reason you are not delighted with it, return it for a prompt and courteous refund (including shipping and handling). PRICE INCLUDES ATARI JOYSTICK + ALPHA INTERFACE + DETAILED INSTRUCTIONS + DEMO PROGRAM. ONLY $39.95

MODEL 3 OWNERS: WE HAVE A **MODEL 3 TO MODEL 1 ADAPTER**

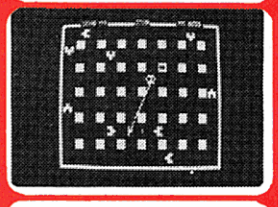
BIG FIVE SOFTWARE SUPER FAST MACHINE LANGUAGE ACTION GAMES



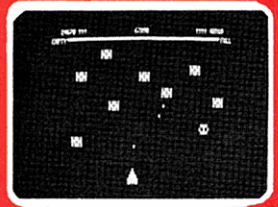
If you and your TRS-80 have longed for a fast-paced arcade-type game that is truly a challenge, then **SUPER NOVA** is what you've been waiting for. In this two player machine-language game, large asteroids float ominously around the screen. Suddenly your ship appears and you must destroy the asteroids before they destroy you! (But watch out because big asteroids break apart into little ones.) The controls that your ship will respond to are thrust, rotate, hyperspace, and fire. All right! You've done it! You've cleared away all the asteroids! But what is that saucer with the laser doing? Quick! You must destroy him fast because that guy's accurate! ◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆ **SUPER NOVA®**



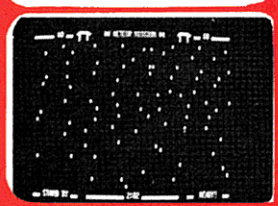
The sound of the klaxon is calling you! Cruel and crafty invaders have been spotted in battle formation warping toward Earth at an incredible speed. Suddenly, your ship materializes just below the huge flock of invaders. Quickly and skillfully you shift right and left as you carefully fire your lasers at them. But watch out! A few are breaking out of the convoy and flying straight at you! As the whine of their engines gets louder, you place your finger on the fire button knowing all too well that this shot must connect—or your mission will be permanently over! With sound effects! ◆◆◆◆◆◆◆◆◆◆◆◆◆ **GALAXY INVASION®**



Your TRS-80 screen has been transformed into a maze-like playfield for this game. As your ship appears on the bottom of the screen, eight alien ramships appear on the top. All of them are traveling at flank speed directly at you! Quickly and boldly you move toward them and fire missiles to destroy them. But the more aliens you destroy, the faster the remaining ones become. If you get too good you must endure the wrath of the keeper of the mazefield: the menacing "Flagship". You must destroy him fast because, as you will find out, that guy's accurate! With sound effects! ◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆ **ATTACK FORCE®**



With thousands of stars whizzing by you, your **SPACE DESTROYER** ship comes out of hyperspace directly under a convoy of aliens. Almost effortlessly, you skillfully destroy every last one. But before you can congratulate yourself, another set appears. These seem to be slightly more intelligent than the first set. Quickly you eliminate all of them, too. But your fuel supply is rapidly diminishing. You must still destroy two more sets before you can dock with your space station. All right! The space station is now on your scanners! Oh no! Intruders have overtaken the station! You must skillfully fire your neutron lasers to eliminate the intruders from the station before your engines run out of fuel and explode! With sound! ◆◆◆◆◆◆◆◆◆◆◆◆◆◆ **COSMIC FIGHTER®**



The second **Big Bang** has occurred and the galaxy is full of stray asteroids and meteors. As you look through your space port you see a belt of asteroids drifting across the screen blocking your path to the safety of the space station above. But be careful because meteor showers, exploding suns and invading aliens may strike your ship and send it hurtling back to ground level. How many times can you and your opponent maneuver through those obstacles before time runs out? With sound effects! ◆◆◆◆◆ **METEOR MISSION II®**

SPECIAL

INTRODUCTORY OFFER:

Order the STICK-80 with one or more games and choose one:

•FREE ALPHA GREEN SCREEN
OR
•FREE MAGIC ARTIST PROGRAM

IN ADDITION YOU MAY:
ORDER 2 GAMES AND TAKE 10% OFF
ORDER 3 GAMES AND TAKE 15% OFF

All games are written in machine language and supplied on cassette.

EACH GAME IS AVAILABLE IN 2 VERSIONS

A SIDE 1: LEVEL II 16K FOR MODEL 1
SIDE 2: LEVEL II 16K FOR MODEL 3 $15.95

B SIDE 1: DISK 32K FOR MODEL 1
SIDE 2: LEVEL II 16K FOR MODEL 1
THE DISK VERSION SAVES THE HIGH SCORES ON DISK $17.95

* ALL THESE NEW JOYSTICK GAMES WILL ALSO WORK USING THE KEYBOARD WITHOUT ANY MODIFICATIONS.

GOOD NEWS: If you already have a non-joystick version of these BIG FIVE SOFTWARE games, send the original tape(s) with your STICK-80 order + $3 per game. We will send you a new Joystick version of your game(s).

ALSO AVAILABLE **DOUBLE-STICK 80:** 2 ATARI JOYSTICKS + ALPHA DUAL INTERFACE + INSTRUCTIONS + DEMO PROGRAM: $59.95

ALPHA Product Co.

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.

85-71 79 St. WOODHAVEN N.Y. 11421 phone: (212) 296 5916

THE ASSEMBLY LINE

by William Barden, Jr.

"I love the Z-80, but some people look upon it as the result of an illicit union between an itinerant 8080 and a marketing representative and do not think kindly of its instruction repertoire."

This column describes a typical computer project—one which should be easy but invariably isn't.

In the process of describing the project, we'll discuss some of the aspects of using the serial interface. We'll also be looking at some Color Computer 6809E programming and a new program to investigate disk data.

The Problem

The Color Computer has a line printer interface built into the hardware and software via the RS-232C (serial) output channel. Possibly Radio Shack will come out with some inexpensive serial to Centronics bus device allowing you to connect the Color Computer to an existing Model I/III line printer. In the interim, however, I wanted listings of Color Computer software.

The options seemed to be:

- Design a hardware device using a UART so that I could go directly from the Color Computer to the line printer.
- Run the serial output from the Color Computer to the RS-232C input of the Model I and thence to the line printer.

Obviously, the second option was most attractive. Everyone knows software is easy. That's why people feel no qualms about copying.

The Color Computer End

The Color Computer RS-232C interface is handled in Microsoft's 8K BASIC interpreter. The 8K version implements serial output only; the extended BASIC will allow normal communication in both directions.

All output is handled by turning a bit of a PIA (Peripheral Interface Adapter) on and off to get the serial output (See Fig. 1). Working storage in page 0 of RAM allows you to select the baud rate and certain other parameters as shown in Table 1.

The serial output connector is a four-pin DIN connector with the signal assignments shown in Fig. 2. Pin 3 of this connector is ground. Pin 4 is the serial output to the printer. A high (greater than +3 volts) represents a ready condition for the the printer; this means that the printer has printed the previous character and is ready to accept the next. A low (less than −3 volts) represents a "busy" condition; this means that the printer is still processing the last character. Pin 1 is not used. This came back to haunt me later.

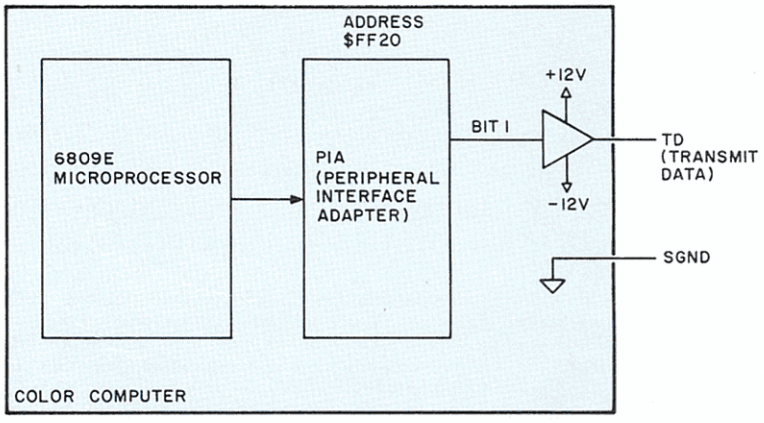


Fig. 1. Color Computer Serial Output

The Model I End

The other end of the 4-connector cable is shown in Fig. 3. It attaches to the Model I RS-232C cable and from there follows a circuitous route past the (unique) connector into the RS-232C board.

The RS-232C board's complexity is widely overrated. It doesn't do that much, but what it does, it does in abstract fashion, especially if you read the RS-232C manual!

In the minimum configuration, three lines are required for serial communication—one is a ground line, one a transmit data (TD) line and one a receive data (RD) line.

The usual signal for asynchronous communication looks like Fig. 4. If the signal is high (positive voltage or logic 1), it is said to be marking; if the signal is low (negative voltage or logic 0), it is spacing. This nomenclature dates back to pre-TRS-80 times when teletypewriting devices were used even without the Color Computer.

The serial interface looks for a change from mark to space. This is called the start bit. As soon as the mark changes to a space (1 to 0), the serial interface says "Aha! A start bit! Data is a-coming in." By prior arrangement, the data coming in must be at an agreed upon baud rate with an agreed upon number of data bits, parity bit, and start and stop bits.

When the start bit is received, the serial interface, using its own clock timing, looks for the subsequent data bits, parity bit, and stop bits. Note that this action is initiated with the start bit; there is no clock on the serial line. Because the bits are coming in at a predefined baud rate, the serial interface clock timing knows

RAM ADDRESS		DEFAULT
$95	BAUD RATE	$00
$96		$57
$97	LINE DELAY	$00
$98		$01
$99	COMMA FIELD WIDTH	$10
$9A	LAST COMMA FIELD	$70
$9B	LINE PRINTER WIDTH	$84

BAUD RATES $ = HEX

RATE	LOCATION $95	LOCATION $96
120	$01	$CA
300	$00	$BE
600	$00	$57
1200	$00	$29
2400	$00	$12

Table 1. Color Computer Printer Parameters

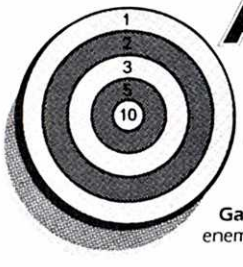
The Resurrection of IAGO™

It is the hour. The clanging of swords is silent—only far off sounds of men struggling are heard. Othello is down. IAGO stands astride—beckoning to all challengers.

IAGO, the classical Shakespearean antagonist to Othello has returned as the ultimate Reversi/Othello game. IAGO contains one of the most powerful strategies ever developed for a Microcomputer system.

IAGO is for the novice to advanced player, and with 7 levels of play that will provide years of challenging enjoyment. So, if computers are your game then ask your Datasoft dealer for a demonstration or contact us for details.

TRS-80 Models I & III
(Diskette) $24.95
(Cassette) $19.95



ARCADE 80

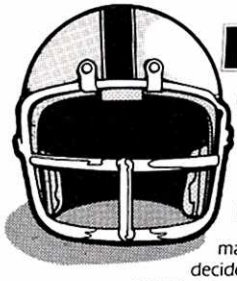
Three high quality arcade games in one simple format. This is home entertainment at its best.

Game 1: Falling Bricks—Balances accuracy against time. If you're not quick enough the bricks come crashing down on you.

Game 2: AstroMines—Seek and destroy the enemy's mines before fuel runs out. Be careful or you will destroy yourself.

Game 3: Star Run—A Star Raiders type game filled with refreshing graphics.

TRS-80 Model I (Diskette) $24.95 (Cassette) $19.95



FOOTBALL CLASSICS

Goes beyond ordinary computer football programs. You can recreate sports history by pitting top teams against each other in their best years. You make the plays and the statistics decide the outcome. And that's not all!!!! With the easy to use updated input file, you can update or create your own statistics. Included with the program are six teams of the past.

TRS-80 Models I & III (Diskette) $24.95 (Cassette) $19.95

Datasoft Inc.™
COMPUTER SOFTWARE

407

16606 Schoenborn Street, Sepulveda, Calif. 91343—(213) 894-9154

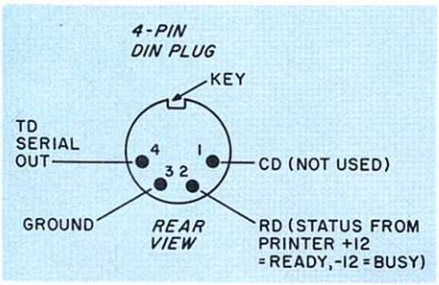


Fig. 2. Color Computer Serial Out Connector

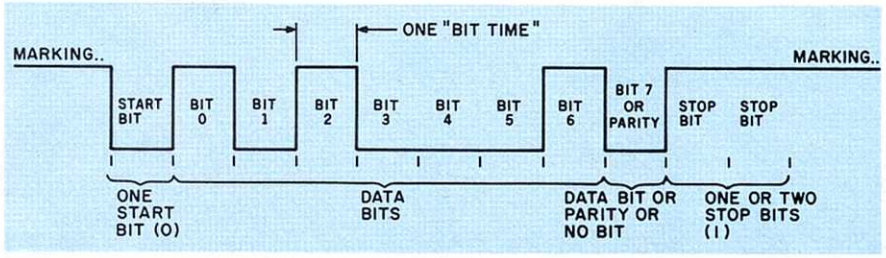


Fig. 4. Asynchronous Communication Signals

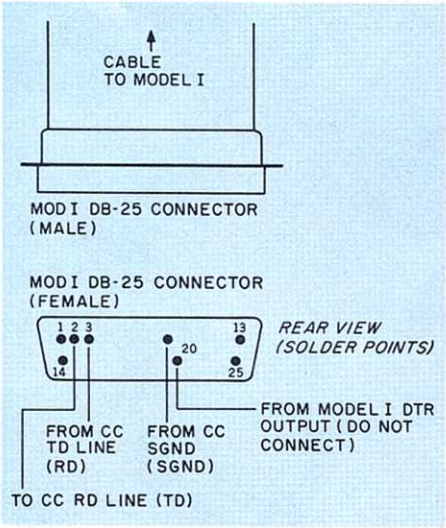


Fig. 3. Model I RS-232C Connector

approximately where to look for the next bit. Obviously if this process were done over 34,768 bit times the serial clock would get out of sync with the incoming data. However, for 10 or 11 bits there is no problem.

That's basically all there is to passing data serially, in the simplest case. If data communication is allowed in both directions simultaneously, then the transmission is called full duplex. One character can be transmitted one direction while another goes the opposite direction. This is a common mode for terminals, where the terminal sends a keyboard character via telephone lines, and the receiving computer immediately returns the same character. The resulting echo verifies that the character has been received properly by the computer. If data transmission is allowed in only one direction at a time, the transmission is said to be half-duplex. A terminal using this mode prints locally with no echo back from the remote system.

The RS-232C board times out incoming characters. After the start bit is received, it starts reading in the remaining bits. The bits are converted from their serial form to an eight-bit value, stripped of parity, start and stop bits. This eight-bit value can then be read into the A register by a simple IN instruction.

Data transmitted from the RS-232C board goes through the inverse process. A parallel byte from an OUT is converted to a serial bit stream and sent out over the TD line. The same serial clock times this operation.

The protocol from computer to RS-232C board is similar to any other device. If a byte has been sent to the board and is being transmitted from the one-byte transmitter buffer, the board cannot accept another character during the transmission time. A "transmitter buffer empty" flag indicates when the RS-232C board can accept another character. A "received data" flag indicates when another received character has been assembled and can be read.

What about those other lines? The other RS-232C signals are standard signals that represent various communication conditions. If a modem is being used, for example, it's convenient to know that the phone is still connected or that someone is calling up your computer to see if he can get a copy of the new Color Computer to Model I System Printer Driver he has heard about. These lines are also used as status lines for a non-communication environment. Clear-to-send (CTS), for example, is commonly used for NEC Spinwriter interfacing, or other serial printers.

Why use other lines when all serial transmission could be handled with three lines? For faster rates. The Spinwriter and some other devices are buffered. They contain memory that will accept data at extremely fast serial transmission rates, much faster than printing speeds. If continuous printing is not done, then the over-

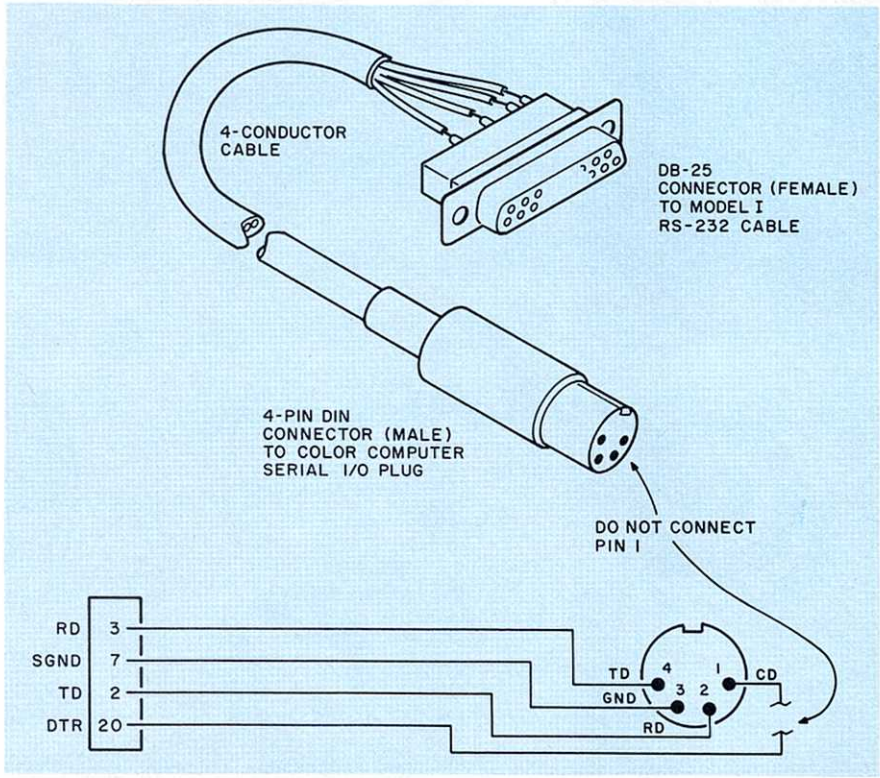


Fig. 5. Color Computer to Model I Cable

THE 80'S BELONG TO SNAPP!

MASTER / SLAVE
This software package was designed to support the transferring of files from one Model II to another, via direct connection or modem/phone line connection. ALL kinds of files, and baud rates up to 9600 are fully supported. Transfer files in either direction, even with the SLAVE Model II UNATTENDED! $150

AUTOMAP
Save time creating a formatted screen for input ease. This ON LINE/OFF LINE utility will display information with simple GET and PUT statement commands. Realize up to 75% time reduction. No user memory. Programming is easier. Input with ease using AUTOMAP. $75

ITOII
A helping hand when converting BASIC programs from the Model I to the Model II. Automatically adjusts PRINT @, and PRINT USING to compensate for differences in the language. Advises you where adjustments are necessary for PEEK, POKE, etc. $25

CONVERT
This remarkable utility converts "V" format files (the sequential format used by the SHACKS COBOL and BASIC Compilers) to the "F" format files (the sequential file format used by the BASIC interpreter and BASCOM), and vice versa. Without this product, programs written for the interpreter will have to be RE-KEYED to be used by the SHACKS Compiler BASIC. $75

DIAL
USR 330D Auto Answer/Auto Dial, Direct Connect Modem. 300 baud, originate /answer 103J compatible. When used in conjunction with our DIAL software is capable of complete origination of communications with remote locations without operator intervention. Special combination price, modem and software. $420
Software only $50

DOUBLE TAKE 3741
This is not a football play but the way to play ball fast in converting IBM 3741 and similar formated diskettes to Radio Shack formated disks or vice versa. Fast is the name of the game. $200

3M SCOTCH DISKETTES
Double density certified 8" Floppies for the Model II. Better quality is not available at any price. Ten diskettes to a box.

Quantity (boxes)	Price Per/Box
1	$35.50
5	$34.50
10	$33.50
20	$32.50

$3.00 shipping charge. This charge is waived if software is purchased on same order.

HOSTII/TERMII
Allows remote control of a Model II from another Model II, or any ASCII terminal. Our Host system, unlike the one supplied with TRSDOS 2.0, supports accurate screen positioning on the Term station. Without this feature, formated displays appear on the screen looking like randomly placed garbage. Requires NO user memory! This system is designed to provide software support to our customer locations without ever leaving the office. Custom versions are now available for most nationally distributed terminals as a $25.00 option. Call for details. $50

SPOOLER Model I, Model II and Model III
Our workhorse! Unlike the one supplied with TRSDOS 2.0, ours requires no special knowledge or training on the part of the operator. Additionally ours performs much better. On the Tandy SPOOLER, everytime a disk is accessed, the printer stops dead! This package is available for Model I, in the TRSDOS/NEWDOS 80 versions, or for the Model II. Greatly enhances system performance when running typical business applications. Many applications have been benchmarked to run nearly TWICE AS FAST with the SPOOLER installed. Installs in minutes and no changes are required to your programs. Preferred Model II versions require NO user memory. An additional feature for the Model II version only is Disk Spooling support which we highly recommend for word processing applications. $100.00

DISK SPOOLING OPTION $50.00

ALL PRODUCTS NOW AVAILABLE FOR THE MODEL III

SNAPP, INC.
SNAPP, INC.
SNAPP, INC.
SNAPP, INC.
SNAPP, INC.

8160 Corporate Park Dr.
Cincinnati, Ohio 45242

Call Toll Free ✓232
VISA 1 - 800 - 543-4628
Ohio residents
master charge
call collect (513) 891-4496

All products now available to run with TRSDOS 2.0.

ULTRA PPD
This is the ultimate Proportional Printer Driver that does the job the others do not. Add to the Electric Pencil and your print will look like its copy has been typeset. No word processor should be without this enhancement. Now available for the DWII and the LPIV. $100

BPRINT
Allows you to access a serial printer simultaneously with the standard parallel printer. Easy interface to BASIC. Drive two printers at once! $75

XPRINT
Print neatly formated hard copy listings of BASIC programs from disk. Programs may be ASCII or compressed. Quick and easy group selection allows you to print many listings with one command. $35

DOSFIX
A collection of patches to TRSDOS and BASIC to enhance their usability and function. Includes our well known BREAK7E patches and facilities to disable verify detect which will increase average disk speed by 30%. Free with any Model II Software Package. Purchased separately $10

TERMS OF SALE:
Credit card customers, add 2% C.O.D. customers add $2. Ohio residents add 5½% sales tax. Shipments normally made the same day we receive your order. Credit granted to governmental agencies, educational institutions and D & B rated business firms. Please include purchase order number when ordering.

OUR GUARANTEE:
If your diskette arrives damaged, we will replace it without charge. If you ever accidentally damage it, we will replace it for a $10 handling charge. For a period of one year, we will provide you with any enhancements or updates for a $10 handling charge. For a period of one year, if errors are discovered in the programs, they will be corrected without charge. In the event we cannot correct an error, you may return the program material for a refund.

Electric Pencil is a trade mark of Michael Schrayer Software, Inc.

TRS-80 and TRSDOS are trademarks Radio Shack division of Tandy Corporation.

NEWDOS and NEWDOS/80 are trademarks of Apparat, Inc.

✓ Reader Service—see page 290

GO WITH A WINNER!

PRINTERS

Okidata Microline 82

	List Price	Your Cost
NEW — Okidata Microline 82 . Features: 80 C.P.S.; Bidirectional; Full forms Control; Choice of Parallel or Serial Interface as Standard; Buffered Serial Optional	$ 960.	Ask For Our Price $850
Epson Model MX-70 . . . . . Features: 80 C.P.S.; Unidirectional; Bit Plot Graphics; Adjustable Sprocket from 4" — 10"	$ 495.	Ask For Our Price
Epson Model MX-80	$ 645.	Ask For Our Price
Okidata Microline 80	$ 800.	Ask For Our Price $695
Anadex DP-9500 & DP-9501	$1650.	Ask For Our Price
Anadex DP-8000	$1095.	Ask For Our Price

INTERFACES —

Epson Apple Plug-in Interface & Cable	$110.
Epson Serial Interface & Cable	$ 90.
Epson IEEE 488 Interface & Cable	$ 80.
Model TRS-80 Expansion Interface Cable	$ 35.
Model 232 Universal CA Cable, Universal Serial Interface	$ 25.
Okidata Microline 80 Tractor	$100.
Okidata Microline 80 RS-232 Interface with 256 Character Buffer	$170.

SYSTEMS FURNITURE —

We stock Systems Furniture tables.

PRINTER STANDS —

We stock Universal printer stands for most printers.

COUPLERS & MODEMS —

We handle the broad Tek-Com line.

BREAKOUT BOX

	List Price	Your Cost
NEW — HAYES TECHNICAL . . . . Full RS 232 Breakout and Patching Interface Box	$195	Ask For Our Price

master charge VISA

ASK FOR OUR INSTANT DISCOUNT
From Roy Hawthorne
Talk To Bill Tokar On Applications

CALL TOLL FREE
U.S.A.
1-800-521-2764
MICHIGAN
1-800-482-8393

Reminder:
We are open
8:30 AM to 5:00 PM EST
Monday through Friday

WRITE TO:
"The Stocking Source" ✓438
NEW 24069 Research Drive
Farmington Hills, MI 48024
313-474-6708

all program runs faster as characters are sent out rapidly to the buffer and the program continues processing instead of waiting for the printer's completion of the next character.

Also, without a status line and an unbuffered printer (there are few of these left) it's possible to lose characters as the printer does a carriage return and line feed at the end of the line. Without status back to inhibit transmission, several characters might have been sent out and ignored.

Physical Considerations

With both the Color Computer and Model I researched, it became laughingly simple to implement the scheme. First was a run down to the local Radio Shack, 35 miles away. No, they didn't have four-pin DIN connectors—plenty of five-pin ones, though... It was DIN, DIN, DIN, for the rest of that day. Finally, at MarVac Electronics—four-pin DINs. The RS-232C female connector to fit the Model I was easy. Any computer or electronics store usually carries them.

A search of my wire supply yielded plenty of four-conductor wire, but all of it in 23-inch lengths. I finally got four-conductor rainbow wire at my local Radio Shack and assembled the cable shown in Fig. 5. The length can be any reasonable length, up to hundreds of feet.

The Software

There is no additional software required on the Color Computer end. Executing an LLIST or a PRINT-2 will send out characters at 600 baud with one start bit, seven data bits, nor parity, and two stop bits, as shown in Fig. 6. The Model I must be configured to this standard. Optionally, the baud rate can be changed in the Color Computer by POKEing into locations 149 and 150 (decimal) as shown in Table 1.

On the Model I end, however, there's plenty of new software—well, plenty compared to the Color Computer. It's shown in Program Listing 1. It is assembled as a CMD file which can simply be loaded and executed from disk.

The CCXFER program first executes an OUT to 0E8H. This simply resets the RS-232C. Next, the RS-232C switches are

```
F000            00100          ORG     0F000H          ;CHANGE FOR YOUR SYSTEM
                00110 ;*****************************************************
                00120 ;*PROGRAM TO USE MODEL I LINE PRINTER FOR COLOR COMPUTER*
                00130 ;*LINE PRINTER OUTPUT.                                 *
                00140 ;*    MODEL I: PIN 3 (RD) CONNECTS TO CC PIN 4 (TD)    *
                00150 ;*             PIN 7 (SGND) CONNECTS TO CC PIN 3 (GND) *
                00160 ;*             PIN 2 (TD) CONNECTS TO CC PIN 2 (RD)    *
                00170 ;*             PIN 20 (DTR) CONNECTS TO CC PIN 1 (CD)  *
                00180 ;*SET MODEL I RS-232 TERM/COMM SWITCH TO TERM           *
                00190 ;*SET RS-232-C BOARD SWITCHES AS FOLLOWS:              *
                00200 ;*             S1  S2  S3  S4  S5  S6  S7  S8          *
                00210 ;*             CL  OP  CL  OP  OP  OP  CL  OP          *
                00220 ;*             -   7-BITS  PD  2 STOP  600 BD          *
                00230 ;*****************************************************
                00240 ;
F000 D3E8       00250 CCXFER   OUT     (0E8H),A        ;RESET RS-232-C
F002 DBE9       00260          IN      A,(0E9H)        ;READ SWITCHES
F004 E6F8       00270          AND     0F8H            ;DISCARD BAUD RATE
F006 323EF0     00280          LD      (IMAGE),A       ;SAVE
F009 F600       00290          OR      0               ;SET BRK
F00B D3EA       00300          OUT     (0EAH),A        ;PUT IN CTRL REG
F00D DBE9       00310          IN      A,(0E9H)        ;GET SWITCHES
F00F E607       00320          AND     7               ;GET BAUD RATE
F011 2136F0     00330          LD      HL,BAUDTB       ;ADDRESS OF BAUD RATE CODES
F014 0600       00340          LD      B,0             ;ZERO B
F016 4F         00350          LD      C,A             ;BAUD RATE TO BC
F017 09         00360          ADD     HL,BC           ;POINT TO CODE
F018 7E         00370          LD      A,(HL)          ;GET CODE
F019 D3E9       00380          OUT     (0E9H),A        ;PUT IN BRG
F01B DBEA       00390 CCX050   IN      A,(0EAH)          ;GET UART STATUS
F01D CB7F       00400          BIT     7,A               ;TEST BYTE AVAILABLE
F01F 28FA       00410          JR      Z,CCX050          ;GO IF NO
F021 3A3EF0     00420          LD      A,(IMAGE)         ;GET IMAGE
F024 F604       00430          OR      4                 ;RESET BRK
F026 D3EA       00440          OUT     (0EAH),A          ;PUT IN CTRL REG
F028 DBEB       00450          IN      A,(0EBH)          ;GET BYTE
F02A CD3B00     00460          CALL    3BH               ;OUTPUT
F02D 3A3EF0     00470          LD      A,(IMAGE)         ;GET IMAGE
F030 E6FB       00480          AND     0FBH              ;SET BRK
F032 D3EA       00490          OUT     (0EAH),A          ;PUT IN CTRL REG
F034 18E5       00500          JR      CCX050            ;LOOP
                00510 ;
F036 22         00520 BAUDTB   DEFB    22H             ;110 BAUD
F037 44         00530          DEFB    44H             ;150
F038 55         00540          DEFB    55H             ;300
F039 66         00550          DEFB    66H             ;600
F03A 77         00560          DEFB    77H             ;1200
F03B AA         00570          DEFB    0AAH            ;2400
F03C CC         00580          DEFB    0CCH            ;4800
F03D EE         00590          DEFB    0EEH            ;9600
F03E 00         00600 IMAGE    DEFB    0               ;SWITCH IMAGE
F000            00610          END     CCXFER
00000 TOTAL ERRORS
```

Program Listing 1. Model I CCXFER Program

✓ Reader Service—see page 290

read. The baud rate portion on the lower three bits is discarded and the remainder of the image saved. This image is output to the control register of the RS-232C (PUT 0EAH,A) to set up the number of data bits, parity, and number of stop bits.

The three bits of the switches define the baud rate. These are used to index the BAUDTB table to pick up the proper code to pass to the RS-232C controller baud rate generator (OUT (0E9H,A).

The code at CCX050 on is the main loop of the program. (The preceding code is executed once to read the switches, so the switches cannot be redefined during execution.) The UART status is read by IN A, (0EAH). If bit 7 of this status is set, the RS-232C controller has received a data byte. If the receiver buffer has a character, an IN A, (0EBH) picks up the character and passes it to the Level II line printer routine at 3BH. The program then loops for the next character.

The remainder of the code is concerned with setting the TD line. This line is used as a ready line to the RD input of the Color Computer. It must be high for a printer ready condition. At high baud rates it's possible to overrun the line printer speed.

The Line Printer III, for example, is a 120 character per second printer, which would not be able to keep up with a 2400 baud rate (240 characters per second). Characters would be lost. The TD line is therefore reset (negative voltage or logic 0) just prior to calling the line printer routine and set (positive voltage or logic 1) just after returning from the routine. The BRK bit passed to the RS-232C control register is controlled by data bit 4. This bit is reset by an OR 4 and set by an AND 0FBH before it is output by an OUT (0EAH), A. Reset generates a 1 output, while set generates a 0 output on the TD line.

Debugging

I love computer periodical articles. Most of them look like the author whipped up the design and program in an hour and a half. This job took twelve hours, including travel. After some misunderstandings about the RS-232C, including an inverted voltage output on TD and an LD in place of an IN, which I just could not see, I was able to print one character properly. Then the Color Computer hung up.

The problem? I had cleverly connected the CD line of the Color Computer to the DTR output of the Model I, thinking ahead to my next project, transmitting files back and forth between the two computers. This signal in some way affects the serial operation.

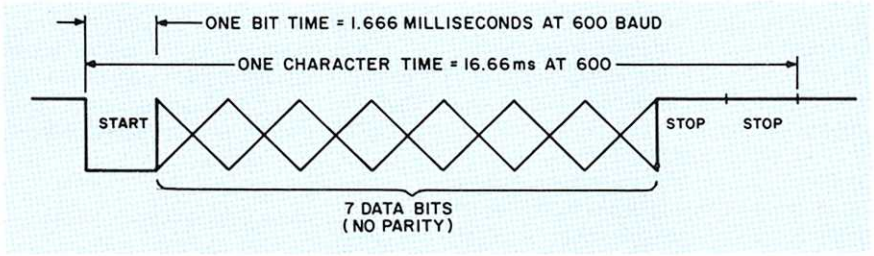


Fig. 6. Color Computer Serial Data to Printer

Operation

To enable printing: connect the cable between the two computers. Turn on or reset the Color Computer and change the baud rate in RAM by a POKE if applicable. Set the Model I switches as shown in Program Listing 1; the baud rate switches must be reconfigured to the Color Computer baud rate if it has been changed. Load CCXFER into the Model I and start execution. Printing to your Model I line printer should now be possible.

I now have printer output from the Color Computer through the Model I. Wasn't it easy? Meanwhile, my next project, transmitting files between the two machines, is proceeding nicely. It should be a snap...

More 6809E Programming

Several columns ago I described some of the aspects of the new Color Computer. Let me describe a few more related to the 6809E microprocessor programming.

The 6809E is an upgrade of the 6800 microprocessor. The 6800 has a nice instruction set built along classical lines. I love the Z-80, but some people look upon it as the result of an illicit union between an itinerant 8080 and a marketing representative and do not think kindly of its instruction repertoire. Those people will want to program on the Color Computer.

The basic 6800 used the registers shown in Fig. 7. The 6809 adds some additional registers to the two accumulators, index register, and stack pointer of the 6800, as shown in the figure. A second index register provides more indexing capability. A second stack pointer points to a user stack. This stack holds user parameters, while the original hardware stack holds return addresses and the environment.

The accumulators can be treated as one 16-bit register to allow 16-bit arithmetic operations (sound familiar?). The direct

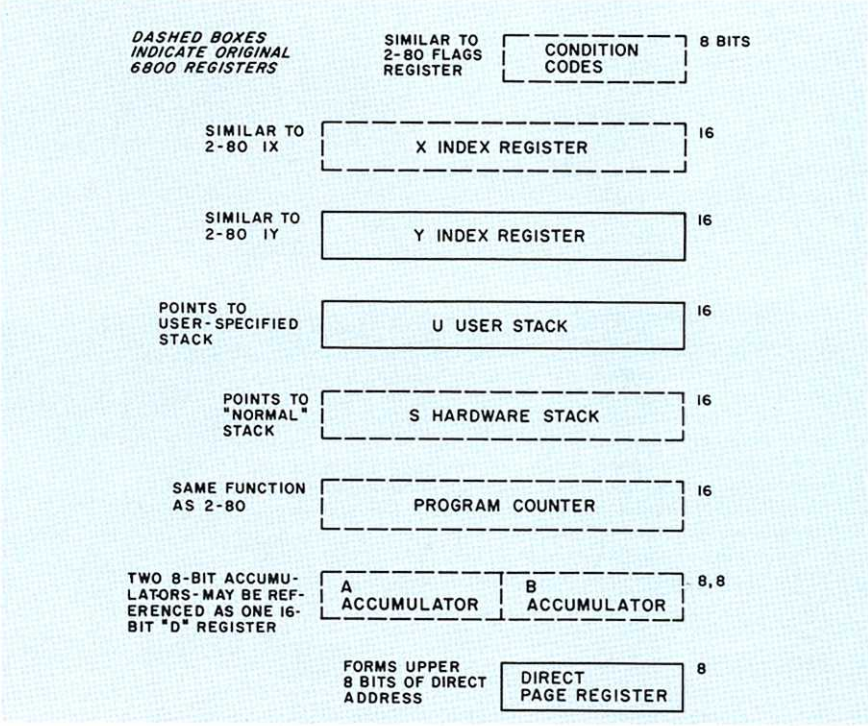


Fig. 7. 6809E Registers

This printer costs less than $450. Beat that... if you can.

Epson.

This is the Epson MX-70. The lowest priced dot matrix printer you can buy. Now, that in itself should make it very attractive to a lot of people. But you ain't heard the half of it.

To begin with, the MX-70 has a lot more in common with our now-famous MX-80 than just the name. Like unequalled Epson reliability. And technological breakthroughs like the world's first disposable print head. But frankly, the MX-80 packs a lot more power than some people need. So we built the MX-70 to be a no-frills printer. At a no-frills price.

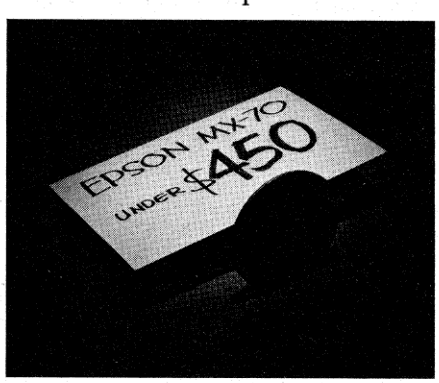
But the MX-70 is still a great little printer. We give you 80 CPS unidirectional printing. Top-of-form recognition. Programmable line feed and form lengths. Plain paper printing. An easy-to-read 5x7 matrix. Self test. And an adjustable tractor feed.

That's what you'd expect from a basic little printer. But here's something you wouldn't expect: the finest graphics package on the market today. Free.

We call it GRAFTRAX II. And it means 480 dots across the page, resolution to 60 dots per inch, and a graphic image free of the jitter and overlap that plagues other printers. You get cleaner grays and finer point resolution.

So now you've got a choice. You want more power and extra functions, you buy the MX-80. You want a basic little printer that prints, and keeps on printing, you buy the MX-70. They're both at your dealer now.

But at this price, you'd better hurry.



✓404

EPSON
EPSON AMERICA, INC.

23844 Hawthorne Boulevard • Torrance, California 90505 • (213) 378-2220

page register is initially set to 0, to address page 0 of the memory (locations 0 through $3FF; the $ is a hex prefix instead of the familiar Z-80 H suffix). It can be changed at any time to point to any 256-byte area, as, for example, a common working storage area.

Two additional condition code bits are related to the number of registers pushed on the stack (E) and a fast interrupt request (F).

The instruction set of the 6809 retains most of the 6800 instructions. The format of some have been redefined on the basis of frequency of use. There are many new instructions related to 16-bit operations. A hardware multiply instruction (8 × 8 bits) is implemented, but no divide.

The most significant changes to the 6800 base have occurred in addressing modes. There are nearly two dozen separate addressing modes. Modes similar to the Z-80 modes include such addressing as immediate addressing, register operations, direct addressing, and the like. There are a variety of indexing modes—offsets (displacements) of 5, 8 and 16 bits can be added not only to the X and Y index registers, but also to the two stack pointers! Another feature is the ability to auto-index. After an index register is used, it is incremented or decremented automatically by one or two to point to the next location.

As I mentioned in the previous column, the hooks are in Microsoft BASIC for assembly language on the Color Computer. I've been digging around in a disassembly of the 8K BASIC and found some very interesting things that would satisfy any assembly language freak. The cassette output of 1200 and 2400 hertz, for example, is synthesized by using the built-in 6-bit digital-to-analog converter. A table of sine wave values is converted to the sine waveform. All RS-232C operations are done in the firmware; there doesn't appear to be any serial input in the 8K BASIC, but this will be in the extended BASIC version.

As of yet Radio Shack doesn't have an assembler, but there are a number of monitors and assemblers available from other sources. Write me and I'll send you a complete list.

If I sound enthusiastic, I am. I'm always fascinated by new toys, especially sophisticated, inexpensive ones. Look into the Color Computer!

Trakcess

As I've stated before in this column, I can't be bought (easily). From time to time, however, I see new products that are interesting. One of these is The Alternate Source's Trakcess. I'm mentioning it here because it is related to a previous column on disk operations. If you would like to continue your education on disk operations down to the disk controller level, I recommend Trakcess.

Trakcess is an assembly language program, written by Roxton Baker, that allows you to perform disk operations such as stepping the head, reading and writing sectors, locating usable sectors, reading and writing tracks (the entire track, not just the data portion!), and building a format track. Using this program, you can build a disk to your own specifications (one sector of 3024 bytes per track, for example). When used with your own assembly language routines, you can design your own operating system (given 10 or 11 man years). Although I have mixed feelings about this controversial subject, you can also use Trakcess to investigate any disk.

Next month we'll have more assembly language topics. If you have strong ideas, or any ideas about what you'd like to see here, write me at: 28182 Palmada, Mission Viejo, CA 92692. ■

80 ACCOUNTANT

by Michael Tannenbaum C.P.A.

Many times in past months, I have been tempted to devote a column to word processing.

When Michael Shrayer introduced the Electric Pencil, the program was light years ahead of the unsophisticated word processors of the time. It was fast, it was convenient and it was easy to use. Since I rarely write long reports, I never ran out of memory.

Because Pencil had been designed for use on a faster processor, it suffered from typing wrap-around. More than once I lost letters as I typed. In general, the good features far outweighed the bad and the Pencil was, and still is, a joy to use.

Scripsit

About this time, Radio Shack introduced Scripsit for the Model I. Scripsit eliminated the wrap-around problems and provided for far more formatting latitude than the Pencil. It also introduced me to the concept of horizontal scrolling.

Horizontal scrolling is rarely required for preparing text. However, if you must prepare any type of schedule where the line length will exceed 64 characters, horizontal scrolling allows you to move the screen left and right over the text. Thus, you can prepare, edit and review a schedule with data lines that can be 132 characters long.

Electric Pencil II for the Model II represents a significant improvement over the Model I release. Although it is still a "string oriented" word processor, you can embed control commands in the text. These control commands dynamically adjust margins, spacing and any other parameter. Thus, Electric Pencil II eliminates most of the problems of dealing with outline reports.

An additional sub menu has been added to facilitate disk commands. From this menu disks are initialized, files loaded into the program buffer and the buffer cleared. It also enables you to view the entire directory by executing a DIR command from the sub menu.

Pencil can also be used to edit a BASIC program, if the program is saved in ASCII format and spaces are available every 30 characters or so. A specially modified version of the Pencil is supplied for this purpose.

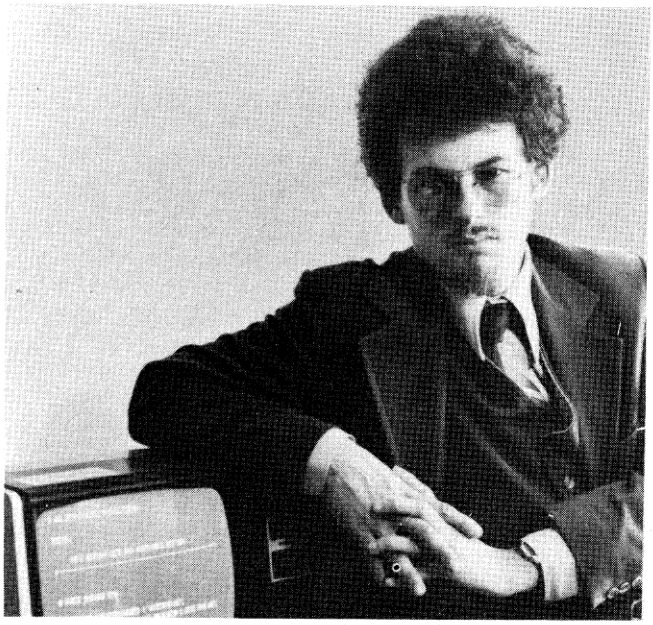
Although the Pencil is suitable for most text preparation, if long files are required, you may run out of memory. Authors of books or long reports should sub-divide the material to ensure adequate memory for each section. If a document requires footnotes, you should force pagination, so that the footnotes will appear at the bottom of a page.

With all of its features, the Pencil II is still basically a text processor; horizontal format schedules cannot be easily prepared, a severe limitation for an accountant.

Scripsit II

When Model II Scripsit became available, it represented as great an improvement over Scripsit I as the Pencil II. The folks at Radio Shack really have a winner. It's SUPER!

Unlike Pencil, Scripsit II is a "page oriented" word processor. This means that text is entered on a page by page basis. Pagination is not automatic and when a page is full, the program prompts



✓ Reader Service—see page 290

you to get a blank page in order to continue. The major advantage to this method of entering text is its similarity to a normal typewriter. If you prepare text that requires a header or footnotes, a page oriented system is the easiest to use.

Although the system is delivered with a cassette augmented instruction course, it is far easier to use than its Model I predecessor. Included with its operating commands is a "help" file. By keying in "ESC" and the letter H, a listing of the control sequences for all commands is displayed.

This advanced feature is only one of many that are included in Scripsit II. One of the most appreciated is the reverse printing option on the video display to highlight words, sentences or paragraphs selected for deletion or moving. Highlighting text in this manner graphically defines the area affected *before* it is irrevocably changed.

Another excellent feature is the command line, displayed on the last line of the video screen. This line constantly displays the name of the text you are working on, the page number, cursor position, line spacing and margin settings.

Because this Scripsit version was written for the Model II, it uses the special function keys F1 and F2. Pressing the F1 key opens up the text at the cursor location to allow the insertion of letters or words. Pressing F2 closes up the text. Pressing F2 alone serves as a character delete key.

Scripsit II is ideally suited to those of you with a Daisy Wheel II printer. With this Radio Shack printer you can overstrike, reverse your line feeds, use superscripts and subscripts. This combination of Scripsit II and the Daisy Wheel printer contains features which rival the best commercial word processing systems.

However, Radio Shack did not stop there. They've made it possible to integrate a file prepared by the Profile II program with a form letter prepared by Scripsit. Key items can be extracted from the Profile file and inserted into the Scripsit file as required. This very powerful feature is of value to any organization that requires regular correspondence with its members and must communicate variable data such as amounts due or events scheduled.

While Scripsit and Profile comprise an impressive combination of systems that can be used on a Model II for word and file handling, there is still a missing link. Neither of the two systems handles arithmetic tables well.

Accountants prepare many reports for clients that are a combination of mixed tabular and non-tabular text. Tables of figures are generally required. While Scripsit can be used in its "horizontal" mode to format the data for printing, it provides no means of proving the arithmetical accuracy of the data entered. This is unfortunate because it means that documents with numerical data prepared by Scripsit will still require proofreading.

Lifeboat's T/Maker

What is available, however, is an interesting system called T/Maker, Lifeboat Associates, Inc., New York, N.Y. T/Maker is a CP/M and CBASIC system which includes both a word processor and a very competent arithmetic package. Because CP/M and CBASIC are available for all models of the 80, T/Maker is usable on any Radio Shack machine, however, Models I and III must have at least 48K and the Model II at least 64K to use the system.

As with all types of non-TRSDOS systems, a preliminary setup procedure is required before the system can be used. While this is not complicated, unless you are familiar with the screen formatting conventions of your computer and CP/M utilities, it is best that this be handled by your software dealer.

Included with the software is a demonstration program. The demonstration covers text entry, formatting, table creation and arithmetic calculations.

T/Maker's ability to chain commands in order to operate on a body of mixed text and tabular data is quite effective. Once text and tabular data are entered into the program buffer, the JCL (job command language) protocol can be used to align text, perform calculations and format the table for printing.

To enter data into the buffer, a file must be created. The program checks to ensure that the name has not been used before and, if acceptable, enters the data on a line by line basis. Lines can be longer than 80 characters, however, the screen must be adjusted to accommodate the longer line before the text is entered. By entering ESC, then control L, the screen image is shifted 30 columns to the right.

In many respects, data entry is similar to Scripsit in its horizontal mode. The computer screen serves as a window that can be positioned over the text as required. Although the calling keystroke sequence of T/Maker is different, it has most of the editing features of both the Pencil and Scripsit. Characters and lines can be inserted and deleted, and the cursor can be moved all over the text.

Because the editor of T/Maker is "line oriented," block moves or deletions are not as convenient as with Scripsit or the Pencil. A block of data can be extracted from the text and moved from one spot to another, but it must be extracted on a line by line basis. Each line extracted is placed in a holding buffer. When all lines to be moved have been entered in the buffer, the cursor is moved to the desired spot; the buffer is then unloaded, completing the transfer operation.

Compared to Scripsit and the Pencil, T/Maker is a cumbersome word processor. However, T/Maker was not intended for this purpose. It was designed to prepare tables and in this task it shines.

T/Maker creates a table using the editor to enter the heading, column captions, row labels, numeric formatting requirements and computational parameters. The range of calculations that can be performed on the data and displayed in the table are enormous. Not only footing and crossfooting are available, but calculations of averages, growth rates and percentage replacements.

Despite 28 pages of examples and demonstrations, you will probably find the computational section of the manual heavy going. However, in time and with practice, all of the program's enhancements should become usable.

After you have specified the arithmetic computations and the data is entered, the command COMPUTE fills in the blanks in the table. Since T/Maker is not a machine language program, computation takes longer than with a program like Visicalc, however the results are the same.

To prepare the table for printing, remove the calculation parameters with the command CLEAN. These parameters are not lost and can be recalled whenever another calculation is required.

T/Maker also contains a sort program. This unusual feature allows the T/Maker user to order tabular data for maximum impact, either before or after a computation has been completed. To the best of my knowledge this powerful command is unique to the T/Maker.

T/Maker is not limited to a text file in the buffer. Disk data files can be added to or concatenated with a buffer file. If desired, arithmetic data on disk can be included in computations and the results unloaded to the disk. Used in this manner the buffer can serve as a mask to process disk file data. With this technique, you can generate a large number of tables if required.

Like Visicalc, which it resembles in many ways, T/Maker is an extraordinary product. If your work requires the preparation of management reports which include tabular data, I urge you to see a demonstration of this system. While you are there, examine the manual. I am sure that you will find it worth your while.■

The book you've been waiting for...

Ever since Radio Shack sold the first TRS-80 Model I users have been searching for detailed information about its inner workings that Tandy would not, or could not, make available. In particular the Level II BASIC from Microsoft contains dozens of subroutines that can be tremendously useful to any programmer, but Tandy Corporation is probably under contractual obligation to Microsoft not to supply information (if they even have it!).

Dedicated users, proficient in assembly language, have disassembled the Level II ROMs and made their own comments. But the majority of users are left with virtually no information, apart from occasional articles and whatever they can decipher on their own.

ENTERPRISING USERS - Several of the more enterprising programmers realized that if they published their own comments a lot of TRS-80 users would buy them. The BOOK, Disassembled Handbook and Supermap are some of the available books giving comments on the ROM set - but they all suffer from serious drawbacks, being either incomplete, unintelligible or even worse inaccurate!

Incomplete books are usually published when the author has not finished understanding what he's writing about. Hence the "continued next book" lines in some publications, translated into english read "buy another book when I've done some more work". Unintelligible books are due to poor editing, or no editing at all! Inaccurate information is a result of not checking with anyone else.

Microsoft BASIC Decoded & Other Mysteries is both complete and understandable. Nearly 7,000 lines of comments for the Level II ROMs, with an additional 6 chapters of useful information, make this the biggest and best book available on the subject.

Written by James Farvour, the comment section took more than a year to finish - it even includes the changes for the latest ROM set in an appendix. Edited by Jim Perry, until recently managing editor of 80 Microcomputing, the text and comments are understandable.

Tested examples are given for virtually every ROM subroutine, showing you how to CALL them from BASIC or use them in an assembly language program. With more than 300 pages Microsoft BASIC Decoded & Other Mysteries is by far the largest book about Level II available.

Copyright - In order to respect Microsoft copyright the actual disassembled code is not printed, but the book is designed to come apart and fit into a standard 3 ring binder with your own disassembly (all pages are pre-drilled).

In short, Microsoft BASIC Decoded & Other Mysteries, is the most complete, understandable and accurate guide to your Level II ROMs that is available bar none!

Complete & Understandable - IJG, publishers of TRS-80 Disk & Other Mysteries, could have published an incomplete or unintelligible book on the ROMs - but chose to wait and do it properly.

Pick one up at your nearest IJG dealer, phone your order in or use the coupon -

✓37

IJG Computer Services 569 N. Mountain Avenue Suite B Upland, CA 91786

Phone Orders (714) 946-5805

Please send me a copy of Microsoft BASIC Decoded & Other Mysteries for $29.95 + $2.00 shipping (CA residents add 6% tax).

Name_____

Address_____

City_____

State_____ Zip_____

Charge my

☐ MasterCard

☐ Visa

Interbank # _____

Expiration Date _____

_____

☐ Check enclosed

☐ Ship COD ($2.00 extra)

Signature_____

* T.M. Microsoft T.M. Tandy Corp.

80 APPLICATIONS

by Dennis Kitsz

". . . the computer is executing machine language already— it only looks *like BASIC."*

The past few months I've been running wild with all sorts of projects and some pretty rich software. Now it's time to lean back, clear the mind, and take another look at how machine lanaguage does its work.

Maybe you are already saying, "Sure, I've been through the binary thing before. First, they tell me about ones and zeros, and the next thing I know, they're deep into the arcane recesses of machine language! It just doesn't connect."

Be patient, and I'll try to forge that connection.

When you power up your computer, BASIC is already "in there", and ready to go. How did it get "in there"? ("In there" is the useful phrase. Keep it in mind whenever we talk about programming the computer.) BASIC is "in there" because the Z-80 is a master of electronic illusion.

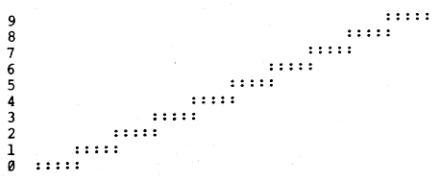
What's a Z-80? This is the microprocessor inside your TRS-80 that runs the show—the "brain" of the computer. It doesn't need the TRS-80's outside shell, but the TRS-80 needs it. How so? Well, apply the proper power to a Z-80 microprocessor and it will charge ahead and operate. But without something to structure what the Z-80 does, it merely produces heat. Not a lot of heat, but that's the only useful work it can do by itself.

The first device attached to it is a "clock". Clock isn't a great word because it makes one think of things with dials and alarms, but it's a lot less fearsome than talking about an "oscillator", which is what this clock really is—a string of pulses, first on, then off.

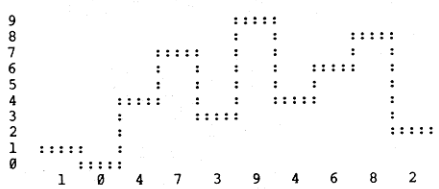
Hook up the clock to the Z-80, and it begins its real work, fetching commands and executing them. (What commands is it fetching? What form are they in? Where do they come from? Those questions are at the heart of explaining a computer, yet there's still some more exploration of the Z-80 to be done.)

The Z-80 is an integrated circuit the size of your thumb, with 40 external connections (pins). That number is arbitrary, just as almost everything to do with computers is arbitrary. Somewhere somebody standardized these things, probably because manufacturing precision is limited. No, I'm not getting off the track. This is central to why all the binary bugaboo came about.

From the viewpoint of ten-fingered humans, it would be easier to do computing in decimal form. Ideally, since electricity is the most likely force to get things done, ten different levels of voltage could be used, thus:



That way, the number 1047394682 could be represented very neatly like this:



Indeed, there are electronic parts manufactured which are capable of distinguishing ten such levels of precision. But they're not easy to make, and hence they're expensive. Getting them to be reliable is a problem, and diagnosing their flaws is virtually impossible.

So it came to pass that someone, somewhere in the dark ages of computer technology, decided that only two voltage levels were reliable enough to use—zero represented by an off (low) condition, and one represented by the on (high) condition. Yup, there's the stuff you've heard before; the ones-'n'-zeros story. But think of it this way: that same someone *might* have decided that a negative voltage was zero, ground potential was one, and a positive voltage was two. You and I would then be struggling with base three instead of base two!

The Z-80 needs power and clock pulses, but the big gambit is having it do wizard work for us, which means putting information into it and getting something back. The information is called data. The smallest value for simple data is zero, and the highest value is . . . well, what is it?

The number of pins assigned to data determine its highest value. Your Z-80 has eight pins set aside to send and receive data. Other microprocessors have sixteen pins, some four pins, and there's even a processor with only one data pin. Making a choice among processors is based on the amount of work to be done and the speed with which it must be done. The bulk of small computer work seems to get along comfortably with the eight-bit data bus, as those pins are known.

The largest number, therefore, is eight ones (1111 1111), or a decimal value of 255. (Aside: How does 255 relate to 1111 1111? Can you see an inherent relationship? How does 180 relate to 1011 0100? Any easy way to remember that? Fast-food for thought: The symbol FF relates to 1111 1111. And B4 relates to 1011 0100. To be continued.)

Numbers up to and including 255 can be sent directly along the Z-80's data bus. Bigger numbers have to be sent in eight-bit chunks. If you're with me so far, then you're about to ask, "How do these data get to the Z-80? How does it use them?" (If you weren't about to ask those questions, do it now, because those are the ones I'm going to answer.)

Simply stated, a processor does its work *in order.* That's a simple statement until you ask the questions, "How does it know the order? Where does it start?" The answers are again arbitrary, but based on utility. It could just start, and we would have to use switches to feed information one piece at a time to the data bus. But that's very inefficient; we might as well do it by hand, or with an abacus, and skip the electronics.

Instead, sixteen of the Z-80's pins are used to specify a number from 0000 0000 0000 0000 to 1111 1111 1111 1111. It can use these numbers to move from command to command, from place to place, starting (when the power is turned on) with

all address lines reading 0000 0000 0000 0000.

It then fetches its first command, executes it, and moves its program counter to the next available space and executes that instruction. And then the next, and the next, in millionths of a second! Ah, a computer! So fast!

Slow down... where did it get the instructions? Embedded in the answer to that is a biggie: once the Z-80's got power and its clock, it needs—you guessed it—memory! The guts of your TRS-80 are wired so that whenever those 16 address lines form a number (from 0 to 65535 in decimal) only one group of eight memory cells responds. Each memory cell represents one bit of information, and eight in tandem provide a single *byte* of data to the microprocessor.

You turn on the TRS-80. The Z-80's counter sets itself to 0000 0000 0000 0000, and sends out a "read the memory" signal. The eight bits of memory that respond only to 0000 0000 0000 0000 make their collective information available to the microprocessor, and the Z-80 follows that instruction. It moves its counter ahead, and fetches and performs the next instruction. And the next. After many hundreds of instructions, and in a mere fraction of a second, the screen reads MEMORY SIZE?

Now you fear the worst; I can sense it. You're saying that he's going to jump into machine language, and leave you behind byting the dust. Well, you're right on one count: This is the jump into machine language. But you know what? You've already made it. You're there. It happened in the previous paragraph.

Okay, your computer's on, but you're operating in BASIC. How do you get to machine language? Well, the computer is executing machine language already—it only *looks* like BASIC. You can wrest the Z-80 away from BASIC: First, you must know how to get to the memory that the Z-80 is going to use, and second, how to get the Z-80 to that memory. Clear the screen and enter this statement:

```
POKE 16000,66
```

Somewhere on the screen a letter B appeared. You didn't really print a B on the screen, but rather placed the value 66 into memory spot 16000. That's what POKE does. It just so happens that the TRS-80 circuits print on screen whatever they find at any memory locations from 15360 to 16383.

Give me another minute before you shrug. Enter:

```
POKE 16000,234
```

Now you've got a graphics character. You placed a different parameter into memory at 16000 in place of the B. Now you'll see what the Z-80 finds when you turn on the power, at memory location 0000 0000 0000 0000:

```
CLS                    <ENTER>
X = PEEK(0)            <ENTER>
POKE 16000,X           <ENTER>
```

The same graphics block, you say? True enough, and you will find the value (PRINT X) to be 243. Two hundred forty three is the first command discovered by the Z-80 when you turn on the power. What this instruction does is unimportant for the moment; look instead at some more of that area of memory:

```
10  CLS                    :REM NICE AND CLEAN
20  X = 15360              :REM FIRST POS'N ON
                            SCREEN
30  Y = 0                  :REM FIRST BYTE Z-80
                            SEES
40  FOR N = 0 TO 1023      :REM TOTAL SCREEN
                            POSITIONS
50  POKE X,PEEK(Y)         :REM MOVE FROM Y TO
                            SCREEN
60  X = X + 1              :REM NEXT SCREEN POSI-
                            TION
70  Y = Y + 1              :REM NEXT Y POSITION
80  NEXT N                 :REM NEXT IN 1024 LOOP
90  GOTO 90                :REM DON'T PRINT
                            "READY"
```

A screen full of garbage grows. But the garbage you see actually is a visual representation of the first 1K (1,024 bytes) of commands that make up the Level II BASIC language built into your computer. What you have really done, then, is to take 1K of the information found at 0 to 1023 in memory, and duplicate it at 15360 to 16383 in memory.

In other words, you have directly affected the memory contents of your TRS-80. Which means you are just about ready to put a machine language program into memory.

Specific machine language instructions and what they do can be found in the manual accompanying Radio Shack's Editor/Assembler. The value for each instruction is given in binary and hexadecimal. Earlier in the column, I asked what possible relationship 1111 1111 could bear to 255, and suggested that there might be a more reasonable relationship between 1111 1111 and FF.

And there is, at least in a visual sense. Whereas it might be possible to learn that 1011 0100 binary is 180 decimal, it is unlikely that a quick glance will reveal the decimal value for 1011 0100 1011 0100 (it is 46260). Working with binary numbers alone is possible (after all, the machine does only that), but it becomes unwieldy for humans.

"Now it's time to lean back, clear the mind, and take another look at how machine language does its work."

Each block of four binary digits may be represented by a single character. Up to the number nine (binary 1001), it is easy, and identical to our decimal numbering. After nine, we run out of single characters to use. Back in the early days, there was some indecision as to what to do, and six extra characters were invented (anybody remember dek for 9 + 1 and el for 9 + 2?) These obscure characters were soon dropped in favor of the familiar letters A through F: Decimal 10 becomes A (1010), 11 becomes B (1011), 12 is C (1100), 13 is D (1101), 14 is E (1110), and 15 is F (1111).

In this way the number 1011 0100 1011 0100 becomes very easy in hex . . . B4B4. And don't forget that data can only move to the Z-80 in eight-bit chunks. If you want to tell the Z-80 to "go to address 1011 0100 1011 0100", how do you do it? In two eight-bit binary chunks, of course, but how do you remember that value quickly when you are programming? 46260 is a single piece, without its component decimal parts of 180 and 180 making themselves particularly obvious. On the other hand, B4B4 snaps into two, neat, byte-sized pieces—B4 and B4.

Like the young violinist who asked a passerby how to get to Carnegie Hall, the answer to learning hexadecimal is just the same—practice.

It's now time to create a simple machine language program, and insert it into memory using BASIC's POKE statements. Here is the program:

```
3E   42   32   00   3E   C3   CC   06
```

Here is where the program might be placed in memory:

```
5000 5001 5002 5003 5004 5005 5006 5007
```

And here is what the program does:

- Places 42 (hex) into Z-80 accumulator register. This can be noted "LD A,42", and spoken "load A with 42". In BASIC it might be thought of as "LET A = 66".
- Moves the character from the accumulator into memory at 3E00. This may be written "LD (3E00),A", meaning "load the memory contents at 3E00 with the value of

A". Again, a rough BASIC equivalent of this action would be "PRINT@512, CHR$(A)".

● Returns to BASIC command mode with a READY message. This instruction can be written "JP 06CC", and said, "jump to memory location 06CC". This is like a BASIC "GOTO".

Had the Radio Shack designers put a hexadecimal number option into Level II BASIC, it would be a lot simpler to write a machine language program. You might have been tinkering with it on occasion. But, easy though it is to work with hexadecimal, you're forced to rethink it all in decimal when working from Level II BASIC. Here is the above program with the hex instruction converted to decimal:

```
62   66   50   0   62   195   204   6
```

And here is the decimal representation and where the instructions go:

```
20480 20481 20482 20483 20484 20485 20486 20487
```

To put this program in memory:

```
POKE 20480,62 : POKE 20481,66 : POKE 20482,50 :
POKE 20483,0 : POKE 20484,62 : POKE 20485,195:
POKE 20486,204: POKE 20487,6
```

An eight-byte program to print the letter B is now "in there". It starts at 20480 decimal. Want to try it? You can send the Z-80 there from BASIC by typing:

```
SYSTEM     <ENTER>
/20480     <ENTER>
```

The READY comes right back. But notice there's a letter B at the left of the screen where there was none before. Done. The Z-80 processor was directed to start executing the program at 5000 hex by our SYSTEM command. And it returned to BASIC command level. That's machine language.

In response to numerous requests, then, here are the steps for converting many machine language listings to BASIC POKE statements:

1. Check to see that the program is right for your computer's memory configuration—4K through 48K RAM. If the author has given only hex addresses, convert a hex address to decimal this way:

```
W*4096 + X*256 + Y*16 + Z
```

Where less than four digits of an address are given, the leftmost are assumed to be zeroes. Remember that A = 10, B = 11, C = 12, D = 13, E = 14, and F = 15.

The topmost address for the four RAM possibilities:

```
4K   20479
16K  32767
32K  49151
48K  65535
```

2. If it will fit in your machine, count the number of data bytes present. If the author has given a hex dump, it is easy. An eight-byte hex dump looks like this:

```
3E  42  32  00  3E  C3  CC  06  ...
```

If the author has given an assembly listing, check to see that it is assembled. An assembled listing will contain six or seven columns, and look like this:

```
Addr  Data      Line#   Label   OpCode   Operand
----  ----      -----   -----   ------   --------
5000  3E42      00100   START   LD       A,42H
5002  32003E    00110           LD       (3E00H),A
5005  C3CC06    00120           JP       06CCH
```

The second column contains the data, and you should break it up into pairs of characters (C3CC06 becomes C3 CC 06) before counting. If the author has not presented an assembled listing, you're out of luck without the Editor/Assembler handbook and a lot of hard work. Most listings in *80* are assembled as a convenience for BASIC and T-BUG users.

3. Once you've counted the data bytes, determine the first memory address in the program. It is identified by a four-character number to the left of the first line in a hex dump. On an assembly listing, it is represented by the label ORG (origin).

This information should be converted to decimal using the formula in Step 1.

4. Create a FOR-NEXT loop using that start address and one less than the number of data bytes, this way:

```
10 FOR X = 20480 TO 20480 + 7
```

This represents the area of memory that a given machine language program might use.

5. Convert all the data to decimal using the formula:

```
Y*16 + Z
```

Remember the values for A, B, C, D, E, and F given above. For example, the value 3E is 3*16 + E . . . 3*16 + 14 . . . 48 + 14 . . . that becomes 62 decimal.

6. Take the converted data and place it in BASIC program lines, beginning with DATA and separated by commas:

```
100 DATA 62,66,50,0,62,195,204,6
```

7. Next, create a line that will read the data and POKE it into memory, as follows:

```
20 READ A : POKE X,A : NEXT X
```

(Early TRS-80's have a bug. Enter POKE 16553, 255 before running any program which reads data.)

8. Find the program's starting address. The author will tell you this in the article, or you can discover it by looking at the END statement in an assembly listing.

```
5000  END  START
```

Convert that address into *two hex* pieces and then convert these each to decimal. For example, to convert 5000 to split decimal, break it into 50 00, and convert each of those to decimal using the formula in Step 5. The result of conversion is 80 and 0.

9. POKE those values into memory locations 16527 and 16526, this way:

```
30 POKE 16527,80 : POKE 16526,0
```

"Had the Radio Shack designers put a hexadecimal number option into Level II BASIC, it would be a lot simpler to write a machine language program."

10. Finally, send the program to the desired routine by using this command:

```
40 M = USR(0) : REM – M MAY BE ANY VARIABLE
```

11. The completed program will look something like this:

```
10  FOR X = 20480 TO 20480 + 7
20  READ A: POKE X,A : NEXT X
30  POKE 16527,80 : POKE 16526,0
40  M = USR(0)
50  DATA 62,66,50,0,62,195,204,6
```

12. For programs with more than one ORG, write a separate FOR-NEXT loop and READ-DATA line for each section. There will be only one entry point to place in 16527 and 16526.

The above steps will take time, but with them, virtually any machine language program that will run in your TRS-80 can be converted to BASIC statements. In fact, you can even write a BASIC program to do the formula conversion work for you. ■

Next Month: Interview with Dr. Lirpa

HARDWARE BREAKTHROUGH

DESIGN SOLUTION INC. presents the AN-SERIES DEVELOPMENT SYSTEM

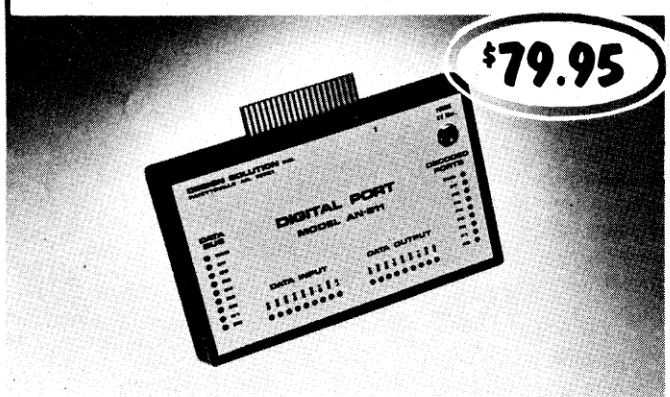
DIGITAL SPEECH PROCESSOR

Experiment with digital speech synthesis and recognition. using a **BASIC** editor program provided with this system. Speech is entered, digitized and stored in memory or in disk files. Vocabulary files are construced using digital core images of your voice. Labeled and indexed sounds, words and phrases are then available for use in your **BASIC** programs. Different vocabulary files can be loaded for different **BASIC** Programs, etc. while the digital speech processor will operate on any model I, Level II **TRS-80*** system. We recommend a level II with disk drive to reduce voice image loading times. Experiment with data compression from **BASIC**, by increasing and decreasing sample rates. Develop your own voice recognition programs. Simulate low pass, band pass, and high pass filters, etc. This is truly a software man's dream-come-true. Enter a string of numeric data and let your **TRS-80** repeat it. Let your **TRS-80** dictate data files for error checking. The sky and your imagination are the absolute limit!

2716-2732 EPROM PROGRAMMER

At last! An affordable **EPROM** programmer for your **TRS-80***. Program either 2716 or 2732 **EPROMS** from a **BASIC** program in less than 300 seconds. With **DESIGN SOLUTION'S SOFTWARE** you are able to program **(BURN IN) EPROM** from **TRS-80** memory, **VERIFY** data transfer, copy **ROM** to **EPROM**, or load **TRS-80** memory from **ROM** or **EPROM**. **THE AN-551** is a must for all small system development packages. Unit comes complete with external power supply and operation manual. The 40-pin edge card connector is **GOLD** plated for extra long life. **ZERO INSERTION FORCE SOCKETS** simply plug into the front panel socket of the **AN-551** if quick changes are required. (ZIF socket not included.)

DIGITAL PORT INTERFACE

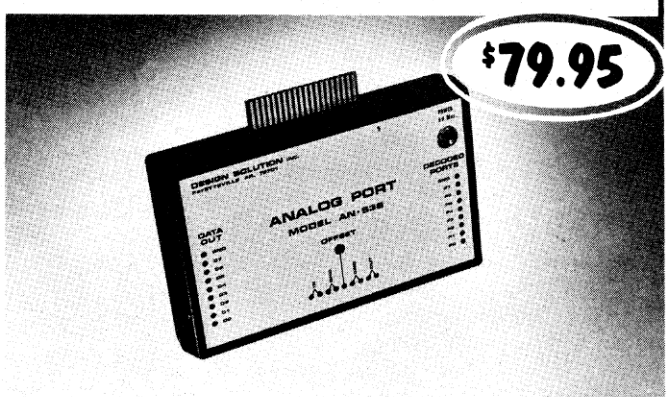


Now you can **BREADBOARD** your digital projects with ease. The **AN-511 DIGITAL PORT** provides all the hardware necessary to interface your digital projects with the **TRS-80***. From **MACHINE LANGUAGE** or **BASIC** programs you can now access and control:

8 BITS OF OUTPUT DATA FROM TRS-80
8 BITS OF INPUT DATA TO THE TRS-80
8 DECODED PORT ADDRESS (DEVICE CONTROL) LINES
8 BIT DATA BUS FROM Z-80 IN TRS-80

Using the **IN** and **OUT** commands in **BASIC** or their machine language counter-parts, 8 bit data values can be moved from external hardware projects into the **TRS-80** and vice-versa. With the 8 decoded port outputs up to 8 additional bytes of input or output data can be accessed. The **AN-511** is supplied with external power supply and complete operation and projects manual.

ANALOG PORT INTERFACE



The **SUPER BOX!** The **ANALOG PORT, MODEL AN-538,** is a very versatile **ANALOG TO DIGITAL AND DIGITAL TO ANALOG CONVERTER.** All conversion operations are under **TRS-80*** program control. Experiment with **MUSIC** and **VOICE SYNTHESIS.** From machine language or **BASIC** programs you can manipulate analog signals, digitize voltages and measure them with the **TRS-80.** Generate various waveforms, noise, speech patterns, etc., and output them as analog signals. The **AN-538** also provides a mini-digital port. The 8 bit output data to the D/A converter is accessable through the front panel connector, along with 8 extra decoded port (device control) lines. All conections are made using standard #22 solid wire. Lines push through the front panel, received underneath by special connectors, ensuring positive connection. The **AN-538** comes complete with external power supply and operation-project manual.

All connections made to **AN-SERIES** products from your breadboard are simply pushed through the front panel. Custom connectors on the P.C.B. provide super reliable connection for thousands of operations. All **AN-SERIES** products are warrented for a full **90-DAYS** under **DSI's** limited warrenty policy. Complete documentation is provided for each model in an attractive folder, including theory of operation, and special interest projects and applications. All units are supplied with external power supply modules that connect through a mini-jack on the front panel. A 40-conductor cable is required for connection with your **TRS-80** and is **NOT SUPPLIED** by **DESIGN SOLUTION**, but is available from your local **RADIO SHACK STORE.** (Components are 1(278-771) 40 conductor ribbon cable and 2(276-1558) edge card connectors.)

***TRS-80** is a registered trademark of **RADIO SHACK, a TANDY COMPANY.**

DESIGN SOLUTION INC. ✓123
BOX 1225, FAYETTEVILLE, AR 72701

C.O.D. add $3.00 — Other add $1.50 S/H.
Orders and Information: (501) 521-0281

✓Reader Service—see page 290

THE ALTERNATE SOURCE

TASMON

Written By Bruce Hansen

- Replace Registers
- Modify Memory
- Hex Memory Dump
- ASCII Memory Dump
- Disassembled Dump
- Disassemble to Printer
- Dump Screen to Printer
- Sum Hex Values
- Subtract Hex Values
- Find 1-4 Consecutive Bytes
- Skip Forward One Instruction
- Back Up One Instruction
- Clear Screen
- Relocate System Programs
- Move a Block of Memory
- Load a System Tape
- Load a /CMD Disk File
- Write a System Tape
- Write a /CMD Disk File
- Disassemble to Disk
- Disassemble to Tape
- ROM or RAM Breakpoints (9)
- Set Breakpoints
- Display Breakpoints
- Clear Breakpoints
- Single Step with CALL in full
- Single Step through CALLs
- Trace at Eight Speeds
- GO Execute Program

Disassembles with Labels!

TASMON is fully relocatable, file oriented, and excellently implemented. Comes with an extensive user manual complete with sample dialogues and command reference card. For use on Level II, Disk System or Model III. 32K recommended.

$29.95

(Specify disk or tape, please)

TRAKCESS

Written by Roxton Baker

- Analyze any TRS-80 disk
- Build your own disk formats
- Direct control of 1-4 drives
- Toggle between drives
- Pencil-like editing features
- Edit in hex or ASCII
- Handles sectors of any length
- Learn about disk structure
- Read and Write Sectors
- Take and Put Tracks
- Scan Track Sectors
- Locate Disk Sectors
- Copy Track
- Duplicate Disk
- Build Format Track
- Edit or Fill Memory
- Figure CRC's
- Hex or ASCII dump to Printer
- Will copy to a blank diskette
- Default values for most input
- Scrolls thru tracks & sectors
- No system DOS in drive zero
- Handles any number of tracks
- Breaks any 'protected' code
- Will copy any 'protected' disk

A Powerful Zap Utility!

Version 1.1 Now Available

- Highly Interactive
- Smoother Operation
- Full Error Checking
- Compatible with Newdos/80

TRAKCESS is approximately 25K of Basic and machine language code. It requires a Model I 48K Disk system; two drives are required for copy functions. Printer optional.

$24.95

KBE KeyBoard Editor

Written by Larry Kingsbury

PROGRAMMABLE KEYS

- Operable under DOS or Basic
- Operable under EDTASM
- Program 191 different keys
- Allows 'nested' softkeys
- Load softkeys from disk
- Save softkeys to disk
- Pre-defined softkeys included
- Toggle softkeys on or off
- Redefine any key at ANY time
- Softkey length unlimited

FULL SCREEN EDITOR

- Operable under Basic or DOS
- Operable under EDTASM
- Edits any line up to 255 chars.
- Softkeys work in Edit Mode
- Easy duplication of lines

DRIVER

- Keyboard debounce
- Auto repeat
- Lower case
- Typewriter keyboard
- Echo video display to printer
- E/P control key support
- Full graphics input
- Prevents printer hangup
- Sets lines per page

KBE is compatible with VTOS, NEWDOS, NEWDOS/80, and TRSDOS 2.3. KBE requires a 32K disk system for operation, and is supplied on diskette with user's manual. KBE is compatible with double density.

$39.95

Don't Forget a Subscription to TAS -- $12.00 for 6 issues!

1806 Ada Street
Lansing, Michigan 48910
517/487-3358
517/485-0344

Add $2.00 for crush-proof box and postage, $1.50 for COD orders, 4% for Master Charge or Visa orders.

( ) Send me TASMON ( ) 6 issues please!
( ) Send me TRAKCESS ( ) Send me KBE

Name: _____
Address: _____
City: _____
State, Zip: _____

EDUCATION 80

by Earl R. Savage

". . . all micro installations in schools are alike. All suffer from a shortage of money."

When the general public thinks of computers in the schools, it is almost certain to be in terms of record-keeping in the office and CAI (computer aided instruction) in the classroom. Administrative records and CAI may be the most important but there is another application of great significance to teachers. Let's look at what one teacher is doing.

Larry Arrington teaches secondary mathematics in Roanoke, VA. He comes as close to getting maximum use of his 80 as anyone. Larry not only uses CAI in his classes, but he uses an 80 to do "paper work" and to communicate with parents and students, saving time and effort for him!

Report Card is one of the programs Larry has written. Student grades are entered for classwork, homework, quizzes, and tests on a daily or weekly basis. Grades are automatically weighted according to classification. At both midpoint and end of each grading period, Report Card generates personalized letters.

Every student gets a letter which includes his grades (date, classification and score), the average grade, an explanation of the grading system, and an offer of special assistance if desired.

Every parent is sent an appropriately worded letter which contains the student grade information. Report Card also generates other reports which are sent to the guidance department and to the school administration. Special reports on individual students or groups can be printed at any time.

Another of Larry's programs is called Teacher's Absentee Report. It generates several different reports concerning student attendance. They are quite useful to the guidance department and administration as well as to Larry.

The state of Virginia administers an annual competency test in mathematics. Larry has devised his own version of a math competency test which he administers to his students before the state test. He puts the results into a special statistical program and turns the 80 loose.

Personalized reports of test results are generated for parents and students, and include information about the test, how to interpret the results, and the degree of strength and weakness in each of the 25 competency skill areas.

That is not the full extent of Larry's non-CAI use of the 80. It is applied in other ways for his classroom students and the participants in co-curricular activities which he sponsors.

It is apparent that the teacher, the students, their parents, the guidance counselors and the school administrators have more information than ever before about Larry's students' achievement and learning needs. It is axiomatic that the more we know about each student, the more effectively we can help each one learn.

A very significant outcome of Larry's 80 use is the great increase in valuable communications with parents. They are impressed and delighted, evidenced by the fact that on parent conference days, there isn't time to talk with all who come to see him.

Once set up, Larry's ongoing operation provides more with less effort than he expended in his old paper and pencil days.

If you would like to know more about what Larry is doing, he has agreed to correspond within the limits of available time. Please enclose an SASE (self-addressed, stamped envelope) when you write him—Larry W. Arrington, 2815 Edison Street, N. E., Roanoke, VA 24012.

Raising Funds

Who is responsible for the microcomputers in your school? What is the source of funds for programs, paper, cassettes, ribbons, accessories and so on? You say there are no funds? Perhaps you can do something about that!

I've found that 80s are in schools under every imaginable set of circumstances. They "belong" to departments (math, science, computer); guidance and administrative offices; obviously, educators in all kinds of specializations are responsible for computer operations.

In spite of this, there seems to be one way in which all micro installations in schools are alike. All suffer from a shortage of money. In fact, some teachers tell me that funds are non-existent.

I thought you might like to know that you are not alone and that you might have some help sitting there on the desk before you. Why not let your 80 earn its keep?

There are several books on how to make money with a micro. Be quite careful implementing money-making ideas—most schools have strict regulations about fund raising ventures. Most businessmen object vigorously to competition from an organization which their taxes help to support.

Consider selling a service within the school community itself. Surely there are possibilities in every school. Here are some ideas for your computer group to consider.

In the area of games and general use, an organization could make its computers available to other students. Before or after school, but more probably at Science Fairs and similar functions, students might be willing to pay a fee to play Star Trek, Adventure, Chess or whatever. This is no way to get rich quick, but it can help in the purchase of programs, supplies, and so on.

The area of data manipulation also provides some income opportunities. You can offer tabulation and statistical analyses of the results of surveys and standardized tests. Clubs, libraries, guidance and administrative offices are engaged in many types of record keeping. Weekly updating and analysis of athletic statistics is a wide-open field. Computer dances have been successful in many schools.

A functional word processing program added to a system which includes a printer opens another wide range of income producing possibilities. If your school prints in house, you can offer camera-ready originals with or without right justification. Schools and clubs print newsletters, bulletins, newspapers, magazines, and programs for athletics and other events.

Adequate funding seems to be a universal problem in schools. If you have found a way to make your computers contribute to their own support, let me know the what and how. Other teachers can profit from your experiences. ■

THE NATIONAL COMPUTER SHOWS

HAVE WE GOT A PROGRAM FOR YOU IN '81

Attend the biggest public computer shows in the country. Each show has 100,000 square feet of display space featuring over 50 Million Dollars worth of software and hardware for business, industry, government, education, home and personal use.

You'll see computers costing $150 to $250,000 including mini and micro computers, software, graphics, data and word processing equipment, telecommunications, office machines, electronic typewriters, peripheral equipment, supplies and computer services.

All the major names are there including; IBM, Wang, DEC, Xerox, Burroughs, Data General, Qantel, Nixdorf, NEC, Radio Shack, Heathkit, Apple, RCA, Vector Graphic, and Commodore Pet. Plus, computerized video games, robots, computer art, electronic gadgetry, and computer music to entertain, enthrall and educate kids, spouses and people who don't know a program from a memory disk.

Don't miss the Coming Of The New Computers – Show Up For The Show that mixes business with pleasure. Admission is $5 for adults and $2 for children under 12 when accompanied by an adult.

Ticket Information

Send $5 per person with the name of the show you will attend to National Computer Shows, 824 Boylston Street, Chestnut Hill, Mass. 02167 Tel. 617 739 2000.
Tickets can also be purchased at the show.

THE SOUTHWEST COMPUTER SHOW

DALLAS
Dallas Market Hall
2200 STEMMONS FRWY
AT INDUSTRIAL BLVD
APRIL 9-12
10 AM TO 7 PM
THURS-SUN

THE MID-WEST COMPUTER SHOW

CHICAGO
McCormick Place
SCHOESSLING HALL
23RD & THE LAKE
SEPTEMBER 10-13
10 AM TO 7 PM
THURS-SUN

THE MID-ATLANTIC COMPUTER SHOW

WASHINGTON, DC
DC Armory/Starplex
2001 E. CAPITOL ST. SE
(E CAP ST EXIT OFF I295
– KENILWORTH FRWY)
ACROSS FROM RFK STADIUM
SEPTEMBER 24-27
10 AM TO 7 PM
THURS-SUN

THE NORTHEAST COMPUTER SHOW

BOSTON
Hynes Auditorium
PRUDENTIAL CENTER
OCTOBER 15-18
10 AM TO 7 PM
THURS-SUN

THE SOUTHEAST COMPUTER SHOW

ATLANTA
Atlanta Civic Center
395 PIEDMONT AVE NE AT
RALPH McGILL BLVD
OCTOBER 29-NOVEMBER 1
10 AM TO 7 PM
THURS-SUN

80 NEWS

Edited by Nancy Robertson

"Cook's operating systems have come to be known both for their underlying brilliance and for user frustration."

Randy Cook and the Thorny Path that Began with TRSDOS 2.0

In the primeval dawn of microcomputers (way back in 1978), Radio Shack created the skeleton of the TRS-80 Model I in hardware. They contracted with Microsoft to give the creature BASIC language and life functions. To give it the option of super intelligence, they contracted with Randy Cook, a young minicomputer programmer, for a disk operating system.

Minimally, a disk operating system (DOS) coordinates I/O between the disks, ROM, interpreter and programs. When it is in use, a DOS controls and orders a micro's thought processes.

If there were a chronicle of the TRS-80, Randy Cook would surely be listed as the father of disk operating systems. Tracking the history of his DOS in the micro market is like going down a path overgrown with brambles and sour grapes. It began with Radio Shack and TRSDOS 2.0, the first DOS they released. After TRSDOS 2.1, Cook brought out a more sophisticated operating system, VTOS, under his own Virtual Technology label.

Then, around January of 1979, a separate manufacturer, Apparat, Denver, CO, produced NEWDOS. It was a patched and enhanced version of Cook's work for the Shack.

In November, 1980 Cook signed a contract with Lobo Drives International, Goleta, CA giving them "nonexclusive" but "total" rights to VTOS, according to company spokesmen. They will be selling LDOS, a reassembled DOS incorporating several parts of the source code of Cook's VTOS.

Brilliant, but Frustrating

Cook's operating systems have come to be known both for their underlying brilliance and for user frustration. TRSDOS 2.1 was riddled with bugs. In a recent letter to *80* Harv Pennington, the author of *TRS-80 Disk and Other Mysteries*, wrote that when TRSDOS 2.1 was the only available operating system, "we saw the message 'NO SYSTEM' almost as often as we saw 'DOS READY.' " Later, the write-protect feature, which made it difficult to make back-up copies of VTOS 3.0, frustrated users. And VTOS 4.0 still lacks a user's manual.

Pennington took a close look at VTOS in *TRS-80 Disk and Other Mysteries*. He has reflected since then, "I liked VTOS except for the protection features that made it so hard to use on an every day basis and the difficulty in recovering a disk once it had been eaten by the system. I did not recommend the use of VTOS 3.0 for those reasons."

"People who have marketed Cook's operating systems consistently point out his lack of support."

"That has changed. VTOS 4.0 has none of the shortcomings of 3.0. It is vastly improved and reliable. It is my opinion that this is a DOS for serious applications and software development. It retains all of the excellent design features of the original TRSDOS and its added features are astounding." Pennington, however, is fortunate to have the knowledge to wade through the DOS without a manual.

People who have marketed Cook's operating systems consistently point out his lack of support. On the other hand, Cook counts the times he's been ripped off by marketers. "If you're looking for a quote, I wish you would quote me. Everybody out there thinks, 'There's Randy Cook. How much did they pay him?' $3000—that's all I received from Radio Shack. I'd like the public to be aware of that. They think I'm a recluse, just sitting here raking in the money. It's not true."

At 32 Randy Cook is currently working for Mostek, Carrollton, TX in strategic marketing. He has been living in the Dallas area since leaving his home state of Oklahoma in 1968.

At home Cook has a mini and several micros. "Everything talks to each other through a central processor." By "everything" he means the oven, the thermostat, the doorbell, etc. "I like chains; I like to combine it—I'm a systems person," he says.

"As much as I like systems, I'm appalled by Big Brother. At times it scares me that I might be contributing to that. I hope not. It depends on how the systems are used."

Simultaneous and remote access truly fascinate him. "Take something like MicroNet," he says. "Right now people can hook up to the system and play games against the central computer. Well, imagine that some day people will hook up and play games with one another—that's simultaneous access."

Chronology

But business problems and user questions do not go away. Chronologically, the story begins in 1978 with Radio Shack and TRSDOS 2.0. Shugart contracted with Radio Shack to provide disk drives for their computers. Part of the deal included Shugart taking the tab for a disk operating system. Cook was working on a DOS for minis with Xerox when Shugart contacted him. He began working on a "minimal" DOS to meet Shugart's specifications, he explains.

According to Cook, he gave Radio Shack DOS 1.2 through 2.0 as preliminary versions. "TRSDOS 2.1 was the first ver-

sion I meant to go out of house," Cook says. But 2.0 reached the market with an announcement that it still had bugs, according to sources at Radio Shack. Several months later TRSDOS 2.1 reached Radio Shack outlets. Customer feedback pointed out that there were still problems.

Following 2.1, Cook claims he contracted directly with Radio Shack to provide "an exact number of features for 2.2 at a set price." In the meantime, he says, Microsoft was having problems with the size constraints of ROM. ROM modifications meant more DOS changes. In addition, Cook says he was asked to add more features. "Miscommunications" became a problem, according to Cook.

Van Chandler, the senior manager of software development at Radio Shack, remarks that "We agreed to pay him a flat fee, and everything went down hill from there." Whatever the cause, Radio Shack and Randy Cook parted company, and TRSDOS 2.2 was redeveloped in-house.

Randy started Virtual Technology, his own company, before he left Radio Shack. He explains that the company is "the marketing arm" of Custom Micro Systems. Cook claims he has "sole proprietorship of Virtual and Custom Micro Systems" and that he is sole owner of his DOS copyright.

It was through Virtual Technology that Cook brought out a DOS on his own. An agreement was also made with Ricky Steele of Automated Computer Software, Nashville, TN for its distribution. It was advertised as "TRS-80 DOS 3.0 by the original author." Reportedly, Automated Computer Software went bankrupt. According to one source, Steele, who owned the company, "got no support for the system from Randy, so he held back on payments. Then Randy quit doing any work." Steele was unavailable for comment. Cook's one comment on the episode was, "They took me for over a million dollars and then went bankrupt."

Following the fiasco with Automated Computer Software, Cook released VTOS 4.0. A distribution agreement was made with Dennis Brent of Quality Software Distributors, Dallas, TX in 1980. A manual to accompany the disk was to be provided by Small Business Systems Group, Westford, MA.

"We were going to split profits in a way that we both thought was fair," Cook said. Brent explains the agreement was "all on a handshake."

Problems developed almost at once. Randy contends he did not receive payments for product deliveries. Brent argues that he could get no support and tried to get bugs corrected on his own. Brent listed patches on the MicroNet bulletin board on a regular basis. Disagreements also developed over the manual.

In the midst of this turmoil, Lobo Drives International, Goleta, CA approached Cook privately. In November of 1980, Lobo signed a contract with Cook for "nonexclusive" but "total" rights to VTOS. They have reassembled parts of the VTOS source code and incorporated it in a new code that they are selling as Lobo Disk Operating System (LDOS).

"Cook is still concerned about royalties he believes should be paid by Radio Shack for TRSDOS. . . and. . . by Apparat for NEWDOS."

Cook complains that LDOS will be "head-on" competition for VTOS, and is not what he expected from the agreement. Before the product reached the market, he called the whole affair "another stab in the back."

Lobo's president Roger Billings and Bill Schroeder, who is working on the LDOS project for Lobo, both believe the contract gives the company the right to use Cook's code, in part or in whole, in a variety of their own products. They point out that the company will be paying Cook royalties no matter how or where they use his material.

Another person at Lobo, who wished to have his name withheld, expressed his overview of Cook. In his opinion, Cook's biggest problem has always been an inability to polish and finish his work. He could not say if payment problems caused the lack of support, or if the lack of support caused the pay disputes. But now Lobo has done the finish work and signed a contract to pay Cook royalties for the first time since his product has been around.

"In my estimation we are providing Randy Cook a huge favor. We are going to make him a very rich man." Lobo, he concedes, stands to make some money on the deal as well.

Cook is still concerned about royalties he believes should be paid by Radio Shack for TRSDOS 2.2 and 2.3, and royalties by Apparat for NEWDOS. "I own the copyright on the DOS and all its derivative DOSs," he says.

Cook believes that the heart of the question of copyright infringements is a distinction between patching the object code and reassembling the source code. Without accessing the source code, debugging and other changes can be effected by adding jumps in the object code. Cook claims, "Anybody can buy the object code (by simply buying the program) and apply patches to make it jump back."

He continues to say that "NEWDOS, in my opinion, was version 2.1 TRSDOS plus a few patches. NEWDOS has not been reassembled." He believes the same is true of TRSDOS 2.3. "When I look at these programs, what I see is that people are making jumps."

To emphasize his point, Cook suggests booting up your system with NEWDOS. Type BOOT/SYS.WHO and hold down the 2 and 6. Cook's copyright message appears on the screen, beginning with this sentence: "This operating system was designed by and is the sole property of Randy Cook." On TRSDOS 2.3 and up, the same command brings up a Tandy Corp. copyright message.

At Apparat, company general manager, Edward Krahmer responds, "We have not removed his copyright so people still have knowledge of the original product. We do not like to see anybody's software bootlegged and we like to see credit given."

Krahmer also said, "We have never said anything else but that all we did was provide some additional support in the form of patches and enhancements to Radio Shack's TRSDOS. To our knowledge we have never marketed an operating system. We gave credit and tried to make sure the customer had purchased TRSDOS. And we have studiously avoided selling to anybody who does not have TRSDOS. Our concern was that we would not violate Radio Shack's rights."

Krahmer added that he had never had any contact with Randy Cook. He would like to avoid a controversy and says, "If there are differences of opinion, it's semantics, because there are no philosophical differences." Copyright protection, in Krahmer's opinion, is vital to all people in the microcomputing market.

Whether or not Apparat and Tandy have actually infringed on Randy Cook's copyright is debatable. Whether or not Randy Cook can support his DOS progeny is also debatable. The conflicts raise some interesting legal questions for the entire micro market. ■

by Nancy Robertson
80 Staff

Kids Say the Darn'dest Things About Learning from Classroom Computers

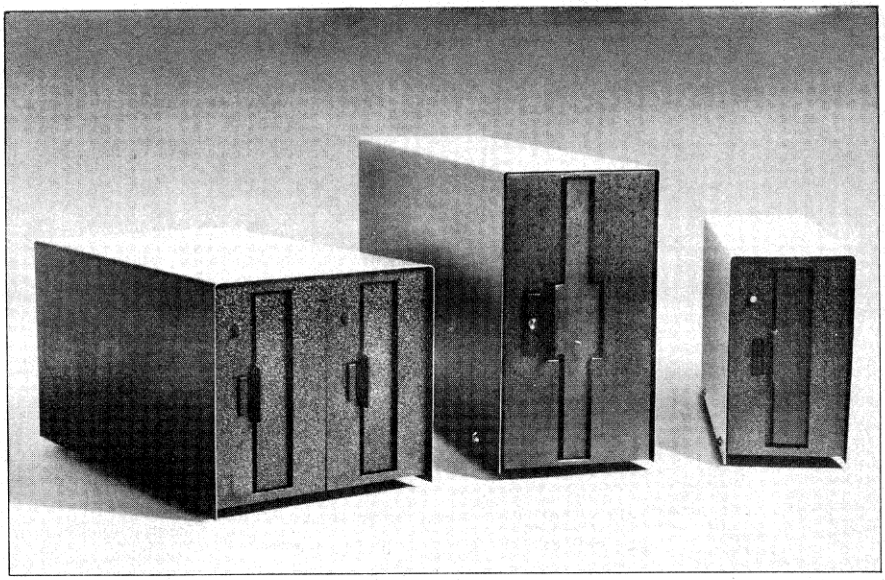
While educators and experts spout off about computers in the schools, we think it is about time that the kids have a chance.

Students at an elementary school in New Hampshire have been using two Level II TRS-80s with cassette for about a year on a Title IV computer literacy grant.

Staff editor Chris Crocker sat at a table in the school library with fourth graders Laurie, Allison, Amy, Kristen, Jeff, David, Doug and Thede. The school requested that its name be withheld. Here's what the kids had to say.

80: What are you learning on the computer?
Several (at once): Programming.
Thede: Well, small things like PRINT"CAT" and stuff like that.
Doug: What do you mean, like what games are we learning—like Blackjack and Mountain?
All: Number Series, Westward Ho, Herkle, Bagles.
80: Are you learning math?
All: Yeah!
Laurie: And social studies.
Doug: In one game we have ten seconds to answer a question in math and each time you get one right you build a rocket and the last time the rocket blasts off.
80: I wish I could play games like that. Do you work alone or do you work together?
Amy: Sometimes alone and sometimes together. Usually together.
80: Is there anyone here who doesn't like the computer?
All (laughing): No.
80: What's your favorite program?
All (around the table): Blackjack, Bagles, Herkle, Westward Ho. Hey, Thede, what tape do you like?
Thede: Westward Ho.
80: What's Westward Ho like?
Jeff: You try to make over a thousand miles up to Oregon, and you have rain and hills and everything.
Doug: And you die to death, and it asks if you want a fancy funeral.
Thede: They give you a thousand dollars and you have to spend it wisely.
80: Sounds like fun. Do you take tests on the computer?
David: There's something we call the Number Series.
Thede: We have an Apple at home, and—
80: I wish I had one. Do you know the number questions that you have to answer really quickly?
All: Awww, yeah, we have that one.
80: Do you think those are tough programs?
Several (at once): No, they're simple.
80: Do you think they're fair?
Jeff: Yeah. Mrs. Bradley sometimes puts three digits in the multiplication tables.
Doug: Well, one thing in Westward Ho, every time you try to shoot the animal, you always miss.
Thede: There's a one out of ten chance that you'll hit it.
David: About one out of 100.
Jeff: Bullseye!
Doug: I only hit it once.
Jeff: Nail it right between the eyes.
Thede: It'd be a miracle to hit it!
80: Who do you think is more fun to learn from, your teacher or the computer?
All (laughing): The computer!
Amy: Yeah, they can't talk back.
David: They can't make you stay in for recess.
Jeff: Yeah, and teachers don't tell you you're right.
Amy: Oh, no!
80: Teachers don't tell you you're right?
Jeff: Well, not always.
Thede: The computer writes on the screen "WRONG, TRY AGAIN."
Amy: The teacher will say, "Stay in for lunch or recess."
80: Do you ever get mad at the computer?
All: Yeah.
80: When?
Laurie: One time I got mad because the teacher typed in the wrong thing. The question was, "What's New Hampshire: country, state or county?" I typed in STATE and it was wrong, so that's when I got mad.
David: Oh, I get mad when I play Westward Ho and they say you starve to death and light your wagon on fire and you get caught in a snowstorm.
Doug: And wild animals attack you.
Jeff: I hate waiting for it to load—you sit there for hours and hours—maybe two minutes!
80: What do you think is a long time for a program to load?
David: Five minutes.
Jeff: Sometimes it takes longer if it doesn't load.
80: Can you all load the programs?
All: Yes.
80: Do you know how to turn on the computer?
All: Yes.
80: So, you think that five minutes is a long time to wait for a load?
Doug: Yeah, the average is about three.
80: Do you get impatient?
Amy: Yeah, sometimes you just sit down and read.
Jeff: I almost fell asleep once when it was loading.
Allison: Are you going to give us a test on how to load it or something?
80: Oh no, I'm not a teacher.
All: Yay!
Thede: Something that I really love about my Apple is it loads fast. You see, I have a disk drive.
80: Do you think that you've learned a lot on the computer?
All: Naah.
Kristen: We've learned a few things.
80: Do you think you're better at multiplication tables?
All (laughing): No.
Amy: The hard ones are when we go up to three digits.
David: Oh, yeah, that was fun. It was that launch the rocket thing. I never launched the rocket that time.
David: Mrs. Bradley wrote it. That's Mrs. Bradley over there.
80: I know, I already talked with her.

Continues to page 55



More power to you.

Disk drives, hardware and software— now more affordable *and* more available!

NEW PRODUCT ANNOUNCEMENT: NEW MODEL III WITH DISK DRIVES!

TRS-80™ Model III with dual MPI B91 80-track disk drives—704K of reliable disk storage on only two drives!
This system features:
- Two MPI B91 double-density 80 track drives
- 48K RAM, double-density Newdos 80
- 3 millisecond access time
- Includes A.M. Electronics' controller board and MAKE80© program
- Complete, ready-to-run

Special Introductory Price..... **$2295**

The Ultimate Small Business Computer: TRS-80™ Model III with four dual-headed MPI B92 80-track disk drives!
This system features:
- Four proven MPI drives
- 48K RAM, double-density Newdos 80
- 3 millisecond access time
- 2.8 Megabytes reliable disk storage
- Includes A.M. Electronics' MAKE80© program and controller board
- Complete, ready-to-run

Special Introductory Price..... **$3795**

CASES AND POWER SUPPLIES

5½-inch enclosures
Single drive unit case and power supply **$85**
Dual drive unit case and power supply **$120**
(Extender cables are $15 each extra)

8-inch enclosures
Single drive unit case and power supply **$150**

Store Hours: Tues. - Fri. 11-7, Sat. 11-5.
Complete Service Center: For speedy repairs, including modified TRS-80's.

Attention dealers, OEM's & distributors
Call us for details on our attractive pricing.

SPECIALS!

SAVE ON COMPLETE DISK DRIVES!

5¼-inch drives
40-track MPI 51 w/case, power supply and extender cable ... **$320**
80-track MPI 91 w/case, power supply and extender cable ... **$425**
Dual-headed 80-track drive with case and power supply..... **$550**

8-inch drives
Single Siemens 8-inch drive with dual case and power supply .. **$695**

INTRODUCING TEAC DISK DRIVES—THE BEST DRIVES WE HAVE SEEN TO DATE!

Teac 40-track disk drive **$299**
Teac 80-track disk drive **$399**
All drives complete with cases, power supplies and extender cables.
(Add $20 for "Flippy" version)

PRINTER SPECIAL!
Epson MX-80 Printer..... **$495**

TRS-80™ SOFTWARE PACKAGES

MAKE80©
Converts standard 35 or 40 track diskettes into 80-track-readable diskettes ... **$14.95**

The power behind the drives©

✓ 452

A.M. ELECTRONICS, INC.
3366 Washtenaw Ave.
Ann Arbor, Michigan 48104
(313) 973-2312

Visit our retail showroom for a "hands-on" look at our TRS-80™ hardware and software

SUPER UTILITY, by Kim Watt.
Stand-alone 24K machine language program for disk includes:

ZAP UTILITY
- Read/modify data regardless of disk protection • One-step track-to-track/sector to sector • Dual cursors; ASCII and Hex. Modify in Hex, Decimal or ASCII • Display disk sectors, display file sectors, copy disk sectors, compare disk sectors, display/modify main memory, search memory or disk for specified string and return location

PURGE UTILITY
- Kill files by file spec or category • Zero out unused directories or sectors
- Compute passwords, change disk's name, date, passwords, protection levels • Directory routine indicates all active and inactive files, their location in directory and status of granules

DISK COPY UTILITY
- Copy *any* TRS-80™ readable disk, regardless of protection

TAPE COPY UTILITY
- Copy *any* TRS-80™ readable tape, regardless of protection or baud rate

DISK REPAIR UTILITY
- *Automatically* repair damaged HIT, GAT or BOOT sectors • Directory check advises of errors • Automatic recovery of killed files • Shows active and inactive files, and their location on the disk

MEMORY UTILITY
- Move memory • Jump to memory • Test memory • Compare memory • Input or output any byte to any port • Zero memory • Exchange memory • Edit memory • Load memory

SUPER UTILITY is now available for **$49.95,** plus $2.50 shipping and handling!

™TRS-80 is a trademark of Tandy Corp.

VISA

master charge

Microcomputer Savvy Aids the Handicapped

1981 has been designated "The Year of the Disabled" by the Federal Government in an effort to focus national attention and energies on the needs of thousands of Americans who are physically or mentally handicapped. In both the public and private sectors of the microcomputing community, the development of applications designed to aid in the daily life and rehabilitation of the handicapped are already underway. Large and small firms and hundreds of individual experimenters are working on hardware, software and systems designs for handicapped applications.

A typical example is Bill Yoreo of the Converse Communication Center in West Hartford, CT. By combining his interest in the handicapped with his technical expertise and a TRS-80, Yoreo has made it possible for many hearing impaired people in Connecticut to communicate by telephone.

Messages for the Deaf

His nonprofit company, quartered in the American School for the Deaf, began in 1970 in response to a need that existed for a message relay service for the deaf. People wishing to contact a deaf person would call the message center and leave their message. It would then be relayed to the center's clients who were equipped with five level Baudot TTY units.

Today, Yoreo and the Converse Center are trying to replace the 700, five level, Baudot TTY units in service with an eight-level ASCII unit based on a TRS-80 Model I. His complete communication system includes an interface, modem, coupler and TRS-80, and sells for $625.

Using standard ASCII transmission formats and specifications Yoreo's hardware allows messages to be sent and received over standard telephone lines. Users may communicate through the Converse Center, or directly with each other using Yoreo's equipment. Over 50 TRS-80-based units have been placed in service by Yoreo in Connecticut; and the gradual replacement of outdated TTY units is proving to be a slow and expensive process.

When interviewed by *80 Micro*, Yoreo discussed some of his projects and problems. He had a common complaint about software development. "The software we require is quite specialized," he said. "Up to now, we have been writing all of it in house and that's a problem. We have neither the time nor the resources to devote to software development that we'd like."

Working Closely with Radio Shack

On the subjects of hardware and his affiliation with Radio Shack, Yoreo said, "The TRS-80 is ideal for our purposes. It is relatively inexpensive and reliable." He added, "We have been working closely with Radio Shack and have found them to be very cooperative. They have allowed us to afix a Converse label to our version of their computer and they have waived some of their prohibitions against modification to allow us to manufacture our system while retaining their warranty. In all, they have been extremely helpful and supportive."

Yoreo's present system, is based on a computer that Radio Shack has discontinued manufacturing. In the future he plans to use Tandy's color machine as the core of his system. He told *80 Micro*, "We are in the process now of switching over from the Model I to the Color Computer. We have one in the lab for evaluation and the prospects look good."

Voice Synthesis

Another innovator in the handicapped applications market is Triformation Systems, Inc., Stuart, Florida. Their most recent offering, in a long line of products for the visually handicapped, is called the Free Scan Speech Terminal. The FSST permits a blind operator to call up pages of information which have been stored in a 6502-based terminal. Audio prompts let the operator know where he is on the page and the spelled speech information is delivered in audible form by a voice synthesizer. Data rates as high as 480 characters per second are possible. The cost of the system is about that of a good used car, $3400.

Joe Kerstey of Triformation told *80 Micro* that, in his opinion, voice synthesis is the way of the future in applications for the blind. He was quick to add, however, "There will always be a need for hard copy at some point. Braille will not be totally replaced because its use is too widespread."

Contest to Help Handicapped

To encourage the rapid research and development of computer-based handicapped applications, the National Science Foundation and Radio Shack have cosponsored a national computing competition. Administered by the Applied Physics Lab of Johns Hopkins University, the contest is billed as "Personal Computing To Aid The Handicapped, The Johns Hopkins First National Search."

The contest is a national search for ideas, devices, methods, and computer programs to help handicapped people overcome difficulties in learning, working and adapting to home and community settings. One hundred awards will be made including a $10,000 grand prize, 15 personal computer systems, other cash prizes and certificates of achievement. Entries from professional computer specialists, high school and college students and from interested hobbyists are encouraged. Ten regional competitions will determine the 100 finalists who will be invited to Washington, D.C. this fall for the final judging.

The three categories of submissions are: 1) computer-based devices (hardware invented, or modified, for the purpose, or working hardware and software which can demonstrate a new application); 2) computer programs (specialized software and concepts for existing computers); 3) system concept/design (written descriptions of ideas not yet implemented). Entries must be received by June 30th, 1981.

A two-day conference and seminar will be held in Washington at the time of the final judging.

Paul Hazen, a researcher at the Johns Hopkins Applied Physics Lab and director of the competition, told *80 Micro* that he expects over 10,000 entries by the June 30th cutoff date. Hazen said, "What we are looking for is technical feasibility. Entries must be practical and useful to the handicapped with a minumum of computer training." Hazen feels that the competition is a challenge to the American people

Continues to page 55

NEVER UNDERSOLD!

That's right, if you can find a lower price in this magazine for any of the items listed in this ad, we will reduce our price below our competitor's price. See each box below to determine how much EXTRA we will cut off of THEIR price if we're not lowest. Please consider the competitor's shipping charges, OUR SHIPPING IS FREE!*

FLOPPY DISKETTES & SUPPLIES

IF YOU CAN FIND A LOWER PRICE IN THIS MAGAZINE ON ANY OF THESE ITEMS, DEDUCT $.50 FROM OUR COMPETITOR'S PRICE, THAT'S OUR PRICE!

Call For Quantity Discounts

Verbatim Diskettes (box of 10)

5¼" MD525-01 soft,10 or 16	$26.50
5¼" MD577-01 quad soft, 10 or 16	$33.00
8" FD34-1000 soft	$30.00
8" FD32-1000 hard	$30.00
8" FD34-8000 double density soft	$44.00
8" FD32-8000 double density hard	$44.00

Printwheels (specify style)

Qume or Diablo	$6.50

Labels

3½"x15/16" (5000 labels)	$18.75
Other sizes and quantities	CALL

Ribbons

Diablo Hy Type I	$4.95
Diablo Hy Type II	$5.25
Qume Sprint	$3.50
Centronics Zip Pack	$3.95
MANY OTHERS	CALL

PRINTERS

IF YOU CAN FIND A LOWER PRICE IN THIS MAGAZINE ON ANY OF THESE PRINTERS, DEDUCT $10 FROM OUR COMPETITOR'S PRICE, THAT'S OUR PRICE!

Paper Tiger IDS-440G	$939
Paper Tiger IDS-460G	$1193
Anadex DP-8000	$855
Anadex DP-9500	$1395
TI-810 Basic	$1625
Centronics 737	$825
NEC 5500 D w/ Bidrctnl Board	$2695
NEC 5530	$2595
VISTA Daisey Wheel Printer	$1834
Qume 5/45,5/55	CALL
Escon IBM Interface	$595

Call For Other Printers

HARDWARE

IF YOU CAN FIND A LOWER PRICE IN THIS MAGAZINE ON ANY OF THESE ITEMS, DEDUCT 5% FROM OUR COMPETITOR'S PRICE, THAT'S OUR PRICE!

Novation CAT Modem	$145
Novation D-CAT Modem	$185
16K Memory kit	$46
Isolators	$49
Shugart 35tr Drive	$349
Pertec or MPI 40tr Drive	$359
Lobo Drives	CALL
Matchless Drives	CALL
Percom Doubler	$209
Percom Seperator	$27
AIM-65 Computer	375
TI-99/4 Computer	$925
California Computer Systems Bds	CALL
Symtec Computer Boards	CALL
Mountain Hardware Boards	CALL
Green Screen	$11

Call for Other Hardware

4636 Park Granada
Calabasas, Ca. 91302

✓69

Alpha Byte Storage

VISA

For phone orders CALL:
(213) 883-8594

SPECIAL #1

If you purchase the "TRS-80 DISK AND OTHER MYSTERIES" Book for the regular price of $22.50
you can buy 10 VERBATIM DISKETTES AND a plastic library case for $22.00
TOTAL $44.50

SPECIAL #2

If you purchase APPARAT NEWDOS+ for the regular price of $99.95
you can buy 10 VERBATIM DISKETTES AND a plastic library case for $5.00
TOTAL $104.95

SPECIAL #3

If you purchase APPARAT NEWDOS/80 for the regular price of $149.00
we will give you 10 VERBATIM DISKETTES AND a plastic library case ..... FREE
TOTAL $149.00

SPECIAL #4

If you purchase the MICROSOFT BASIC COMPILER for the REDUCED PRICE of $190.00
we will give you 10 VERBATIM DISKETTES AND a plastic library case ..... FREE
TOTAL $190.00

SPECIAL #5

If you purchase the new "MICROSOFT BASIC REVEALED" book for the regular price of $29.95
you can buy 10 VERBATIM DISKETTES AND a plastic library case for $20.00
TOTAL $49.95

SPECIAL #6

SOFTWARE DEALS--If you want to purchase any software (including Utilities, Operating Systems, Games, Buisness Programs, etc.) we will automatically deduct from 10% to 40% off the regular retail price.

*FREE shipping on all orders over $20. Visa and Master Card accepted. All never undersold offers good as supply lasts. Please add 2.00 for all COD orders. Please call for items not listed. We gladly answer any questions on all of our hardware, software, and supply needs. Quanitity discounts available. School purchase orders accepted. Please remember to figure competitors shipping and handling charges when arriving at never undersold price.

Model II, Tandy II & the Vertical Market

Tandy Corp. has another way to market the Model II business microcomputer. It's called the Tandy II, and is not available from Radio Shack stores.

The Tandy II has a blue rather than a black face, and can sport the name of the company marketing it on the logo panel. Furthermore, Radio Shack service centers will repair the Tandy II.

Vertical Marketing

Tandy II equipment (including the Model II, and peripherals such as disk drives and printers) has been available for about a year now, according to Charles Phillips, senior vice president of special markets at Tandy. The equipment is sold only through a select group of systems houses addressing a "vertical market." This means those firms who deal directly with a particular type of business—lawyers or doctors, for example. Phillips said that they are trying to focus on the business consumer who is less likely to purchase equipment at a retail store.

According to Irwin Taranto of Taranto & Assoc., one of the systems houses selling Tandy II equipment, "Some people don't understand just what they need. We sell them the whole system."

Firms selling the Tandy II equipment can sell the microcomputer with whatever peripherals they choose, and with their software—a custom-tailored approach. According to Taranto, "the profit margin on the machine itself is not that great." Profits are made on the value added to it, such as the software.

Drew Bourrut of Bourrut Consulting Corp., Smithtown, NY, said "If I'm selling large equipment I can sell Pertec. If I can sell something smaller, I can sell the Tandy II. You're wasting your time in this business until you can sell hardware."

Tandy does not suggest a retail price for the Tandy II according to Phillips. "They can sell it for anything they want. We'd like to think that they're selling a package."

"According to Taranto, 'the profit margin on the machine itself is not that great.' "

Tandy provides an abbreviated warranty on Tandy II equipment. The Radio Shack warranty on TRS-80 hardware is for 90 days. The warranty on Tandy II equipment is for 45 days from date of shipment. Some systems houses, such as Bourrut, will extend the warranty for an additional period, paying the difference.

According to Taranto, his company is doing well by the Tandy II equipment. "This is where we think the industry is going," he said.

Tandy, Too—Three?

Tandy II contract marketing is successful for Tandy, as well. According to Phillips, Tandy plans to be increasingly more aggressive in the future. They are considering marketing the Model III in a similar fashion. "It started off slowly, but has really been picking up," said Phillips.

Though unwilling to comment on the number of systems houses marketing the Tandy II, Phillips did say, "It's good business for us."■

by Chris Crocker
80 Staff

Banks Can File IRS Form 1099 on Magnetic Media

In a recent press release the Internal Revenue Service encouraged users of IRS form 1099 to abandon the practice of filing paper information returns. In place of paper, the IRS wants to receive magnetic media in the form of computer tape, diskettes or disk packs. The use of magnetic media was urged due to the substantial savings in time and money that result for the IRS when they receive computer-ready returns.

Form 1099 is used primarily by banks, brokerage houses and savings and loan institutions to report the payment of interests and dividends. When received by the IRS, the information of form 1099 is used for document matching and is compared with that supplied by individuals on their personal income tax returns. Before any computerized comparisons take place, however, all form 1099 information must be entered into the government's computers. When form 1099 information is received on tape or disk, the lengthy and error prone data entry process is eliminated.

Magnetic Media Returns

The practice of filing magnetic media returns is slowly catching on with those required to file form 1099. In 1976, approximately 38,400 magnetic media filings of form 1099 were made. In 1978, close to 46,800 filings were made in computer-ready form. The IRS expects well over 50,000 this year.

Besides reducing paper handling costs and saving considerable data entry time, the IRS gains additional benefits from the use of magnetic media. The reduced space demands made on the government's storage facilities are almost reason enough to justify the use of disks and tape. Magnetic media has also proved to be more accurate for the IRS. Their experience has shown that information in electronic form has required less correction than its paper counterpart.

To facilitate the switch to magnetic media filing the IRS furnishes programming guidelines and technical assistance free to anyone interested. Information on magnetic media reporting can also be found in a free IRS booklet. Publication 882, "Magnetic Media Reporting of Information Returns," can be obtained by calling the tax forms/information number listed in local telephone directories.

Not for Personal Computer Users

While encouraging their larger customers to file magnetically, the IRS does not encourage individuals with microcomputers to do so. At present, the domain of magnetic media filing belongs to large institutions using forms like 1099. IRS spokesmen were reluctant to comment on when, if ever, personal income tax forms like 1040 or 1040A would be available on magnetic media. IRS media feature service representative, Johnell Hunter said, "Right now the last thing we want is 200,000 microcomputer owners sending us disks and tapes."

It appears that this April, as in past years, TRS-80 owners should keep their pencils sharp, their subtraction accurate and their computer in the closet. The IRS is not quite ready for the microcomputer yet.■

Chris Brown
80 Staff

Coming Home to the Farm

Computer information networks and Radio Shack products, which have already brought electronic news and banking into the home, are now bringing commodity reports down home to farmers. Professional Farmers of America, Cedar Falls, IA is selling subscriptions to a 24-hour-a-day farm information service; and Project Green Thumb, Lexington, KY has been offering a similar service through a federally funded experiment. Both agriculture networks are using terminals and software designed by Radio Shack.

Instant Update, the network offered by Professional Farmers, is distributing Videotext terminals with 16K memory to their subscribers. Video display is through a standard television.

Stewart Cross, who is the market manager for Instant Update, explains that his company began offering a monthly newsletter about price fluctuations several years ago. As commodity prices began to vary more frequently, the newsletter changed to weekly distribution. When price fluctuations became daily, Professional Farmers initiated a telephone hotline.

The quantity of incoming information relevant to farmers, the continual change of commodity prices, the demand for the hotline service, and awareness of Project Green Thumb lead Professional Farmers to the idea of a computer network to handle the information glut.

Besides quoting commodity prices from the Chicago Board of Trade, Cross says Instant Update also includes "market-sensitive news" and "strategies and tactics." These services include such things as charting and technical analysis of the movement of individual commodities, seasonal and regional weather predictions, and same-day reporting on USDA (United States Department of Agriculture) decisions. The service is scheduled to begin March 2nd.

Green Thumb

Ironically, funding for Project Green Thumb expires March 3rd. The pilot project in Shelby and Todd Counties, KY was designed to make market and weather information more accessible for farmers. Using the host computer already established at the University of Kentucky, Green Thumb has offered farmers reports from the Chicago Merchantile Mart, the Chicago Board of Trade, the National Weather Service, the USDA's Agricultural Marketing Service and studies from the university's agriculture specialists.

Although the project became operational March 3, 1980, S. H. Phillips, who has been involved with the project from its early stages, explains that there was a lot of preparatory work beforehand. Western Union developed processors to be located at the county's farm extension offices. Radio Shack designed a terminal called the green thumb box (a less sophisticated version of the Videotext hardware and software). The grant covered a year of operation and paid for the processors and home terminals.

According to Phillips, several private businesses and the American Farm Bureau have shown interest in Green Thumb's potential. "The project is going through a transitional period," he says. While it is hard to predict what lies ahead for Green Thumb, it is apparent that electronic networks have found yet another market. ■

Handicapped

Continued from page 52

to use their skills and computer savvy. He added, "Just as important will be the opportunity provided to inventors and developers to make contact and form partnerships with the handicapped in a way that can lead to the wide acceptance of the use of the new computer technology."

Contest Info

Further information on the computer competition can be obtained from Personal Computing to Aid the Handicapped, Johns Hopkins University, P.O. Box 670, Laurel, MD 20810.

In both public and private sectors efforts are being made to integrate the microcomputer into the handicapped environment. The suitability of the machine in such settings is obvious; its range of applications limited only by the imagination and creativity of its users. ■

By Chris Brown
80 Staff

Kids Say

Continued from page 50

All: Oh.
80: How many of you have used the social studies program? (Four hands go up.) Have you learned a lot from that?
Amy: No.
80: What do you mean, no?
Thede: It's like a test of ten out of ten.
80: How many have you gotten?
All (going around the table): Ten. Nine. Ten. Eight—but then I got ten.
Jeff: He said that Portsmouth was the biggest city in New Hampshire.
80: Would you rather take the test on computer or with a pencil and paper?
All: Computer.
Allison: You don't get yelled at if you get something wrong.
Jeff: She'll (the teacher) make us write it 50 times until we get it right.
David: The computer just says, "NOPE. SORRY, TRY AGAIN."
80: How many tries do you get?
Doug: Three. No, two.
80: Do you get a grade for that?
Laurie: No, it just says eight out of ten.
80: Do you like that better?
All (at once): Yes. No.
80: Would you like to have a computer for a teacher?
Several: Yes.
Several: No.
All (together): Yes.
David: Yeah, I would. I'd like to have a walking robot.
80: What would happen if something went wrong with it?
Jeff: That would be great.
Doug: Yeah, then you could go home.
80: Permanent recess, right?
Amy: *Very* long.
80: What would happen if it told you you were wrong when you knew that you were right?
Jeff: I'd kill it.
80: Would you like a computer in every class?
All: Yeah!
David: I'd like to have a computer in my desk. I'd get answers from it.
80: What if you had a computer that only asked questions?
Several: I'd destroy it. I'd make it go bozo.
Amy: Is the tape running?
David: Do you think your story will be in the magazine?
80: It might be.
Jeff: Do you think we'll all be in it?
80: If my editors like the story.
Doug: If they like the story, put it in the paper!
Jeff: Put it in *Time Magazine.*
David: *Boston Globe.*
Amy: Put it in *National Geographic*!
80: I'll see what I can do. ■

Two New DOSs of the Near Future

There we were, ready to go to press, and what should come through the mail but a proof copy of Lobo Drives International's LDOS manual. The LDOS disk had not reached the market at press time, but advertising was already running, and company spokesmen promised a mid-February release. At *80 Micro* we can't resist a challenge, so we decided to slide in a preliminary overview of the latest breaking disk operating system—or at least an overview of its manual. And to give you a complete perspective, we are including news of yet another DOS that is expected to reach the market by late spring.

According to Bill Schroeder at Galactic Software, Mequon, MI, Lobo contacted Galactic and Misosys, Alexandria, VA when their disk operating project stalled. Schroeder has since guided the development of the DOS with Roy Soltoff of Misosys acting as the systems analyst. The three companies will share distribution of the product.

The LDOS Manual

The manual itself is written in terms which can be understood by laymen (including a glossary of words which will become all too familiar quickly enough) as well as advanced programmers. LDOS and its utilities are covered in a format similar to the TRSDOS manual.

The preliminary version of the manual is well over 200 pages in length. It includes a section on the LDOS library commands and extended utilities, a section on the job control language, explanation of building files to control the automatic operation of a sequence of events, a section on LBASIC, etc.

Lobo also emphasized their planned user support in cover material that came with the proof of the manual. A toll free telephone number (800) 559-6901 is available so that no one need feel marooned in the event of difficulties. For the first year of ownership, the original disk may be traded for the latest upgrades at the cost of a dollar. The cover matter also states that a quarterly newsletter will be published and a MicroNet bulletin board service will be sponsored by Lobo Drives for LDOS users.

Another DOS—Not a Patch

But LDOS may not be the only new operating system coming out this spring. Kim Watt, the author of SuperUtility, explains that he and Vernon Hestor, the author of ULTRADOS, have "taken ideas from other DOSs and have started from scratch writing a new source code." The DOS they are developing will not be a patch to any other established DOS.

Features will include device independence, machine independence and upper and lowercase. As yet no decision has been reached on the product name. It is likely to be released through Breeze Computing, Inc., Berkley, MI.■

by Jake Commander

NEW PRODUCTS

Edited by Chris Crocker

Program Proofreads Text

Microspell, a proofreading program, looks up each word of a text file in its dictionary and alerts you every time it finds a potential misspelling. The program replaces misspellings with your corrections. The Microspell package contains a dictionary of 20,000 common words and has the capability to add 14,000 words.

Microspell is available for Model II with 48K RAM and with 150K disk capacity. It is sold for $249 from Lifeboat Assoc., 1651 Third Ave., New York, NY 10028.

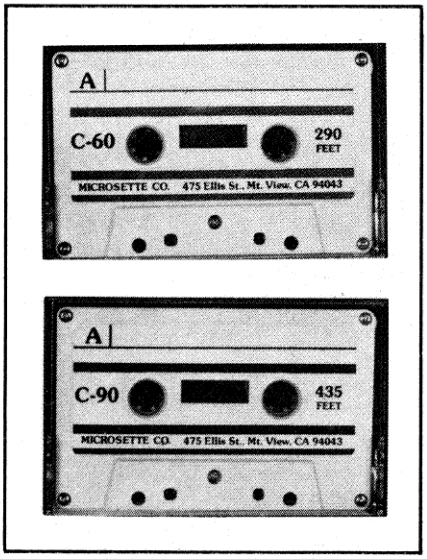
Reader Service ↙ 182

Index Abstracts Magazine Articles

Microcomputer Index indexes and abstracts articles, reviews and columns for 18 periodicals in the microcomputer field. Abstracts are arranged in the same order as they appear in the magazine.

Microcomputer Index is published quarterly, starting with April-June 1980 for $5.95 per issue. For more information contact Microcomputer Information Services, 3070 Adams Way, Santa Clara, CA 95051.

Reader Service ↙ 176



Microsette C-60 and C-90 Cassettes

Touch-sensitive Display

A capacitance-sensitive display kit allows data to be input to a data processing system by touching the appropriate area on the screen with a fingertip. Standard kits are provided in 32 touch position configurations for 12 and 15 inch diagonal CRT monitors. Electronic outputs are provided for parallel or serial interfaces.

For more information contact Interaction Systems, Inc., 24 Munroe St., Newtonville, MA 02160.

Reader Service ↙ 179

Longer Cassettes

Microsette's C-60 and C-90 are 62 and 93 minute recording tapes, usable for both microcomputers and high fidelity recording equipment. Both cassettes have leaders to protect tapes during handling and shipping.

The C-60 length cassette costs $2, and the C-90 length costs $2.50. For more information, contact Microsette Co., 475 Ellis St., Mountain View, CA 94043.

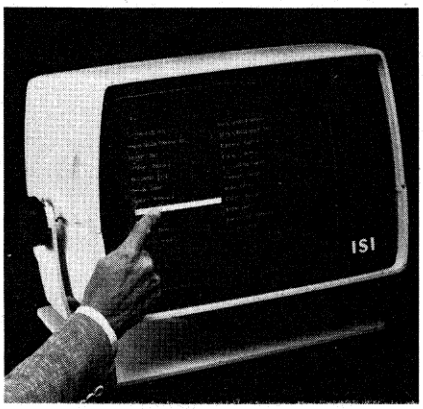
Reader Service ↙ 165

Home Business Programs

A four-program package from Blechman Enterprises is designed for the small home-based business with no employees. Twelve-column Ledger prints out a ledger of income or expense. Speed Letter is a simple word-processing program allowing 150 text lines. Three-across Mailing Labels maintains and prints a small mailing list. Auto Dialer holds 500 names and phone numbers in memory.

The package is available on cassette for 16K Level II BASIC for $25 from Blechman Enterprises, 7217 Bernadine Ave., Suite B, Canoga Park, CA 91307.

Reader Service ↙ 181



Touch-sensitive CRT Kit

Program Manages Small Business Accounts

Ann Rose, an accounts receivable program for the TRS-80, manages 328 accounts per pair of diskettes. The program provides customer lists either alphabetically or in order of account number, and prints statements individually, totally, or partially.

The program requires a Model I or II with 48K, two disk drives and a printer. Prices were not released. For more information, contact Sturdivant and Dunn, Inc., Box 277, 124 Washington St., Conway, NH 03818.

Reader Service ↙ 164

The New Products section is intended to inform our readers of new products on the market. All information in the section is taken from product releases sent by manufacturers. Because of the volume of product releases, we cannot attest to the quality of the products listed.

Books Teach BASIC, Pascal

Inside BASIC Games, by Richard Mateosian is a book which uses eight games as a framework for learning BASIC programming. The book costs $13.95.

Fifty BASIC Exercises by Jean Pierre Lamoitier teaches BASIC through fifty graduated exercises. Each exercise includes explanations, flowcharts, comments, and programs. *Fifty BASIC Exercises* costs $12.95.

Introduction to Pascal is a tutorial guide to UCSD and Standard Pascal. *Introduction to Pascal* is authored by Rodnay Zaks and costs $12.95.

The Pascal Handbook, by Jacques Tiberghien is an alphabetical dictionary of Pascal symbols, reserved words, identifiers and operations for most existing versions of Pascal. For more information, contact Sybex, Inc., 2344 Sixth St., Berkeley, CA 94710.

Reader Service ↙ 167

CP/M Sort Utility

Ultrasort II is a CP/M machine language program that sorts, merges, selects records from data files and finds the number of logical records in a file. It can be used as a stand-alone utility and is written in CBASIC 2. Ultrasort II sorts on up to five keys, each independently ascending or descending.

Ultrasort II is available from Computer Control Systems, Inc., 298 21st Terrace S.E., Largo. FL 33541.

Reader Service ↙ 169

Guide to New Computers

The *Microshopper* is a consumer's guide to microcomputers from PGI Publishing. It includes articles and write-ups of hardware and software on the market, as well as statistical comparison charts about printers and CRTs.

The *Microshopper* costs $9.95 and is available from PGI Publishing, 1425 W. 12th Pl., Tempe, AZ 85281.

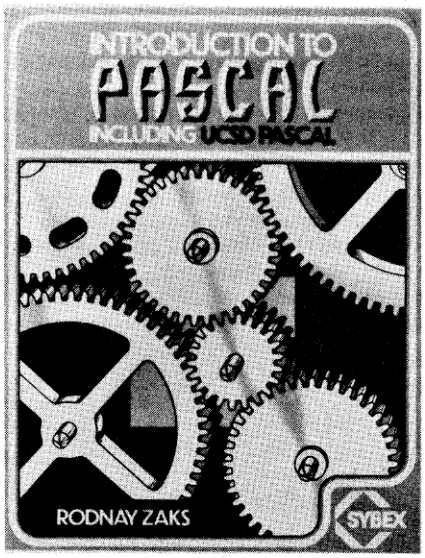
Reader Service ↙ 160

System Manages Database

Versatile Information Manager (VIM) is a four-program database management system. The database manager allows interactive entry, update, deletion and examination of records. The sort program allows sorting in ascending or descending order. The report generation program permits printing of one-up, two-up and three-up mailing labels or columnar listings.

VIM runs on the Model I with at least 32K of memory, an expansion interface and one or more disk drives. The package runs on TRSDOS 2.2 or NEWDOS and is available for $99.95 from Microcosm, Inc., P.O. Box 2034, Dearborn, MI 48123.

Reader Service ↙ 178



Introduction to Pascal

TRS-80 Host Program

Super-host is a host program allowing any type of system to communicate with the TRS-80, according to Programs Unlimited. The program will conform itself to run on TRSDOS, NEWDOS 2.1, or NEWDOS-80 and fixes the calendar function on these systems for proper date advancing.

Super-host also allows limited access through use of passwords. The program is available on disk for $29.95 from Programs Unlimited, P.O. Box 265, Jericho, NY 11753.

Reader Service ↙ 180

Educational Software Catalog

Selected Microcomputer Software, 1980-81 is a catalog from Opportunities for Learning. The catalog lists educational software for the TRS-80, Apple and PET microcomputers in math, science, social studies, language arts and foreign language, computer programming, games, music, art, as well as programs to assist teachers and administrators.

The catalog is available free from Opportunities for Learning, Dept. L2, 8950 Lurline Ave., Chatsworth, CA 91311.

Reader Service ↙ 172

FORTH for TRS-80

TFORTH is a version of FORTH extended for the TRS-80. TFORTH contains a built-in operating system, assembler, text editor, a floating point math package, I/O package, phoneme assembler (support for voice synthesizer) and graphics links to Radio Shack routines.

TFORTH is available on 35, 40, or 80 track disks for $130 from Advanced Technology Corp., 1617 Euclid Ave., Knoxville, TN 37921.

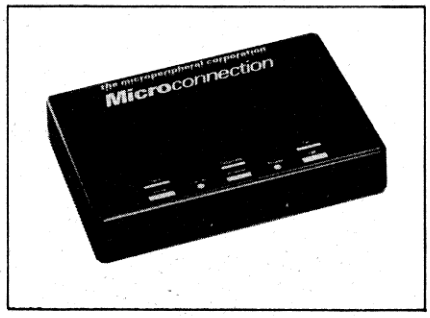
Reader Service ↙ 184

Modem Connects to Keyboard

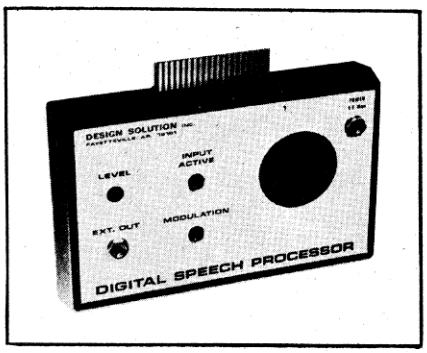
The TRS-80 Connection, one of the Microconnection interfaces from The Microperipheral Corp., connects the TRS-80 Model I or PMC-80 to the telephone network. The direct connection eliminates problems associated with acoustic coupled modems, according to Microperipheral.

The modem attaches to the data/address bus to decode RS-232 information. It can connect directly to the keyboard, eliminating the need for the expansion interface or serial RS-232 I/O board. Prices start at $199.95. The autodial/auto-answer option costs $79.95 from The Microperipheral Corp., 2643 151st Pl. N.E., Redmond, WA 98052.

Reader Service ↙ 168



Microperipheral's Microconnection



Design Solution Speech Processor

Voice Interface, EPROM Programmer

The Digital Speech Processor System from Design Solution, Inc. is a voice interface for the TRS-80. Voice signals through the front panel-mounted transducer are digitized and saved in core images in RAM or on disk files. The Speech Processor System costs $89.95.

Also from Design Solution is the Model AN-551 EPROM Programmer. The Model AN-551 allows dumping TRS-80 RAM to EPROM, verifying EPROM data transfers, copying ROM to EPROM and loading TRS-80 RAM from EPROM. The AN-551 includes software documentation and costs $89.95. For more information contact Design Solution, Inc., Box 1225, Fayetteville, AR 72701.

Reader Service ↙ 175

Printer Ribbon Replacement

A multistrike replacement ribbon for NEC printers is available from Aspen Ribbons. The cartridge is available in black, blue, brown and red.

Aspen Ribbons also will reload used cartridges with new ribbon at a reduced cost. No prices were released. For more information, contact Aspen Ribbons, Inc., 1700 N. 55th St., Boulder, CO 80301.

Reader Service ↙ 166

Utility Adds Time Intervals

Timetrak is a utility for adding time intervals. The start and stop boundaries are 12-hour clock times (in hours and minutes). When T is pressed, the program processes all of the start/stop time pairs. It then displays, in hours to five decimal places, the sum of the intervals spanned.

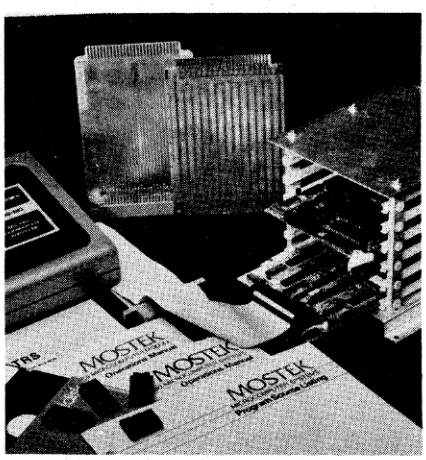
Timetrak is available on cassette for Level II 16K for $25. It is also available on disk for the Model II at $35 from Omni Systems Co., P.O. Box 29347, Minneapolis, MN 55430.

Income Tax Program Book

The 1980 Gooth Income Tax Book, an aid for computing taxes on microcomputers, includes more than 40 income tax form programs, line by line, as well as other programs used in tax work.

Programs are written in Model I BASIC. The book costs $16.95 from Gooth Software, 931 S. Bemiston, St. Louis, MO 63105.

Reader Service ↙ 171



TRS-Proto system from QC Microsystems

STD Bus System

The TRS-Proto system consists of the components necessary for a stand-alone standard bus system using the TRS-80 Model I for editing, assembling, and console control operations during program development.

The standard package includes the Mostek MDX-CPU2 card and the Xitex XTD-TRS interface card, a six-slot standard card cage with wire wrap and extender cards, static RAM memories with custom address PROM and interface cable for the TRS-80, Mostek DDT-80 debug operating system in ROM, and a 5 1/4 inch disk containing driver software.

The package is available in a 2.5 MHz configuration for $895 and a 4 MHz configuration for $1050. Both systems operate from a single +5 V power supply. For more information, contact QC Microsystems, P.O. Box 401326, Garland, TX 75040.

Reader Service ↙ 170

Solitaire Fantasy Game

Lords of Karma is a solitaire fantasy adventure. According to Avalon Hill, the program takes you through forests, mountains and caverns while completing tasks for the Lords of Karma.

Lords of Karma requires a 48K Level II TRS-80 and costs $20 on cassette. For more information, contact The Avalon Hill Game Company, 4517 Harford Rd., Baltimore, MD 21214.

Reader Service ↙ 162

Electronic Circuit Analysis

Passive Electronic Circuits Analysis (PECA) from Adventure International will calculate all currents (magnitude and phase) in any circuit with up to twenty meshes containing resistors, capacitors, and inductors. Using Ohm's Law relationships, other circuit parameters may then be found.

The program requires a Model I 16K Level II and costs $19.95. It is available from Adventure International, Box 3435, Longwood, FL 32750.

Reader Service ↙ 163

Space Conquest Game

A strategic simulation entitled Conquest is set in outer space, where two players fight for control of a newly discovered planet.

Conquest costs $14.95 on cassette for Level II 16K from Lakefront Software, P.O. Box 5240, Willowick, OH 44094.

Reader Service ↙ 173

Advanced Speech Driver

Speak-2-Me-2 is a printed circuit card that installs in a Texas Instruments Speak & Spell, and by providing the interface between the Speak & Spell and the Model I or III, allows the user to add speech to business and game programs.

A driver program, besides providing a direct output routine for words and

phrases of the Speak & Spell vocabulary, also includes routines which will create new words using syllables of Speak & Spell words. The advanced speech driver and speech games disk costs $29.95. The Speak-2-Me-2 adapter costs $69.95 including cable. For more information, contact Percom Data Co., Inc., 211 N. Kirby, Garland, TX 75042.

Reader Service ↙ 183

Accounts Receivable System

AR is an invoicing and monthly statement system which keeps track of current and aged accounts receivable. The package maintains a file for each customer consisting of name, address and phone number, along with type of account, current balance and tax rate.

AR operates on TRSDOS 1.2 on the Model II and requires 64K and dual disk drives. For more information contact Micro Architect, Inc., 96 Dothan St., Arlington, MA 02174.

Reader Service ↙ 185

Tiny Pascal Patch

Paspatch adds utilities and enables disk input and output for Radio Shack's Tiny Pascal for the Model I. The new commands are entered under the Tiny Pascal monitor system.

Paspatch is a patch for Tiny Pascal—both tapes are necessary to make a disk based system. Both 32K and 48K versions of Paspatch on the same cassette are available for $14.95 from Modular Software Assoc., 3533 Prospect Ave., Glendale, CA 91214.

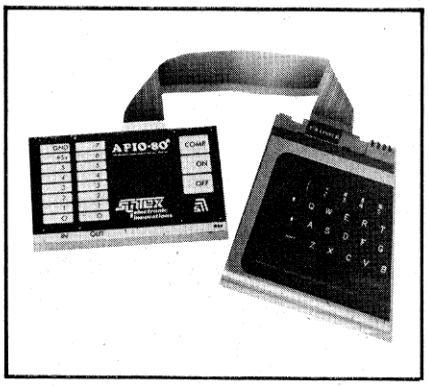
Reader Service ↙ 177

Program Maintains Checking Accounts

Microcheck-80 is a software package that maintains a checking account using a TRS-80 Model I with 32K and one disk drive. The program reconciles bank statements and maintains a complete file of cleared checks. The package permits retrieval by category, month, or any combination of the two.

Microcheck-80 is available for $39.95 from Suma Microware, 1110 W. 41st St., La Grange, IL 60525.

Reader Service ↙ 326



Syntex Electronic Innovations APIO-80

All-purpose I/O Port

The Syntex Electronic Innovations APIO-80 is an all-purpose input-output port for the TRS-80. The port allows software control of any electronic device connected to it, according to Syntex. The port can be used with Level II BASIC or Level I T-Bug machine language programs.

The APIO-80 is an assembled and tested printed circuit board that plugs into the expansion port of the TRS-80. The package costs $45, and includes the *Syntex APIO-80 Newsletter* from Syntex Electronic Innovations, P.O. Box 4034, Lancaster, PA 17604.

Reader Service ↙ 161

PMC-80 32K Expansion

The Expander (Model EXP-100) provides 32K memory expansion, mini-floppy disk interface for four drives, Centronics parallel printer interface, RS-232C interface and an S-100 bus for the PMC-80.

The PMC-80 with the Expander will run TRSDOS software as well as other systems designed for the TRS-80. For more information contact Personal Micro Computers, Inc., 475 Ellis St., Mountain View, CA 94043.

Reader Service ↙ 328

Book Features Screen Graphics

Computer Videographics: Color, Design, Typography by Edmund Van Deusen presents the basic principles of graphic design as they apply to the monitor screen. Guidelines are provided for the use of animation, typography and blank space.

Computer Videographics is available for $18.50 from CC Exchange, Box 1251, Laguna Beach, CA 92652.

Reader Service ↙ 325

Program Aids in Income Property Analysis

INCOPROP is a real estate program for income property analysis. The analysis generates a property operating statement and a 10 year cash flow analysis. A built-in I.R.S. tax table aids in calculating an investor's projected after-tax income stream.

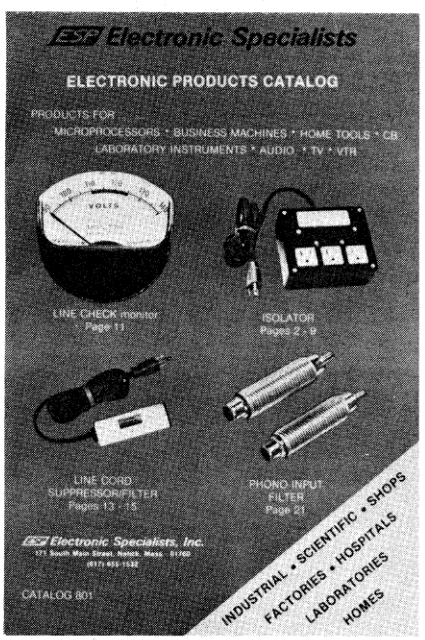
INCOPROP operates on the TRS-80 Model I or III. The package is on cassette and includes a pad of data worksheets for $120. For more information, contact E-Z Software, P.O. Box 591, Novato CA 94947.

Reader Service ↙ 329

Catalog Lists Electronic Products

Catalog 801 from Electronics Specialists lists interference control products, protective devices and other electronic products. Contact Electronic Specialists, Inc., 171 S. Main St., Natick, MA 01760.

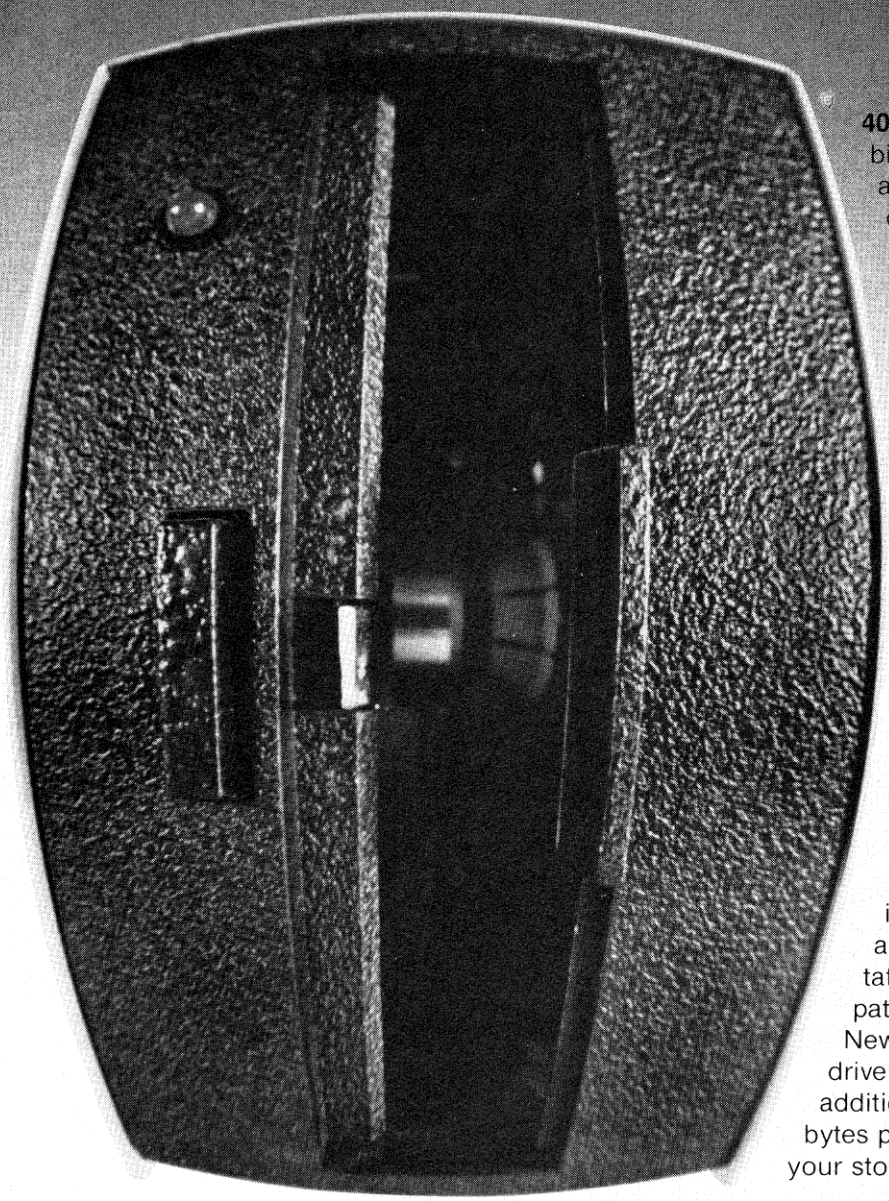
Reader Service ↙ 327



Electronics Specialists Catalog

Apparat introduces

More bytes per buck

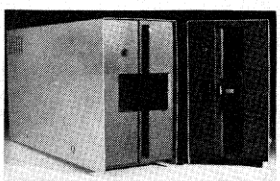


405 K/bytes of storage. Apparat has combined its Newdos/80 operating system and a dual-sided 80 track mini-floppy drive to give you 405,000 bytes of storage in a single volume. Modification patches to Newdos/80 expands the capability of single density drives, so you'll have greater applications for your TRS-80 model 1.®

Drives plug directly into an expansion interface with no modification required so you can now have over 1 megabyte of storage on-line with standard mini-floppy diskettes. Each drive has 316 free grans, for a total of 948, on a maximum of three 80 track drives, which can be added to a TRS-80.

Upgrading to double density is possible by running under most double density controllers.

And, you can choose either an MPI or Tandon Drive Mechanism. Drives come complete with case, power supply, interface cable and documentation including patches to Newdos/80. Either drive mechanism is priced at only $839 with additional drives available at $789. At 482 bytes per buck, it just might be the answer to your storage problems.



Apparat, Inc.

4401 S. Tamarac Pkwy. • Denver, CO 80237 • (303) 758-7275

(303) 741-1778

800 525-7674

Not for use in Colorado, Alaska, or Hawaii

VISA master charge

264

All prices cash discounted / Freight: FOB factory. Ask for our free catalog.

In this rebuttal to an 80 letter.
Author Harv Pennington asks his critic,
"Is English your third language?"

Badinage and Discipline

TO: John Grass
FROM: H.C. Pennington, author *TRS-80 Disk & Other Mysteries*
SUBJECT: Book Review letter, December, *80 Microcomputing*, p. 18.

John, I would like to reply to your letter and perhaps clarify a few things for you. I will take each of your points in turn.

The title of the book is *TRS-80 Disk and Other Mysteries.* The key word here is DISK, *not* disk operating system. This book is about D-I-S-K. This book is not about the PROGRAM CODE that is on the disk but the CODES that the system programs use to determine where things go—passwords, file tapes and so forth.

To take me to task for not disassembling the PROGRAM CODE for the system is, in my opinion, a bit unfair. However, I suspect that you are a novice user (if not somewhat naive) to expect that the disk is the same as the disk operating system, a totally different subject.

To those of us who suffered with TRSDOS 2.1 and 2.2 the necessity of recovering the files on the disk was a daily, if not hourly, occurrence. In my own case, the computer was in use twelve to fifteen hours a day. The disk drives were only 35 tracks, and every available byte of disk space was precious. Everyone using the computer knew in their heart-of-hearts that it *should* work and work reliably. But the system would boot in the middle of a

SAVE, or when you entered the last change to a 25K program; or it would "eat the disk" for unexplained reasons.

Radio Shack's response to these problems was that we had modified the system somehow (not true, at the time), that the hardware was "bad" or that we had some "power problem."

Little did the user know that most of the problems were in TRSDOS 2.1 itself.

Accidental Apparat

Now, hark back again to those early days of 1978-79. At the time, TRSDOS 2.1 was all there was and I, along with thousands of others, was going crazy because of its bugs and quirks. In those dark and ancient days we saw the message, "NO SYSTEM" almost as often as we saw "DOS READY." There were no explanations, no help and no hope in sight.

Along came a couple of guys from Colorado with something called NEWDOS. You refer to them in your letter to *80 Microcomputing* with tongue in cheek as, ". . . a pair of heroes in Colorado who have saved the day for all us TRS-80 owners." You couldn't have been more accurate. They were a pair of heroes. They did save the day. NEWDOS transformed my TRS-80 into a serious and reliable computer. The so-called hardware problems disappeared and, not only that, for the first time the user had access to the SYSTEM and the disks!

Originally NEWDOS was written only to make the system work, because those guys in Colorado with the TRS-80 were having the same problems as everyone else. The difference between them and all the rest of us "early users," was that they had the expertise to fix the DOS. The formation of Apparat was quite literally an accident. In those days there was no *80 Microcomputing* or other national magazine that even recognized the existence of the TRS-80. NEWDOS was then known as APRDOS and was only available from someone who already had a copy. You couldn't buy it! It was only the demand for an operating system that worked that literally forced the formation of Apparat.

Now back to correcting your misconceptions about my book. You say you expected to find in this book how the disk knows where to start and when to stop. I will refer you to chapter 6.0, page 36. There is a general discussion about the directory. Under "GAT SECTOR"—SECTOR 0, about two-thirds down the page, you will find a SUPERZAP display of the Granule Allocation Table. Following that there is a discussion of the GAT sector and its purpose. Briefly it explains, as you have put it, ". . . how the disk knows where to start and when to stop."

Following that is a discussion of the HIT SECTOR. This tells you how the system locates available directory space and does a quick scan of the directory to find a file. After that is the discussion of the actual directory entries.

Now, here is where I take great and severe umbrage over your letter to the editor. I do explain how, once again in your own words, ". . . the system knows where all the pieces of a fragmented program are located." In fact, I explain that in great detail. (See page 41.)

Now, about your last bitch in paragraph one: ". . . how a multiple disk system knows which drive to use." This is a function of the Level II operating system (nothing to do with the disk whatsoever) and the disk operating system code. Neither of these things have anything to do with the book's subject: the disk's codes, the disk's organization, the representation of files on the disk, and the methods used to recover all or parts of damaged files, disk codes and entries made to the disk.

The parting shot in your first paragraph says that you had hopes that I would have given hints on changing code on the disk so that you could change the "start-up-process." Once again, I must say that the program code *stored* on the disk is *not* the disk. The disk is a storage medium. The program code is stored on the disk. The code does not work *on* the disk—it works *in* memory.

Let's finish paragraph two. You refer to comments of mine about Radio Shack as ". . . the crums in Fort Worth" and the ". . . stupes who put TRSDOS together." I NEVER said that! I did take Tandy/Radio Shack to task for ignoring us, giving us bum dope, non-admission of operating system errors, and generally having a damn superior attitude.

TRS-80 Disk & Other Mysteries has around 43,000 words of text and, on the whole, I believe my comments and preachments about Radio Shack's software are in proper proportion. I would also point out that I only give them hell on their software, *not* their hardware.

You complain that I only crow about the errors in TRSDOS. On page 86 is a lengthy description of one of the major errors found in TRSDOS 2.1. It wasn't included in one of the original manuscripts of the book. In the first manuscripts, I merely said that there was a problem with the GAT sector's allocation code that eventually led to the disk being "clobbered" and made un-usable. My reviewers raised a great hue-and-cry that a detailed explanation was needed because of the severity of the problem.

Illustrated Problem

Here is an illustration of what I am talking about. Let's say you are developing a program that creates and uses files; maybe it is a payroll system. Each day you key-in and debug some code. Every night you back-up your disks, just to be on the safe side. You are using TRSDOS 2.3.

Now, you have been at the machine for several hours, coding, running and saving each change. During each program run, as you are debugging your program, your program creates files. As a test, you have to kill the files your program has created, in order to see if they are created properly each time the program runs.

You must remember that the files *are not closed every time you hit break or when a stop or end is encountered.* That means, that as you look at your video screen and its ready message, *your file is still open!*

"Now," you say, "I'll kill that payroll master file and the employee master, fix the bug in line 20000, and run the program again." You kill the files and, then, make your corrections, type RUN and ENTER. The disks mysteriously start. The drives (you have two, in this story) click a couple of times and then —DAMN—IT—the system hangs!

You curse, swear and snort. Oh well, you press the reset, DOS READY appears, and you go back to BASIC, load your program and continue working. "It must have been a power glitch," you say to yourself. "That's three times this week"

What you do not know, and will never know, is that TRSDOS 2.3 has been depositing garbage into the directory's GAT and HIT sectors each and every time you kill one of those files you forgot to close while you were developing the program. What is worse, every time you back up the disk, you are backing up the errors! Each error makes the errors that follow it worse!

Suppose that you finally get your payroll system developed. You back it up to a new disk, kill the old files that you do not need, and start using it. When the disk gets full, the system bombs! The files disappear from the system, and you can no longer access the data files on the disk.

"Thank God, I back up the disk every time I use it," you think to yourself. "I only have to type in this week's payroll again." Well, it will only take 30 minutes or so and you'll have everything recovered. Once again, you run the program. Once again, it

bombs!

"*%##### **** !!," you say. "Who the hell is zapping my power?"

The logical conclusions that you may arrive at are: a) the computer is not reliable; b) the incoming power is flakey; c) Apple is a better computer; d) your program has a bug that you cannot find; or e) you are being punished for some crime against the gods.

None of the above are true. The problem is the operating system and your own ignorance of what is happening *on* the disk.

In order for you to know about the subtle errors that are occurring, you will need a SUPERZAP type of program and the DIRCHECK utility or one like it. With DIRCHECK you would have detected those GAT and HIT errors. (If you want to know what GAT and HIT mean, and how they work, read pages 36 to 41.) With your SUPERZAP you would have the tool to fix the problem and with my book, you would have the information you need to fix it.

If you had put your program *and* the data on the disk in drive zero, you may have gotten the cryptic message, "NO SYSTEM" at some point. No amount of swearing and thrashing would have caused that disk to boot again. If you take your problem to a Radio Shack store or call Fort Worth, I'll bet you money that you will be told that you have a "hardware problem." If you take your machine in for service, they will undoubtedly find a problem, and you will pay for it. It is not out of malice that this is done. It is ignorance. The hardware people do not understand (as a rule) the software problems.

If you have started with an absolutely virgin disk (either in Drive zero or one) and had COPYed the files to the virgin disk, you may never encounter any of the problems I have described.

Test Program

Here is a helpful test program for those of you with both TRSDOS and NEWDOS:

```
10  REM ********************************
20  REM **              KILLTEST        **
30  REM **------------------------------**
40  REM **  A PROGRAM TO GENERATE DIR-  **
50  REM **   ECTORY ERRORS BY 'KILLING  **
60  REM **          OPEN FILES          **
70  REM **                              **
80  REM ********************************
90  REM
100 FOR X = 1 TO 10
110 OPEN"R",1,"TEST1"
120 OPEN"R",1,"TEST2"
130 OPEN"R",1,"TEST3"
140 KILL "TEST1"
150 KILL "TEST2"
160 KILL "TEST3"
170 CLOSE
180 NEXT
```

Follow the test procedure exactly or the results and conclusions will be tainted.

TEST PROCEDURE FOR THE "KILLTEST" PROGRAM
TEST SOFTWARE: 1. TRSDOS 2.3
2. NEWDOS 2.1 (with current ZAPs)

TEST PROCEDURE: (PART I)

1. Make a backup copy of TRSDOS 2.3. WRITE PROTECT the DOS backup. (We'll remove it in part II of the test.)
2. Make a backup copy of NEWDOS 2.1 WRITE PROTECT the DOS backup.
3. Format a diskette on Drive 1.
4. Run DIRCHECK and verify that the formatted disk is absolutely error free.
5. Boot-up the system under TRSDOS.
6. Type: 'BASIC' <ENTER>
7. Answer TRSDOS's MEMORY? and FILES? with <ENTER>
8. Type: RUN "KILLTEST" and <ENTER>. KILLTEST has a loop counter with X = 1 to 10. TRSDOS will hang and not complete the loop. You will have to repeat steps five thru eight to complete ten passes of the program.
9. After you have RUN KILLTEST ten times go to NEWDOS and boot-up (unless you have it on TRSDOS).
10. After DOS READY, type: DIRCHECK and answer its prompts.
11. You will now see a DIRCHECK's evaluation of the errors created on the test disk's directory. The following is an actual DIRCHECK output after running KILLTEST under TRSDOS.

```
KILLTEST 12/02/80
00  BAD EXTENSION ENTRY
00  EXTENSION ENTRY HAS BAD BACK IN-
    DEX CODE
C0  BAD EXTENSION ENTRY
C0  EXTENSION ENTRY HAS BAD BACK IN-
    DEX CODE
E0  BAD EXTENSION ENTRY
E0  EXTENSION ENTRY HAS BAD BACK IN-
    DEX CODE
01  BAD EXTENSION ENTRY
01  EXTENSION ENTRY HAS BAD BACK IN-
    DEX CODE
70  FREE GRANULES
KILL TEST DIRECTORY CHECK & LIST COM-
PLETED
```

12. Format a diskette on Drive 1.
13. Run DIRCHECK and verify that the formatted disk is absolutely error free.
14. Boot-up the system under NEWDOS 2.1.
15. Type: 'BASIC' <ENTER>.
16. Type: RUN "KILLTEST" and <ENTER>. KILLTEST has a loop counter with X = 1 to 10. Under NEWDOS 2.1 the system will NOT hang.
17. After KILLTEST looped ten times, go to DOS with CMD"S".
18. After DOS READY, type: DIRCHECK and answer its prompts.
19. You will see the following DIRCHEC output after running KILLTEST under NEWDOS 2.1.

```
KILLTEST 12/02/80
BOOT/SYS SIP=6 EOF = 5/0 1 EXTS 5 SECTORS
DIR/SYS SIP=5 EOR = 10/0 1 EXTS 10 SECTORS
  67 FREE GRANULES.
  KILLTEST DIRECTORY CHECK & LIST COMPLETED
```

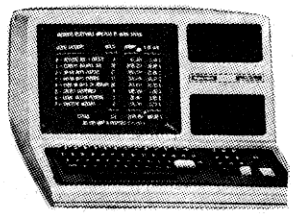
Conclusions Drawn from Part I

In each of the above DIRCHECK printouts, you will notice that the number of free GRANULES is noted. In the first direc-

MODEL II

$ DISCOUNT $
TRS-80® DEALER A301

MODEL III



COMPUTER SPECIALISTS

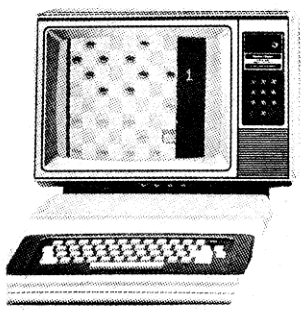
26-4002
64K 1 Drive
$3440.00

26-4160 1 Drive EXP.	$1035.00
26-4161 2 Drive EXP.	1575.00
26-4162 3 Drive EXP.	2115.00
26-4501 Gen. Ledger.	180.00
26-4502 Inventory.	180.00
26-4503 Payroll.	360.00
26-4554 Acct. Rec.	180.00
26-4701 Fortran.	270.00
26-1157A Daisy Wheel.	2290.00
26-1158 Daisy Wheel II.	1799.00

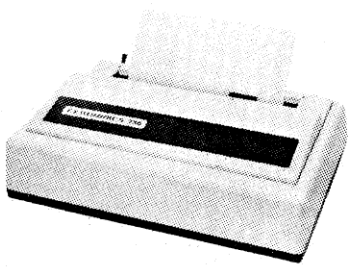
26-1155 Quick Printer II.	$187.00
26-1145 RS-232 Board.	84.00
26-1140 "O" K Interface.	249.00
26-1141 "16" K Interface.	359.00
26-1142 "32" K Interface.	469.00
26-1160 Mini Disk - Drive O.	419.00
26-1161 Mini Disk - Additional.	419.00
26-1154 Lineprinter II.	699.00
26-1165 Line Printer V.	1710.00
26-1159 Lineprinter IV.	859.00
26-1166 Line Printer VI.	1080.00
26-1563 Scripsit - Disk.	79.00
26-1566 Visicalc.	83.00
26-1562 Profile.	72.00

26-1061 4K I.	$630.00
26-1062 16K III.	888.00
26-1063 32K III 2-Drives, RS232.	2225.00

COLOR



26-3001 4K.	$360.00
26-3002 16K.	540.00
26-3010 Color Video.	360.00
26-1206 Recorder.	54.00
26-3008 Joysticks.	22.50



CENTRONICS

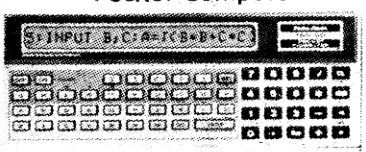
Fast 100 CPS Centronics
730 Printer. 577.00
Text Quality Centronics
737 Printer. 737.00

Model II Cobol Compiler $360.00
Cobol Run Time Package $36.00

ALL OTHER R.S. SOFTWARE FURNITURE, STANDS, CABLES AND ACCESSORIES AT DISCOUNT FROM CATALOG PRICE.

Novation Cat Modem. $149.00
CCA Data Management System. 72.00
Adventure Games Games 1-9 each. 14.00

Pocket Computer



26-3501 1.9K P.C.	$225.00
26-3503 Cassette I/F.	45.00
14-812 Recorder.	72.00

Acorn Software Products, Inc.

GAMES:

Alien Invasion	$9.00
Stock Market	9.00
Star Trek	9.00
Block Em	9.00
Ting-Tong	9.00
UTILITIES:	
System Savers	14.00
EDUCATION:	
Language Teacher	18.00

FREE: COMPUTER CATALOG UPON REQUEST

1-800-841-0860 Toll Free Order Entry

MICRO MANAGEMENT SYSTEMS, INC. ✓72

No Taxes on Out Of State Shipments

Immediate Shipment From Stock on Most Items

DOWNTOWN PLAZA SHOPPING CENTER
115 C SECOND AVE. S.W.
CAIRO, GEORGIA 31728
(912) 377-7120 Ga. Phone No.

R.S. 90 Day Limited Warranty
F-48 Form Provided

Largest Inventory In the S.E. U.S.A.

*TRS-80 is a registered trademark of the Tandy Corp.

EVEN SIGMON CRAVES A LITTLE SECS.

SIGMON

SIGMON is a complete machine Language development tool for your TRS-80 Color Computer and it's super powered 6809E microprocessor.

Features:

MONITOR: Display memory in HEX and ASCII, Direct HEX and Decimal entry into memory/registers, Move, Find, Tape Read and Write, Output to printer and more.

DISASSEMBLER: Display Addresses, Opcode operand, Mnemonics, and Address expressions for specific range of instructions.

MINI ASSEMBLER: Allows you to enter instructions in Symbolic form and have them converted to Machine Language.

DEBUGGER: Allows Stepping and Break point execution of Machine Language programs.

With SIGMON you can awaken the awesome powers that lurk within the dark recesses of your color computer.

Cassette and Source Code for the TRS-80 Color Computer $29.95

S.E.C.S.

(Screen Edit Control System)

SECS adds new power to your TRS-80 Color Computer. If you have been waiting for Enhanced BASIC and it's capabilities, then you are ready to take the next logical step.

Features:

FULL SCREEN EDITOR: Full Cursor control with auto repeat, Insert, Delete, Join lines, Relocate lines and Audible error warning.

HI RES GRAPHICS: Two modes of Hi Res graphics, Set point, Set line, Set color, Set programmable character, Set screen, and Toggle between Low Res and Hi Res/Normal and Inverted Screens.

HI RES CHARACTER GENERATOR: Sixty-four definable characters, Save and Load Characters sets to and from tape, Define Characters, and Place on screen in any color and in any position.

The entire system is linked into BASIC and is completely transparent to the user. As a BASIC programming tool, SECS is unsurpassed.

Cassette for the TRS-80 Color Computer $29.95

Software Utilities for the TRS-80 Color Computer

For further information about these and other Datasoft programs, visit your local software dealer, or call us.

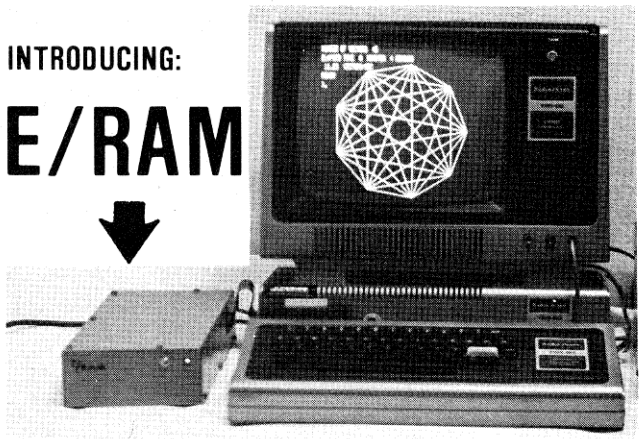
Datasoft Inc. T.M.
COMPUTER SOFTWARE

16600 Schoenborn Street * Sepulveda, Calif. 91343
(213) 894-9154 ↙407

HI-RESOLUTION GRAPHICS FOR TRS-80*

INTRODUCING:

E/RAM



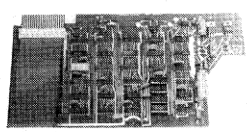
E/RAM Graphics is a unique hardware/software package, which will integrate high-speed, high resolution graphics into any Level II TRS-80 system. E/RAM hardware is a fully plug-compatible box, which installs in minutes, and requires absolutely **no** modifications to the TRS-80 system. E/RAM software is a compact, relocatable set of utilities which provides the user with easily accessible graphics functions. For instance: the user pokes the end point coordinates of a line into certain locations, does a USR call, and an optimized dot-raster line is automatically drawn on the screen at very high speed (less than 10 milli-seconds for a medium length line).

E/RAM does not require the purchase of an additional monitor CRT. The high-resolution graphics video is syncronized with the TRS-80 video and appears on the screen with the normal TRS-80 display. Alphanumerics, TRS-80 graphics, and E/RAM high-resolution graphics may be displayed simultaneously or individually.

E/RAM hardware contains its own 6144 byte video memory, which provides a true 256 x 192 matrix of **independent** graphic elements. (E/RAM is **NOT** a programmable character generator type graphics system. Character generator systems have serious limitations in full screen graphics applications.)

E/RAM will operate with or without an expansion interface, and with any standard memory configuration (4k through 48k).

E/RAM is **fast**. "E/RAM" is an acronym for **E**xtended **R**andom **A**ccess **M**emory, a very short description of the Patent-Pending method of I/O employed by this device, which gives it memory-mapped speed without interfering with the memory space used by the TRS-80.



The installation of E/RAM will not affect normal operation of the TRS-80. High resolution ON/OFF is under program or manual control (a switch is provided). An expansion card edge connector is provided so that other peripherals may be used on the TRS-80 bus.

E/RAM software package is compact (less than 1000 bytes), fast, easy to use, and very flexible. A relocating loader is provided. The user can delete unneeded routines if more memory space is required. Lines can be drawn as fast as 13 per second using BASIC USR calls, and as fast as 200 per second using assembly language programs.

Routines usable through USR of BASIC, and of course an assembler CALL are:

INIT	-	Sets up display
PLOT	-	Plots a point
READ	-	Reads a point from the screen
BLACK	-	Sets drawing mode to black (off)
WHITE	-	Sets drawing mode to on
CLEAR	-	Clears the high-resolution graphics screen
LINE	-	Draws a line

As an example, after the utilities package is loaded and you desire to draw a line, the following sequence of BASIC instructions could be executed:

U=USR(0)	Return the communications area
POKE U+1,X0	Provide the beginning X coordinate
POKE U+3,Y0	Provide the beginning Y coordinate
POKE U+5,X1	Provide the ending X coordinate
POKE U+7,Y1	Provide the ending Y coordinate
V=USR(4)	Draw the line (Current speed is approximately 13 vectors/second)

The complete E/RAM package is available for only $349.95, and includes case, power supply, cables, software cassette, and complete documentation.

To order, or for further details, write or call:

VERN STREET PRODUCTS THE COMPUTER STORE

We handle a full line of Radio Shack products ↙432

4949 South Peoria Ave
Tulsa, Okla 74105
(918) 747-2550

Send $10.00 for a set of the manuals provided (applicable towards purchase)
Dealer inquiries are invited.
Terms: COD Welcome, check, money order, Master Charge, or Visa
Delivery: Stock to 60 days.
E/RAM was designed, and is manufactured by KEYLINE COMPUTER PRODUCTS, INC. 13 East 6th Street, M/C 200, Tulsa, Oklahoma 74119.

*TRS-80 is a registered trademark of Radio Shack, a Tandy Corporation.

tory check it says there are 70 GRANULES free. Wait a minute. Arithmetic is needed here. TRSDOS has only 35 tracks and there are only two GRANULES per track. Seventy divided by two means that all of the tracks have free space. What on earth happened to the Directory and the BOOT/SYS program?

A quick look at the Directory with SUPERZAP will show you that the SYSTEM file area of the directory as well as other parts of it, has been wiped out (clobbered) with trash all over it. The test diskette used with NEWDOS 2.1 has no errors noted by DIRCHECK and an examination with SUPERZAP will confirm that none exist.

Conduct the test again. You will find, in all probability that TRSDOS 2.3 will make the errors *but they will be different errors* and will occur in different places in the directory. Not only that but the number of errors will vary considerably. If you continue to use the same disk without re-formatting it, you will be amazed at the number of errors.

PART II OF TEST

1. Remove the write protect tabs from both DOS disks.
2. Repeat the procedure outlined above, only now the program will OPEN, KILL and CLOSE on the SYSTEM disk in Drive zero.
3. How many times the TRSDOS 2.3 disk will function will depend on how lucky you are. If it succeeds in making enough errors, the program will terminate with a "SYNTAX ERROR," or it may hang. When you BOOT nothing will happen. In any case, after ten OPENs, KILLs and CLOSEs, BOOT the system. Ninety-nine times out of 100, nothing will happen. The video will look at you with a very blank stare. If you are very lucky, it will at least say, "NO SYSTEM" or "DISK ERROR."
4. Repeat the test with NEWDOS 2.1. Make it really tough. Change the loop from 10 to 100.

Conclusions Drawn from Part II

Killing open files is a bad practice with anybody's DOS. I don't recommend it. However, a good DOS should be able to detect that the kill is being issued before the close and take the proper steps to intercept and correct the condition before the data is wiped out.

I have run the above test over 200 times and not once has NEWDOS wiped out the directory or any parts of it! TRSDOS deallocates bytes in the GAT table, does not clean up the HIT table and deposits garbage in the DIRECTORY sectors. It is still the same primary problem I describe on page 86 which was taken from the Apparat documentation.

So, not even TRSDOS 2.3 solves the problem completely.

Final Conclusions

1. The "guys in Denver are a couple of heroes."
2. TRSDOS 2.3 is worth exactly $14.95.
3. I should have given NEWDOS more plugs in the book.
4. TRSDOS 2.3 is OK for trivial programming tasks.
5. TRSDOS 2.3 is OK for running games.
6. You don't use your system very much.
7. I will not be loved in Fort Worth, especially now.

The story preceding the test is not something I made up. The story is a composite of dozens I've heard, if not hundreds. I have seen it time, after time, after time. I have even received programs from well known vendors with errors built right into the disk directory!

The Third Paragraph

Paragraph three, John, is where you really put your foot into it. You say, "Most of the rest of this book tells us about the mistakes in Radio Shack's TRSDOS and the inadequacies of some of the other disk systems and how great NEWDOS is. We learn that NEWDOS works and fixes all the mistakes. The book then describes how to use NEWDOS commands and features." I won't quote the rest of it but I'll damn sure take you to task on this!

Apparently you don't read very well or carefully. Perhaps your comprehension is impaired, or worse, English is your third language. SUPERZAP is not, repeat, NOT part of the NEWDOS operating system. It is a utility that will run under TRSDOS 2.1, TRSDOS 2.2, TRSDOS 2.3, VTOS 3.0, VTOS 4.0, NEWDOS 2.1, NEWDOS/80, DOSPLUS 3.0 and Level IV DOS.

I devoted one chapter to the commands and functions of SUPERZAP, *not NEWDOS.* When the book was written, SUPERZAP was the only utility devoted to the reading and repairing of the disk. It was not *one of many,* but the *only* one.

Additionally, there were a couple of other utilities that were of some value for anyone needing to examine the disk or recover data. These programs are also discussed in that same chapter. The most valuable of these utilities is DIRCHECK, also by Apparat. As far as I know, this is still the only utility of its kind. Let me repeat: These utilities will run on any DOS, including TRSDOS 2.3!

Now for my final reply: *Nowhere* in the book is there one single line of text that is documentation for the NEWDOS operating system! With the exception of chapters one and two, the book is devoted to the disk's organization, data recovery, the disk's codes, and other helpful utilities.

H. C. Pennington
Upland, CA

Some DOS facts for the uninitiated.

The Disk DOSsier

by Jake Commander

What is a Disk Operating System? If the whole magazine were available, plus another two, then it might just be possible to scratch the surface of this fascinating subject. They are comparatively simple or complex, depending on the size of the computer they control.

The sort of operating system you find on an IBM or Honeywell machine takes man-years to write, and is a highly complex set of intermeshing pieces of software. These monstrous systems control a multitude of functions such as looking after timesharing users who want to access the main frame computer from a remote terminal via a telephone line. These users must be logged-on to the main machine, their accounts updated, and then slotted in with others who will all be competing for the central processor.

The operating system will also automatically schedule batch work. Batch work consists of separate jobs submitted to the computer one after another, such as payroll, invoicing, sales ledger and the like. These programs can be input to the computer using punched cards and would probably include a number of related programs running one after another (often referred to as a suite). The operating system will read all the input from cards or other media and, after determining the job's priority, allocate it in a queue.

Also to be taken into account will be the job's memory and I/O requirements. Is enough memory available to run the program concurrently with other programs which may be executing? Will the printer be required? Can the printer output be sent to a disk to be printed later when the printer is available? Using the limited resources of the main computer in as efficient manner as possible is sometimes known as spooling and is another function of the operating system.

This batch mode of operation is most like a TRS-80 system, except that input comes from keyboard, tape, or disk. So far, none of the above may sound much like your familiar TRS-80, but as we progress through the layers of a giant operating system, we'll start to see and recognize some familiar functions.

Time Slices

Remember those remote timesharing users? All of them might be using the resources at the same time as the batch jobs are running. New users are given special priority by means of a time-slice in line with all the other remote stations.

These time-slices are usually the result of setting a hardware timer, and allowing the user a few milliseconds of the computer's valuable time. Although this may seem like rather a small amount, remember that he may receive his few milliseconds many times a second. Thus, to that particular person he seems to have constant use of the computer. These time-slices are terminated either when the user requires an unavailable resource such as a disk file, or when his time is up, in which case a hardware interrupt occurs.

Hardware interrupts give us our first parallel with the TRS-80. The DOS on our tiny micro is unlikely to have to worry about multiple users, so interrupt tasks will be limited to minor jobs like updating a real-time clock, displaying a program counter location, or scanning to see if a particular key is pressed, in which case a particular course of action will be followed. This is how Debug gets control if Break is hit during the execution of an independent machine code program.

The number of interrupt tasks that are performed is usually definable by the individual user. You may not want a real-time clock, but your DOS may perform a special action if it receives an interrupt which it recognizes as an impending power failure. On the largest main frames this sort of interrupt is given the highest priority of all, and causes of spontaneous dump of the computer's memory on to disk. This is usually done in milliseconds which means that all data is completely saved by the time power actually fails.

One of the most important functions performed by any of the disk operating systems for the TRS-80 is control of I/O. For instance, anyone who's programming in BASIC doesn't want to worry about whether the disk in any drive has a locked-out track. He just wants to read or write a file. This is where our trusty DOS comes in.

Once a BASIC programmer says, "write that information to my file" in whatever dialect of BASIC he cares to say it, the DOS takes over. Firstly, the DOS must know the physical location of the file from the disk directory, make sure the file is open and a data control block available. The data control block or DCB tells the DOS various details about any I/O device (in this case a disk).

The programmer can be completely oblivious to details without which the DOS is lost; for example, which disk contains the file? How big is the file? Is the file split into separate segments (referred to as extents on the TRS-80)? Where are we pointed to in the file—the beginning? The end? Are we attempting to read more than was actually written to the file? All these questions must be answered by the DOS.

If any error occurs, then the person operating the computer should have as many options as possible.

This is one of the areas where the available DOSs fail miserably. If, for example, you get a read error on a disk, you're usually stuck. Imagine someone is running a payroll program and is reading employee details from disk. A hard read error occurs due to a scratch on the surface. What does the DOS do?

By current standards, the DOS will probably go through a sequence of attempts to read that sector, over which the operator has little or no control. This sequence is usually something like this: Set the offending drive to track zero (in case a seek error caused the head to be improperly posi-

'One of the most important functions performed by any of the disk operating systems for the TRS-80 is control of I/O."

tioned); Try another read—no good—try another seek, and another read, and so on. This may go on for between ten and twenty attempts, while the operator watches helplessly. Eventually the program will drop into his lap with a message such as "DATA RECORD NOT FOUND DURING READ."

A complex operating system would now ask for human help. Does the operator want to try all those seeks every time the sector is re-read? Probably not. If you have read error on track 40, the last thing you may want is to watch twenty times as the disk oscillates between track 0 and track 40 all to no avail.

But were you asked? Did any special error recovery routine come in to RAM and give you any choice? Not likely.

The options offered on a main frame would include changing the disk, skipping the sector or re-trying the I/O. The attempt to find the error would never be abandoned unless the operator specifically requested it.

Most DOSs available for the TRS-80, show no such persistence. Certain DOS's for the 80 can tailor your system to your needs. A DOS may know the number of disks you have, the number of tracks per disk, track stepping rates, etc. All of which optimize the use of your own particular configuration.

Another DOS utility redefines a device. You could redefine your printer in the event

How Your TRSDOS Works

In the beginning, all is darkness. The Z-80 microprocessor lies dormant in the TRS-80 keyboard unit. Hundreds of electronic logic gates are waiting. Power up, and it springs to life.

The whole purpose of a central processing circuit is to step, byte-by-byte, through memory, picking each byte and asking, "what am I to do next?" Each byte represents a unique instruction to the central processing unit, a portion of which is dedicated to decoding each byte encountered in a special instruction register. The decoding circuitry guides the CPU chip through an electronic sequence of internal events which result in that particular operation code (opcode) being carried out.

The Program Counts

The actual byte executed is fetched from memory at a location which is pointed to by another special register in the CPU. This register is appropriately named the program counter or PC register. Each time an opcode byte is decoded, the program counter is automatically incremented to point at the following byte in memory.

Thus, the chip steps through memory, executing each byte pointed to by the PC register. If the chip is instructed to jump somewhere else, then the program counter is loaded with the value of that "somewhere else" and continues executing from there.

All that remains is to know which byte performs which operation in the CPU, arrange them in a sensible order and we have a machine code program. This is what we have in the ROM.

When the Z-80 is awakened from its slumber, its first inclination is to execute the opcode that's pointed to by the PC register. Unfortunately, this register will have a totally random value because of its power up state. This would cause the computer to go careering at breakneck speed through a random sequence of operations.

The designers of any chip have to make provisions to control this initialization. In the TRS-80 this is accomplished by holding the CPU in a reset state (doing nothing) for a few hundred milliseconds until a small capacitor charges up and releases the chip. The reset condition helps out by setting the PC register to zero, so the chip does what it was built for and executes the byte at memory location zero which just happens to be ROM. The ROM is, of course, pre-programmed, so we're at last on our way to the DOS. Now that we have absolute control, things happen in a nice orderly fashion.

The CPU does a few housekeeping tasks to prepare the RAM for later use. Then it scans to see if the Break key is pressed. If it is, then Level II BASIC is initialized and the familiar MEMORY SIZE? message is displayed. If no Break key is pressed, then it checks to ascertain whether or not a floppy disk controller chip exists in the expansion interface. If no interface is connected, then no controller chip exists, in which case the MEMORY SIZE? message is displayed.

Whether or not a disk is available, instructions will be exeucted from ROM which tell the floppy disk controller to set drive zero to track zero and read the first sector. This is the bootstrap sector on the disk and is always read into memory starting at location 16896. The routine to do all this is contained in 42-byte section of ROM which has now performed its duty, so the CPU is told to jump to location 16896. This location was just filled during that sector read, so the program now executed will be as determined by the first sector on the disk.

The Beginning

If a TRSDOS disk was just booted in, the story has just begun. The sector now in memory is 256 bytes long in RAM, an improvement over a 42-byte ROM routine. The first action of this freshly loaded routine is to clear the screen. If this much happens, then you at least know the bootstrap sector was read.

When this bootstrap sector was written to the disk by the FORMAT or BACKUP program, the third byte was set to a value which tells the DOS where the directory track is on the disk. This byte now tells the bootstrap loader where to look for SYS0. The directory entry for this DOS executive program is always assumed to be in sector 4 of the directory. This entry tells where on the disk to look for SYS0, so after extracting this information from the directory, the executive is loaded.

The SYS0 program is in the same standard format as any machine code program on disk, and contains bytes of information which tell the loader where in memory to load it. A special byte then signifies an execution address to the loader, and the load process is terminated. This execution address is now loaded into the PC register and SYS0 takes over.

The first thing SYS0 does is to check

"The whole purpose of a central processing circuit is to step, byte-by-byte, through memory, picking each byte and asking 'what am I to do next?' "

that it is unavailable. If you have an important printout to make, this can be really useful. The DOS would redefine your printer to be a disk file, in which case all printer output would go to the disk to be printed on another machine.

Finally, a DOS may offer a number of general purpose utilities to help run your system that much easier. These would include tape to disk transfer, program relocation, debugging and diagnostic utilities. These are extras, however, and not really part of the DOS itself.

All of these functions and utilities add up to a lot of machine code. On a TRSDOS system disk, 34K is required for the various DOS routines. Obviously, this would not fit into a 16K or 32K system. By using an executive DOS routine in a small amount of RAM, all the necessary DOS functions can be called into memory from disk. Thus, the operating system becomes a system of overlays, all under the central control of the executive. This is why a system disk is always required in drive zero—that's where the executive looks and expects to find any module it feels necessary to call.

This all-important central module is the SYS0/SYS on a DOS disk and is loaded only once when the DOS is booted. Incidentally, that term "boot" comes from the fact that when the computer is initialized, a very simple routine reads in a more complicated routine which, in turn, reads in an even more complicated routine, the executive itself. The executive will then load in the overlay which says DOS READY or whatever, and await operator input. Thus, the software, it can be said, has picked itself up "by its own bootstraps."

DOS for the 80

The development of disk operating systems for the TRS-80 has made a fascinating scenario. The original DOS was written for Radio Shack by Randy Cook in Texas. He performed the gargantuan task of pulling together the keyboard, expansion interface and disks via TRSDOS. The original version was reputed to be TRSDOS 1.0, but no one seems to be prepared to actually admit its existence.

TRSDOS 2.1 was all too real. When Radio Shack released this version, they released a swarm of bugs onto the market. According to Cook, he hadn't finished the project.

With this wonderful first attempt, many of us pioneering users found ourselves in the grips of a system whose main penchant was forgetting to keep the drives rotating! This incurred what was to become known as "silent death." All one could do was reboot the computer and lose any BASIC program that was in memory.

TRSDOS 2.2 came to the rescue. This was an updated version from Randy Cook and cured quite a few of the former bugs.

Randy Cook and Radio Shack had a falling out about this time and the history of TRSDOS becomes hazy.

Radio Shack promptly released TRSDOS 2.3, which purported to be an upgrade from 2.2. However, the main change seemed to be the removal of all Randy Cook's copyright notices throughout the disk. They were replaced by Radio Shacks own.

While all this was going on, other things were afoot in Colorado. A couple of programmers took TRSDOS 2.1 plus attendant bugs, and thoroughly rectified most of them. To aid them in this task, they wrote some useful utilities such as a disassem-

---

the amount of memory in the system, preset various RAM addresses, and arrange for handling interrupts. The TRSDOS title is then printed and finally, a check is made to see if an AUTO command file is specified in the first sector of the directory. The portion of SYS0 that performs the initialization just described is only needed once, so it is loaded into an area of RAM which is used by DOS for transient module overlays.

The rest of SYS0 contains the code for determining which overlay is to be loaded at which time, and where to find its entry in the directory. It also contains such things as the interrupt task routines, keyboard debounce routines, and any other software which is always needed in RAM and thus, is not a suitable candidate for overlaying. SYS0 now loads the AUTO command file, if specified, or calls SYS1 into the overlay area.

The reason for using overlays is simple. All told, the operating system disk contains around 34K of machine code. This doesn't leave much room in a 48K machine. Not only that, but it's not necessary to have all that code in memory at one time. For instance, if you are opening a file, it's obviously not necessary to load any code which deals with killing or closing a file. Thus, these modules are written as separate overlays and only loaded when needed.

Using small modules saves memory, but saves it only at the expense of the time required to read the modules. This overlay system dictates that they must always be available, and the operating system expects to find them in drive zero. This explains why single drive users can't use disks that don't contain a DOS, thereby forfeiting 34K of usable disk space. After SYS1 is loaded, it is executed at an entry point which prints DOS READY.

SYS1 is the module which interprets keyboard input. If a command is typed, such as DIR or FREE, this is recognized as a library command, and the library module (SYS6) is loaded into a special overlay area, leaving SYS1 intact. The library function is then carried out. This may involve using other overlays, which is why the library module loads into its own special area.

If SYS1 doesn't recognize the keyboard input as a library command, it assumes that a machine code command file was specified and attempts to find it in the directory. It does this by appending a CMD extension as a default to the keyboard input and using it as a file specification. So, if you type BASIC, SYS1 will look for a file called BASIC/CMD. If it finds it, SYS1 relinquishes control to the DOS executive (SYS0), which opens the file using SYS2.

When the file is successfully loaded into memory, and its entry point determined, the program is executed. In this case Disk BASIC asks, HOW MANY FILES?

Now you can load a BASIC program file using, once again, SYS2 to open the file, SYS0 to load it and, if a disk error occurs, SYS4 to process the error. The amazing thing about all this is that it's still faster than loading a tape!

The modules contained on a TRSDOS disk are as follows:

BOOT/SYS—Bootstrap loader. (Only executed at startup).

SYS0/SYS—Resident TRSDOS I/O routines and executive.

SYS1/SYS—Command interpreter.

SYS2/SYS—OPEN, INIT, or EXTEND a file.

SYS3/SYS—CLOSE or KILL a file.

SYS4/SYS—Disk error message handler.

SYS5/SYS—DEBUG module.

SYS6/SYS—Library utilities such as APPEND, ATTRIB, etc. ■

by Jake Commander

When DecisionMaster™ speaks everybody listens.

Let's face it. We all have to make decisions. Decisions that can change our lives. Decisions that can make us happy or unhappy. Decisions that could win us fame or fortune. Now, DecisionMaster can help you make the best decisions of your life.

Use Bayesian theory to peer into the future...even if you've never heard of the Bayes' Rule. Do a complete weighted factor analysis...without knowing what one is. Use discounted cash flow to compare investment alternatives without bothering with present value tables. These and other sophisticated theories that were once the exclusive domain of professors and top business executives are *built into* DecisionMaster's algorithms... so you can use them at the touch of a key!

DecisionMaster is easy to use. It features:

- A fully documented manual developed by an authority in the field.
- A unique program-controlled cross reference system.
- A powerful formatted-screen data entry system.

You'll use DecisionMaster in hundreds of routine decisions, as well as more important ones such as • Buying a house • Changing jobs • Selecting investment • Evaluating insurance policies • Expanding product lines • Leasing vs. purchasing.

If you buy only one computer program this year, make it DecisionMaster. And when it speaks, *listen*.

DECISION?MASTER™

I-302

To order, see your software dealer or return this form with your check to: DecisionMaster, Dept. 80, 10428 Westpark, Houston, Texas 77042 Add $1.50 for shipping and handling

Diskette for: ☐ Apple II* (32K) $29.95 ☐ TRS-80** (Level II-32K) $29.95 ☐ TRS-80** (Model III) $39.95

Name

Address

City State Zip

CHARGE by phone toll-free: **1-800-231-5768 Ext. 306**
(In Texas: 1-800-392-2348) or return this form:

☐ VISA ☐ MasterCard Bank Code

Account No.

Signature

Expiration Date

*Apple II is a registered trademark of Apple Computers, Inc.
**TRS-80 is a trademark of the Radio Shack Division of Tandy Corp.

bler, enchanced editor/assembler, and disk zap utility. This whole package was then made available as NEWDOS, or NEWDOS + if you purchased the utilities.

This lifted a few heads in the industry. Would Radio Shack sue? After all, it was their upgraded TRSDOS. Would Microsoft sue? Apparat survived by pointing out that to use their system the purchaser had to buy TRSDOS to have any documentation. Thus no-one was robbed of any royalties. Besides, theirs was the only DOS that worked properly.

Randy Cook was not idle. He formed his own software company, Virtual Technology, and had written VTOS 3.0. This was probably the direction TRSDOS would have taken had Radio Shack and Randy Cook remained friends. Anyone who has disassembled TRSDOS has seen the small pieces of unfinished code that point to this fact. However, Randy made a small marketing error with VTOS 3.0. He'd watched the increasing frequency with which illegally copied disks were circulating and decided to give his VTOS some backup protection. He did this by making it impossible to back up a copied disk. Thus, only the purchased master could be used to make backup. A lot of potential customers were scared off by this, so its success was somewhat stifled.

Meanwhile, Apparat pursued its own success. They had completely rewritten NEWDOS, removing any possibility of copyright transgressions as it now contained only Apparat's original code. (Incidentally, if you haven't already seen the proof of the NEWDOS 2.1 pudding, boot up, type BOOT/SYS. WHO, and then hold down the 2 and 6 keys. You will be shown.) This NEWDOS 80 was assured of success because of the justifiably high reputation Apparat had gained with NEWDOS 2.1.

Randy Cook struggled onwards. Out came VTOS 4.0 which even outshone 3.0. A superb operating system, at this moment, VTOS 4.0 probably offers more features than any other DOS.

Entrepreneurs have been watching the DOS debate with eyes wide, propped open by dollar signs. TRS-80 users now have DOSPLUS, ULTRADOS, OPSYS and probably more operating systems to choose from, all offering their best attempts to be better than the last. From all reports, NEWDOS 80 appears to be doing well. TRSDOS 2.3 appears to be standing still, VTOS will soon become LDOS from Lobo Drives International, and the merry-go-round continues onwards. ■

EXCITING NEWS FOR INVESTORS WITH TRS-80 32K DISK-BASED SYSTEMS!

Standard & Poor's proudly announces STOCKPAK, a unique software and data system to help you meet your investment goals like a Wall Street professional.

STOCKPAK not only delivers a "stand-alone" Portfolio Management System but also gives you the software for Standard & Poor's monthly Common Stock Data Service (available to TRS-80 owners on a subscription basis). With STOCKPAK and the Data Service you command one of the most powerful and versatile investment tools available.

Here's How STOCKPAK Will Help You:

A 900 COMPANY DATA BASE SERVICE

Monthly Data Service subscribers receive a diskette containing 30 vital financial items on 900 of the most widely traded stocks (S&P "500" and 400 NYSE, ASE and OTC issues). Accompanying this monthly diskette is an Investor's Newsletter highlighting important financial news and investment strategies, with suggestions for maximizing the usefulness of the system.

STOCKPAK SELECTION SYSTEM

The heart of STOCKPAK is a powerful, analytical stock selection tool which enables investors to choose stocks which meet their investment criteria. For example, you may wish to select only those oil and gas stocks with price/earnings ratios of less than 7 and yields of 6% or more. Once a group of stocks has been selected, you can store it as a separate data file for continuing use.

REPORT WRITER

You can define the report formats you would like to see on those stocks meeting your investment objectives. Hundreds of calculations and ratios that you define can be sorted, averaged or totalled, and displayed on video screen or optional printer.

PORTFOLIO MANAGEMENT SYSTEM

Now you can effectively evaluate and manage your own stock portfolio of up to 100 securities with as many as 30 transactions for each. You can record "buy" and "sell" transactions, price and dividend information and stock splits for instant retrieval, for record keeping and tax purposes. You can measure actual performance or create hypothetical situations to help you make "buy" or "sell" decisions.

HOW TO ORDER STOCKPAK

STOCKPAK is designed exclusively for TRS-80 users with 32K business systems with two mini-disk drives. You can obtain the basic software and sample Data Base, plus a comprehensive User's Manual from your local Radio Shack Store for only $49.95. The STOCKPAK Monthly Data Updating Service can be ordered directly from Standard & Poor's for $200 annually, or from the order form provided in the basic package you purchase from Radio Shack.

Standard & Poor's Corporation
25 BROADWAY, NEW YORK, NY 10004 (212) 248-3994/3374

A side by side comparison of five popular operating systems.

DOS Talk

Stewart E. Fason
John Burgan
115 Via La Selva
Palm Beach, FL 33480

Remember that feeling of power and euphoria after you upgraded from Level I 4K to Level II 16K? A new world awaited you. Things loaded twice as fast; you could edit lines and acquired many new powerful commands.

You were satisfied for awhile, until someone showed you a disk system. Wow! Sargon loaded in seven seconds. That convinced you. You started figuring how much it would cost to upgrade—at least $800.

Initially, you spent about $500 for your 80, but now you are approaching two kilobucks. It's been well worth it though. What once was a nifty little machine that played blackjack turned into a giant, limited only by your imagination and pocketbook.

Disk Investment

You found that your machine would need a Disk Operating System (DOS) from which to take instructions. The standard DOS for the TRS-80 is Radio Shack's TRSDOS. The original TRSDOS 1.0 was weak and full of bugs. Randy Cook wrote it, and rewrote it, again and again, until he got it right. Today's TRSDOS 2.3 works exactly as the manual indicates—first time, every time. The operating manual that comes with it is typical Radio Shack documentation—outstanding.

During the last year or so, there has been a plethora of ads extolling the virtues of new DOSs. These ads tell us that for an additional $50 to $150 we can have an operating system that will do many new and wonderful things. Although those new capabilities alleged by vendors look interesting, how is one to know whether these new systems are really worth the additional investment?

The answer we reached, after hundreds of hours, is a qualified "perhaps!" An operating system is like an automobile; one's worth depends on your needs. The following is an examination of the DOSs currently on the market.

This article is a collaboration between two very different people. John Burgan, a bright engineering student, deeply into his computer, is a first class nitpicker/bugfinder. Stewart Fason is a successful businessman (bourgeois, robber baron type), who rates a simple "automatic break key disable" feature higher that any sophisticated zapper—for his own protection. The ideal system to Fason is one which can be mastered by a secretary without sending her to computer school. Burgan the serious programmer, finds versatility of paramount importance. Features which have great appeal to Fason may just cause Burgan to yawn.

First, however, a word of caution is in order. All vendors advertise their systems to be TRSDOS upward compatible. Don't be misled into believing that means all programs which run on TRSDOS will run on their DOSs. We found that over 95 percent of the programs made for TRSDOS were 100 percent compatible. If you have a program that is critical to your application, find out if it is supported by the DOS you are considering before spending money.

In discussing the relative merits of the advanced DOSs, we shall discuss only those features which we found to be potentially useful. They may be simply an enhancement of an existing TRSDOS feature, or they may be totally new (vis-a-vis TRSDOS).

The speed ratings in Table 1 may seem of little consequence because, in any DOS, things happen relatively fast. For any given operation, speed doesn't matter. However, one should recognize that after 1000 or 10,000 operations speed can be quite meaningful. The savings of wear and tear on your drives and disks will add up over the years.

The operating systems that we tested are in alphabetical order:

DOSPLUS, from Micro Systems Software, Hollywood, Florida
NEWDOS +, from Apparat, Inc., Denver, Colorado
NEWDOS-80, from Apparat, Inc., Denver, Colorado
ULTRADOS, from Level IV Products, Inc., Livonia, Michigan
VTOS 4.01, from Virtual Technology, Dallas, Texas

The features which we have reviewed are included in the chart in Fig. 1.

"In discussing the relative merits of the advanced DOSs, we shall discuss only those features which we found to be potentially useful."

DOSPLUS

DOSPLUS includes several features that make it less troublesome to use than TRSDOS. For example, when it is booted in, it automatically checks for a lowercase display, and enables a lowercase driver routine if it finds one. It also includes a repeating keyboard that repeats the last key that is pressed if it's held down for more than one-half a second. DOSPLUS also has a screen print feature that sends the contents of the screen to the printer when shift/clear are pressed.

DOSPLUS is also user proof. That is, if you ask for DIR :5, or DIR :1 without a disk in the drive, the system won't hang up, but will come back with a DEVICE NOT READY instead. The printer is similarly supported. No more calls from your secretary saying, "It froze up, what do I do?"

Several new commands are included in this DOS, some of which are useful, and some which just seem to take up extra space.

BOOT reboots the system and reinitializes the DOS on command. The reset button is easier.

BUILD constructs a file containing a sequence of commands to be executed with a Do statement.

CLEAR places a file in the directory and sets aside a specified amount of space for it.

DO executes a sequence of commands entered by the Build command. This can also be used when booting up with the auto feature.

FORMS sets the printer driver to a specified number of lines per page, characters per line, and page length.

FREE displays a free map similar to the Model II and III computers. This looks nice but it's not too useful.

PAUSE is used in Do files after prompting the operator to input a disk, etc. It pauses until the enter key is pressed.

RS232 displays, but does not set the RS232 board baud rate. It is basically a reminder function for those of you with bad memories.

In addition to the commands, several utilities are included:

PURGE allows the user to kill a group of files with one command, by displaying each name and asking for approval to kill it. This is an easy way to get rid of useless files quickly. If you accidentally Purge the wrong file, Restore brings it back from the dead.

DISKZAP is a sector oriented disk modification. It allows the user to examine and modify disk sectors and patch programs. It works well and is easy to use.

CLRFILE zeros a file, but does not change its allocation in the directory. Big deal?

COPY 1 is a single drive copy utility. It allows the user to copy any disk file that will fit in memory with a single drive system. This is a must for one-drive systems.

CRUNCH is a space compression utility. It removes spaces and REM statements from BASIC programs.

TRANSFER copies all programs from one disk drive to another. A real timesaver.

SYSTEM (DOS)	TRSDOS	NEWDOS +	NEWDOS-80	ULTRADOS	VTOS	DOSPLUS
Price (List) $	$17	$99	$149	$119	$99	$99
Doc. Completeness	EXCEL	GOOD	EXCEL	EXCEL	GOOD	FAIR
Documentation Clarity	EXCEL	FAIR	EXCEL	GOOD	FAIR	FAIR
Command Simplicity	EXCEL	FAIR	FAIR	GOOD	FAIR	EXCEL
*BASIC Program Load Time	8	4.5	3.5	4.1	9.5	4.1
CMD File Load Time	3.2	3.5	3.4	3.5	7.2	3.4
Relative Speed	5	2-3	1	4	6	2-3
Repeating Keyboard	NO	NO	NO	NO	AUTO	AUTO
Single Drive Copy	NO	NO	CMD	NO	CMD	UTIL
Purge	NO	NO	CMD	UTIL	CMD	UTIL
Transfer	NO	NO	CMD	UTIL	CMD	UTIL
Editor/Assembler	NO	UTIL	UTIL	UTIL	NO	NO
Disassembler	NO	UTIL	UTIL	UTIL	NO	NO
Machine Lang. Offset	NO	UTIL	UTIL	UTIL	NO	NO
Device Not Ready Ck.	NO	AUTO	AUTO	AUTO	AUTO	AUTO
High Speed Mod Support	NO	NO	AUTO	NO	AUTO	AUTO
Disk Modification Utility	NO	UTIL	UTIL	UTIL	NO	UTIL
Automatic Patch Utility	NO	NO	NO	NO	CMD	NO
Device Routing	NO	NO	NO	NO	FULL	PARTIAL
DO/Chain	NO	NO	CMD	UTIL	CMD	CMD
Lowercase MOD Support	NO	NO	UTIL	AUTO	AUTO	AUTO
Print Spooler	NO	NO	UTIL	NO	CMD	NO
RS232 Driver	NO	NO	NO	NO	UTIL	CMD
System Size (Grans)	14	17	23	17	34	17

UTIL = DOS Utility—Separate program from operating system. Can be easily removed from disk to recover that space for the user.
CMD = DOS Command—Function built into the DOS system files (/SYS) that can't easily be removed to recover that file space.
AUTO = Automatic—Feature that activates itself automatically without intervention of the user, usually on powerup.

Table 1. Comparison of Operating Systems

"The feature of the BASICs are identical to the TRSDOS BASIC, with a substantial savings in memory."

DOSPLUS rates high for simplicity. The manual is okay, but should go into more detail with some sample program applications for its special features. For example, the manual only briefly mentions one of DOSPLUS's dynamite features. It is the Auto-Do-Build function which can be used to allocate tasks to the computer when booting up.

When your secretary powers up in the morning, the following happen automatically: Check data file and print out appointments for the day; print any special dates within the next two weeks which may have been forgotten; phone the company computer in New York to see if any messages are keyed for you; and then, disable the break key. All this with no operator input. This one feature is well worth the price.

We found DOSPLUS to be a user's DOS. Serious machine language programmers will probably look elsewhere.

NEWDOS+

NEWDOS+ is basically a reworked TRSDOS 2.1 operating system with several new features and utilities added. We found this system to be basically bug-free. However, if any do appear, Apparat has included a disk modification utility, so the user can apply patches if they are necessary. New features and utilities added include:

COPY/BACKUP utilities are functions which are combined into one file. It seems to run more efficiently and use less room on the disk this way.

DOS functions are available from BASIC and return to BASIC after execution, without losing any of the RAM normally available. Also included in BASIC is an Open E option to add to the end of a sequential file.

JKL sends the screen contents to the printer.

DIRCHECK is a directory check and verification of utility. It lists any errors in the directory.

LMOFFSET utility loads the module offset program and allows the user to put machine language programs on the disk that normally conflict with DOS memory.

EDTASM is the Editor/Assembler from Radio Shack which has been modified to load and save programs to disk.

SUPERZAP/BAS is a disk sector oriented modification utility written in BASIC. It allows the user to patch files on disk.

LEVEL 1 Level I BASIC relocated into Level II RAM. This is not a Level I to Level II conversion routine, but actually Level I BASIC.

DISASSEM is the Z-80 disassembler program.

LV1DSKSL utility is a Level I disk save/load routine. It allows the user to save a Level I BASIC tape onto disk and load it into a Level I program.

NEWDOS-80

In the NEWDOS-80 documentation Apparat suggests you spend several hours studying the manual before touching the diskette containing the system. Many hours will be necessary before you fully understand its capabilities.

NEWDOS-80 is one of the more powerful DOS systems. It has a completely rewritten BASIC that is upwardly compatible from TRSDOS BASIC, but adds so many new types of data files that we have not yet been able to use all of them. It seems that this version of BASIC was written to make it easy for the business programmer to do anything.

The entire operating system appears to have been written from scratch, and seems to have very few bugs. Though it claims to be upward compatible from TRSDOS, we found some programs that wouldn't execute directly with it. Apparat seems to have patches to make them all work, either by modifying the program or the DOS.

The DOS is more versatile, but more complex than the others; anyone closely following the instructions, however, should have little trouble understanding the commands. This DOS also supports eight-inch drives, and the Lobo Expansion Interface.

The new features added to the DOS are:

BASIC, which supports the DOS commands with return to BASIC, as well as new file structures, including variable length records, add to end of sequential file and many others.

CHAIN executes a series of commands entered in a Do file.

HIMEM sets the top memory location available to DOS and BASIC (equivalent to setting memory size in BASIC).

JKL sends the contents of the screen to the printer.

MINIDOS is a miniature DOS that allows you to execute many DOS commands without disturbing programs stored in memory. Pressing numerals 1, 2, 3 simultaneously calls up this function. For example, if you need a directory from Scripsit, you can press 1, 2, 3 simultaneously, and you enter MINIDOS; by typing DIR, and then MDBORT MINIDOS abort), you are back in Scripsit.

PDRIVE sets the parameters for the disk drives in the system (number of tracks, size, directory location and directory size), allowing the user to set directory size. It also lets users of 77 or 80-track drives set up a larger directory, but not free space on the disk.

PURGE kills selective groups of files on a disk after asking permission for each.

SYSTEM allows the user to configure special system options. For example: enable/disable passwords; enable/disable debounce; enable/disable JKL and 1, 2, 3 keyboard routines; enable/disable lowercase driver; set number of drives in the system so DOS does not search all four drives; set default drive to other than 0 for DIR and other commands. Disassembler and Editor/Assembler programs are included.

SUPERZAP is a sector and file modification utility. Most other disk mod utilities allow you to modify sectors, but first you have to find your file. This utility handles that for you. All you have to know is the file name.

LMOFFSET loads the module offset routine and lets the user load machine language programs that conflict with DOS and store and execute them from disk.

LEVEL 1 utility is Level I BASIC relocated into high memory, and it allows Level I programs to be executed without converting them to Level II.

ASPOOL utility is an automatic print spooler program, which allows the computer to store format print output on disk and dump the disk file to the printer while the computer is doing something else.

DIRCHECK utility checks the directory for invalid entries and lists information about the directory entries.

This DOS is the fastest and most powerful DOS for the serious programmer. It is also the highest priced system, but comes with by far the most complete manual.

ULTRADOS

ULTRADOS is an easy to use operating system that is almost completely command compatible with NEWDOS+ (not NEWDOS-80). It does have some additional features in addition to what NEWDOS+ adds, and is slightly easier to use than NEWDOS+. New features are:

CLEAR writes zeros to memory locations from 5200 to the logical end of memory pointed to by the TOPMEM pointer set by DOS on powerup or by the BASIC memory size question.

NAME THAT SONG

A revolutionary sound program for the 32K TRS-80 disk system co-authored by the musical director of Broadway's "Annie." The game comes with over 100 tunes and you can add your own using the provided utility. This multi-player game has fast action and machine language subroutines. Just like the popular TV show– even has commercials! ONE OF THE MOST ENJOYABLE GAMES I'VE SEEN AND HEARD FOR THE TRS-80"– Richard Taylor, author of TRS-Opera. 32K disk $19.95

STELLAR ADVENTURE

Explore the galaxy and fight the deadly Kyraxans in this real-time graphic game with sound.

Level II 16K tape $14.95
32K disk $19.95

CITY ADVENTURE

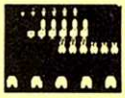
Travel from a peaceful suburb to the hustle and bustle of the big city. Standing in your way are muggers, runaway cars and numerous other obstacles. A knowledge of the program "Interludes" is helpful. Rated PG.
Level II 16K tape $9.95
32K disk $14.95

MICRONOPOLY

A full scale version of the famous board game Micro-nopoly is the only program we know of that plays by the rules, allows trading and doesn't require you to have a board.
Level II 16K tape $9.95
32K disk $14.95

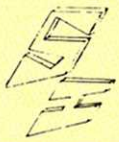
BREAKOUT

In this machine language game with sound you must destroy the graphic blocks with your bouncing ball. This simulation of the popular arcade game has 64 variations.
L 2 16K tape $9.95 32K disk $14.95



ALIEN INVASION

Shoot down the space invaders as they pass over your laser cannon. This is a new improved version available only from Software Innovations. With sound.
Level II 16K tape $9.95
32K disk $14.95



Software Innovations
320 Melbourne Rd. ↙478
Great Neck, NY 11021
(516) 482-6004
MNET-70330,205

Add $1.00 for shipping.
Free catalog available.
Terms: Check, M.O,
VISA, MC.
NY res add sales tax.
Dealer Inquires Invited.

ELECTRONIC ORDERING IS HERE

↖ INNOV-80 CBBS (516) 482-8491

↙ CONNECTION-80 CBBS (516) 588-5836

COMMUNICATE!!

MICROCONNECTION– $249.00
For Model I or III, decodes the TRS-80 buss and creates an RS-232 port. Eliminates need for expansion box and/or RS-232 board. Provides serial printer output. Requires software written for port 208. Requires adapter cable for Model III.

RS232CONNECTION– $199.95
Works with virtually any computer or terminal having an RS-232 serial I/O port. Replaces obsolete acoustic coupled modems. Compatible with all TRS-80 terminal software. Requires RS-232 cable.

COLORCONNECTION– $199.95
Converts Radio Shack Color Computer into a communications terminal. Provides serial printer output. Does not require any additional memory or extended basic. Complete with cables.

AUTOCONNECTION– $79.95 (option)
This option permits CONNECTIONS (except Color) to answer the telephone and/or dial numbers automatically. Supplied with a dialer demonstration program.

EUROPEANCONNECTION– $20.00 (option)
This option modifies the above units for European Standards.

MODEL III ADAPTER– $29.00
Cable converts Model III 50 pin buss to 40 pins used by Model I peripherals.

RS-232 CABLE– $24.00
Four-foot ribbon cable terminated with 25 pin male DB25 plugs.

SMART-80 and SMART III
By Dick Balcom

Smart Terminal Programs for the TRS-80+ Model I and Model III Computers! This is the ONLY Smart Terminal Program available for the Model III as of this printing! These programs are available in either cassette or disk versions for the Mod I and Mod III. These programs allow you to up AND down text and programs to your TRS-80+ from other users, Timeshare Systems (source, MicroNET), and free public BBS (CONNECTION-80, etc.). This software represents the State Of the Art in terminal software for the TRS-80+.

Smart-80(C), Smart III(C) automatically converts basic programs saved on Tokens to ASCII code, and back again. EDTASM tapes can also be loaded thru the terminal!

Smart-80(D) is the Disk based version of Smart-80 with the ability to transfer diskfiles to another user over both phone and radio with the use of a modem. These programs support Auto Log-on, as well as other features not found in other programs.

Smart-80 (C) Model I $79.00
Smart III (C) Model III $99.00
(1500 Baud Tape)
Smart-80 (D) Model I $79.00
Smart III (D) Model III $99.00

B.T. Enterprises ↙380
171 Hawkins Road
Centereach, New York 11720
Dealer Inquiries Welcome

Add $2.00 S & H
NYS res. add appr. tax

(516) 981-8568 (Voice)
(516) 588-5836 (Data)
MNET-70331,105

DEAD writes zeros to memory locations from 5200 to the physical end of memory, ignoring the TOPMEM pointer.

TOPMEM sets the logical end of memory to other than the physical end as set by DOS on powerup. This DOS has three separate BASICs, with unique characteristics as follows:

BASIC is the ULTRADOS full featured super BASIC, similar to TRSDOS BASIC, but using less memory (more available for user programs), and adding more features as follows:

CMD "C" compresses the BASIC program in memory and removes spaces and linefeeds, leaving more free memory and allowing faster execution.

CMD"DOS COMMAND" executes the DOS command and returns to BASIC.

CMD"O" allocates a new file buffer in addition to those allocated when BASIC was initialized.

CMD"X" returns to Level II BASIC in ROM to preserve the program in memory, and also to allow return to the Disk BASIC at a later time. If you ever had a Level II BASIC program that wouldn't run right in Disk BASIC, you will appreciate having this feature.

CROSS REFERENCE will list the variables or line numbers in the BASIC program.

RENUMBERING allows the user to renumber a BASIC program while in BASIC.

SHORTHAND allows several commonly used commands (list, edit, for example) to be entered in shorthand form as only one character followed by ENTER.

In addition to Super BASIC, two other BASICs are included, Short BASIC (BASICS), and Tiny BASIC (BASICT) that give up some of the commands of Super BASIC, and leave more free memory. The memory available for the three BASICs are as follows:

Super BASIC (BASIC) 40414 (2124 less than TRSDOS)
Short BASIC (BASICS) 41204 (2914 less than TRSDOS)
Tiny BASIC (BASICT) 41980 (3690 less than TRSDOS)

The features of the BASICs are identical to the TRSDOS BASIC, with a substantial savings in memory. The Tiny BASIC went one step further and removed the file handling of TRSDOS, and kept only disk load and save commands, for yet more memory savings. We consider these good features and wonder why the other vendors can't seem to come close. In addition to BASIC, other utilities included are as follows:

DA Utility— disassembler program.
EA Utility— disk based editor/assembler program.
LM Utility— load module offset utility.
LC Utility— lowercase driver routines.
RS Utility— ROM search utility, searches the Level II ROM for a 16-bit word specified by the user.
UZ/BAS Utility— Ultra-zap sector oriented disk modification utility.
GR Utility— graphics keyboard driver, allows graphic characters to be entered from the keyboard.

Knife/BAS Utility - program that allows the user to "perform surgery" on a BASIC program. It does much more than just remove space, it also allows the user to:

Change all or part of variable names.
Change all or part of constants, data list items, or strings. Change graphic codes from the CHR$(XXX) format into compressed strings (the funny looking ones that list as a bunch of garbled key words, but are really graphic strings)
Merge adjacent lines into one longer line.
Split one line into two shorter ones.
Excess space removal.
Remove REM statements.
Change reserved words.

Between the Super BASIC and these two utilities, this will be a very good system for the BASIC programmer. The documentation that comes with this DOS is in every way as good as the TRSDOS manual from Radio Shack. This DOS lists for $20 more than NEWDOS+, but is well worth the difference.

BOOT reboots and reinitializes the system on command.

BUILD allows the user to enter a set of commands to be executed by CHAIN.

CHAIN executes a series of commands entered with BUILD.

MEMORY allows the user to directly set the TOPMEM pointer to reserve high memory.

PURGE selectively kills a group of files, after asking permission to kill each.

RUN allows the user to load and execute a program on a non-VTOS disk, or a non-system disk, with only a single drive system.

How's your love life?

A little dull around the edges? Routine? Predictable? Boring? Maybe all it needs is a little Interlude. Interlude is the most stimulating computer game ever conceived. It combines a computer interview, an innovative programming concept, and a one-of-a-kind manual to turn your love life into exciting, adventurous, delicious fun!

Interlude is: romantic...playful...outrageous...a fantasy. Interlude is: ■ A Bed of Roses (Interlude #1) ■ Mata Hari (Interlude #49) ■ The Chase (Interlude #7) ■ Rodeo! (Interlude #71) ■ The King and I (Interlude #60) ■ Some Enchanted Evening (Interlude #84) ■ Caveman Caper (Interlude #82) ■ From Here to Ecstasy (Interlude No. 30) ■ Satin Dreams (Interlude #72).

More than 100 Interludes are included in the program. Most are described in detail in the accompanying manual, but several surprise Interludes are buried in the program awaiting that very special time when your interview says you're ready. (When you learn secret Interlude #99, your love life may never again be the same!) Interlude can give you experiences you'll never forget. Are you ready for it?

Interlude™
The Ultimate Experience.

INTERLUDE, 10428 Westpark, Houston, Texas 77042. I'm really ready. Send my Interlude today.

Apple II (16K)*
☐ Cassette ($16.95)
☐ Diskette ($19.95)
☐ Diskette—Pascal or DOS 3.3 ($19.95)
Add $1.50 for shipping and handling.

TRS-80 (Level II-16K)**
☐ Cassette ($16.95)
☐ Diskette ($19.95)

Poster
☐ 20"x 24" reproduction of this ad without ad copy ($4.95—includes shipping charges)

Available for immediate shipment.

Please enclose your check payable to INTERLUDE or complete the charge information:

☐ MASTERCARD ☐ VISA All charge customers must sign here. _____
Account No. _____ Expiration date _____ MasterCard Bank Code _____

CHARGE CUSTOMERS: Order by phone toll-free! **1-800-231-5768 Ext. 306** (Texas: 1-800-392-2348 Ext. 306)

Name _____ Age _____
Address _____
City _____ State _____ Zip _____

*Apple II is a registered trademark of Apple Computers, Inc. **TRS-80 is a registered trademark of Radio Shack, a Tandy Co.

"An operating system that suits your needs will prove to be a good investment, not over a week or month, but over an extended period of time."

SYSTEM allows the user to set some hardware dependent features, and optionally, configures a version that automatically sets them on power up. For example, high speed mod enable/disable, lowercase enable/disable, blinking cursor enable/disable/rate, JKL screen print enable/disable/graphics, on/graphics off, default drive set, delay time after drive motor turn on set, drive step rate set.

XFER is similar to COPY, but will copy a file from one disk to another with a single drive system.

In addition to the added commands, the following utilities are also included:

PATCH takes care of only one of the functions of the ZAP programs on the other systems, but it does so without the user being required to know anything about machine language, or the operating systems. It allows the user to modify and/or correct errors in files on the disk. For example, BASIC is not supplied with VTOS, but by patching TRSDOS BASIC, you have VTOS BASIC.

VTCOMM utility is a communications utility allowing communication between machines via the RS232 interface/modem.

KSR utility is a keyboard send-receive terminal emulator similar to the Term program from Radio Shack.

In addition to the utilities, several data files are included to support the patch, KSM and set functions.

VTOS is a very powerful operating system, but its designers seem less concerned with human engineering than the others, and therefore it takes more practice before VTOS can be used efficiently.

VTOS 4.01

VTOS 4.01 seems to be written for the high level business user, with features so advanced that many minicomputer operating systems don't have them. Unfortunately, with all this power, there is more room for bugs, as well as more memory overhead for the CPU. This results in the most powerful business DOS, but also one of the slowest.

This version of VTOS did eliminate the password that plagued earlier versions, but the backup utility still gets upset when handed a disk formatted by another system. One of the powerful features of this DOS is its device specs. Some examples are: *DO = display output, *PR = printer output, *KI = keyboard input, *RO = RS232 output, *RI = RS232 input. They are used essentially like filespecs, and their equivalent of a directory command is the little used device command. With these, you can use commands such as:

ROUTE, which allows you to route data destined for one device to either another device or to a disk file.

SET allows you to set up a device spec using a device driver program supplied with the DOS or created by the user. (For example, set *CL to RS232/DVR.)

SPOOL allows the user to route data destined for a device (any device, not just the printer) to another device through a file. It does it in such a way that data for the device first goes into a file (or memory) and from the file (or memory) to the device when it is ready. In this way, the computer continues running the user program until the printer becomes available.

RESET resets the devices to their original drivers and essentially cancels the above commands. In addition to the device handling commands above, other commands are included.

LINK allows the user to link devices together. For example LINK *DO to *PR causes everything displayed to go to the printer also.

FILTER is similar to SET, but also allows the user to filter the data before it reaches the device or file. For example, it will eliminate graphic and control characters from data before it reaches a printer that doesn't support graphics or control characters.

ALLOC preallocates disk space for a file adjacent to it.

Bugs and Zaps

There are bugs in every system. As they are found, the seller fixes them. Apparat is especially good at this. They should be. It takes over 40 "zaps" to bring an original NEWDOS-80 up to current standards. Zaps, or changes, do not necessarily mean flaws in the system. Many of them support some new or existing utility. For example, the popular Electric Pencil requires a zap for most systems. Each vendor assures the purchaser he will be notified of all new zaps. Those of you with bootleg copies (shame on you) are on your own.

Only you can determine which DOS is best fitted to your needs. However, after extensive use of the systems, one point becomes quite clear. An operating system that suits your needs will prove to be a good investment, not over a week or month, but over an extended period of time. The authors of this article applaud the work of these DOS writers who, each in his own way, have indeed built better, albeit, different mousetraps. ■


SEND FOR FREE CATALOG
OVER 150 ITEMS
Ahoy Dealers!
CALL OR WRITE TODAY
FOR YOUR FREE
DEALER PACKAGE!
SOFTWARE AUTHORS!
We want to sell your programs!
SEND YOUR TAPE OR DISK
(with any documentation)
TO OUR SOFTWARE
REVIEW BOARD - Dept. R
Adventure International
SOFTWARE PUBLISHING DIVISION OF Scott Adams INC.
ADVENTURE INTERNATIONAL BOX 3435 LONGWOOD, FL 32750 (305) 862-6917

80 Interviews

Lew Kornfeld

Interview by Nancy Robertson

They're changing the guard at Tandy Plaza. After ten years as president of the corporation, Lewis Kornfeld is retiring from Radio Shack. He will be 65 July 31, and is retiring by choice. In the interim, Mr. Kornfeld is serving as vice chairman and John Roach has stepped up to the presidency.

In an interview with 80 at the Boston computer show, Kornfeld traced the history of Radio Shack and discussed his personal plans for the future.

Kornfeld's background was in English lit and reporting when he came to Radio Shack in 1948 as advertising manager. From advertising he moved into marketing. The man responsible for Radio Shack's development and marketing of the TRS-80 Model I, Kornfeld says computers and their intricacies were way over his head when he made the decision to go ahead with production. His remembrances of his 32 years with Radio Shack are marked by his strong opinions about business, his company loyalty and an old-fashioned patriotism.

With his retirement, Kornfeld looks forward to more time with his wife and the pleasures of home. He plays the piano and the recorder. He wants to get back to "serious reading" and to start work on a book about advertising. And he wants to take up a new instrument. "I bought a flute," he says. "I haven't taken a lesson, or opened it up. I don't even know how to put it together. But the day I retire, I'm going to take it out."

80: I got the press release saying you were retiring.
Kornfeld: Yes, Ma'am. As planned by me and my wife for quite a few years. July 31 I'll be 65, and I've been working 45 years, you know, overall. Thirty-two—at that time it will be almost 33—for Radio Shack. And you know, there was other assorted work, like teaching English lit, writing for a newspaper and five years in the Marine Corp.

80: Where did you teach English lit?
Kornfeld: At the University of Denver. I was an instructor while I was getting my MA in 1941. And I've got lots and lots of hobbies. I can afford to retire, which is a little more than I anticipated a few years ago. I thought it might be very difficult to retire, but why wait until they drag you out?

80: It doesn't look like you're completely retiring.
Kornfeld: I'm going to stay on the board. I'll drop all titles and become an outside director.

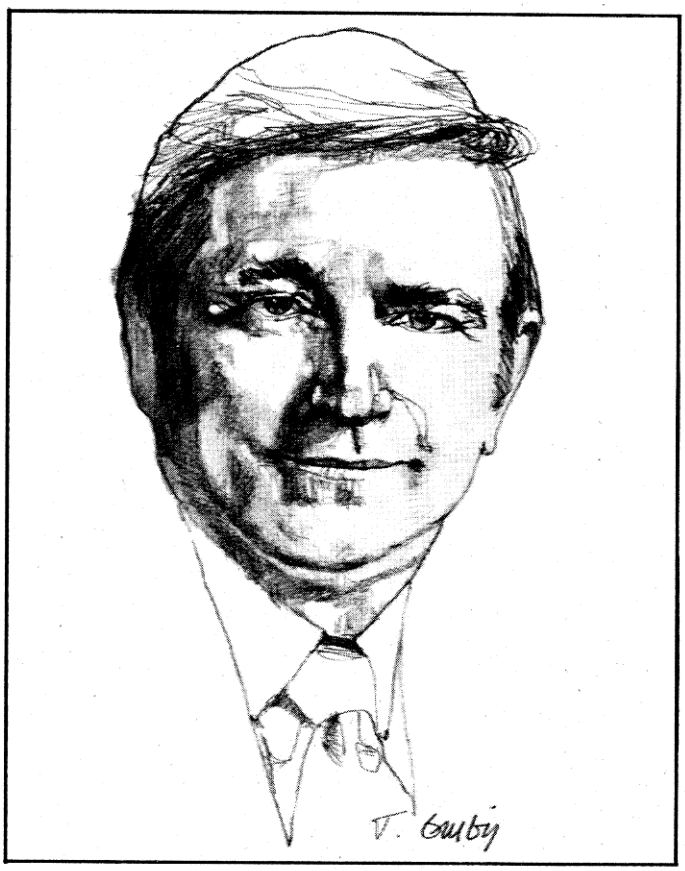
80: How does somebody who starts by teaching English end up the president of a large manufacturing company?
Kornfeld: Manufacturing and retail. How it happens is, in my case, I went into advertising. So, my first job at Radio Shack was as advertising manager. That was 1948. Then, in '54 I became V.P. of advertising—which meant they didn't have to give me a raise. In this period, I discovered a latent ability to select merchandise for the marketplace. I became V.P. of merchandising and advertising in 1958, and then president in 1970.

80: How long has Radio Shack been around?
Kornfeld: Well, Radio Shack technically grew out of a company that was founded in Boston in 1921. That's the year we use as our start-

Continued to page 86

80 Interviews

John Roach



Interview by Nancy Robertson

John Roach explains that Radio Shack is "a company that has grown primarily from within. We very rarely bring in someone from the outside." But maybe his attitude towards his work has as much to do with his recent promotion to president of the company as his 14 year history with the company.

In a recent telephone interview with 80, Roach answered questions about Radio Shack's future and its commitment to computers. If the greatest strength of retired president Lew Kornfeld was his marketing outlook, Roach's forte is his familiarity with computers.

In college, he majored in math and followed through with a masters in business management and a computer studies fellowship from Texas Christian University. With his degrees in hand, he began working in the data processing field. Two years later, he joined Radio Shack and became manager of their data processing division. When the first Model I rolled off the production line, he was the vice president in charge of manufacturing.

A native of Texas, Roach lives with his wife and daughters Amy, 14, and Laurie, 8, in the Fort Worth area. But there isn't much time outside of work. He told 80, "What we do here (at work) is so much fun, you don't have to go looking for a lot more."

While observers have seen Roach's strong computer background as an indication that Radio Shack is gearing itself for a computer-dominant future, Roach disagrees.

80: You were saying that the significance of data processing was that you had experience in several different fields of Radio Shack's business.

Roach: Well, being the data processing manager, I've had a very good introduction to all parts of our operation. Of course, I don't think it's insignificant that since I had a feel for the computer business I became intimately involved in Tandy's entry into the microcomputer business. I was in charge of warehousing, manufacturing, and, then, executive vice president of Radio Shack.

80: You don't feel, then, that your appointment is a sign of Tandy's commitment to computers?

Roach: Certainly I think the natural evolution of Tandy is into more technology products of which computers are a very important part. But certainly, it's our desire, goal and ambition to help bring technology to the consumer in all phases of consumer electronics and not exclusively computers.

80: I wonder if you could tell me how your presidency will differ from Kornfeld's.

Roach: Well, I don't know that it will differ significantly. We, basically, have a lot of the same, or essentially the same, philosophies. Many came down from Charles Tandy. Both of us worked closely with Charles. While Lew is a little bit more advertising-oriented in his experience, I hope that there isn't any real significant difference in our outlook towards the future.

80: I also wanted to ask you about some new directions the company seems to be taking in marketing. For one thing, I noticed that you now have a non-exclusive agreement with Random House to distribute your products outside of Radio Shack outlets. This is new. I wondered if you could comment on that.

Roach: Well, we have a number of marketing activities in consumer electronics that do not follow precisely our normal pattern of Radio Shack stores or Radio Shack Computer Centers. The Random House agreement is strictly to give us an avenue to approach the educational market, which we feel is one of the major segments of the microcomputer market.

80: Since the Random House agreement is non-exclusive, that would imply there will be other agreements.
Roach: Certainly, since it is non-exclusive, there is opportunity for other agreements. I don't know that we have any other agreements at the present time.

80: Another new approach that I've noticed in your marketing recently has been the selling—within Radio Shack outlets—of software packages designed by other companies. There's the Personal Software Space Warp program, which is also a non-exclusive agreement. And this is something I find unusual for Radio Shack.
Roach: I don't think it's really all that unusual. If you study the history of the business, we have these products manufactured by others for sale under our label. So I think our software activities are following very closely to what I think our historical hardware activities have been. You know, we manufacture somewhere around 46 percent of everything we sell. And so without knowing the breakdown on software, we make some of it ourselves, and we buy some from others.

80: I was under the impression that generally Tandy buys exclusive rights. In contrast, companies like Adventure International and Instant Software will market one another's programs non-exclusively. This Personal Software venture with Space Warp is the only time I've come across a Radio Shack non-exclusive rights situation. That's why I was especially curious about it.
Roach: I don't really know personally that much about either of those software packages or our arrangement with the people. But I don't consider it to be a major philosophical change.

80: While we're talking about marketing, I'm also very interested in what may be happening with your European market. When I was interviewing Mr. Kornfeld, he was comparing your movements in Europe and Australia to the Japanese, in that they are willing and able to take a loss for several years. What is the situation now? Is the European market up and going for you?
Roach: Well, it's relatively public knowledge, at least in the financial community, that we did lose substantial amounts of money in our overseas operations up until about 1978. Those operations are now profitable and, in general, on a very good growth pattern. We plan to aggressively expand in some of these overseas areas.

80: Any place in particular?
Roach: Well, of course, we already have in Australia and Canada similar penetrations of the market to what we have in the United States. Our most aggressive expansion at the present time, overseas, is in the U.K. and Belgium and in France.

80: I've seen what looks to me to be the three major uses of micros right now. One, of course, is in the home. Another is business, and the third seems to be education. I wondered if, on each of these points, you had any ideas or predictions on the sociological impact of microcomputers.
Roach: I think it's our position that we make the hardware and, to some extent, the software available. What people use them for and what the ultimate impact of the machine will be is up to the users. Obviously, I think that without any question computers are going to be pervasive in everyone's lifestyle over the next decade and a half. That doesn't mean that everybody is going to recognize and appreciate that they're using the computer. Many of them (computers) might be very dedicated in purpose. But, none the less, it's inevitable that they're going to appear everywhere.

80: Well, speaking in terms of home applications, I know that Tandy's involved in Knoxville with an experiment to put the Color Computer into homes to be used, first of all, for banking. Have you had any feedback on that yet?
Roach: Well, I look at all of these things as being additional utility. Probably the very first computers that anybody attempted to use in the home were for hobby, or maybe you call it hobby recreation. I think now we're progressing to the point where they're not only recreational, but educational. The Knoxville banking is not an experiment: It's the real thing. This and our Videotext efforts in general lend another bit of utility to the home terminal. If the various software and hardware enhancements are made and the utility continues to increase with each enhancement, at some point you'll have the right services at the price people are willing to pay to use it.

Now, I don't think it's going to be, at least in the short term, that suddenly just *everybody* has to have a computer. But, there are a few more people that thought they had to have a computer this year than thought they had to have one last year. And with the building awareness, with the increase of utility—utility with respect to price—this is going to continue.

80: How does Tandy expect to influence the market in education?
Roach: I think that in education computers have a place as an alternative, or, not really an alternative, an additional tool that can be used to help the teacher train students—independent of age.

But, probably, as far as the computer industry itself is concerned, the biggest thing that's going to happen, by the mass introduction of computers into the schools and into the homes, is that we're going to have a lot more young people who select computer science and programming as an occupation. And we will, I think, see some of the pressure for more and better software relieved by some really great programming by people fresh out of high school, fresh out of college. Because, really, young minds normally work best anyway.

80: I'm also interested, knowing that Tandy/Radio Shack has recently developed an education division, how they are going about meeting the software demands of the education market. Let me preface that a little bit. There was a conference at Harvard on education recently. It seems that a lot of the teaching profession is leery of too much drill-and-practice, and feel that there are more applications that can be applied. However, the software is not coming through the marketplace that does these things.
Roach: I don't claim to be an expert on computers in education. I do know that we're developing, or people are developing, a number of educational software packages that deal not only with drill and practice, but the teaching of programming and mechanically helping teachers with their grading and statistics. There is, I guess they call it the course writer type program, which lets teachers place their own material in the machine easily.

I'm not saying we're doing all the right things. I'm saying we're do-

KING OF THE HILL!

We've taken artistic license with our illustration in order to make a point: MYCHESS is the most powerful microcomputer chess program on the market, bar none.

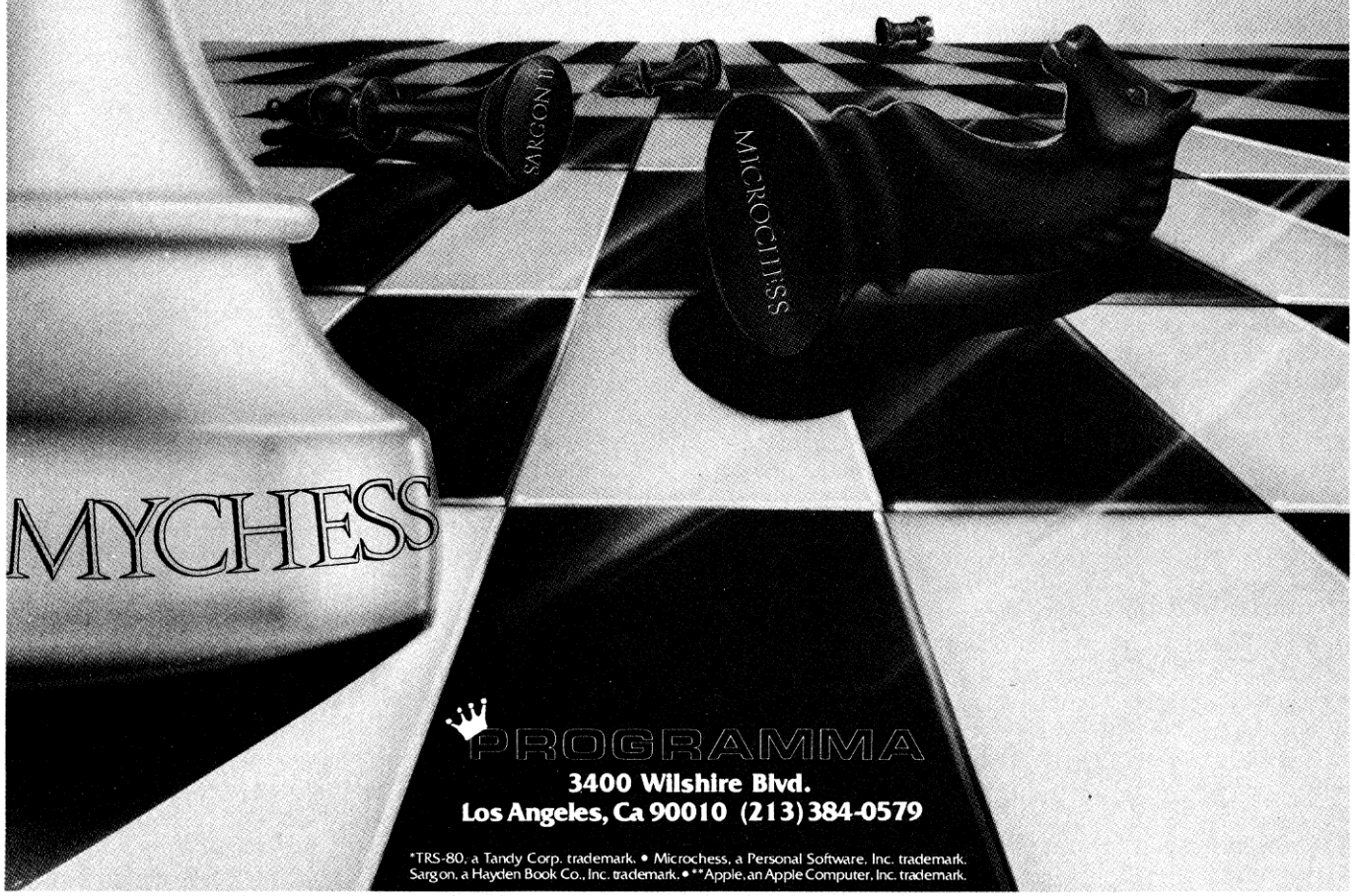
Proof? All you want and then some. For example, MYCHESS was the winner of the "Fifth West Coast Computer Fair". At the "Third World Computer Chess Championship" in Linz, Austria, it was the highest finishing micro... in addition to winning the special Blitz Tournament (5 to 1) against six top players. Add to this its USCF rating of 1565, and you know you're dealing with the King of the Hill.

You'll find MYCHESS is the perfect companion or opponent whether you're an advanced player, or starting your first game. For it lets you set the difficulty of the game from level 1 to 9. And, you can change levels of play as you go ... or even change sides. Want to set time limits for moves? MYCHESS can do it. Want to save a game for later? MYCHESS will store up to 6 games. And, for added interest, it will even predict the upcoming line of play.

If you're a player, you'll appreciate the MYCHESS challenge. If you're a beginner, you'll enjoy learning from a master. Either way, when it comes to superior chess, make your move ... to MYCHESS. Available for the TRS-80* with 32K, for $34.95 including disk, documentation and backing by Programma International. Apple** version coming soon.

Can **you** beat

MYCHESS



PROGRAMMA

3400 Wilshire Blvd.
Los Angeles, Ca 90010 (213) 384-0579

*TRS-80, a Tandy Corp. trademark. • Microchess, a Personal Software, Inc. trademark. Sargon, a Hayden Book Co., Inc. trademark. • **Apple, an Apple Computer, Inc. trademark.

ing a number of things; that we can't possibly hope to do everything, and I'm sure professional educators and independent software houses, and a lot of other people will develop software that will be important to making computers more useful in education.

80: I would also like to know what you feel the effect of electronic mail will be on the nation.
Roach: That's a hard question. Certainly, there are a lot of communications which could be handled, maybe, just as efficiently via telecommunications or electronic mail. And certainly, as people have a home terminal, the facility is going to be there for this to happen, but how quickly it's going to happen is a little bit hard for me to visualize.

80: What about electronic news?
Roach: Electronic news. Of course, with our bargaining arrangement with CompuServe we are providing, and plan to provide substantially more over the next 12 months, electronic information including news. I think it remains to be seen how important a medium it becomes. I believe that just like you get your news today from the radio and the television, and magazines and newspapers, etc., that as time goes along, one more place you'll get instant information—say an instant update of those stories you're interested in—will be through your home terminal.

80: The FCC recently gave Tandy a reprieve on the radio frequency regulations for Model I interface. Our readers, I'm sure, would like to know what Radio Shack might do to continue to protect their Model I.
Roach: Well, we have additional software in development for the Model I. We plan to continue to sell the peripherals, including the expansion interface for the Model I. I think it will be treated very much like any other product; as long as there's a demand for support for the Model I, they can expect it to be there. Also, like any other product, the maintenance and repair service will be available for literally years.

80: *Fortune* reported recently, and I'm quoting, "U.S.-made computer chips were judged in one test this year to be three times as likely to fail as Japanese chips." Basically, it was an article about quality control of American-made goods vs. Asian goods. I was wondering how Radio Shack falls in this category. Are your chips Japanese or are they American?
Roach: We have those American or domestically produced chips and we also use foreign produced chips that are Japanese. I think (the survey) is referring to fallout, what's called infant mortality (those chips that fall out within the first 24 to 48 hours). Burn-in procedures and testing procedures in the manufacturing process are designed to find those chips independent of who buys the manufacturing.

I don't think there's any question but what the Japanese very aggressively have pursued a quality image in general. Not only in the manufacture of semiconductor chips but in many of the other products that they're manufacturing. And I think it has had a positive effect, particularly on the U.S. semiconductor industry, which I believe is taking steps to make their quality equal to, or better than, their Japanese counterparts.

80: Within this question, there's the question of where Tandy stands in relation to the Asian market. I went over this with Mr. Kornfeld, but it would be good to get a perspective on it from you as well.
Roach: We are selling some computers in Asia. We will continue to sell there. Whether we will ever be number one, you might say, is probably not too likely, because in business, like everything else, the Japanese are very nationalistic.

80: What about the effect of the Asian products on our market?
Roach: I think it's generally conceded that Asian products will become increasingly more important. The only thing that is really necessary for them to become more important is for them to bring some important things to the marketplace. I think in a few segments of the market they will do this relatively well. In other segments of the market they may never do it very well, or it may take a long time before they do it very well. We will see a continual increase in the amount of Japanese produced products available on the U.S. market.

80: Is Tandy the sort of company where people form close and lifelong friendships? It seems to me to be interesting the way Lew Kornfeld, for instance, spoke very fondly of Mr. Tandy.
Roach: We're a company that has grown primarily from within. We very rarely bring in someone from the outside. Some of us have worked together 10, 15, 20—Lew's been here 30—or over 30 years. Mr. Tandy was a very brilliant, dynamic character and certainly would fall into the category of one of the most unforgetable people you'd ever want to meet. . . whether you worked for him or not. Certainly we were all very fond of Mr. Tandy.

80: About your own life outside of Tandy; I mean, there is life outside of Tandy, isn't there?
Roach: Well, what we do here is so much fun you don't have to go looking for a lot more. My life is very distinctly split into two parts. A very significant part of it is spent on our business and the other part is spent with my family. For instance, I don't play golf because I don't want to take that very little time I've got available away from the kids and my wife. When I'm home, I'm home, and when I'm at work I'm at work.■

Continued from page 82

ing year. It was born in Boston. It became Texas-headquartered when Tandy Corp. acquired it in 1963. . . . Then in '75 Tandy Corp. spun off to the stock holders (that is, got rid of to the share holders) all the other companies except the electronics companies. These were Radio Shack and a small company known as Allied Electronics, which was an industrial electronics company. We sold that in about 1977 or '78.

80: I think our readers would be interested in a retrospective look at the decision to go into computers.
Kornfeld: O.K. The plain simple truth of how we got into the business: First of all, it couldn't have happened if it hadn't been for a number of situations. In other words, it isn't just one juncture; it's the coincidence of a number of junctures.

First, we had to have been in manufacturing, understand the economics of manufacturing, and also have a lot of confidence in our ability to manufacture. Having started the modern Radio Shack manufacturing program, I never had any doubt that we had the capa-

bility. We started making the TRS-80 Model I about 1976. That's when the decision was actually made to put it into production.

We knew we had a very formidable marketing arm in terms of our stores. We knew we were very good at sourcing. We had just celebrated the twentieth anniversary of our import and export company in the Far East, actually in Tokyo, which I helped establish in 1955. So, we had Far East sourcing; we had domestic sourcing. We were self-manufacturing, at the time, probably 60 to 65 percent of our stuff in the U.S. and 30 to 40 percent overseas, mostly in the Orient.

Then, we had one person in particular, who was a buyer on the merchandising staff, and who was a computer hobbyist. Nobody egged him on to be more of a hobbyist. Nobody told him to stop messing up his office with all that junk. My office, for example, was only about 50 feet away from his (in our old Fort Worth offices). I used to look in, and I'd tell him, at first jokingly, "When I can play chess with that, and you don't have to hold the wires or explain why it's not working when it was just working, I'll put it into manufacturing." One day, in essence, that's what happened.

Very fortunately, we'd had a change in manufacturing personnel, and the fellow who is now president of Tandy Corp., John Roach, was my V.P. in charge of distribution. John was there, I liked him, he understood computers because he had run the corporate computer section (the data processing division). I appointed him head of manufacturing, and then we built a nucleus of a team.

The game was played basically by my rules. And I'm not a computer anything. But my rules were: No kit, we wanted to sell something that worked out of the box. No funny names, you know, like apples or oranges or lemons. As I put it, no racing stripes. We wanted it to look like a piece of business equipment. I wanted to start out by selling a complete system at a price. And I wanted—I insisted—that this system be expandable to whatever limit it had, which was well over my head. But that's exactly what we did.

Then we put the price on it. We spoke to a number of people. How much is it worth? I had suggestions maybe as high as $1400. I had low side suggestions, as low as about $900. And I said to myself, the hell with them. Because I'd been a pricer for 25 years or more at that time, I said to myself, I'm going to price it low enough so that if it fails to sell at a desirable rate, we will never look at each other and say, "What if we had sold it for less?" I didn't want that to be an excuse. So, it was priced practically, like most of our equipment. Then it was pushed out to stand on its own. In retrospect, we probably could have gotten another couple hundred dollars for it.

I priced the entire system, the 12-inch monitor, the CPU keyboard, the power supply, the cassette recorder and manual all for 599 bucks. . . . The fact is, at the time, the 599 was a very satisfactory number. We ultimately cut it to 499. This was after about two full years of selling. By then we had the scale of economy on our side.

As it turned out, I didn't really have any idea just how much back-up and support a computer—seriously marketed—actually needs. But I remember telling Mr. Tandy, who was our late chairman and founder of the modern company, that if I could compare it to a 747 airplane, we were going to need some big airports and some long landing strips.

80: One thing I often hear is, "Yeah, Radio Shack sells a good computer, but their stereo equipment is lousy!"
Kornfeld: It's absolutely, totally, catastrophically false. Just for example, we are reputed by others to be the country's largest, in terms of unit sales, seller of headphones and speakers. We've been selling our own brand of Realistic speakers for 25 years. I don't know how many millions of them have actually been sold. You just can't do it for 25 years. You might get away with it for three or four, but you can't continue on.

80: People have been making comments to me. They have to do with the future of Tandy and predictions that the company has a bad turn ahead.
Kornfeld: A bad turn ahead! God damn, where do you reporters get all that crazy news? What are you reading?

80: I've heard this around the show today. People are saying that they think you are going to have trouble because the Japanese imports will flood the market.
Kornfeld: I'll answer you—if you'll tell me that you're not quoting from the majority feeling. You just picked it up.

80: I admit it.
Kornfeld: Listen, I used to be a reporter myself.

80: You're a gossip monger, then you go back to the source to try to find out if it's true.
Kornfeld: And you hope it is, because then the story will be a little bit more juicy.

80: But it's always good to hear how somebody can respond to something like that.
Kornfeld: I'll respond. Ask me a question.

80: How do you think the Japanese imports are going to affect Tandy? Do you think the Asian-Japanese imports will be a bad thing for you?
Kornfeld: Well, Tandy is in a very interesting position, as is pretty well known. We bring upwards of half a billion dollars merchandising cost over from the Orient. So, we are partly Asian, if you want to look at it that way. We have hundreds—not hundreds, thousands—of Asian employees.

We have the ability, because of our dual citizenship, so to speak, to move with the tide in commerce. If, in fact, things in Asia shape up following the lines of audio, radio and tape recorders (to name three principal areas where the trend shifted away from this country and Europe and into Asia), we're there already. We will sense it. We will know if we have to shift, which I sincerely hope we won't have to do. God damn it, I'm a chauvinist first—in the sense that I don't like to see this tremendous (domestic) drain.

I feel that of all the manufacturers of computer electronics in the field, we're the most flexible and most ready to move with any change in the pattern of the country of origin. There are already some strong computer products coming from Asia. Printers, for example. Do you know why? Quality for the price. And it ought to be here (in the U.S.)—it was here. I'm not going to get on a soapbox, but it's true. The Japanese computer companies are a can-do group. If they decide they can and want to do it, then, they will do it.

80: How and why was John Roach chosen as your successor?
Kornfeld: He just happened to be my personal choice, but that didn't mean he had to be *the* choice. Why is Ronald Reagan president? Because he has the majority.

You can't just walk out on anything, you know. Naturally, I wanted to be as sure as I could be that there was somebody prepared to take

over. John had the widest range of training: the corporate computer division, distribution, merchandising, manufacturing. His corporate computer experience indicates he's pretty good at numbers. His distribution experience is critical and also the manufacturing. There he was, the right experience at the right moment.

80: What are your retirement plans?

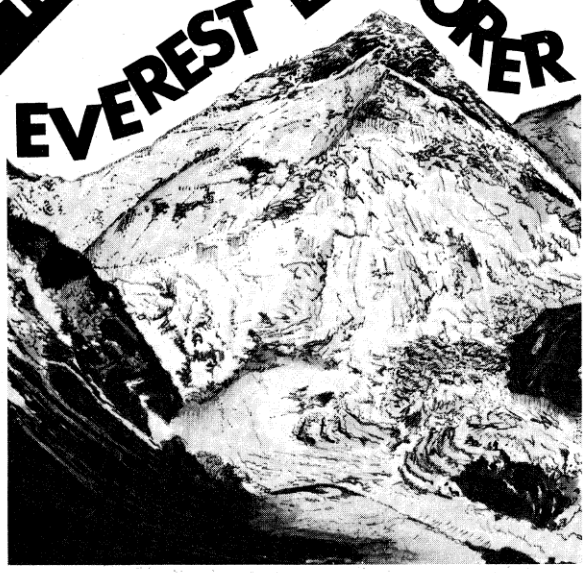
Kornfeld: First of all, I write all kinds of stuff and I intend to keep on with it. Second of all, I'm a photographer. Thirdly, I travel and my wife likes to travel. Fourthly, I like to play musical instruments. I play the piano and the recorder, the alto recorder. And I bought a flute. I haven't taken a lesson or opened it up. I don't even know how to put it together. But the day I retire, I'm going to take it out and say, "How do I assemble it?"

I obviously like reading, and to tell you the truth, for the last several years, I haven't been able to seriously read anything. I read spy books, science fiction books—anything I can read on an airplane in one pass over a country or over an ocean. But to get back to serious reading, that I want to do.

I'm a person, also, who likes to have projects, and I've got a project, which is to write a book on advertising. I think its got a number of publishers interested. I want to tell them, just facetiously: "Warnamaker once said that for every dollar he spent for advertising, he knew half of it was wasted, but he didn't know which half. I'm going to partfully, hopefully, tell that."■

FOR TRS-80*

EVEREST EXPLORER



by William Godwin & Don Knowlton

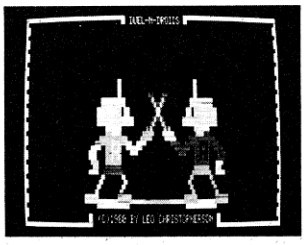
Towering above the clouds, the summit of Mount Everest is a forbidding spot. Only a handful of daring adventurers have made it to this five-mile-high pinnacle where the air is too thin to breathe, violent storms erupt with little warning, and danger lurks at every footfall.

Everest Explorer lets you challenge the world's highest mountain without ever leaving home. This remarkable simulation goes beyond most adventures, pitting you against challenges more terrifying than dragons and dungeons: the *real-to-life* horrors of one of man's most dangerous endeavors. It's an adventure so real that you may want to don a parka and climbing boots while playing.

Assigned an expedition budget, you must select the manpower, food, fuel, shelter and oxygen supplies you will need to support your quest. Now the adventure begins as you conquer the elements and terrain, establishing ever higher encampments.

Weather, route, season, camp placement, climbers' condition and morale all play key roles in the final question: Will you reach Everest's 29,028 foot summit alive?

Available for TRS-80* Level II, 16K for $14.95 on tape. 32K disk version, including "save game" feature and other enhancements, $20.95.



DUEL -N- DROIDS

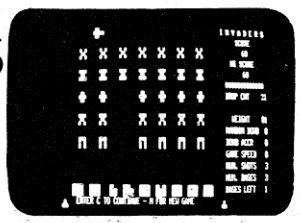
by Leo Christopherson

Your 'droid has already learned NIM, so now it's time to teach it how to wield a laser sword! Leo Christopherson, author "Android NIM," "Dancing Demon" and other animations, has developed a new type of animation and high-quality sound in his latest work.

Your 'droid starts out as a lowly clown. You teach it how to use a laser sword by controlling its movements—advance, attack, even retreat if necessary. After training it to be a "Grand Master," you enter the tournament against the program's skilled 'droid. Revel in the fanfares of the victorious—or hear the funeral dirges of the defeated! Entertainment for all ages.

Requires 16K. Available for $14.95 for tape version, $20.95 on disk.

INVADERS FROM SPACE



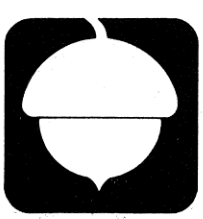
by Carl Miller

A fast machine language approach to this classic (and addictive) space game. As you play, the aliens drop bombs, move from side to side, and try to overrun your bases. Hold them off—and score—by shooting them down. But, just as you think you've got it all under control, the action speeds up.

Choose the game speed, enemy bomb frequency and accuracy, shots on screen and the number of your bases. Move your base and simultaneously fire at the invaders—you cannot do this in most similar games. Full sound effects add even more excitement to the incredible speed and action of INVADERS FROM SPACE. Fun for all ages and skill levels.

Available for TRS-80* 16K Level II for only $14.95 on tape or $20.95 on disk.

*TRS-80 is a trademark of Tandy Corp.



✓34

Acorn

Software Products, Inc.

634 North Carolina Avenue, S.E., Washington, D.C. 20003

These and other popular Acorn programs are available now at fine computer stores. Ask for them.

DEALER INQUIRIES INVITED

René Descartes * Homer * Jeremy Bentham * Plato * John Adams * Pliny the Elder *

* Thomas Hobbes * Leibnitz * Ludwig van Beethoven * Michelangelo * Sigmund Freud * Galileo Galilei *

Computer\n

1: Device designed to execute a sequence of mathematical operations.

Education for the Home

Beginner's Russian

This package consists of three programs that graphically display the Cyrillic alphabet. The programs are arranged so that you progress from one to the next—building your knowledge as *you* progress. It includes instructions on proper pronunciation of the letters and even an introduction to simple Russian words.
Order No. 0136R $9.95

Everyday Russian

Everyday Russian will acquaint you with the Russian words relating to: foods, places to eat, everyday signs, and the names of common stores. You will also learn the order of the Cyrillic alphabet. Each of the three divisions of this package will teach you the words and then quiz you on comprehension. You can even practice typing in Russian, using your TRS-80 keyboard as a "Cyrillic typewriter."
Order No. 0137R $9.95

The Russian Disk

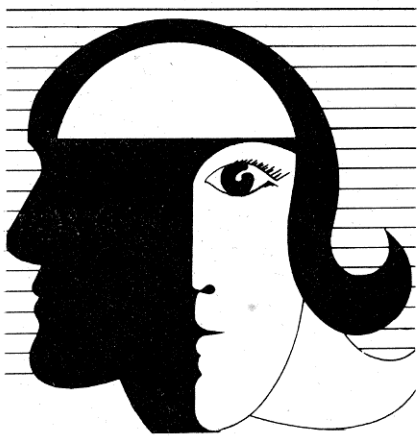
Now you can have *both* the Beginner's Russian and Everyday Russian packages on floppy disk! Requires an Expansion Interface with 16K and one disk drive.
Order No. 0212RD $24.95

Teacher

This program allows you to input any number of questions and answers. The computer will prepare tests, give quizes, provide up to three hints per question and even give (optional) graphic rewards for correct answers. Perfect for parents, teachers, or anyone faced with learning a lot of data in a short time.
Order No. 0065R $9.95

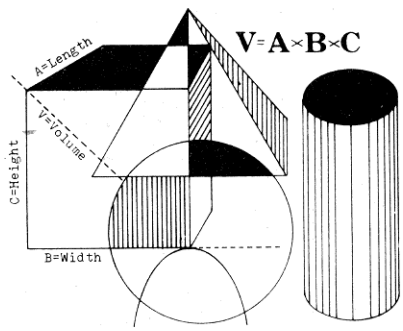
Wordwatch

Four programs for budding lexicographers, etymologists, or anyone else who uses words. In WORD RACE, you must choose the proper definitions. Find the misspelled word in HIDE N SPELL. Take a pre-recorded quiz in SPELLING BEE, in which the words are played aloud! Meet variations on proper spelling in SPELLING TUTOR.
Order No. 0111R $7.95



IQ Test

Are you smart enough to buy this package? IQ Test will administer and score an intelligence test in 30 minutes flat! There are three equivalent tests, each consisting of 35 questions, designed to test your general knowledge and problem solving abilities. Most of us claim a "touch of genius"—here's your chance to prove it!
Order No. 0157R $9.95



Archimedes' Apprentice

A tutorial software package that will teach you the formulas used to find the volume of *any* solid object. It covers parallelopipeds (cubes and rectangular solids), prisms, pyramids, cylinders, cones and spheres. It can even quiz you on how well you learned the lesson.
Order No. 0092R $9.95

Video Speed-Reading Trainer

You can increase your reading speed and comprehension. How? By practicing, that's how! This three-part program will flash characters or words on the screen, then you must echo what you saw. You can begin at a relatively slow rate, because the computer will advance your speed automatically as your speed and comprehension increase. It will train you with numbers, letters, words and phrases.
Order No. 0100R $9.95

Typing Teacher

A complete seven-part package that guides you from familiarization with the keyboard, through typing words (and phrases), to mastery of touch typing. Your video monitor becomes a bottomless page for typing practice!
Order No. 0099R $9.95

All packages listed are for the TRS-80 Model I Level II; they require 16K of memory and are cassette-based unless otherwise indicated.

Instant Software™

PETERBOROUGH, N.H. 03458
603-924-7296

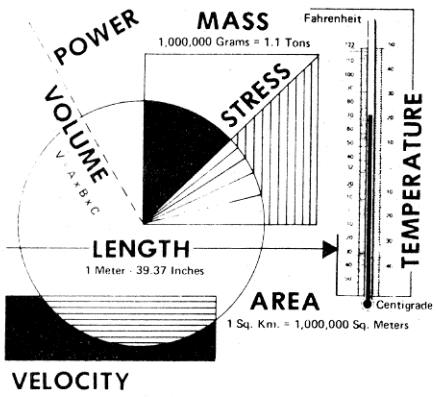
Rebbe Akiva * John Stuart Mill * Pythagoras * Niccolo Machiavelli * Sun Tzu Wu *

William Shakespeare * Thomas Edison * Aristotle * William of Occam * Thomas Jefferson * Baruch Spinoza * Sri Ramakrishna * Thomas Gainsborough * Hippocrates * Blaise Pascal * Socrates * Voltaire * El Greco * Aeschylus * Oliver Wendell Holmes * Ptolemy *

Education\n

1: The action or process of training and developing knowledge.

Education for the School



Omni-Calculator

This package is a brilliant solution to the problems involving the complex world of different units of measurement. It provides a rapid means of conversion from one unit of measurement to another in any of 10 categories: length, volume, mass, velocity, area, density, power, energy, pressure/stress, and temperature.

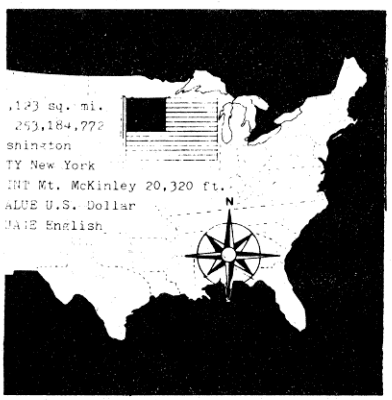
The Omni-Calculator is an invaluable package for scientists, electricians, teachers, doctors, surveyors, builders, plumbers, civil engineers, and pharmacists.

Order No. 0211RD $19.95

The Elements

This program can be used to introduce students to the periodic table of the elements or for review by students or adults who want to refresh their memory in chemistry. It includes each of the elements' name, atomic number, weight, and symbol, its acid base and normal physical state. Enrich your knowledge of science with the PERIODIC TABLE OF THE ELEMENTS package.

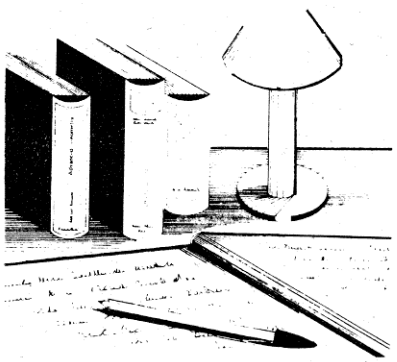
Order No. 0216R $9.95



Geography Explorer: U.S.A.

Travel the country and learn vital facts about each of our 50 states. This package offers the most fascinating way of learning yet. Learn each state's name, capital, largest city, nickname, etc.... And as a bonus, this package offers the capability of a light pen.

Order No. 0071RD $49.95



Teacher's Aide

Now you can have the benefits of Computer Assisted Instruction (CAI) in your own home. The Teacher's Aide program will let you create a teaching system for any conceivable subject. The program allows you to create a question and answer lesson (you can input up to 8000 characters per lesson). You can then save this lesson on the disk and create an entire sequence of lessons.

Your lessons can be tailor-made for you or your students. The options available are: (1) review the material prior to taking the lesson, (2) provide hints to help answer questions, and (3) offer a graphic display as a reward for correctly answering all the questions. The Teacher's Aide program will even allow for spelling errors!

The Teacher's Aide package is perfect for parents, teachers, and students who need the unlimited patience and undivided attention only a computer can provide. Readin', writin', and 'rithmatic will never be the same—now that you have the Teacher's Aide package from Instant Software.

This package requires the following minimum system:

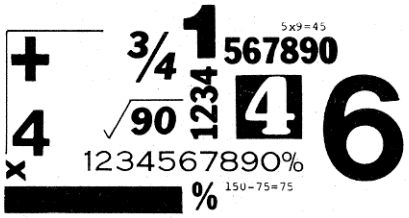
1. A TRS-80 Level II with 16K RAM.
2. An Expansion Interface with 16K RAM.
3. One disk drive.
4. Any compatible Disk Operating System.

Order No. 0214RD (disk-based) $39.95

All packages listed are for the TRS-80 Model I Level II; they require 16K of memory and are cassette-based unless otherwise indicated.

SEE YOUR LOCAL INSTANT SOFTWARE DEALER OR

Just Call Toll-Free 1-800-258-5473



Math Master

Math Master is an educational package designed to help you understand the math concepts of addition/subtraction/multiplication and division, through repetitive problem solving. This package also aids in fraction problem solving, and would be an asset to your program library.

Order No. 0257R $9.95

Instant Software™

PETERBOROUGH, N.H. 03458 603-924-7296

Reader Service—see page 290

The end of the beginning—some afterthoughts on style.

Into the 80's

Ian R. Sinclair
89 Alexandra Rd.
Sible Hedingham
Halstead, Essex, England C093NP

Seven months from the start of this series, you've lost your beginner's status. It's time to look at a few items which had to be left aside earlier, and the first of these is program planning.

Many BASIC programs grow untidily and haphazardly from an idea or from another program. We've all done this, looked at someone else's program and thought, "Hey—I could really make something out of this." After a lot of work you can have a program that pleases you, but it's what I call a Stein program—Frank N. Stein—made from bits and pieces, and full of GOTOs.

When you start programming, you're glad to write a program that works, and you don't really care too much about how it was planned and what it looks like. You should now start to care about these points, not least because there is a considerable saving in time that can be made.

I don't have much time to spend in front of the computer. Most of my program work has to be done in other places, at lunch breaks, where and when I have odd moments. Because computer time is precious, I don't want to spend it sorting out syntax errors, NEXT without FOR, and other needless errors. This is particularly important on the Level II TRS-80 because every time you edit you lose any variable values which are in the program, so the whole program must be run again from the start. It makes sense, therefore, to have all syntax errors sorted out before you enter a program.

Why *syntax* in particular? When the TRS-80 detects a syntax error, which might be very trivial, it hangs up the run, displays the SN error message, and then enters edit mode automatically, with the offending line number displayed and waiting for you to edit. If you make any attempt to edit, even L for LIST, you reset all your variables. On some programs this wouldn't matter, but if your program involved reading in data from tape or entering a number of items from the keyboard, you won't want to lose it if there's any way of avoiding it. Make a note of the line number and then press the RESET button; this takes you out of the EDIT mode without losing the variables.

You can now type GOTO (next line) and processing will take up from there, unless there is a NEXT whose FOR was in the previous line. If the line you lost had an important command in it, substitute with a direct command. For example, if the line read 510 PRRINT"VALUE IS";V: N = V↑2 + 2*C—6*L, and it hung up because of the double R in PRINT (when will you do something about that key bounce?), press RESET, type N = V↑2 + 2*C − 6*L ENTER, and then type GOTO 520, assuming that the next line is 520, and ENTER again. Your program should then continue. Practically all the commands of the TRS-80 can be used either from the keyboard directly or within a program with the same effect.

You can spend a lot of time at the keyboard sorting out flaws which never should have got that far. Program planning should make your keyboard hours more productive. All programs are wanted yesterday!

A program should start with an outline plan of what you want it to do. If it's a game program, you need to consider what the strategy of the game is to be, and write down the rules. This is the hardest part of any game program, and it's why there are several thousand versions of "Hangman." If you start with simple, established rules, you've saved yourself months of effort. You can't start programming until you know what you need to program.

Once you have a clear idea of what the program is supposed to do, write it down. It's only too easy to make a lot of alterations to a program which will leave you at the next session wondering what it was you wanted to do, and why you did it.

Lots of professional programmers use flowcharts, and flow-charting is urged on every trainee programmer. I dislike flowcharts. They complicate rather than simplify for me, causing too much visual clutter. You can find plenty of reading matter about flowcharts elsewhere—I'm going to describe how to work without them. To be fair, flowcharts can be very useful when you are working in other programming languages. I feel that they aren't really appropriate to BASIC.

I start by writing down what I expect the program to do—at what stages I need to put in information and at what stages I expect to see information on the screen (or the printer). This is my equivalent of flowcharting, but in words rather than in pictures.

Once I'm sure what I want the program to do, I sit down with stacks of paper. As I use a sheet, I title it and give it a page number. Next, I design any menu stages. I also note what is going to happen when each choice is made.

Construction

The next step is program construction! I usually go for a very short 10–20 line numbers) main program, with the choices are ar-

"POKE graphics need practice, but they run a lot faster than SET or CHR$( ) routines, and they are very useful when animations are needed."

ranged as subroutines, so that I can alter them as much as I want to later. Listing 1 shows an example of this—each part which might need changing is a subroutine, and the main program consists of only five lines. All the INKEY$ steps, YES/NO decisions, and so on are left as subroutines, since they can be standard subroutines which are used in several programs. I keep a tape of all my subroutines and run that in as a starter for any program I am entering.

If the title is short, the method of underlining I've shown in Listing 1 is quicker than using STRING$. If the complete program is long, it's a good idea to delete the instruction lines if instructions are not needed. This can be done immediately after the decision step by using the fact that a NO answer returns M = 2. A line such as 100 IF M = 2 then DELETE 8000-8100 will delete instruction lines in 8000 through 8100.

This leaves you free to design your instructions after the program is running as you want it. Any alterations you make then in the instructions won't affect the main program. If you made the instruction lines as lines 20 through 120, adding more instructions would have to be done carefully. In addition, it's difficult to follow what the program does when it gets cluttered with extra lines.

The rest of the program is designed in the same way, with a menu display followed by the INKEY$ routine letting you choose an item. This subroutine would normally include any error-trapping you needed. Line 40 is the menu subroutine cross-roads. Line 50 is needed because after any of these subroutines has been used, the program will return at line 50. We have to offer the choice of a return to menu or ending the program.

Not every program can be put into this form; there are programs which need no menu choices, and which use very few subroutines.

Once the main program has been written you can start designing the subroutines. Each of these should be treated just like a main program.

Watch Your Variables

Each time you make use of a variable, N, A$ or whatever it is, write down what you use it for and in what lines it's used. For example, it's easy to get into the habit of using N in FOR...NEXT loops. If you use N for anything else within a loop, you will wreck the loop. If you start a loop as: 2400 FOR N = 1 to 200: INPUT N$(N) and then follow it up with something like. 2420 PRINTCHR$ (23) L$(N): FOR N = 1TO1500: NEXT and then several lines later you have 2450 NEXT, don't be surprised if odd things happen.

At 2400, N will take the value 1 and you input the first string. At 2420, this string is printed, and a delay loop is used to keep the large letters on the screen for a time. The trouble is that the delay loop also uses N, so that at the end of the delay loop, N will be set to 1501. At line 2450, the NEXT command will find that N is 1501, much greater than the jump-out-of-loop value which was set in line 2400, so the loop stops. You could spend a lot of time wondering why only one value ever got itself input and displayed, especially if you save memory by writing long lines with the variables buried deep inside.

The rest of the program is designed in the same way, with a menu display followed by the INKEY$ routine letting you choose an item. This subroutine would normally include any error-trapping you needed. Line 40 is the menu subroutine cross-roads. Line 50 is needed because after any of these subroutines has been used, the program will return at line 50. We have to offer the choice of a return to menu or ending the program.

Not every program can be put into this form; there are programs which need no menu choices, and which use very few subroutines.

Once the main program has been written you can start designing the subroutines. Each of these should be treated just like a main program.

Watch Your Variables

Each time you make use of a variable, N, A$ or whatever it is, write down what you use it for and in what lines it's used. For example, it's easy to get into the habit of using N in FOR...NEXT loops. If you use N for anything else within a loop, you will wreck the loop. If you start a loop as: 2400 FOR N = 1 to 200: INPUT N$(N) and then follow it up with something like 2420 PRINTCHR$ (23) L$(N): FOR N = 1TO1500: NEXT and then several lines later you have 2450 NEXT, don't be surprised if odd things happen.

At 2400, N will take the value 1 and you input the first string. At 2420, this string is printed, and a delay loop is used to keep the large letters on the screen for a time. The trouble is that the delay loop also uses N, so that at the end of the delay loop, N will be set to 1501. At line 2450, the NEXT command will find that N is 1501, much greater than the jump-out-of-loop value which was set in line 2400, so the loop stops. You could spend a lot of time wondering why only one value ever got itself input and displayed, especially if you save memory by writing long lines with the variables buried deep inside.

Also note how each subroutine uses variables from the main program or from other subroutines, and what it does with them. For example, if you have a set of strings stored as array L$(N), and these are used by a subroutine and changed in the subroutine, make a note of this. If you need to use the unchanged value in another subroutine, you will have to use a different variable name for the changed value.

After using this method of constructing programs for some time, you'll have a good stock of useful subroutines. Some of these (neat printing routines, tabulated displays,

```
10 CLS:PRINT@154,CHR$(23)"TITLE":PRINTTAB(13)"=====":FO
     R N=1TO1500:NEXT:GOSUB200:REM  200 IS SUBROUTINE W
     HICH ASKS IS INSTRUCTIONS ARE NEEDED
20 CLS:PRINTTAB(35)"MENU":PRINT:PRINTTAB(2)"1. ENTER NE
     W DATA":PRINTTAB(2)"2. REPLAY DATA CASSETTE":PRINT
     TAB(2)"3. PROCESS DATA":PRINTTAB(2)"4. RECORD DATA
     ":PRINTTAB(2)"5. END PROGRAM"
30 GOSUB 500:REM INKEY$ ROUTINE TO FIND CHOICE
40 ON K GOSUB 1000,2000,3000,4000,5000
50 PRINT "DO YOU WANT TO RETURN TO THE MENU?":GOSUB 600
     :IF M=1 THEN 20 ELSE 5000:REM YES/NO RUTINE IS AT
     600
5000 END
```

Program Listing 1

```
10 FOR N= 17129 TO 17139:PRINT PEEK(N):NEXT
```

Program Listing 2

```
10 FOR N=17129 TO 17179:PRINT PEEK(N);"   ";:NEXT
```

Program Listing 3

". . . they make any machine language program look highly exotic until you get accustomed to them."

record and replay of packed data, etc.) are likely to be used in every program you write. Keep them together on a cassette, with a note of what they do and what line numbers are used. Make sure you also note what variables each subroutine uses, what variables have to be passed to it and what variables it passes back.

If you write programs which use many subroutines, then you can update and improve it easily. Got a tape replay subroutine which is too slow? Some day you'll come across a faster one, and you'll be able to rewrite it in subroutine form and use it to replace the old one. Even if these new routines need more lines, the subroutine methods allow you to leave plenty space so there's no need to try to shoe-horn a new routine in between lines 40 and 50, for example.

PEEK and POKE

Everything that goes on inside the TRS-80 involves the use of machine code. Each command used in BASIC calls up a machine code subroutine which in turn calls other subroutines which do the work. The BASIC used by the TRS-80 is *interpreted* BASIC, which means that when you RUN, each command calls up the machine code subroutines one by one as the program progresses. This is a lengthy and clumsy way of using BASIC, and mainframe machines use a program called a compiler, which converts all the BASIC commands of a program into one large machine code program, rather than operating piece by piece. Compiled BASIC runs a lot faster, but it's not easy to edit and chop around, which is why it's not used much in microcomputers.

A good machine language program runs a lot faster and takes up much less memory space than the equivalent BASIC program, which is why so many long programs are written in machine code. I want to show you how you can make use of machine language programs.

A machine language program consists entirely of numbers between 0 and 255. A lot of books and articles show these numbers in the hexadecimal scale, which is based on sixteen, as compared to the normal scale based on ten. Hexadecimal numbers use letters A through F to represent the decimal numbers 10 through 15, and they make any machine language program look highly exotic until you get accustomed to them.

The use of hexadecimal codes is a hangover from early microprocessor units, and the TRS-80 displays every number in ordinary decimal form. It therefore seems pointless to keep converting numbers into hexadecimal and back again just to represent machine code programs, which the computer converts into binary code anyway. A lot of information in the TRS-80 codes comes in a mixture of hex and decimal. Conversions always cause mistakes, so I work entirely in decimal unless I am writing new machine code programs, and I'll use the decimal codings exclusively in this article.

The Program Line

The PEEK command lets you find code stored at any memory location inside the computer. Using PEEK does not alter any of the codes, so you can PEEK to your heart's content. One very instructive PEEK is at a program instruction line, and Program Listing 2 lets you do just that. The number 17129 is the address number of the first byte of free memory into which BASIC programs are entered. When you PEEK there, you are looking at the first code of any program that has been entered. No matter what number you give to the first line of your program, the first code is always stored at 17129. Our program lets you look at the first ten code numbers.

"Avoiding short lines wherever possible. . . saves considerable amounts of memory, sometimes making the difference between being able to fit the program and getting the dreaded OM error signal."

For the moment, ignore the first two numbers in the series, and concentrate on the third and fourth, which are 10 and 0. These constitute the line number. Two codes are used because we can use line numbers up to 65529, which uses two bytes. The line number is stored in two parts: a lower part which comes first, and an upper part which comes second. The actual decimal number is determined by adding the lower part to 256 multiplied by the upper part—in this case, it's 10 + 256 × 0. Line 1000 would be 232,3, because 232 + 3 × 256 = 1000. Fig. 1 shows how to convert decimal numbers to numbers in TRS-80 coded form.

The first and second numbers are the address number of the *next* line, so that the TRS-80 can pick up its directions at the start of each line. The numbers in our example should be 9,67, because the next line should start at address number 9 + 67*256 = 17161. When you PEEK at a piece of completely unused memory, you will find that the codes are alternately 0 and 255, which are the numbers that are set into the memory by the power-on sequence. Some bytes (low in memory address number) are set to other values when the break key is pressed.

Whenever you type a line number, you use up five bytes of memory, consisting of a zero which is placed at the end of the previous line (to indicate that it has ended), the two bytes of the next line address, and the two bytes of the line number. Avoiding short lines wherever possible, even if it means using a lot of ELSE..IF statements, saves considerable amounts of memory, sometimes making the difference between being able to fit the program and getting the dreaded OM error signal.

Code number five on our list is 129. That's the TRS-80 code meaning FOR. You might expect this to be stored as three codes, the ASCII codes for F, O and R, but the TRS-80 uses memory-saving single codes for all its commands. Table 1 shows a list of the command codes.

After the FOR code, there's a blank (ASCII 32) because you typed in a blank between FOR and N (didn't you?). The computer does not need this space, and you can save memory by omitting all such spaces. There are only a few statements which can cause trouble if you do this to them—check the examples in the Manual.

N appears as the seventh item in our list, stored as its ASCII equivalent, 78. The eighth item is the code for =, which is 213. This is *not* the same as the ASCII code for =, because we're not using = as a character to be printed but as a command to be carried out. Since you left a gap between = and 17129, the ninth code number is 32.

The detective work begins to look interesting, and we would like to look beyond the tenth code. Program Listing 2 was a rather wasteful printing method, and we'll get more information on to the screen by using Program Listing 3, which prints the codes in lines separated by commas. We can now see the whole program in its coded form.

We recognize the first nine codes from the previous examples. The numbers 17129 and 17179 are stored in ASCII form, needing five bytes each. This is a wasteful method: if you use a number such as 17129 frequently in a program, you can save memory by declaring it as an integer variable. Use a command such as A% = 17129; the % sign means that the number is an integer, and can be stored in two bytes using the code method we have seen used for line numbers. Alternatively, by using DEFINT A at the start of the program, we could command A = 17129 and achieve the same

continued to page 98

✓ *Reader Service—see page 290*

Dec. Code	BASIC Keyword	Dec. Code	BASIC Keyword
129	FOR	167	LOAD
130	RESET	168	MERGE
131	SET	169	NAME
132	CLS	170	KILL
133	CMD	171	RSET
134	RANDOM	172	RSET
135	NEXT	173	SAVE
136	DATA	174	SYSTEM
137	INPUT	175	LPRINT
138	DIM	176	DEF
139	READ	177	POKE
140	LET	178	PRINT
141	GOTO	179	CONT
142	RUN	180	LIST
143	IF	181	LLIST
144	RESTORE	182	DELETE
145	GOSUB	183	AUTO
146	RETURN	184	CLEAR
147	REM	185	CLOAD
148	STOP	186	CSAVE
149	ELSE	187	NEW
150	TRON	188	TAB
151	TROFF	189	TO
152	DEFSTR	190	FN
153	DEFINT	191	USING
154	DEFSNG	192	VARPTR
155	DEFDBL	193	USR
156	LINE	194	ERL
157	EDIT	195	ERR
158	ERROR	196	STRING$
159	RESUME	197	INSTR
160	OUT	198	POINT
161	ON	199	TIME$
162	OPEN	200	MEM
163	FIELD	201	INKEY$
164	GET	202	THEN
165	PUT	203	NOT
166	CLOSE	204	STP
205	+	231	CVS
206	−	232	CVD
207	*	233	EOF
208	/	234	LOC
209	↑	235	LOF
210	AND	236	MKI$
211	OR	237	MKS$
212	>	238	MKD$
213	=	239	CINT
214	<	240	CSNG
215	SGN	241	CDBL
216	INT	242	FIX
217	ABS	243	LEN
218	FRE	244	STR$
219	INP	245	VAL
220	POS	246	ASC
221	SQR	247	CHR$
222	RND	248	LEFT$
223	LOG	249	RIGHT$
224	EXP	250	MID$
225	COS		
226	SIN		
227	TAN		
228	ATN		
229	PEEK		
230	CVI		

Table 1

1. Divide the number by 256 and discard anything which follows the decimal point. What's left is the upper byte (which always comes second in the coding sequence). Multiply the upper byte number by 256 and take the result away from the original number. The answer is now the lower byte (which comes first in the coed sequence).

EXAMPLE: Convert 23478 to TRS-80 code
3478/256 = 91.710937. . . . all we want is the 91 (upper byte)
91 × 256 = 23296. . . . . take this away from 23478 and we get 182 (lower byte)
The number in TRS-80 coding is therefore 182 91

```
10 FOR N=17129 TO 32767: IF PEEK(N)<>136 AND PEEK(N+1)<>19 THEN
NEXT ELSE PRINT N
20 END
5000 PRINT"THIS IS LINE 5000"
READY
>_
```

Program Listing 4

```
10 FOR N=0 TO 99:READ J:POKE 32667+N,J:NEXT
20 DATA:REM NEED 100 NUMBERS BETWEEN 0 AND 255
```

Program Listing 5

```
1 CLEAR300:ZZ$="":FOR I=1TO107:READ J:ZZ$=ZZ$+CHR$(J):N
     EXT:POKE16422,PEEK(VARPTR(ZZ$)+1):POKE 16423,PEEK(
     VARPTR(ZZ$)+2)
5 REM INTO THE 80'S FIG 7.8
5000 DATA255,243,121,254,13,40,3,254,32,216,245,229,197
     ,6,9,55,245,245,33,1,252,205,33,2,33,222,0,43,124,
     181,32,251,241,31,245,48,19,33,2,252,24,19,14,3,17
     5,13,40,2,24,219,0,0,0,0,24,47,198,0,33,1,252,205,
     33,2,0,0,33,222,0,43,124,181,32,251
5001 DATA16,212,17,222,0,203,74,40,11,33,2,252,205,33,2
     ,27,122,179,32,251,241,241,254,13,40,198,183,40,19
     7,193,225,241,201
```

Program Listing 6

result. Each time we need 17129 in the program, we can now use A, saving memory space.

The TO part of the FOR...TO...NEXT loop is stored, as usual, as a single code number. Conversion of certain words, like FOR, NEXT, RUN, and so on, into single byte numbers means that you have to be careful *not* to use these words as variable names in a program.

By checking the code numbers in Table 1, and the ASCII codes, you can trace how the instructions are coded—but this is just coded BASIC, not true machine code. The code numbers on a BASIC line are instructions to the interpreter. This introduces you fairly painlessly to the idea of instructions stored as number codes, and it shows beautifully how the TRS-80 line is coded.

You can now see how it's possible to change line numbers. The first line number will always be found at memory locations 17131 and 17132. The bytes in 17129 and 17130 will indicate the address of the start of the next line; the third and fourth along from that number will give the next line number and so on.

Program Listing 4 searches through 16K memory and stops when it finds the address of line 5000. 5000 in TRS-80 code is 136,19, so we set the IF...statement to detect the sequence 136,19 anywhere in the memory. If you have a reference to this number as a variable earlier than line 5000, you'll turn up the wrong address, but it's simple to modify the program so that it lists every reference to 5000 (by adding a :NEXT at the end of line 10).

PEEK allows us to find the codes that are stored in the memory, which are always printed in decimal form. One of the functions of the type of program called a monitor or bug is to print out these codes, usually in hexadecimal form, as a "dump". As part of my "help-stamp-out-hex" campaign, I don't use a monitor of this type.

Poke

POKE is the companion command to PEEK, and has to be followed by two numbers. The first number following POKE is

1	2
4	8
16	32

5 = 4+1

SO POKE 15360,5 LIGHTS UP THE TOP LEFT CELLS (4&1)

Table 2

PROFESSIONAL

HALF A MILLION TAX RETURNS CAN'T BE WRONG!
(OR THEY HAD BETTER NOT BE)

INCOME TAX SYSTEM
FOR TRS-80* MODEL I OR II

Our system, which prepared 500,000 1979 returns, features the following:

1. Full interactive user control, **in tax-form language only,** line-by-line.
2. Screen display of full 1040 and all schedules, prior to printout.
3. Change of a single amount item automatically changes and re-computes entire return.
4. All printout formats IRS and state approved.
5. Stores Preparer's Identification for automatic printing at bottom of page 2.
6. Built-in Validation Check tests entire system, hardware and software.
7. Special Printer Adjustment routines, Line Length, etc.
8. Selection of closed or open output formats—for standard Form 1040 or open name-box types.
9. **Software control of text position on page.** Makes forms-alignment simple. Permits use with non-adjustable printers.
10. Fills in pre-printed Forms or you can use overlays. Your choice.
11. Automatically computes: Tax - SDI Overpayment - Wages Total from W-2's - Earned Income Credit - Income Averaging - Maximum/Minimum Tax - Least Tax Method - All Percentage of Income Limitations - All Fixed Limitations - many, many more.
12. Full support through the tax season — no charge.
13. Inexpensive yearly updates in accordance with tax-law changes.
14. Modular construction — lets you order only the type and size system you need.

PRICING STARTS AT $189.95 (1040 & SCHEDULE A)

25-PAGE DESCRIPTIVE MANUAL $7.50 (Refunded on Order)

MINIMUM SYSTEM REQUIRED: MODEL I, 32K, 1 DISK DRIVE

*TRS-80 IS A TRADEMARK OF TANDY CORP.

CONTRACT SERVICES ASSOCIATES

✓10

706 SOUTH EUCLID **ANAHEIM, CA 92802**

TELEPHONE (714) 635-4055

★ ★ ★ 20 YEARS OF SERVICE ★ ★ ★

"After using this method of constructing programs. . . you'll have a good stock of useful subroutines."

the address which is to be used. The second number, separated from the first by a comma, is the number between 00 and 255 which is to be inserted into that address in memory. Unlike PEEK, POKE changes the code stored in memory (unless you have used identical numbers), so it's a command you have to be careful about. You can't change the ROM which operates BASIC, but you can change a lot of codes elsewhere and end up with some very strange effects. A lot of poorly-planned POKE instruction will cause the computer to start up again, with the words MEMORY SIZE? appearing. Unless you use the reset button at the back, you will lose any program which you had in memory, including machine code programs as well as BASIC.

Most machine code programs are loaded from system tapes, and we covered the techniques for loading these in Part 2. It always irks me to have to use a system tape for a short program, so I've rewritten common routines as BASIC programs, by using POKE. If I POKE a code number to a place in memory, it's as surely there as if I had put it there from a system tape. Program Listing 5 demonstrates the form of the BASIC program needed—this reads 100 bytes from a data line and POKEs them one by one into memory locations 32667 upwards.

If the machine language program is short (less than 255 bytes), a more convenient method, illustrated in the Level II Manual, is reading the characters into a string. Program Listing 6 shows an example of this for a printer routine. The complete machine code program exists as the long string ZZ$, and the program is accessed by using the address of the string in memory. The advantage of using a string for this purpose is that it doesn't need any answer to the MEMORY SIZE question, so nothing goes wrong if you forget to reserve memory. The disadvantage is that the address of a string in memory is not fixed. If the memory starts to fill up, the strings will be shifted about, and the computer will keep a track of each. If a string gets shifted between two of the VARPTR instructions, however, the memory will get completely scrambled. For short routines and programs which use very little memory, it's fine.

For either of these methods, you still have to obtain machine code in the form of data lines. There are two ways of doing this. The first is to print out a decimal dump of the machine code, using a short BASIC program like Listing 5, setting the memory address numbers to the first and last addresses of the machine code program which you placed in protected memory. This works well if the program is only twenty bytes or so, but it becomes decidedly tedious for longer programs.

For longer programs I use a very neat little program which reads the machine code and creates data lines directly. The rest of the program then erases, leaving the data lines, so a new program can be written around it. This program is available in England from A.J. Harding (Molimerx), 28 Collington Avenue, Bexhill-on-Sea, East Sussex, England. The program is also available from associates in the U.S.

POKE Graphics

You may also use the POKE instruction to obtain quick graphics. The video display of the TRS-80 is memory mapped, which means that every part of the screen corresponds to a piece of data in the memory—yes, there are Brand X computers which don't do this! Because of the memory mapping, we can create shapes on the screen through POKE instructions to video memory, which runs from 15360 to 16383.

Address 15360 controls the top left-hand side of the screen. The addresses go in bundles of 64 per line on the screen for 16 lines to 16383, which is at the bottom right-hand side. At each memory address, you can POKE numbers which will light up a screen block, just as SET does. Fig. 2 shows what numbers correspond to which cells. If you type POKE 15360,5, you will light up the cells shown in the example at that part of the screen.

POKE graphics need practice, but they run a lot faster than SET or CHR$() routines, and they are very useful when animations are needed. To continue your study on POKE graphics, there's a book called *An Introduction To TRS-80 Graphics* by Don Inman, and it comes straight off the *80 Microcomputing* bookshelf.

Program Listing 6 introduced a new TRS-80 BASIC function (and you thought you knew it all!). VARPTR(variable) is a form of the PEEK command, and is applied to a named variable. If you have a number variable such as N in a program, the command PRINT VARPTR(N) puts the number, which is the memory address at which the variable N is stored, on the screen. If you then enter PRINT PEEK(VARPTR(N)), you'll get one byte of the variable itself—the byte which is stored at the VARPTR(N) address number.

Integer variables need two bytes, so you'll have to PEEK(VARPTR(N) + 1) as well, and find the value by using the well-worn formula: lower byte + 256 × upper byte. Single precision numbers need four bytes of storage, double precision numbers need six bytes of storage, but the address of the first byte can always be obtained from the VARPTR command.

There's more information which you can get using the VARPTR command, but the address numbers are the most important. If

Surplus TRS-80*
RAM Memory Chips

Dealer and Quantity Pricing Available.

DUE TO CHANGES IN PRODUCT STRUCTURE AMERICAN BUSINESS COMPUTERS IS OFFERING SEVERAL THOUSAND 200 NANOSECOND RAM MEMORY CHIPS AT CLOSE-OUT PRICES.

AMERICAN BUSINESS COMPUTERS GUARANTEES ALL MEMORY CHIPS TO BE BRAND NEW AND FREE FROM DEFECTS FOR 180 DAYS. PRICE ($35) INCLUDES 8 CHIPS. QUANTITY DISCOUNTS AVAILABLE.

$35
(per 16K set) ✓484

TEAC Disk Drives

FULLY TRS-80 COMPATIBLE 40 TRACK DRIVES WITH POWER SUPPLY AND CABINETS

275.00 ✓483

TWO-DRIVE UNITS ONLY 540.00

THE MX-80 WITH DISPOSABLE PRINT HEAD.

✓317

A LOT OF PRINTERS COSTING A LOT MORE CAN'T TOUCH THE MX-80'S PERFORMANCE. IT GIVES YOU A CHOICE OF 40, 80, 66 OR 132 COLUMNS OF PRINTING IN AS MANY AS FOUR DISTINCT PRINTING DENSITY MODES. MORE THAN HALF THESE MODES GIVE "CORRESPONDENCE QUALITY" PRINTING. CALL FOR QUANTITY DISCOUNTS.

SAVE $50!

COMPUTERIZED MUSICAL CAR HORN

Hooks to your car in minutes, and plays the most recognizable bars of your favorite tunes, such as:

$99.95
(Regular Price $149.95)

"Starwars" Theme
"Pink Panther" Theme
"Lone Ranger" Theme
Anchor's Aweigh
Star Spangled Banner
Folsom Prison Blues
Ghost Riders in the Sky
King of the Road
"Dallas" Theme
USC Fight Song

Plus 54 more well-known show and college songs.
You'll Love It! Satisfaction Guaranteed.
(add $4.95 shipping & handling) ✓396

OKIDATA MICROLINE 80 PRINTERS

80 characters per second ● Up to 3 part copy ● 9 x 7 dot matrix ● 80, 132 columns ● 64 shapes for graphs and charts ● 6 or 8 lines per inch ● Head guaranteed for 200 million characters ● Friction feed standard—Pin feed optional.

479.00

American Business Computers believes that the Okidata Microline 80 printer is one of the finest printers available. If your needs call for a small reliable printer, contact us **TODAY**. Quantity discounts available.

VISA

For COD ✓397
Add 5%

master charge

HARD DISK DRIVES

For TRS-80* Model II Users ✓192

Up and running—and available for immediate delivery

AMERICAN BUSINESS COMPUTERS IS NOW ABLE TO OFFER HARD DISK (WINCHESTER) DISK DRIVES FOR SALE. THESE DRIVES ARE AVAILABLE FOR TRS-80 MOD II, TRS-80 MOD I, S-100, AND APPLE COMPUTERS.

SEVERAL DIFFERENT DRIVES AND CONTROLLERS ARE AVAILABLE FOR THE TRS-80 MOD II. ALL DRIVES ARE SUPPLIED WITH HARD DISK CPM OPERATING SYSTEMS. CAPACITIES RANGE FROM 5-66 MEGABYTES. CALL OR WRITE FOR PRICES.

32K-EXPANSION INTERFACES

SPECIAL OFFERING. Due to a very special purchase, American Business Computers is able to offer a limited number of Radio Shack* Expansion Interfaces at the lowest price ever.

For TRS-80* Model 1

399.95

American Business Computers guarantees Expansion Interfaces to be Brand New—still in original boxes with original documentation and in perfect working condition.

AMERICAN BUSINESS COMPUTERS
118 SOUTH MILL ST. ✓322
PRYOR, OKLA. 74361
918-825-4844

*TM Tandy Corp.

WE HONOR master charge

For COD service add 5%.

you use the VARPTR command on a string variable, PEEK(VARPTR(string)) returns with the length of the string (a number of bytes not exceeding 255). The address of the string in memory is obtained from VARPTR(string) + 1, the lower byte, and VARPTR(string) + 2, which is the higher byte. This is the scheme which was used in Listing 6.

An important point about routines held in a string is that you must not erase the data or POKE instructions, because the TRS-80 resets all string variables each time you EDIT, CLEAR or RUN. If the machine code program is in high protected memory, then only switching off or giving a new answer to the MEMORY SIZE question will delete it.

USR(0) is used to insert a machine code program in the middle of a BASIC program. For example, suppose you have the Radio Shack KBFIX program bytes stored in high memory at address 32600, but the program has not been run. The problem is how to make the program run without having to go back to the SYSTEM command, type a slash, and enter the number 32600. The solution takes two steps; first place the address of the start of the machine code into reserved memory at addresses 16526 and 16527. This lets the computer know where to find the machine code program. To go into the machine code program, use the command USR(0) with some other command, such as PRINT. The statement PRINT USR(0) would result in the machine code program being run immediately after this statement was encountered. The PRINT part of the command is a dummy command, there only because the computer refuses to recognize USR(0) as an instruction by itself. Statements such as A = USR(0) are equally acceptable.

The manual briefly illustrates the way to pass a value from BASIC to the machine code program and back again, and wisely remarks that it's better not to attempt this unless you have a pretty good grasp of machine code and how to use it. That's just about the understatement of the decade. Presumably someone at Radio Shack knows in detail what the procedures are, so why they don't publish an additional manual is quite beyond me.

We're now approaching the end of the last part in this series. I've purposely omitted quite a few instructions which you'll find useful later, simply because they're not particularly useful to you at this stage—perhaps, Editor willing, I'll write a few articles later. The time has now come to sort out a few items which have been left over, and to give some advice on where to go from here, now that the keys of your TRS-80 look a little dull with wear.

By this time you've probably further explored the EDIT capabilities of the TRS-80. The manual is useful, with helpful examples, so you'll appreciate how powerful these EDIT subcommands are. Making full use of them can greatly reduce the time you spend programming, but you have to memorize the commands. Remember that whenever you use an EDIT command, you will lose all the values of variables.

Don't type a lot of precious data into a program until you're sure that all the syntax errors are cleared, because the computer goes into edit mode automatically when a syntax error is detected. You can prevent this in various ways: remember to press Q (for quit) whenever a syntax error appears, or a line such as 2 ON ERROR GOTO 5000 and put STOP in line 5000 early in your program. If any error occurs, the program will jump to line 5000 and break without losing variables. You can then find the error code number by typing PRINT ERR/2 + 1; the manual has a list of the error codes. The line number of the error can be found by typing PRINT ERL. These error-trapping routines can be used inside programs also to help break out of errors which arise inevitably from the program, like reading too many data bytes; examples are given in the manual.

Where do you go from here? There's any amount of BASIC programming to do. Even if you run out of BASIC programs for your own use, there's a fair chance that there will be hundreds of folk around you who need a BASIC program but have no idea of how to write one. Lots of people earn a respectable living by writing programs, or by adapting existing programs, and as your skill improves you might find (as I do) that this is interesting and rewarding work, daylight or moonlight.

The other way you can go is into machine language, which is a much more difficult path. If, like me, you started computing in machine language before BASIC was invented, the path is easier, but for the complete beginner, the problem is to find a book which starts right at the beginning.

Lance Leventhal has written a very useful book on programming the Z-80, which is the microprocessor that is used in the TRS-80, but we lack a real start-from-zero book on machine language programming using the TRS-80. I keep reading about a book by William Barden, but it never appears at my local Tandy store. With regret, I must report that the Radio Shack books which come with their Editor/Assembler package are not for the beginner, but the articles which are appearing in *80 Microcomputing* as I write this should be a step in the right direction. I have also found a book called *Machine Language Programming from the Ground Up*, by Hubert S. Howe, Jr., which is excellent. For some reason, it's published in England by A.J. Harding, but since Mr. Howe lives at New City, N.Y., just down the Palisades Parkway and beyond Nanuet, the book just has to be available on this side of the Big Pond as well. Look out for it.

One final point. In this business you never stop learning—no one ever knows it all. No matter how long you have been using the TRS-80, you'll be able to thumb through *80 Microcomputing* some day and be struck dumb by some piece of information or some smart subroutine that had never struck you before. That's the best thing of all because for me, while I'm learning, I'm living. So long.■

NEWDOS/80®

VTOS 4.0®

DOUBLE-ZAP/II™

Unleash your **NEWDOS/80** or **VTOS 4.0** power into double density!

Double-Zap is a disk program which zaps **NEWDOS/80** or **VTOS 4.0** for double density operation when used in conjunction with the **PERCOM DOUBLER**.

Double-Zap will completely zap **NEWDOS/80** on a one or two drive diskette, it will completely zap **VTOS 4.0** on a two-drive disk system, and for both **NEWDOS/80** and **VTOS 4.0** it will run single and double density, mix and match.

Double-Zap provides you with ***ADR*** **(Automatic Density Recognition)**.

Acting as an extension to **NEWDOS/80** or **VTOS 4.0**, ***ADR*** allows all drives to be used without regard to whether they contain single or double density diskettes. With ***ADR*** each drive is changed dynamically to match the density of the diskette that it contains. For dedicated applications it is also possible to "Lock" a drive into permanent single or double density operation.

After Double-Zap has run, you will get an extra 64,000+ bytes on the original diskette including the original programs from either **NEWDOS/80** or **VTOS 4.0**. Double-Zap will read any single density **TRSDOS**, **NEWDOS** or **VTOS** files and convert them to double density. Double-Zap is the ONLY double density conversion for **NEWDOS/80** and **VTOS 4.0** authorized by PERCOM DATA to be used with the **PERCOM DOUBLER**, **Required 32k RAM**.

Double-Zap I	runs double density only	**$29.95**
Double-Zap II	for NEWDOS+ owners	**$39.95**
Double-Zap II	runs single and double density	**$49.95**

Software Etc... ✓42
1839 Chamberlain Drive,
Carrollton, Texas 75007.
Phone Orders: (214) 492-0515

Get into INKEY$ and sing the praises of promptless programs.

Do Not Enter

J. I. Weintraub
690 Mtn. View Rd.
El Cajon CA 92021

When I upgraded my TRS-80 from Level I to Level II, I entered the Target Practice game (page H/5 of the manual). The same illustrates the INKEY$ function. Cute, I thought.

As I continued to study the manual in relation to the needs of my students, I discovered that INKEY$ could be one of the most valuable educational programming functions available. Interestingly enough, the function is not generally available on other small computers, where machine language or PEEKs and POKEs must be used to achieve the same result.

INKEY$ is an alternative input procedure to the enter key. More than that, it allows you to define the time span in which a response must be made. In basic fact programs, for example, a student exhibits mastery by responding within a few seconds, without counting on his fingers or using other artificial means to arrive at the answer. Other kinds of programs need this function also. For example, if the student doesn't respond at all, INKEY$ can detect this and provide an alternative procedure. Without it, *nothing* would happen.

Program Listing 1 is an actual program that demonstrates the use of INKEY$ in an addition drill. You will note that the problems and their answers are in data statements.

Setting C$ = INKEY$ in line 70 is necessary in order to clear any input that might be residing in INKEY$. The need for this will become clearer as we go on, but at this point you should know that if you accidentally press the enter key (or any key, for that matter), INKEY$ would retain this input until you call the function again. At this point you would be giving INKEY$ an erroneous input, because it would receive the left-over information.

Line 80 prints the problem. Lines 90 and 125 set the time limits for the first and second digits of the answer. Lines 95 and 130 wait for the input. If there is none for the do-loop number of increments, lines 100 and 135 take over and the program goes to line 190.

The ASCII code for the left arrow is 8, and for enter it is 13. Lines 105, 110, 140, and 145 reprint the problem if either of these keys is struck. Pressing the left arrow would result in an error unless we did this. Lines 105 and 140 do not actually backspace; by reprinting the problem they give the appearance that a backspace had occurred.

Line 115 skips the routine for the second digit if the answer is a one-digit number. Line 120 prints the first digit of a two-digit answer. Note the semicolon, which causes the second digit to print immediately after the first. If there is no second digit, the program goes to line 160, printing the single digit answer on the screen. Unlike an INPUT routine, the input to an INKEY$ function does not automatically appear on the screen. In most cases this is fine, but when a child is typing his answer you certainly want him to be able to see what he is doing.

Line 150 combines the two separate digits into a single numeral, and line 165 compares it to the answer in memory. Note especially that both the INKEY$ input and the value in memory are string variables.

In Listing 2, the addends in the addition problems are generated at random. To avoid comparing a string to a numerical value, line 165 converts the string input into a numerical value, which is then compared to the correct answer in line 166. Listing 3 deals with words of different lengths. The student sees three words and must type the word that comes first in alphabetical order.

Line 76 sets the maximum number of letters in a word. E may be set to any number larger than the longest word in the program. Lines 87 and 160 set up a loop containing E increments. Should the student input the maximum number of letters, the program will go to line 165.

Line 120 will print each new letter in turn, just as if a regular INPUT had been used; line 155 "absorbs" each new letter into a new string, B1$.

The student presses enter when he has completed his input. However, should the time run out before enter is pressed, line 100 will send the program to line 190 and the student is informed that his time is up.

The time limits used with the INKEY$ function must be determined by trial and error with the students who are using the program. In all three examples the FOR X = 1 TO something could be changed to use a variable; FOR X = 1 TO T, where T is set by the teacher at the beginning of the program as I explained in the section dealing with timing loops.

"Records themselves do not prove that a child has learned anything. . ."

Record Keeping

Records themselves do not prove that a child has learned anything; without records we have no way of knowing how the child has interacted with the program.

There are several reasons for building a record-keeping system into educational programs. First is to determine the student's mastery of the lesson. Second, the student himself should be aware of his progress. Another reason is to determine the appropriate level at which the student should be working.

Programs that keep records do not require the services of a teacher or aide to grade papers or to keep written records. Several record keeping systems will be illustrated in the following material: recording scores; keeping track of problems missed, for later study; determining whether the student should stay at the same level, move up a level, or move down a level; using hardcopy print-outs with a line printer; keeping records on a data tape using a cassette recorder; and keeping records on a disk.

```
10 CLEAR500
15 CLS
25 PRINT STRING$(63,61)
30 PRINT"THE INKEY$ FUNCTION--EXAMPLE 2"
35 PRINT:PRINT"     QUESTIONS GENERATED AT RANDOM"
40 PRINT STRING$(63,61)
45 PRINT "PRESS <ENTER> TO BEGIN.":INPUT A$
50 E=RND(10)-1:F=RND(10)-1
70 X=0:J=J+1: C$=INKEY$
75 CLS
80 PRINT@7*64, "(";J;")  ";E;" + ";F;" ="
85 PRINT"TYPE THE ANSWER: ";
90 FOR X=1 TO 450
95 B$=INKEY$:IF B$=""THEN NEXT X
100 IF X>450 THEN 190
105 IF B$=CHR$(8) THEN 80
110 IF B$=CHR$(13) THEN 80
115 IF E+F<10 THEN 160
120 PRINT B$;
125 FOR X=1 TO 150
130 B1$=INKEY$:IF B1$=""THEN NEXT X
135 IF X>150 THEN 190
140 IF B1$=CHR$(8) THEN 80
145 IF B1$=CHR$(13) THEN 80
150 PRINT B1$
155 B$=B$+B1$:GOTO 165
160 PRINTB$
165 P=VAL(B$)
166 IF P<>E+F THEN 180
170 PRINT "CORRECT!"
175 GOSUB 215: GOTO 50
180 PRINT "WRONG ANSWER."
185 GOSUB 215:GOTO 50
190 PRINT"     TIME'S UP!":GOSUB 215:GOTO 50
195 PRINT@7*64,"THIS IS THE END OF THE LESSON."
200 PRINT"GOOD-BYE FOR NOW."
205 PRINT "GO BACK TO YOUR CLASS."
210 FOR X=1 TO 2000: NEXT: RESTORE   'USE 9000 IN PRACTICE
211 J=0:GOTO 50
215 FOR X=1 TO 1000:NEXT
220 IF J>4 THEN 195
225 RETURN
```

Program Listing 2

```
10 CLEAR500
15 CLS
25 PRINT STRING$(63,61)
30 PRINT"THE INKEY$ FUNCTION--EXAMPLE 1"
35 PRINT:PRINT"     QUESTIONS AND ANSWERS IN DATA"
40 PRINT STRING$(63,61)
45 PRINT "PRESS <ENTER> TO BEGIN.":INPUT A$
50 I=0:J=0
55 FOR I=1 TO 5
60 READ Q$(I),A$(I)
65 NEXT I
70 X=0:J=J+1: C$=INKEY$
75 CLS
80 PRINT@7*64, "(";J;")  ";Q$(J)
85 PRINT"TYPE THE ANSWER: ";
90 FOR X=1 TO 450
95 B$=INKEY$:IF B$=""THEN NEXT X
100 IF X>450 THEN 190
105 IF B$=CHR$(8) THEN 80
110 IF B$=CHR$(13) THEN 80
115 IF LEN(A$(J))=1 THEN 160
120 PRINT B$;
125 FOR X=1 TO 150
130 B1$=INKEY$:IF B1$=""THEN NEXT X
135 IF X>150 THEN 190
140 IF B1$=CHR$(8) THEN 80
145 IF B1$=CHR$(13) THEN 80
150 PRINT B1$
155 B$=B$+B1$:GOTO 165
160 PRINTB$
165 IF B$<>A$(J) THEN 180
170 PRINT "CORRECT!"
175 GOSUB 215: GOTO 70
180 PRINT "WRONG ANSWER."
185 GOSUB 215:GOTO 70
190 PRINT"     TIME'S UP!":GOSUB 215:GOTO 70
195 PRINT@7*64,"THIS IS THE END OF THE LESSON."
200 PRINT"GOOD-BYE FOR NOW."
205 PRINT "GO BACK TO YOUR CLASS."
210 FOR X=1 TO 2000: NEXT: RESTORE: GOTO 50  'USE 9000 FOR STUDENT
    S
215 FOR X=1 TO 1000:NEXT
220 IF J>4 THEN 195
225 RETURN
230 DATA "2 + 2 =","4","9 + 9 =","18","3 + 8 =","11"
235 DATA "5 + 9 =","14","8 + 1 =","9"
```

Program Listing 1

Recording Scores

Your programs will check a student's response against the correct answer to a problem. The program will then tell the student whether or not the response was correct. At this point you can count correct and incorrect answers. See Program Listing 4.

Definition of variables: A = the student's response (line 60); B = the correct answer (lines 55 & 65); R = the number of correct answers (line 90); W = the number of incorrect answers (line 75).

W and R are incremented each time the student responds. Line 105 prints this information at the end of the lesson.

This listing is an example of a primitive method of scorekeeping. In Listing 4 no attention was given to the need of the student who answered incorrectly. One way to correct this is to repeat any problem the student misses. Program Listing 5, line 80, returns the program to the missed item each time the student misses it.

In this case, the total answers recorded will exceed the number of problems in the lesson because wrong answers are counted each time they are wrong and counted again when they are answered correctly. There are cases in which this information is

"Recording correct and incorrect answers does not produce a diagnosis of the child's actual needs."

```
10 CLEAR500
15 CLS
25 PRINT STRING$(63,61)
30 PRINT"THE INKEY$ FUNCTION--EXAMPLE 3"
35 PRINT:PRINT"     VARIABLE LENGTH WORDS"
40 PRINT STRING$(63,61)
45 PRINT "PRESS <ENTER> TO BEGIN.":INPUT A$
50 I=0:J=0
55 FOR I=1 TO 5
60 READ Q$(I),A$(I)
65 NEXT I
70 B1$="":X=0:J=J+1: C$=INKEY$
75 CLS
76 E=LEN(A$(J))
80 PRINT@7*64, "(";J;")  ";Q$(J)
85 PRINT:PRINT"TYPE THE WORD THAT SHOULD COME FIRST"
86 PRINT"IN APLPHABETICAL ORDER."
87 FORE1=1 TO E
90 FOR X=1 TO 150
95 B$=INKEY$:IF B$=""THEN NEXT X
100 IF X>450 THEN 190
105 IF B$=CHR$(8) THEN 80
110 IF B$=CHR$(13) THEN 80
120 PRINT B$;
155 B1$=B1$+B$
160 NEXT E1
165 IFB1$<>A$(J) THEN 180
170 PRINT "     CORRECT!"
175 GOSUB 215: GOTO 70
180 PRINT "     WRONG ANSWER."
181 PRINT"THE ANSWER IS... ";A$(J);"."
185 GOSUB 215:GOTO 70
190 PRINT"     TIME'S UP!":GOSUB 215:GOTO 70
195 PRINT@7*64,"THIS IS THE END OF THE LESSON."
200 PRINT"GOOD-BYE FOR NOW."
205 PRINT "GO BACK TO YOUR CLASS.":PRINT:PRINT:PRINT:PRINT
210 FOR X=1 TO 2000: NEXT: RESTORE  'USE 9000 IN PRACTICE
211 J=0:GOTO 50
215 FOR X=1 TO 1000:NEXT
220 IF J>4 THEN 195
225 RETURN
230 DATA "COME   RAIN   WOOD","COME","MOTHER   BIG   AROUND"
235 DATA "AROUND","RUN   GO   SEE","GO","JUMP   JULY   JUNE"
240 DATA "JU     BRING   BABY   BECAUSE","BABY"
```

Program Listing 3

exactly what the teacher requires. If you want to count each problem only once, some minor changes need to be made. (See Proram Listing 6.)

Introduce variable W1: line 71 will increment W1 each time a problem is missed, but W1 will equal 1 only once, so line 75 will define it wrong for W only once. W1 is reset to zero in line 45 each time the problem is answered correctly.

The number of correct answers must be derived from incorrect answers, otherwise a problem missed once and correct the next time will be counted both right and wrong. Therefore, line 90 was deleted from Listing 5 and line 105 was changed to read 5-W (the 5 represents the total number of questions in the program) instead of R.

A side comment: line 40 in Listing 6 uses both a PRINT statement and an INPUT statement instead of the more compact form: INPUT, "PRESS ENTER";A$. This was done deliberately. I do not want my students to see 'PRESS ENTER?' on the screen. The unnecessary question mark bothers me. In a later section I'll discuss punctuation in detail.

```
10 CLEAR500
15 CLS

25 PRINT STRING$(63,61)
30 PRINT"TALLYING CORRECT & INCORRECT ANSWERS."
31 PRINT"     EXAMPLE 1:  TALLY PROGRAM"
35 PRINT STRING$(63,61)
40 PRINT "PRESS <ENTER> TO BEGIN.":INPUT A$
45 CLS
50 N=N+1
55 X=RND(10)-1:Y=RND(10)-1:B=X+Y
60 PRINT@7*64, X;" + ";Y;" =  ";:INPUT A
65 IF A=B THEN 85
70 PRINT"SORRY, WRONG ANSWER!"

75 W=W+1
80 GOTO 95
85 PRINT"THAT'S RIGHT!"
90 R=R+1
95 IF N>4 THEN 105
100 PRINT"PRESS <ENTER>":INPUT A$:GOTO 45
105 PRINT"YOU GOT";R;" RIGHT AND";W;" WRONG."
110 FOR X=1 TO 1000:NEXT
115 PRINT@7*64,"THIS IS THE END OF THE LESSON."
120 PRINT"GOOD-BYE FOR NOW."
125 PRINT"GO BACK TO YOUR CLASS."
130 FOR X=1 TO 2000:NEXT   'USE 9000 IN PRACTICE
135 N=0:R=0:W=0
140 GOTO 45
145 END
```

Program Listing 4

Missed Problems

Recording correct and incorrect answers does not produce a diagnosis of the child's actual needs. If your program uses levels, and all the problems at each level are equally significant, it is sufficient to provide a means of keeping a record of the level attained by each student. There are cases in which you would like to know which problems were missed by each student, however.

One way to do this is to number the problems. If the student misses a problem, its number is printed on the screen or printer, or it can be stored on a data tape or disk storage. This approach can be used only if the problems reside within the program in data statements. Problems generated at random could not be identified because you would have no way of knowing what the content was.

Another way to identify missed problems is to reprint them as they actually appeared in the program in one of the four modes listed above. See Program Listing 7.

Line 51 converts numerical values to strings using STR$. If you are using string values you do not need this line. Line 66 enters items missed (B$) into the array D$(D). D = D + 1 increments the array, so that the first item missed is D$(1), the second D$(2), etc. Line 75 counts the number of wrong answers in W. Line 107 prints D$(D), where D is incremented until W is reached. Line 106 instructs the computer to skip the printout if no items were missed.

Program Listing 8 adds the level concept to Listing 7. Line 65 asks the student for his level. His input controls the variables that will make up his problems in lines 85-105. Line 200 handles the case where the student misses less than two problems at level

Continued to p. 110

FMG CORPORATION NOW CARRIES GRAHAM-DORIAN & PEACHTREE SOFTWARE

FMG CORPORATION
P.O. Box 16020
Fort Worth, Texas
76133
(817) 294-2510

VISA · master charge THE INTERBANK CARD

✓12

M-530

All FMG Software Products Include All Necessary Manuals

NEW VERSATILITY For Your TRS-80

CP/M®2

CONTROL PROGRAM FOR MICROCOMPUTERS ENABLING YOU TO RUN SOFTWARE PUBLISHED FOR CP/M 1.4 ON THE TRS-80

CP/M is considered the industry standard disk operating system because it gives you the hardware-independent interface you need to make your computer work for you. CP/M 2.0 is the latest in the evolution of a proven reliable and efficient software system. FMG CORPORATION NOW OFFERS THE CP/M 2.0 FOR THE TRS-80 .

It features an enhanced upward compatible file system and powerful new random access capabilities. The CP/M 2.0 from FMG provides the ability to run software published for the CP/M system, on the TRS-80 Model II. From minidisks, floppy disks, all the way to high-capacity hard disks, the flexibility of CP/M 2.0 makes it a truly universal operating system. The package includes an 8" system disk, editor, assembler and debugger for the TRS-80

Available in Format A, B, C, G only . . . **$200/$25**

MP/M™

MULTI-PROGRAMMING MONITOR

NEW INDUSTRY STANDARD

A deluxe operating system that provides big computer facilities at small computer prices. MP/M is a monitor program which operates with your microcomputer to provide multi-terminal access with multi-programming at each terminal. Best of all, it's CP/M compatible which means you can run a wide variety or programming languages, applications packages, and development software.

You can run simultaneous editors, program translators, and background printer spoolers. Or you can use MP/M for data entry or data-base access from remote terminals. Or you can use MP/M real-time features to monitor an assembly line and automatically schedule programs for execution throughout the day. MP/M makes an excellent focal point for a cluster of connected microcomputers. The possibilities are limitless.

(Format B) . . . . . **$450/$35**
(Format G) . . . . . **$300/$35**

*CP/M and MP/M are trademarks of Digital Research.
Z80 is a trademark of Zilog, Inc.
TRS-80 is a trademark of Tandy Corp.
Pascal/M is a trademark of Sorcim.

· MICROPRO INTERNATIONAL

SOFTWARE WITH MANUAL / MANUAL/ALONE

(M) **SUPER-SORT I** — Sort, merge, extract utility as absolute executable program or linkable module in Microsoft format. Sorts fixed or variable records with data in binary, BCD, Packed Decimal, EBCDIC, ASCII, floating & fixed point, exponential, field justified, etc. Even variable number of fields per record! . **$225/$25**

SUPER-SORT II — Above available as absolute program only . . . . . **$175/$25**

SUPER-SORT III — As II without SELECT/EXCLUDE . . . . . **$125/$25**

WORD-STAR — Menu driven visual word processing system for use with standard terminals. Text formatting performed on screen. Facilities for text paginate, page number, justify, center and underscore. User can print one document while simultaneously editing a second. Edit facilities include global search and replace. Read/Write to other text files, block move, etc. Requires CRT terminal with addressable cursor positioning . . . . . **$495/$40**

WORD-STAR Customization Notes — For sophisticated users who do not have one of the many standard terminal or printer configurations in the distribution version of WORD-STAR . . . . . **NA/$95**

WORD-MASTER Text Editor — In one mode has superset of CP/M's ED commands including global searching and replacing, forwards and backwards in file in video mode, provides full screen editor for users with serial addressable-cursor terminal . . . . . **$150/$25**

FLOPPY SAVER — Protection for center holes of 5" and 8" floppy disks. Only 1 needed per diskette. Kit contains centering post, pressure tool and tough 7 mil mylar reinforcing rings for 25 diskettes.
5", Kit **$14.95**
5", Rings only **$7.95**
8", Kit **$16.95**
8", Rings only **$8.95**

HEAD CLEANING DISKETTE — Cleans the drive Read/Write head in 30 seconds. Diskette absorbs loose oxide particles, fingerprints, and other foreign particles that might hinder the performance of the drive head. Lasts at least 3 months with daily use.
8" **$32.00**
5¼" **$30.00**

(M) · **DESPOOL** — Allows flexibility and efficiency. (Disk file printing can be accomplished while simultaneously using the computer for other tasks) Slower printers do not tie up the computer. Requires 32K minimum . . . . . **$75/$10**

(M) · **SCREEN EDIT** — Text editor for program entry — allows user the ability to see entries as they are being made. Has command which enables user to move the viewed position of the file anywhere within the current data file OR add information anywhere in the file. Requires 16K minimum . . . . . **$125/$25**
(Also available in TRS DOS format.
Specify model or TRS-80)

(M) · **MAC** — Disk-based, powerful macro assembler utilizes Standard Intel Mnemonics. Includes macro processor.
The CP/M 8080 Macro Assembler reads assembly language statement from a diskette file and produces an Intel "HEX" format object file on the disk suitable for processing in the TRS-CP/M environment. Requires 32K minimum and CP/M . . . . . **$100/$25**

(M) · **ZSID** — Efficient and reliable program testing system for Z80 microcomputers. Capabilities include traceback and histogram facilities. Allows real time break points.
ZSID is a symbolic debugger which expands upon the features of the TRS-CP/M standard debugger, providing greatly enhanced facilities for assembly language program check-out. Requires 32K minimum and CP/M . . . . . **$99/$25**

(T) **MAIL LIST** — Mailing list maintenance package. No sorting required to print normal address labels in zip code sequence. Supports new larger zip code. Sorts and selects on multiple fields. Labels may be printed in user selectable formats. Includes sort and select utilities **$300/$35**

· PEACHTREE SOFTWARE SYSTEMS

(T) **GENERAL LEDGER** — Records details of all financial transactions. Generates a balance sheet and an income statement. Flexible and adaptable design for both small businesses and firms performing client writeup services. Produces reports as follows: Trial Balance, Transaction Registers, Balance Sheet, Prior Year Comparative Balance Sheet, Income Statement, Prior Year Comparative Income Statement and Department Income Statements. Interactive with other PEACHTREE accounting packages. Supplied in source code for Microsoft BASIC . . . . . **$990/$30**

(T) **ACCOUNTS PAYABLE** — Tracks current and aged payables and incorporates a check writing feature. Maintains a complete vendor file with information on purchase orders and discount terms as well as active account status. Produces reports as follows: Open Voucher Report, Accounts Payable Aging Report and Cash Requirements. Provides input to PEACHTREE General Ledger. Supplied in source code for Microsoft BASIC . . . . . **$990/$30**

(T) **ACCOUNTS RECEIVABLE** — Generates invoice register and complete monthly statements. Tracks current and aged receivables. Maintains customer file including credit information and account status. The current status of any customer account is instantly available. Produces reports as follows: Aged Accounts Receivable, Invoice Register, Payment and Adjustment Register and Customer Account Status Report. Provides input to PEACHTREE General Ledger. Supplied in source code for Microsoft BASIC . . . . . **$990/$30**

(T) **PAYROLL** — Prepares payroll for hourly, salaried and commissioned employees. Generates monthly, quarterly and annual returns. Prepares employee W-2's. Includes tables for federal withholding and FICA as well as withholding for all 50 states plus up to 20 cities from pre-computed or user generated tables. Will print checks, Payroll Register, Monthly Summary and Unemployment Tax Report. Provides input to PEACHTREE General Ledger. Supplied in source code for Microsoft BASIC . . . . . **$990/$30**

(T) **INVENTORY** — Maintains detailed information on each inventory item including part number, description, unit of measure, vendor and reorder data, item activity and complete information on current item costs, pricing and sales. Produces reports as follows: Physical Inventory Worksheet, Inventory Price List, Departmental Summary Report, Inventory Status Report, The Reorder Report and the Period-to-Date and Year-to-Date reports. Supplied in source code for Microsoft BASIC . . . . . **$1,150/$30**

(T) **MAILING ADDRESS** — Keeps track of name and address information and allows the selective printing of this information in the form of mailing lists or address labels. Allows the user to tailor the system to his own particular requirements. User-defined format and print-out system uses a special format file which tells programs how to print the mailing list or address labels. Standard format files are included with system. Automatic sorting of data uses indexed file management routines which allow the name and address information to be sequentially retrieved and printed without file sorting. Supplied in source code for Microsoft BASIC . . . . . **$790/$30**

· GRAHAM-DORIAN SOFTWARE SYSTEMS

(T) **GENERAL LEDGER** — An on-line system; no batching is required. Entries to other GRAHAM-DORIAN accounting packages are automatically posted. User establishes customized C.O.A. Provides transaction register, record of journal entries, trial balances and monthly closings. Keeps 14 month history and provides comparison of current year with previous year. Requires CBASIC-2. Supplied in source . . . . . **$995/$35**

(T) **ACCOUNTS PAYABLE** — Maintains vendor list and check register. Performs cash flow analysis. Flexible — writes checks to specific vendor for certain invoices or can make partial payments. Automatically posts to GRAHAM-DORIAN General Ledger or runs as stand alone system. Requires CBASIC-2. Supplied in source . . . . . **$995/$35**

(T) **ACCOUNTS RECEIVABLE** — Creates trial balance reports, prepares statements, ages accounts and records invoices. Provides complete information describing customer payment activity. Receipts can be posted to different ledger accounts. Entries automatically update GRAHAM-DORIAN General Ledger or runs as stand alone system. Requires CBASIC-2. Supplied in source . . . . . **$995/$35**

(T) **PAYROLL SYSTEM** — Maintains employee master file. Computes payroll withholding for FICA, Federal and State taxes. Prints payroll register, checks, quarterly reports and W-2 forms. Can generate ad hoc reports and employee form letters with mail labels. Requires CBASIC-2. Supplied in source . . . . . **$590/$35**

(T) **INVENTORY SYSTEM** — Captures stock levels, costs, sources, sales, ages, turnover, markup, etc. Transaction information may be entered for reporting by salesman, type of sale, date of sale, etc. Reports available both for accounting and decision making. Requires CBASIC-2. Supplied in source . . . **$590/$35**

(T) **JOB COSTING** — Designed for general contractors. To be used interactively with other GRAHAM-DORIAN accounting packages for tracking and analysing expenses. User establishes customized cost categories and job phases. Permits comparison of actual versus estimated costs. Automatically updates GRAHAM-DORIAN General Ledger or runs as stand alone system. Requires CBASIC-2. Supplied in source **$995/$35**

Sample Program Disk For Each Graham-Dorian Business Package. Specify Package **$45**

The sale of each proprietary software package conveys a license for use on one system only.

Prices F.O.B. Fort Worth, Tex. Shipping, handling and C.O.D. charges extra.

· MICROSOFT PRODUCTS

(M) **BASIC-80** — Disk Extended BASIC, ANSI compatible with long variable names, WHILE/WEND, chaining, variable length file records . . . . . **$350/$25**

(M) **BASIC COMPILER** — Language compatible with BASIC-80 and 3-10 times faster execution. Produces standard Microsoft relocatable binary output. Includes MACRO-80. Also linkable to FORTRAN-80 or COBOL-80 code modules . . . . . **$395/$25**

(M) **FORTRAN-80** — ANSI 66 (except for COMPLEX) plus many extensions. Includes relocatable object compiler, linking loader, library with manager. Also includes MACRO-80 (see below) . . . . . **$500/$25**

(M) **COBOL-80** — Level 1 ANSI '74 standard COBOL plus most of Level 2. Full sequential, relative, and indexed file support with variable file names. STRING, UNSTRING, COMPUTE, VARYING/UNTIL, EXTEND, CALL, COPY, SEARCH, 3-dimensional arrays, compound and abbreviated conditions, nested IF. Powerful interactive screen-handling extensions. Includes compatible assembler, linking loader, and relocatable library manager as described under MACRO-80 . . . . . **$450/$25**

(M) **MACRO-80** — 8080/Z80 Macro Assembler. Intel and Zilog mnemonics supported. Relocatable linkable output. Loader, Library Manager and Cross Reference List utilities included . . . . . **$150/$25**

(T) **XMACRO-86** — 8086 cross assembler. All Macro and utility features of MACRO-80 package. Mnemonics slightly modified from Intel ASM86. Compatibility data sheet available . . . . . **$300/$25**

(T) **PASCAL/M*** — Compiler generates P code from extended language, implementation of standard PASCAL. Supports overlay structure through additional procedure calls and the SEGMENT procedure type. Provides convenient string handling capability with the added variable type STRING. Untyped files allow memory image I/O. Requires 56K CP/M . . . . **$150/$20**

(T) **PASCAL/Z** — Z80 native code PASCAL compiler. Produces optimized, ROMable re-entrant code. All interfacing to CP/M is through the support library. The package includes compiler, Microsoft Compatible relocating assembler and linker, and source for all library modules. Variant records, strings and direct I/O are supported. Requires 56K CP/M and Z80 CPU. . . . . **$395/$25**

(M) **PASCAL/MT** — Subset of standard PASCAL. Generates ROMable 8080 machine code. Symbolic debugger included. Supports interrupt procedures, CP/M file I/O and assembly language interface. Real variables can be BCD, software floating point, or AMD 9511 hardware floating point. Version 3 includes Enumeration and Record data types. Manual explains BASIC to PASCAL conversion. Source for the runtime package requires Digital Research's MAC. Requires 32K . . . . . **$250/$30**

(M) **CBASIC-2** Disk Extended BASIC — Non-interactive BASIC with pseudo-code compiler and run-time interpreter. Supports full file control, chaining, integer and extended precision variables, etc. . . . . **$110/$15**

(M) **BSTAM** — Utility to link one computer to another also equipped with BSTAM. Allows file transfers at full data speed (no conversion to hex), with CRC block control check for very reliable error detection and automatic retry. We use it! It's great! Full wildcard expansion to send ★.COM, etc. 9600 baud with wire. 300 baud with phone connection. **Both** ends need one. Standard and Ⓜ versions can talk to one another. . . . . **$150/$5**

(M) (T) **SELECTOR III-C2** — Data Base Processor to create and maintain multi Key data bases. Prints formatted sorted reports with numerical summaries or mailing labels. Comes with sample applications, including Sales Activity, Inventory, Payables, Receivables, Check Register, and Client/Patient Appointments, etc. Requires CBASIC-2. Supplied in source . . . **$349/$30**

GLECTOR — General Ledger option to SELECTOR III-C2. Interactive system provides for customized COA. Unique chart of transaction types insure proper double entry bookkeeping. Generates balance sheets, P&L statements and journals. Two year record allows for statement of changes in financial position report. Supplied in source. Requires SELECTOR III-C2, CBASIC-2 and 52K system . . . . . **$250/$25**

(M) **TEXTWRITER III** — Text formatter to justify and paginate letters and other documents. Special features include insertion of text during execution from other disk files or console, permitting recipe documents to be created from linked fragments on other files. Has facilities for sorted index, table of contents and footnote insertions. Ideal for contracts, manuals, etc. Now compatible with Electric Pencil* prepared files. . . . . **$125/$20**

FORMATS AVAILABLE:

ORDERS MUST SPECIFY DISK SYSTEMS AND FORMATS:

(A) TRS-80 Model I (M) Keys Only
(B) TRS-80 Model II
(C) TRS-80 Model III (M) Keys Only
(D) HEATHKIT H89 (M) Keys Only
(E) NORTH STAR
(F) SUPER BRAIN QD
(G) STANDARD UNIMPLEMENTED

(M) Modified version available for use with CPM as implemented on Heath and TRS-80 Model I computers.

(T) For all (T) items listed above, the recommended system configuration consists of 48K CPM 2 full dize disk drives. 24 x 80 CRT and 132 column printer.

FMG's LIBRARY:

PASCAL USER MANUAL & REPORT
(2nd) Edition by K. Jensen and N. Wirth

- **Tutorial Manual and Concise Reference Report for Both Programmers and Implementors**
- **Includes Helpful Examples to Demonstrate the Various Features of PASCAL**

The book consists of two parts: the user manual and the revised report. The manual is directed to those who have some familiarity with computer programming and who wish to get acquainted with the PASCAL language. The report defines standard PASCAL, which constitutes a common base between various implementations of the language.

Stock No. #B21 — **Price $9.95**

PASCAL PRIMER Problem Solving

This book has three major goals:

- **To introduce all aspects of the programming and problem solving process (includes problem specification and organization, algorithms, coding, debugging, testing, documentation and maintenance);**
- **To teach good programming style and how to produce a high quality finished product; and**
- **To teach the syntax of the PASCAL programming language.**

Numerous examples are employed throughout the text. PASCAL is used as a vehicle to teach various aspects of programming techniques.

Stock No. #B24 — **Price $18.95**

Microcomputer Problem Solving Using Pascal
by Kenneth L. Bowles

- **A Book Designed for Both College Courses AND Individual Self-Study**
- **Ideal for use with UCSD Pascal**
- **Includes Extensions to Standard PASCAL**

This book is designed both for introductory courses in computer problem solving at the freshman and sophomore college level, and for individual self-study. It includes many examples and actually executable programs. It includes information on the necessary functions and procedures for handling graphics and strings.

Stock No. #B22 — **Price $14.95**

BEGINNER'S MANUAL FOR UCSD PASCAL SYSTEM

- **An Enlightening Introduction to UCSD PASCAL**
- **Demonstrates How to Use the UCSD PASCAL System and How to Program in PASCAL**
- **Includes Many Practical Examples of PASCAL Programs**

This book is intended to be used as an introduction and reference manual for people just beginning to use the UCSD Pascal Software System. Whether you have never used a computer before or whether you are an experienced programmer who is unfamiliar with UCSD PASCAL, this book will provide a relatively easy, yet thorough, introduction to UCSD PASCAL.

Stock No. #B25 — **Price $11.95**

PROGRAMING IN PASCAL
by Peter Grogono

- **An Excellent Introduction to One of the Fastest Growing Programming Languages Today**
- **Sections on Procedures and Files PLUS a Chapter on Dynamic Data Structures such as Trees and Linked Lists**

The text is arranged as a tutorial, containing both examples and exercises to increase reader proficiency in PASCAL. Concepts are illustrated by examples, ranging from the Tower of Hanoi problem to circumscribing a circle about a triangle. PROGRAMMING IN PASCAL is sure to hold the reader's interest.

Stock No. #B23 — **Price $14.95**

UCSD Reference Book

- **A Reference Guide to the Complete UCSD PASCAL System**
- **Includes Information on Compiler Basic, Assembler and Editor**
- **Lists Actual P-Machine Codes**

This reference book can be a valuable and time-saving guide to through information on the UCSD PASCAL system. The easy-to-read manual provides fast access to pertinent data.

Stock No. #B26 — **Price $25.00**

```
10 CLEAR500
15 CLS
25 PRINT STRING$(63,61)
30 PRINT"TALLYING CORRECT & INCORRECT ANSWERS."
31 PRINT"     EXAMPLE 2: REPRINTING PROBLEMS MISSED."
35 PRINT STRING$(63,61)
40 PRINT "PRESS <ENTER> TO BEGIN.":INPUT A$
45 N=N+1
50 X=RND(10)-1:Y=RND(10)-1:B=X+Y
55 CLS
60 PRINT@7*64, X;" + ";Y;" =  ";:INPUT A
65 IF A=B THEN 85
70 PRINT"SORRY, WRONG ANSWER!"
75 W=W+1
80 PRINT"PRESS <ENTER>":INPUT A$:GOTO 55
85 PRINT"THAT'S RIGHT!"
90 R=R+1
95 IF N>4 THEN 105
100 PRINT"PRESS <ENTER>":INPUT A$:GOTO 45
105 PRINT"YOU GOT";R;" RIGHT AND";W;" WRONG."
110 FOR X=1 TO 1000:NEXT
115 PRINT@7*64,"THIS IS THE END OF THE LESSON."
120 PRINT"GOOD-BYE FOR NOW."
125 PRINT"GO BACK TO YOUR CLASS."
130 FOR X=1 TO 2000:NEXT    'USE 9000 IN PRACTICE
135 N=0:R=0:W=0
140 GOTO 45
145 END
```

Program Listing 5

```
10 CLEAR500
15 CLS
25 PRINT STRING$(63,61)
30 PRINT"TALLYING CORRECT & INCORRECT ANSWERS."
31 PRINT"     EXAMPLE 3: REPRINTING PROBLEMS MISSED."
32 PRINT"                CORRECTING THE TALLY."
35 PRINT STRING$(63,61)
40 PRINT "PRESS <ENTER> TO BEGIN.":INPUT A$
45 N=N+1: W1=0
50 X=RND(10)-1:Y=RND(10)-1:B=X+Y
55 CLS
60 PRINT@7*64, X;" + ";Y;" =  ";:INPUT A
65 IF A=B THEN 85
70 PRINT"SORRY, WRONG ANSWER!"
71 W1=W1+1
75 IF W1=1 THEN W=W+1
80 PRINT"PRESS <ENTER>":INPUT A$:GOTO 55
85 PRINT"THAT'S RIGHT!"
95 IF N>4 THEN 105
100 PRINT"PRESS <ENTER>":INPUT A$:GOTO 45
105 PRINT"YOU GOT";5-W;" RIGHT AND";W;" WRONG."
110 FOR X=1 TO 1000:NEXT
115 PRINT@7*64,"THIS IS THE END OF THE LESSON."
120 PRINT"GOOD-BYE FOR NOW."
125 PRINT"GO BACK TO YOUR CLASS."
130 FOR X=1 TO 2000:NEXT   'USE 9000 IN PRACTICE
135 N=0:R=0:W=0
140 GOTO 45
145 END
```

Program Listing 6

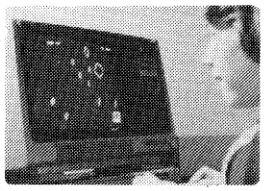
```
10 CLEAR500
15 CLS
25 PRINT STRING$(63,61)
30 PRINT"KEEPING RECORDS"
31 PRINT"     EXAMPLE 1: ON THE SCREEN AT THE END OF THE LESSON."
35 PRINT STRING$(63,61)
40 PRINT "PRESS <ENTER> TO BEGIN.":INPUT A$
45 N=N+1: W1=0
50 X=RND(10)-1:Y=RND(10)-1:B=X+Y
51 B$=STR$(X)+" +"+STR$(Y)
55 CLS
60 PRINT@7*64,B$;" = ";:INPUT A
65 IF A=B THEN 85
66 D=D+1:D$(D)=B$
70 PRINT"SORRY, WRONG ANSWER!"
71 W1=W1+1
75 IF W1=1 THEN W=W+1
80 PRINT"PRESS <ENTER>":INPUT A$:GOTO 55
85 PRINT"THAT'S RIGHT!"
```

Program continues

Games from BIG FIVE will turn your computer into a

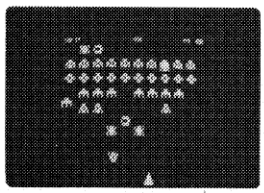
TRS-80 HOME ARCADE

SUPER NOVA©



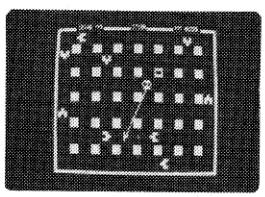
If you and your TRS-80 have longed for a fast-paced arcade-type game that is truly a challenge, then **SUPER NOVA** is what you've been waiting for. In this two player machine-language game, large asteroids float ominously around the screen. Suddenly your ship appears and you must destroy the asteroids before they destroy you! (But watch out because big asteroids break apart into little ones.) The controls that your ship will respond to are thrust, rotate, hyperspace, and fire. All right! You've done it! You've cleared away all the asteroids! But what is that saucer with the laser doing? Quick! You must destroy him fast because that guy's accurate!

GALAXY INVASION©



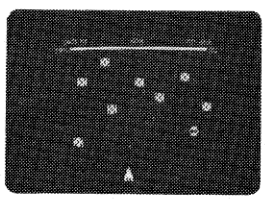
The sound of the klaxon is calling you! Cruel and crafty invaders have been spotted in battle formation warping toward Earth at an incredible speed. Suddenly, your ship materializes just below the huge flock of invaders. Quickly and skillfully you shift right and left as you carefully fire your lasers at them. But watch out! A few are breaking out of the convoy and flying straight at you! As the whine of their engines gets louder, you place your finger on the fire button knowing all too well that this shot must connect—or your mission will be permanently over! With sound effects!

ATTACK FORCE©



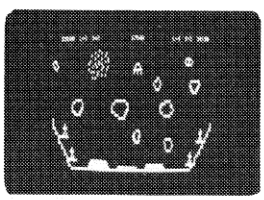
Your TRS-80 screen has been transformed into a maze-like playfield for this game. As your ship appears on the bottom of the screen, eight alien ramships appear on the top. All of them are traveling at flank speed directly at you! Quickly and boldly you move toward them and fire missiles to destroy them. But the more aliens you destroy, the faster the remaining ones become. If you get too good you must endure the wrath of the keeper of the mazefield: the menacing "Flagship". You must destroy him fast because, as you will find out, that guy's accurate! With sound effects!

COSMIC FIGHTER©

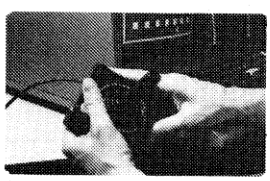


With thousands of stars whizzing by you, your **SPACE DESTROYER** ship comes out of hyperspace directly under a convoy of aliens. Almost effortlessly, you skillfully destroy every last one. But before you can congratulate yourself, another set appears. These seem to be slightly more intelligent than the first set. Quickly you eliminate all of them, too. But your fuel supply is rapidly diminishing. You must still destroy two more sets before you can dock with your space station. All right! The space station is now on your scanners! Oh no! Intruders have overtaken the station! You must skillfully fire your neutron lasers to eliminate the intruders from the station before your engines run out of fuel and explode! With sound!

METEOR MISSION II©



As you look down on your space viewer you can see the stranded astronauts that are crying out for you to rescue them. But first you must maneuver your shuttle down through the asteroids & meteors before you can reach them. Great! You've got one! But now can you get back to the space station to save your fellow shipmate or will you crash and kill both of you? You can fire your lasers to destroy the asteroids, but watch out, because there could be an alien FLAGSHIP lurking behind! Includes sound effects!



NEW JOYSTICKS!

For $39.95 it's now possible to have the famous ATARI joystick interfaced with your Model 1. All of our tapes are now completely compatible with the joystick. Packaged with complete instruction, you can even use it with your own programs! If your old tapes do not say "Joystick Version" on them and you wish to exchange them for new Joystick versions, enclose $2 and your old tape. (Call or write for info on Mod 3 joysticks.)

✓357

BIG FIVE SOFTWARE

P.O. Box 9078-185, Van Nuys, California 91409

For MC/Visa or COD orders, call (213) 782-6861 from 9am-5pm Pacific time.

Prices per game: Level 2, 16K Mod 1/Mod 3—$15.95
Level 2, 16K Mod 1/Disk—$17.95

10% discount for 2 items, 15% for 3 or more. (Including Joystick)
Please add $1.50 postage/handling, Calif. residents add 6% tax.
All games are written in machine language and supplied on cassette.
Disk versions save high scores to your TRSDOS or NEWDOS diskette.

Cassette versions require 16K memory, disk versions require 32K.
We accept checks, money orders, and MC/Visa orders.
All games ©1980 by Bill Hogue & Jeff Konyu.
TRS-80 & TRSDOS are trademarks of Tandy Corp.
NEWDOS is a trademark of Apparat, Inc.
Dealer inquiries invited.

✓ Reader Service—see page 290

```
95 IF N>4 THEN 105
100 PRINT"PRESS <ENTER>":INPUT A$:GOTO 45
105 PRINT"YOU GOT";5-W;" RIGHT AND";W;" WRONG."
106 IF W>0 PRINT:PRINT"COPY AND STUDY:"ELSE 115
107 FOR D=1 TO W:PRINT D$(D):NEXT
110 PRINT"PRESS <ENTER> WHEN YOU ARE FINISHED...":INPUT A$
115 CLS:PRINT@7*64,"THIS IS THE END OF THE LESSON."
120 PRINT"GOOD-BYE FOR NOW."
125 PRINT"GO BACK TO YOUR CLASS."
130 FOR X=1 TO 2000:NEXT   'USE 9000 IN PRACTICE
135 N=0:W=0:D=0
140 GOTO 45
145 END
```

Program Listing 7

```
10 CLEAR500
15 DIMD$(20)
20 CLS
30 PRINT STRING$(63,61)
35 PRINT"KEEPING RECORDS"
40 PRINT"     EXAMPLE 2: SHOWING NEW LEVEL TO WORK ON."
45 PRINT STRING$(63,61)
50 PRINT "PRESS <ENTER> TO BEGIN.":INPUT A$
55 CLS
60 PRINT@6*64,"TO THE STUDENT:"
65 INPUT"WHAT LEVEL ARE YOU ON";L
70 IF L<1 OR L>5 PRINT"TYPE YOUR LEVEL, 1 TO 5.":GOTO 60
75 N=N+1: W1=0
80 ON L GOTO 85,90,95,100,105
85 X=RND(3)-1:Y=RND(3)-1:GOTO 110
90 X=RND(3)-1:Y=RND(2)+2:GOTO 110
95 X=RND(2)+2:Y=RND(5)+2:GOTO 110
100 X=RND(5)+2:Y=RND(7)+2:GOTO 110
105 X=RND(6)+3:Y=RND(6)+3
110 B=X+Y
115 B$=STR$(X)+" +"+STR$(Y)
120 CLS
125 PRINT@6*64,B$;" = ";:INPUT A
130 IF A=B THEN 160
135 D=D+1:D$(D)=B$
140 PRINT"SORRY, WRONG ANSWER!"
145 W1=W1+1
150 IF W1=1 THEN W=W+1
155 PRINT"PRESS <ENTER>":INPUT A$:GOTO 110
160 PRINT"THAT'S RIGHT!"
165 IF N>4 THEN 175
170 PRINT"PRESS <ENTER>":INPUT A$:GOTO 75
175 PRINT"YOU GOT";5-W;" RIGHT AND";W;" WRONG."
180 IF W>0 PRINT:PRINT"COPY AND STUDY:"ELSE 195
185 FOR D=1 TO W:PRINT D$(D):NEXT
190 PRINT"PRESS <ENTER> WHEN YOU ARE FINISHED...":INPUT A$
195 CLS:PRINT@6*64,"THIS IS THE END OF THE LESSON."
200 IF W<3 AND L=5 PRINT"CONGRATULATIONS! YOU'RE DONE!":GOTO 215
205 IF W<3 PRINT"NEXT TIME WORK ON LEVEL";L+1;".":GOTO 215
210 PRINT"NEXT TIME WORK ON LEVEL";L;"."
215 PRINT"GOOD-BYE FOR NOW."
220 PRINT"GO BACK TO YOUR CLASS."
225 FOR X=1 TO 1000:NEXT  'USE 9000 INSTEAD OF 1000 IN REAL PROGRA
    M
230 N=0:W=0:D=0
235 GOTO 55
240 END
```

Program Listing 8

5. Line 205 is for the student who misses less than two and will move up a level. Line 210 keeps the student at the same level because he missed more than two problems at his current level.

Record Saving

The record keeping methods illustrated up to now are displayed on the screen, and will be lost when the next student starts to use the program or the computer is turned off. A simple procedure for saving such records is to have the student copy the information off the screen onto a record card of some sort. In practice, this method has proved effective. The student or the teacher can then use the information as a diagnosis of the student's work, as a guide to the level to work on next, or to determine whether the student needs instruction by the teacher. If you are concerned that the student may not accurately copy the information, I can assure you that this is rare in actual practice. However, there will be times when you want permanent records maintained by the computer.

You should give the teacher the option of using any of the record keeping systems you are offering. See Program Listing 9.

Lines 2100-2125 contain the instructions for using a printer. Lines 2200-2225 tell the user how to prepare a data tape to receive the information. Lines 2300 and 2305 are very brief instructions related to a disk file. Lines 2400-2415 describe the student maintainance of the record without any permanent record being made. Lines 2500-2510 allow the user to change his mind if he may have selected an inappropriate method for record keeping. Lines 95 and 145 send the program to the subroutine in lines 3000 to 4000, that take care of record keeping each time something must be recorded.

Note that in lines 90 and 140 the strings N$ and B$ are saved in X$; in line 3005, X$ is converted to R$. This is to permit the record systems to use the same variable name (R$) regardless of which variable is being recorded at the moment. This technique is necessary when the number of items recorded for each student is itself a variable.

In other words, if John misses three, Mary misses five, and Tony misses none, you need a routine to handle the different number of records for each child. One way would be to record correct answers as well as incorrect, so that each child would have the same number of records following his name.

I have chosen to illustrate the procedure for using a variable number of records because I think it illustrates a simpler and more powerful use of data recording methods (Program Listing 9).

Line 3100 immediately prints the data on the printer. Lines 3200-3215 save data on a cassette tape. CMD'T' and CMD'R' are necessary when using a disk operating system simultaneously with a cassette tape: these commands turn the real-time clock off and on. You must do this because the ticking of the clock interferes with the data going to or from the tape.

Lines 3300-3315 save data to a sequential file on a disk, using the Apparat Disk Operating System. TRSDOS is far more complicated because it does not add data to a sequential file. (See pg. 7-62 of the Radio Shack TRSDOS Manual, Catalog No. 26-2104 for details regarding this problem.)

In both the tape and disk routines, there is only one variable format, R$, in lines 3205 and 3310.

Lines 120-1390 recover data from the records. Lines 1255-1280 present the steps the teacher must follow to recover data from a tape. Once again, CMD'T' and CMD'R' are necessary to stop the clock. Line 1300 opens the relay that starts the cassette recorder. Without this instruction, the

"If you don't erase the file, it will simply add your new data to the end of the file."

keyboard is disabled until you press the RESET button or turn off the computer.

One interesting benefit of the data tape routine is that you can rewind or fast-forward the tape while it is under the control of the computer feeding data back through an INPUT# – 1 command. This feature is especially useful when you have a great deal of data on a tape filed by date, in alphabetical order, or with any other filing system. This works only if you have used just one variable in the PRINT# – 1, INPUT# – 1, and PRINT statements.

Lines 1350-1385 recover data from the disk file and print it on the screen for review. Once you are finished with the file you must erase it by entering KILL'RECORDS/TXT' while the computer is in DOS mode. If you don't erase the file, it will simply add your new data to the end of the file.

Use of Audio

The cassette recorder that came with your TRS-80 can be used as an ordinary cassette recorder/player, with the added feature of being controlled by the computer. If the play key is depressed and the black plug removed from the ear socket, the recorder can be activated with the command OUT 255,255, and deactivated with OUT 255, 0.

Can you imagine how useful this feature might be with young children? Instructions to operate the computer or to run a program, spelling words, foreign language material, phonics lessons, reading material, questions to be answered, and other aural material can be presented!

How is spelling taught with a computer? You can use either the visual or the aural mode. While most teachers test their students by saying the word and having the student write it, tests that are used nationwide use visual cues only. Thus, it would be wise to provide practice using both modes. I believe that students' test scores are artificially lowered because they are taught in one mode and tested in another.

Here are three test examples commonly found in nationwide tests:

- The TRS-80 is a comp--ter.
  Circle the missing part:
  oo u ew eu
- Is the word in parentheses spelled correctly?
  The TRS-80 is a great (computer).
  yes no
- Place a check in front of the word that is spelled correctly:
  ___compeuter
  ___compooter
  ___computer
  ___compewter
  ___none of these

Students should have the opportunity to practice with all forms of questions. Are you

```
10 CLEAR 500
15 CLS
25 PRINT STRING$(63,61)
30 PRINT"RECORD KEEPING"
35 PRINT"     EXAMPLE 3: COMPOSITE OF ALL SYSTEMS"
40 PRINT STRING$(63,61)
45 PRINT"PRESS <ENTER> TO BEGIN.":INPUT A$
50 GOSUB1000
55 GOSUB2000
60 CLS:PRINT@6*64,"STUDENT PROGRAM STARTS HERE."
65 PRINT"RECORD KEEPING SELECTION";R
70 PRINT"(THE NEXT INSTRUCTION IS FOR THE STUDENT.)"
75 PRINT"PRESS <ENTER>.":INPUT A$
80 CLS:PRINT CHR$(23):PRINT@4*64,"HI, THERE!":PRINT:PRINT"PRESS <E
   NTER>.":INPUT A$
85 CLS:PRINT CHR$(23):PRINT@6*64,"WHAT'S YOUR FIRST NAME?":INPUT N
   $
90 X$=N$
95 GOSUB3000
100 N=N+1:W1=0
105 X=RND(10)-1:Y=RND(10)-1:B=X+Y
110 B$=STR$(X)+" +"+STR$(Y)
115 CLS
120 PRINT@6*64,B$;" = ":INPUT A
125 IF A=B THEN 165
130 D=D+1:D$(D)=B$
135 PRINT"SORRY, WRONG ANSWER!"
140 X$=B$
145 GOSUB3000
150 W1=W1+1
155 IF W1=1 THEN W=W+1
160 PRINT"PRESS <ENTER>.":INPUT A$:GOTO 115
165 PRINT"THAT'S RIGHT, ";N$;"!"
170 IF N>4 THEN 180
175 PRINT"PRESS <ENTER>.":INPUT A$:GOTO 100
180 PRINT"YOU GOT";5-W;"RIGHT AND";W;"WRONG."
185 IF W>0 PRINT:PRINT"COPY AND STUDY:"ELSE GOTO 200
190 FOR D=1 TO W:PRINT D$(D):NEXT
195 PRINT"PRESS <ENTER> WHEN YOU ARE FINISHED...":INPUT A$
200 CLS:PRINT CHR$(23):PRINT@6*64,"THIS IS THE END OF THE LESSON."
205 PRINT"GOOD-BYE FOR NOW, ";N$;"."
210 PRINT"GO BACK TO YOUR CLASS."
215 FOR X=1 TO 2000:NEXT  'USE 9000 IN PRACTICE
220 N=0:W=0:D=0
225 GOTO 80
1000 CLS
1005 PRINT@6*64,"FOR PROGRAM INSTRUCTIONS TYPE 1."
1010 PRINT"TO RECOVER DATA TYPE 2."
1015 INPUT S
1020 ON S GOTO 1100,1200
1025 PRINT"TYPE 1 OR 2 PLEASE!!":GOTO 1005
1100 CLS:PRINT@6*64,"<<PROGRAM INSTRUCTIONS TO BE ENTERED HERE.>>"
1190 PRINT:PRINT"PRESS <ENTER>.":INPUT A$:RETURN
1205 PRINT@6*64,"TO RECOVER DATA FROM TAPE, TYPE 1."
1210 PRINT"TO RECOVER DATA FROM DISK, TYPE 2."
1215 INPUT T
1220 ON T GOTO 1250,1350
1225 PRINT"TYPE 1 OR 2 PLEASE!!!":GOTO 1205
1250 CLS
1255 PRINT@4*64,"FOLLOW THESE STEPS IN ORDER:"
1260 PRINT"     1. REWIND DATA TAPE."
1265 PRINT"     2. PRESS <PLAY> KEY ON RECORDER."
1270 PRINT"     3. PRESS <ENTER> KEY ON TRS-80."
1280 PRINT"TO ABORT THIS SEQUENCE, TYPE Q."
1285 INPUT A$:IF A$="Q" THEN 1000
1290 CMD"T"
1295 INPUT#-1,R$:PRINT R$
1300 IF R$="END" THEN 1320
1310 GOTO 1295
1320 CMD"R"
1325 PRINT"IF NO ENTRIES APPEAR AFTER A NAME STUDENT GOT 100%"
1330 END
1350 'FOR NEWDOS ONLY
1355 OPEN"I",1,"RECORDS/TXT"
1360 IF EOF(1) THEN 1380
1365 INPUT#1,B$
1370 PRINT B$
1375 GOTO 1360
1380 CLOSE
1385 PRINT"IF NO ENTRIES APPEAR AFTER A NAME STUDENT GOT 100%"
1390 END
2000 CLS
2005 PRINT@6*64,:INPUT"DO YOU WANT RECORD KEEPING (Y/N)";A$
2010 IF A$="Y" THEN 2025
2015 IF A$="N" THEN R=4:GOTO 2400
2020 PRINT"ANSWER Y OR N    P L E A S E ! !":GOTO 2005
2025 PRINT"TYPE 1 FOR PRINTER."
```

Program continues

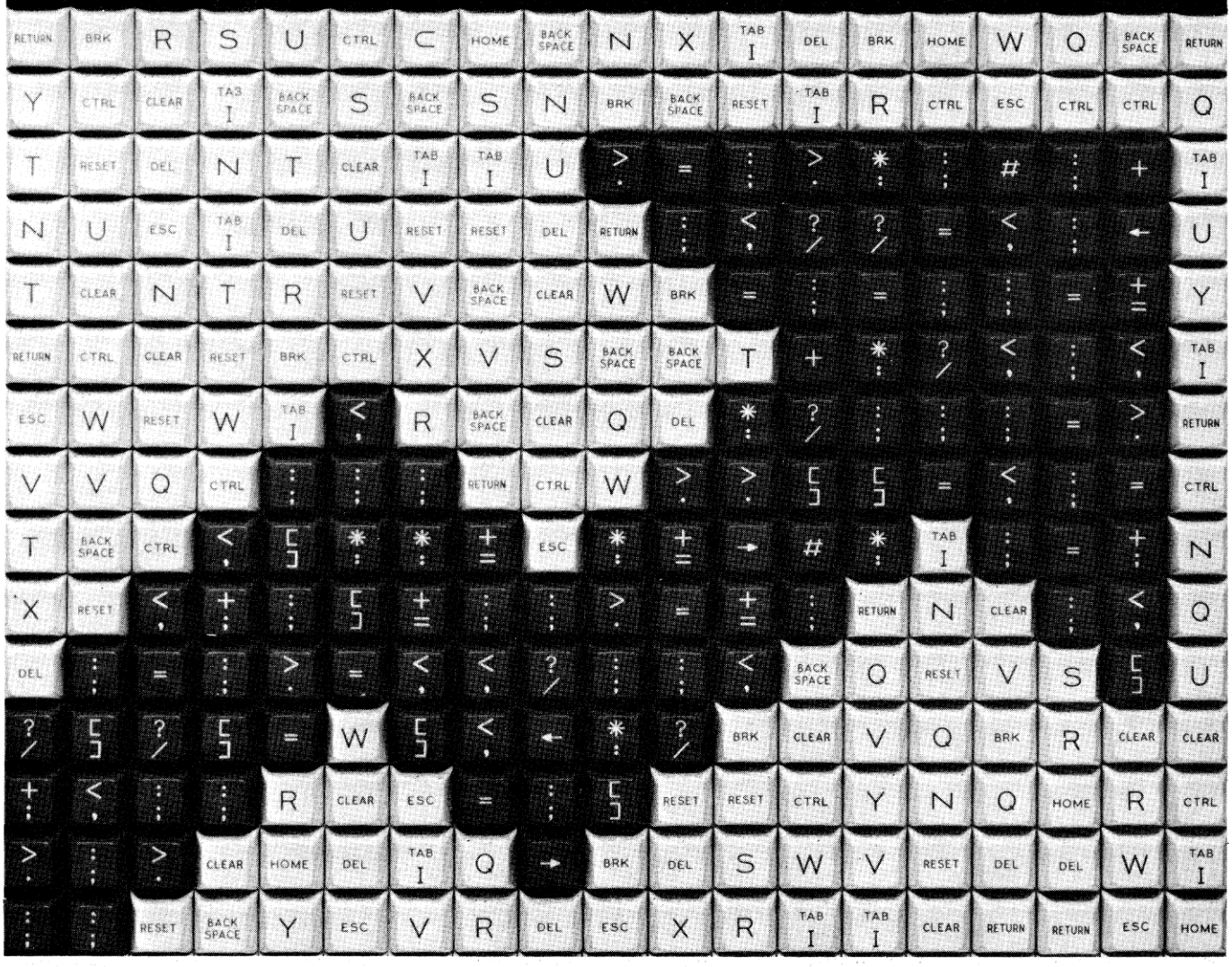
```
2030 PRINT"TYPE 2 FOR DATA TAPE."
2035 PRINT"TYPE 3 FOR DISK SAVE."
2040 PRINT"TYPE 4 FOR STUDENT RECORD KEEPING."
2045 INPUT R
2050 ON R GOTO 2100,2200,2300,2400
2055 CLS:PRINT"TYPE 1, 2, 3, OR 4.":GOTO 2025
2100 IFPEEK(14312)<>63PRINT@13*64,"YOUR PRINTER IS NOT
     READY.
EITHER TURN IT ON OR CHOOSE ANOTHER ENTRY.":GOTO2045
2105 PRINT@6*64,"BE SURE YOUR PRINTER"
2200 CLS:PRINT@4*64,"PLACE AN ERASED TAPE IN THE RECORDER."
2205 PRINT"BE SURE THE TAPE LEADER IS PAST THE RECORD HEAD..."
2210 PRINT"DEPRESS THE <PLAY> AND <RECORD> KEYS."
2215 PRINT"LEAVE THEM DEPRESSED UNTIL ALL STUDENTS HAVE FINISHED."
2220 PRINT:PRINT"AFTER THE LAST STUDENT HAS FINISHED"
2225 PRINT"TYPE 'END' WHERE STUDENT'S NAME IS ASKED FOR."
2230 GOTO 2500
2300 CLS:PRINT@6*64,"BE SURE THERE IS A DISK IN THE DRIVE"
2305 PRINT"AND THAT IT HAS ROOM FOR THE FILE ON IT."
2310 GOTO 2500
2400 CLS:PRINT@4*64,"THE STUDENT'S RECORD WILL BE PRINTED"
2405 PRINT"ON THE SCREEN AT THE END OF HIS LESSON."
2410 PRINT"BE SURE THE STUDENT UNDERSTANDS HE IS TO"
2415 PRINT"COPY IT AND HAS MATERIALS FOR THIS."
2500 PRINT:PRINT"IF YOU WANT TO CHANGE YOUR MIND TYPE 1."
2505 PRINT"OTHERWISE PRESS <ENTER>"
2510 INPUT A$:IF A$="1" THEN 2000
2515 RETURN
2520 END
3000 'RECORD KEEPING METHODS
3005 R$=X$
3010 ON R GOTO 3100, 3200, 3300, 3320
3100 LPRINT R$
3105 RETURN
3200 CMD"T"
3205 PRINT#-1,R$
3210 CMD"R"
3215 RETURN
3300 'FOR NEWDOS ONLY
3305 OPEN"E",1,"RECORDS/TXT"
3310 PRINT#1,R$
3315 CLOSE
3320 RETURN
4000 END
```

Program Listing 9

```
10 ' PROGRAM COPYRIGHT 1979
15 ' BY JEROME I. WEINTRAUB
20 ' EL CAJON, CA.
25 ' WORDS IN DATA FROM "BASIC SPELLING"
30 ' PUB. BY J. B. LIPPINCOTT COMPANY
35 INPUT"HOW FAST";H1    'REPLACE THIS LINE WITH H1=7100
40 DIMA$(2800)
45 CLS:PRINT CHR$(23)
50 PRINT@6*64,"PLEASE REWIND THE CASSETTE TAPE."
55 PRINT"I'LL WAIT FOR YOU..."
60 OUT 255,255
65 FOR Z=1 TO H1:NEXT
70 OUT 255,0
75 CLS
80 PRINT CHR$(23)
85 PRINT@6*64,"TIME FOR SPELLING!"
90 PRINT:PRINT"TYPE YOUR NAME, PLEASE..."
95 INPUT E$
100 PRINT:INPUT"WHAT LESSON ARE YOU ON (1-11)";A
105 IF(A<1)OR(A>11)THEN 100
110 D=10*(A-1)+1
115 C=0:CLS:PRINT CHR$(23)
120 PRINT@6*64,"PRESS THE PLAY KEY"
125 PRINT"ON THE CASSETTE PLAYER."
130 PRINT:PRINT"PRESS ENTER WHEN YOU ARE READY.":INPUT A$
135 PRINT"TYPE THE WORD YOU HEAR"
140 PRINT"NEXT TO THE NUMBER"
145 PRINT"ON THE SCREEN."
150 N=0:F=D-1
155 FOR C=1 TO D+9
160 READ A$(C)
165 NEXT C
170 FOR E=D TO D+9:B$=""
175 F=F+1
180 OUT 255,255
185 FOR H=1 TO H1:NEXT
190 OUT 255,0
195 PRINT CHR$(23)
```

Program continues

KEYS TO PRODUCTIVITY



If you plan, make, use, or buy computer systems or services, you can't afford to miss the upcoming **National Computer Conference,** McCormick Place, Chicago, May 4-7.

The theme of this year's conference is "Keys to Productivity", how to use computers and computer services to help increase output and control costs.

Come see the latest in new equipment and money-saving techniques at over 300 exhibits. Take your pick of over 100 technical sessions and 21 Professional Development Seminars. Enjoy the Personal Computing Festival, now a part of the main conference.

Save time. Save money. Pre-register now for the most dynamic NCC ever. You can:

- Avoid long registration lines
- Save $15 on the full 4-day program
- Secure preferential housing and discount travel.

For Conference Information, call or write **National Computer Conference c/o AFIPS, P.O. Box 9658, 1815 N. Lynn Street, Arlington, VA 22209, (703) 558-3610.**

For discount air fares, airline reservations, hotel and show information, or to pre-register, call toll-free (800) 556-6882.

✓381

NCC'81

✓Reader Service—see page 290

"By using the OUT function to control the cassette recorder, you can help the student learn orally as well as visually."

```
200 PRINT F;:INPUT B$:IF B$=""THEN 200
205 IF B$<>A$(E) THEN N=N+1:A$(N)=A$(E):GOSUB 325
210 IF N=5 THEN 270
215 NEXT E
220 PRINT"VERY GOOD, ";E$;"."
225 IF N>0 THEN 245
230 PRINT"YOU GOT A PERFECT SCORE!!!"
235 PRINT "10 OUT OF 10!!"
240 GOTO 260
245 PRINT"YOU GOT ONLY";N;"WRONG"
250 PRINT"OUT OF 10!!"
255 PRINT"STUDY...":FOR G=1 TO N:PRINTA$(G):NEXT G
260 PRINT"YOUR NEXT LESSON IS NO.";A+1;"."
265 GOTO 285
270 CLS:PRINT CHR$(23):PRINT"THAT'S ALL FOR TODAY."
275 PRINT"STUDY...":FOR G=1 TO N:PRINTA$(G):NEXT G
280 PRINT"YOUR NEXT LESSON IS NO.";A;"."
285 PRINT"TELL YOUR TEACHER."
290 PRINT"GOODBYE, ";E$;"."
295 FOR X=1 TO H1:NEXT
300 CLS:PRINT CHR$(23)
305 PRINT@6*64,"INSTRUCTIONS:"
310 PRINT:PRINT:PRINT"PLEASE TYPE RUN AND"
315 PRINT"PRESS <ENTER> KEY."
320 END
325 CLS:PRINT CHR$(23)
330 PRINT"THE WORD IS.....";A$(N);"."
335 PRINT:PRINT"TYPE IT ONCE"
340 PRINT"AFTER EACH ? MARK."
345 FOR L=1 TO 5:C$=""
350 INPUT C$
355 PRINT C$;:IF C$<>A$(N) PRINT" IS WRONG. TYPE ";A$(N);".":GOTO
    350
360 PRINT" IS RIGHT, ";E$;"!"
365 IF L=5 PRINT"READY FOR THE NEXT WORD..."
370 NEXT L
375 RETURN
380 END
500 DATA INK,OCTOBER,BRICK,ACT,MARKET,TICKET,ARITHMETIC
505 DATA BUCKET,STRIKE,BLANKET,SUBTRACT,INSECT,SUBJECT
510 DATA ACRE,ARCTIC,DRINK,THINK,CRICKET,THICKET
515 DATA BISECT,DISSECT,TRACTOR,ATTRACTIVE,ACREAGE
520 DATA ACTIVITY,SECTION,COUNTERACT,ANTARCTIC
525 DATA PEOPLE,FRIEND,SUIT,FRUIT,BUILT,MOVIE,BUILD
530 DATA FIELD,PIECE,JUICE,POEM,CHIEF,BRUISE
535 DATA SHIELD,POETRY,BUILDING,BRUISING,SHIELDING
540 DATA WIELD,YIELD,FRIENDLY,CHIEFLY,PEACE
545 DATA FIERCE,SUITABLE,JUICY,CRUISE,HANDKERCHIEFS
550 DATA SON,MONDAY,BECOME,OVEN,MONTH,FRONT,TON
555 DATA DOZEN,BOTTOM,APRON,DOCTOR,SECOND,RIBBON
560 DATA BUTTON,COMPANY,NAPKIN,PERSON,PRISON
565 DATA PHANTOM,COMPANION,ACCUSTOM,MILLION,WEAPON
570 DATA OPINION,DIAMOND,PIGEON,REASON,POISON
575 DATA EARTH,EARLY,GREAT,LEARN,WEATHER,DEATH
580 DATA HEALTH,THREAD,BREAK,BREAKFAST,INSTEAD
585 DATA HEAVY,STEAK,PEARL,HEARD,STAKE,BRAKE
590 DATA HEAVIER,WHETHER,EARN,EARNESTLY,ENDEAVOR
595 DATA BREATHTAKING,DEAFENING,DREADFUL,STEADILY
600 END
```

Program Listing 10

one of those persons who has to write a difficult word before you are sure you spelled it correctly? If so, you are a visually oriented speller. Many experts state that spelling is primarily a visual skill, because so many sounds have a variety of spellings. I'm not selling the visual approach today, but it merits more attention than it receives.

In order to use the aural approach with your TRS-80, you need to record the words on a cassette tape and synchronize it with the computer. Follow these instructions:

To ensure proper synchronization, you *must* use leaderless cassette tapes. Many tapes prepared for computer use can be purchased this way. The tape must be leaderless so that you will know almost exactly where the audio will begin when the student activates it. Find a quiet place to work so that no extraneous sound can reach the tape. You can use a watch with a second hand or a stop watch to time your inputs, but I recommend a computer program (illustrated). I say this because I found it confusing to clock the time and record simultaneously; also, the use of a program will insure closer synchronization because the recorder starts and stops gradually, which makes it difficult to time with a watch.

```
10 OUT 255, 255
20 PRINT "START  D I C T A T I N G . . ."
30 FOR X = 1 TO 7100 (Approximately 15 seconds)
40 OUT 255, O
50 PRINT "STOP. PREPARE FOR THE NEXT WORD."
60  INPUT "PRESS <ENTER>";A$
70  GOTO 10
```

Fifteen seconds is enough time to say the word, a sentence containing the word, and the word again: 'INK. . . . . WRITE THE LETTER IN INK. . . . . INK.' A loop of 1 to 7100 may not come out to exactly 15 seconds, but if you use the same loop in the program as you use recording the words, you will have perfect synchronization, even if it is not a specific number of seconds. The spelling program is shown in Program Listing 10.

Line 35 is a convenience for the programmer. If H1 were set to 7100 while I was writing and debugging the program, I would have had to wait 15 seconds each time a word was asked for. By using a variable, I can set it to zero if I want to, or I can set it to 7100 to test the running of the program.

Lines 50 and 55 instruct the student to rewind the cassette tape. The lesson number is provided in line 100. Lines 120 to 145 are further directions to the student. Lines 60 and 180 turn on the cassette recorder, while lines 70 and 190 turn it off.

By using the OUT function to control the cassette recorder, you can help the student learn orally as well as visually. ■

PROGRAMMING TOOLS FOR YOUR TRS-80

INSIDE LEVEL II

The Programmers Guide to the TRS-80 ROMS

INSIDE LEVEL II is a comprehensive reference guide to the Level II 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 including tape formats, RAM useage, relocation of Basic programs, USR call expansion, creating SYSTEM tapes of your own programs, interfacing of Basic variables directly with machine code, a method of greatly increasing the speed at which data elements are stored on tape, and special precautions for disk systems. INSIDE LEVEL II is a clearly organized reference manual. It is fully typeset and packed with nothing but useful information. It does not contain questions and answers, ROM dumps, or cartoons. **INSIDE LEVEL II.....$15.95**

4 SPEED OPTIONS FOR YOUR TRS-80

The SK-2 clock modification allows CPU speeds to be switched between normal, an increase of 50%, or a 50% reduction; selectable at any time without interrupting execution or crashing the program. Instructions are also given for a 100% increase to 3.54 MHz, though the TRS-80 is not reliable at this speed. The SK-2 may be configured by the user to change speed with a toggle switch or on software command. It will automatically return to normal speed any time a disk is active, requires no change to the operating system, and has provisions for adding an LED to indicate when the computer is not at normal speed. It mounts inside the keyboard unit with only 4 necessary connections for the switch option (switch not included), and is easily removed if the computer ever needs service. The SK-2 comes fully assembled with socketed IC's and illustrated instructions. **SK-2.....$24.95**

TELECOMMUNICATIONS PROGRAM

This machine language program allows reliable high speed file transfers between two disk-based computers over modems or direct wire. It is menu driven and extremely simple to use. Functions include real-time terminal mode, save RAM buffer on disk, transmit disk file, receive binary files, examine and modify UART parameters, program 8 custom log-on messages, automatic 16-bit checksum verification of accurate transmission and reception, and many more user conveniences. Supports line printers and lowercase characters. With this program you will no longer need to convert machine language programs to ASCII for transmission, and you will know immediately if the transmission was accurate. **TELCOM.....$29.95**

PROGRAM INDEX FOR DISK BASIC

Assemble an alphabetized index of your entire program library from disk directories. Program names and free space are read automatically (need not be typed in) and may be alphabetized with a fast Shell/Metzner sort by disk or program. The list may also be searched for any disk, program, or extension; disks or programs added or deleted; and the whole list or any part sent to the printer. Finally, the list itself may be stored on disk for future access and update. 'The best thing since sliced bread' (January issue of '80 Microcomputing). Works with TRSDOS, NEWDOS, and NEWDOS/80. One drive and 32K required. **INDEX.....$19.95**

RAM SPOOLER AND PRINT FORMATTER

This program is a full feature print formatting package featuring user defineable line and page length (with line feeds inserted between words or after punctuation), screen dump, printer pause control, and baud rate selection. In addition, printing is done from a 4K expandable buffer area so that the LPRINT or LLIST command returns control to the user while printing is being done. Ideal for Selectric or other slow printers. Allows printing and processing to run concurrently. Output may be directed to either the parallel port, serial port, or the video screen. **SPOOLER.....$16.95**

INSTANT ASSEMBLER

The INSTANT ASSEMBLER is a new, powerful tape-based editor/assembler and debugger for the TRS-80 Model I. It features immediate detection of errors as the source code is entered, assembly to memory as well as to tape, a built-in single-stepping debugger, a compactly coded source format that uses 1/3 as much memory as standard source, the ability to produce relocatable code modules, and the ability to link-load independently written modules. In addition, the INSTANT ASSEMBLER has many operational features including single stroke entry of DEFB and DEFW, continuous editing of succesive lines, alphabetic listing of symbol table, separate commands for listing error lines or the symbol table, block move function, and verification of source tapes.

INSTANT ASSEMBLER includes three separate programs. The assembler itself includes the single-stepper and debugger. In this mode you may have full register displays, decimal or hex entry, forward or backward memory displays, disassembly of object code in memory, memory display in ASCII format, and hex-to-decimal or decimal-to-hex conversion. The single-stepper will step one instruction at a time or at a fast rate to any defined address. During assembly you may quickly switch from assembler to debugger and back again without losing the source code. This makes INSTANT ASSEMBLER an excellent learning tool for machine language programming. Also included on the tape are two versions of the linking loader which allow you to write your programs in smaller modules and link them together for final assembly.

INSTANT ASSEMBLER occupies 8375 bytes of memory. In a 16K machine this will leave you more than 7000 bytes which is enough to write assembly language programs of around 2000 bytes. This makes it ideal for users with only 16K machines. While this version was written specifically for tape systems, we will soon have a disk version as well. The instruction manual may be purchased separately for $5, which will apply towards the purchase of the INSTANT ASSEMBLER. **INTASM.....$29.95**

SINGLE STEP THROUGH RAM OR ROM

STEP80 allows you to step through any Basic or machine language program one instruction at a time, and see the address, hexadecimal value, Zilog mnemonic, register contents, and step count for each instruction. The top 14 lines of the video screen are left unaltered so that the 'target program' may perform its display functions unobstructed. STEP80 will follow program flow right into the ROMs, and is an invaluable aid in learning how the ROM routines function. Commands include step (trace), disassemble, run in step mode at variable step rate, display or alter memory or CPU registers, jump to memory location, execute a CALL, set breakpoints in RAM or ROM, write SYSTEM tapes, and relocate to any page in RAM. The display may also be routed to your line printer through the device control block so custom print drivers are automatically supported. **STEP80.....$16.95**

MACHINE CODE FAST FOURIER TRANSFORM

This complete package includes 3 versions of the machine language FFTASM routine assembled for 16, 32, and 48K machines, a short sample Basic program to access them, a 10K Basic program which includes sophisticated interactive graphing and data manipulation, and a manual of instructions and examples. The machine language subroutines use variables defined by a supporting Basic program to make data entry and retrieval extremely fast and easy for custom implementation. They perform 20 to 40 times faster than their Basic equivalent (256 points in 12.5 seconds), and require less than 1550 bytes of memory. The FFT is useful in analyzing stock market and comodity trends as well as for scientific information. **FFTASM.....$49.95**

DUPLICATE SYSTEM TAPES WITH CLONE

Make duplicate copies of any tape written for Level II. They may be SYSTEM tapes or data lists. The file name, load address, entry point, and every byte (in ASCII format) are displayed on the video screen. **CLONE.....$16.95**

MUMFORD MICRO SYSTEMS

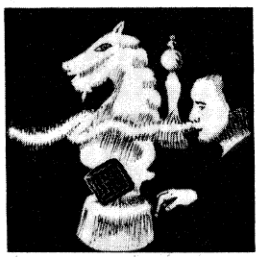
✓144

ORDERING: Complete satisfaction is guaranteed or a full refund will be made. All Model I programs are shipped on cassette unless $5 is included for a formatted (no system) disk. Include $1 postage and handling. California residents add 6% sales tax. Visa, Mastercharge and COD orders accepted.

Box 435-E Summerland, California 93067 (805) 969-4557

THE PROGRAM STORE

GAMBIET/80



By Win Rens from Microtrend
The new "King of micro chess" is here! Fresh from a victorious introduction at the London Tournament (which included top-ranked programs like Sargon II, Rook, Boris, Albatross and Fofner), GAMBIET/80 is rated the best chess program on the market. But don't let its credentials fool you: It is suitable for players at all levels, and is an excellent learning tool as well.

This new champion offers six levels of play, a chess clock (if equipped with RS expansion interface), "take-back" facility, continuous display of moves, and printout capability for recording games. GAMBIET/80 averages 30 moves in 60 minutes in tournament mode -- more than twice the speed of Sargon II. And the display not only indicates each move GAMBIET/80 considers, it shows which one is currently best. Play GAMBIET/80 and learn the meaning of the word "awesome!"

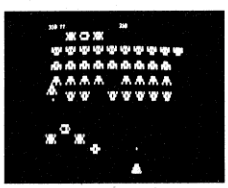
16K tape (transferrable to disk)...$39.95

LORDS OF KARMA

From Avalon Hill
A new adventure through the magical universe of Karma. While just staying alive may be tough enough, the Lords of Karma are watching your every move for deeds of kindness and bravery. You must explore the verdant forests, twisting trails, rugged mountains , and labyrinthine caverns in order to learn their secrets and complete your tasks. If successful, you will earn your place among the Lords, if not . . . ?

48K TRS-80, 32K Pet & Apple II
Tape.....$19.95

GALAXY INVASION



By Hogue & Konyu from Big-Five
"The rage of the arcades" is now available for TRS-80! Exciting sound effects add to the action as the invaders swoop down to destroy your base. Even while you have your hands full battling the aliens, you have to watch out for the Flagship! Super graphics, super action, super fun!

Level I or II, tape...$14.95

TRS-80 Level II 16K unless otherwise noted

EVEREST EXPLORER

By W. Godwin & D. Knowlton from Acorn
Challenge the world's highest moutain without ever leaving home. This remarkable simulation goes beyond most adventures, pitting you against challenges more terrifying than dragons and dungeons: the real-to-life horrors of one of man's most dangerous endeavors.

Assigned an expedition budget, you must select the manpower and supplies you will need to support your quest. Now the adventure begins as you conquer the elements and terrain, establishing ever higher encampments. Will you reach Everest's 29,028 foot summit alive?

Level II, 16K tape....$14.95
32K disk version (with "save game" feature and other enhancements)....$20.95.

Unbelievable Realtime 3-D Graphics!

FLIGHT SIMULATION

From Sub-Logic
The wait is over! If 3-D graphics seem impossible on the low resolution TRS-80, you haven't seen this brilliant program. During FLIGHT SIMULATION, you instantly select instrument flight, radar, or a breathtaking pilot's-eye-view. But be sure to strap yourself in -- you're liable to get dizzy!

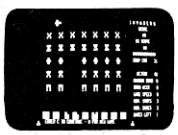
Once you put in some air time learning to fly your TRS-80, head for enemy territory and try to bomb the fuel depot and airstrip while fighting off five enemy warplanes. Good Luck!

Level I or II Tape...$25.00

SAVAGE ISLAND

By Scott Adams from Adventure International
A small island in a remote ocean holds an awesome secret. Will you be the first adventurer to uncover it? Wander the beach, uncover a bottle of rum, and make friends with a mean ol' bear. This is Scott Adams at his best -- and it's only the first installment of a new, multi-part adventure!

16K tape...$14.95 32K disk...$20.95



INVADERS FROM SPACE

by Carl Miller from Acorn
A fast machine language approach to this classic (and addictive) space game. As you play, the aliens drop bombs, move from side to side, and try to overrun your bases. Hold them off -- and score -- by shooting them down. But, just as you think you've got it all under control, the action speeds up.

Choose the game speed, enemy bomb frequency and accuracy, shots on screen and the number of your bases. Move your base and simultaneously fire at the invaders -- you cannot do this in most similar games. Full sound effects add even more excitement to the incredible speed and action of INVADERS FROM SPACE. Fun for all ages and skill levels.

Protected tape....$14.95
Protected disk....$20.95

STAR WARRIORS

By John Freeman from Epyx
You're on your own, light years from Earth in intergalactic space. If you thought saving the princess in "Rescue at Rigel" was a challenge, wait 'til you pit yourself against STAR WARRIORS! Order the second edition of the Starquest Series now.

TRS-80 16K tape, 32K disk; Apple II 48K disk $39.95 each

POKER TOURNAMENT

By J. Warshawer from Adventure Int.
Meet Tex, Slim, Bart, Doc and "The Kid" as you play five-card draw against these computer opponents. This delightful program lets you play out a favorite fantasy: a table-stakes game against a group of crafty high rollers. Not a bad way to brush up on your poker skills, either.

TRS-80 Level II, 16K tape (transferrable to disk)..$12.95
Apple II 24K disk...$19.95

OMNIKEY

From Discovery Bay
Bring your keyboard to life with new OMNIKEY. Single keystroke entry of 26 Basic statements, auto-repeat on all keys, and selectable upper/lower case (if your keyboard is so equipped) speed Basic program entry immensely. In edit mode, OMNIKEY's arrow-controlled cursor can be placed anywhere on the screen to perform auto-insertion, deletion, statement merging, even line repitition and repositioning! Once you've used OMNIKEY, you won't program without it.

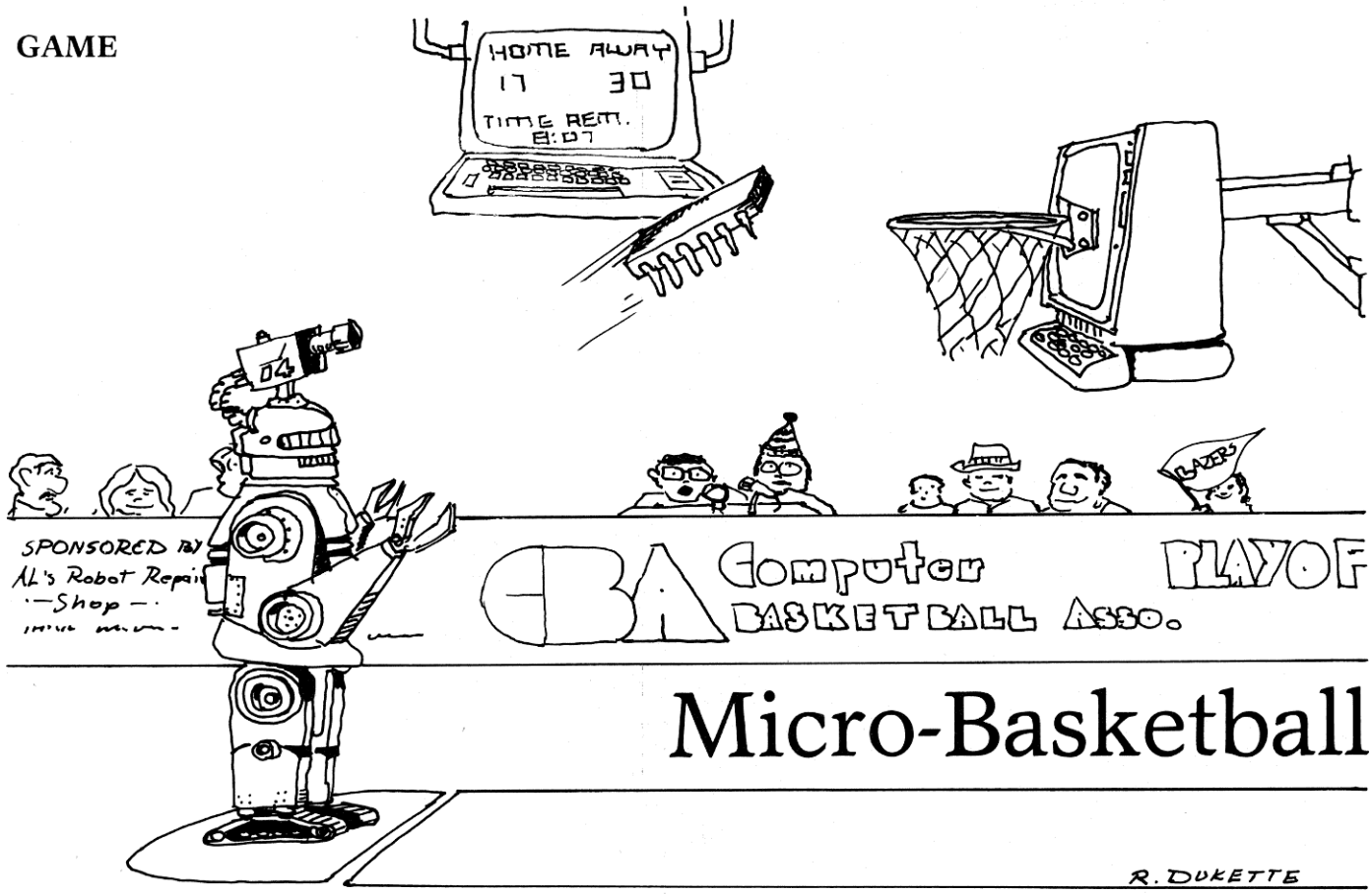
Tape (transferrable to disk)...$22.95

Visit Our New Store: W. Bell Plaza – 6600 Security Blvd · Baltimore, MD

TO ORDER CALL TOLL FREE 800 424-2738 For information Call (202) 337-4691

THE PROGRAM STORE
4200 Wisconsin Avenue NW, Dept. K9 Box 9609
Washington, D.C. 20016

MAIL ORDERS: Send check or M.O. for total purchase price, plus $1.00 postage & handling. D.C. residents, add 6% tax. Charge card customers: include all embossed information on card.



Micro-Basketball

Charles E. Weindorf
2828 Legion Rd.
Erie, PA 16506

I stepped onto the court to play another game of "death" basketball. Death meant no rules, no referees, and no sanity. I was at a disadvantage because my 130 pound, five foot-seven inch body didn't send the opposing six foot-seven inch center into a tailspin. I then asked, "Why kill myself? I like basketball, but not terminal injuries." Thus, the search for a good basketball game program began. Finally, I said, "I'll make it myself."

The Program

The Micro-Basketball program is designed for a Level II 16K TRS-80, and uses about 15.5K of memory to run. It should be input via cassette. For those willing to suffer a little keyboard cramp, the following hints should aid in the typing. Line 4330, for example, has 15 spaces between two words. Counting all those spaces slows the typing process. Before typing the instructions (and cursing at me) here is a simple way to avoid the ordeal. First, leave out lines 3360-4550; then change lines 3360, 3400, and 3750 to CLS: RETURN. The program still offers you directions and scorebook, but it only redraws the court at these selections.

Instructions

Most computer games are a battle of wits between man and computer, but this game's instructions make the reader wonder "In this battle of wits, am I out of ammunition?" The instructions to Micro-Basketball in some parts cannot be easily understood.

My friends who have played the game tell me that it is best to read the instructions quickly, and then refer back as needed. With this advice in mind, I made the program with an option to refer to the instructions before any offensive or defensive play. If you choose to leave the instructions out, however, refer to this article to explain the procedures in the game.

The instructions are split in two parts: the directions (the total set of instructions) and the scorebook.

Offensive Choices	Defensive Choices
outside shot (1)	3-2 defense (1)
inside shot (2)	2-1-2 defense (2)
choice shot (3)	man-man defense (3)

Table 1. Choices

Directions

The directions explain the graphics and the game's limits. Included are the offensive and defensive courts, and the set of players assigned to each team. Your team (Visitors) always has thin players, while the computer has fat ones. Each game is limited to two 15-minute halves (unless the game is tied and goes into overtime). Each offensive play uses up 20 seconds on the clock.

Scorebook

The scorebook section explains strategy and its effect on the shooting percentage. The three factors that change the shooting percentage are team set-up, team status, and a random number.

Team set-up determines the major portion of the shooting percentage (starts at 50 percent each play). Table 1 displays the three offensive and three defensive choices. Each offense against a certain defense has a different probability of success (just as in a real game). Table 2 gives all the possible offensive/defensive combinations and shows which are favorable to the offense or the defense.

Any combination that is favorable to the offense increases the shooting percentage by 15 percent. Any combination favorable to the defense decreases the shooting percentage by 15 percent. Any other offensive/defensive combination increases the

Favored Offensive Set-Ups		
(1) off. vs. (2) def.	(2) off. vs. (3) def.	(3) off. vs. (1)def.
Favored Defensive Set-Ups		
(1) off. vs. (1) def.	(2) off. vs. (2) def.	(3) off. vs. (3) def.
Others		
(1) off. vs. (3) def.	(2) off. vs. (1) def.	(3) off. vs. (2) def.

Table 2. Set-Ups

"The directions explain the graphics and the game's limits."

chance that a foul will occur from 15 to 35 percent.

Team status also changes the shooting percentage. Before each play, team status (aggressive or safe) is chosen along with team set-up. To explain team status, I will use an example from a real basketball game.

The defensive team has decided to play safely to stop any offensive plays that are designed to penetrate near the basket. The offensive team's play is aggressive (designed to penetrate) and the defense is able to stop the play before it is fully executed, forcing the shooter to take an uncomfortable shot. The shot is not likely to go in. The next time down the court, the defense decides to play aggressively, (to stop any shots away from the basket), but the offense decides to play aggressively also. This allows the offensive team to break through and give the shooter a comfortable shot. This shot probably goes in.

In Micro-Basketball, there are small arrows that signify aggressive and safe play. The arrows may seem confusing at first, but this determines if the enemy is playing aggressively or safely: If the arrows for a team are pointed at the other team, they play aggressively; if the arrows are pointed away, they play safely.

The effect upon the shooting percentage is this: If the offensive status and defensive status are the same, the shooting percentage increases by 10 percent. If they differ it decreases by 10 percent.

A random number is the third factor in shooting percentage. This random number from one to ten is added to the shooting percentage to make the game a higher scoring contest.

For example, the offense is playing an inside shot while the defense is playing the three-two. In Table 2, you find that it is one of the other setups that has no effect on the shooting percentage. Both offense and defense are playing aggressively, so the shooting percentage increases 10 percent and there is a random number of three.

Continues to p. 131

Right Guard (G)		G	
Right Foreward (R)	R		
Point Man (P)			P
Left Foreward (L)	L		
Center (C)		C	

Table 3. Positions

Program Listing

```
10 CLEAR200:CLS
20 DIMDF(2,3,5),OF(2,5),C(2,6):GOTO4620
30 GOSUB3330
40 T=15:S=00
50 TE=1:GOSUB1030
60 PB=170
70 IFT<10THENPA=171:GOTO90
80 PA=170
90 GOSUB3300
100 PRINT@242,"WHAT PLAY?";:PRINT@306,"(1)OUTSIDE";
110 PRINT@370,"(2)INSIDE";:PRINT@434,"(3)CHOICE";
120 PRINT@498,"(4)SCOREBOOK";:PRINT@562,"(5)DIRECTIONS";
130 K$=INKEY$:IFK$=""THEN130
140 PL=VAL(K$):IF PL<=0ORPL>5THEN130
150 FORA9=242TO562STEP64:PRINT@A9,"                ";:NEXT
160 PRINT@242,"WHAT MAN?";
170 K$=INKEY$:ONPLGOTO180,230,280,340,350
180 PRINT@306,"(1)POINT MAN";
190 PRINT@370,"(4)CENTER";:PRINT@434,"(5)R. GUARD";
200 K$=INKEY$:IFK$=""THEN200
210 ML=VAL(K$):IF ML><1ANDML><4ANDML><5THEN200
220 GOTO 360
230 PRINT@306,"(2)R. FORWARD";:PRINT@370,"(3)L. FORWARD";
240 PRINT@434,"(4)CENTER";
250 K$=INKEY$:IFK$=""THEN250
260 ML=VAL(K$):IFML><2ANDML><3ANDML><4THEN250
270 GOTO360
280 PRINT@306,"(1)POINT MAN";:PRINT@370,"(2)R. FORWARD";
290 PRINT@434,"(3)L. FORWARD";:PRINT@498,"(4)CENTER";
300 PRINT@562,"(5)R. GUARD";
310 K$=INKEY$:IFK$=""THEN310
320 ML=VAL(K$):IFML<=0ORML>5THEN310
330 GOTO360
340 GOSUB3750:GOTO50
350 GOSUB3400:GOTO50
360 K$=INKEY$:FORA9=242TO882STEP64:PRINT@A9,"                ";:NEXT
370 PRINT@242,"STATUS";:PRINT@306,"(1)AGGRESSIVE";:PRINT@370,"(2)S
    AFE";
380 K$=INKEY$:IFK$=""THEN380
390 SL=VAL(K$):IFSL><1ANDSL><2THEN380
400 K$=INKEY$
410 FORA9=242TO370STEP64:PRINT@A9,"                ";:NEXT
420 REM *** COMP DEF. CHOICE
430 PC=RND(3):SC=RND(2)
440 PA$=P1$:PB$=P3$:PC$=P2$:PD$=P4$
450 K$=INKEY$
460 GOSUB1510
470 FW=194:FX=218:FY=706:FZ=718:KK=1:TA=241:GOSUB 1870
480 IFTM=1THENTM=0:GOTO570
490 GOSUB2780
500 IFML=1THEN530
510 GOSUB1730
520 IFBK=1THENGOSUB2280:BK=0:GOTO550
530 GOSUB2840
540 GOSUB2390
550 PRINT@TA+450,"HIT (P)LAY";
560 K$=INKEY$:IFK$<>"P"THEN560
570 GOSUB1260
580 K$=INKEY$
590 TE=2:GOSUB1030
600 PB=128
610 IFT<10THENPA=129:GOTO630
620 PA=128
630 GOSUB3300
640 PRINT@192,"WHAT DEFENSE?";:PRINT@256,"(1)3-2";:PRINT@320,"(2)2
    -1-2";
650 PRINT@384,"(3)MAN-MAN";:PRINT@448,"(4)SCOREBOOK";
660 PRINT@512,"(5)DIRECTIONS";
670 K$=INKEY$:IFK$=""THEN670
680 PC=VAL(K$):IFPC<=0ORPC>5THEN670
690 ONPCGOTO 720,720,720,700,710
700 GOSUB3750:GOTO590
710 GOSUB3400:GOTO590
720 K$=INKEY$:FORA9=192TO512STEP64:PRINT@A9,"                ";:NEXT
730 PRINT@192,"STATUS";:PRINT@256,"(1)AGGRESSIVE";:PRINT@320,"(2)S
```

Program continues

```
     AFE";
740 K$=INKEY$:IFK$=""THEN740
750 SC=VAL(K$):IFSC><1ANDSC><2THEN740
760 FORA9=192TO320STEP64:PRINT@A9,"                ";:NEXT
770 REM ** COMP OFF. CHOICE
780 PL=RND(3):SL=RND(2)
790 ONPLGOTO800,840,860
800 XX=RND(3):IFXX=1THENML=1:GOTO880
810 IFXX=2THENML=4:GOTO880
820 ML=5
830 GOTO880
840 ML=RND(3)+1
850 GOTO880
860 ML=RND(5)
870 GOTO880
880 K$=INKEY$
890 PA$=P2$:PB$=P4$:PC$=P1$:PD$=P3$
900 GOSUB1510
910 FW=226:FX=250:FY=750:FZ=762:KK=-1:TA=193:GOSUB1870
920 IFTM=1THENTM=0:GOTO1010
930 GOSUB2780
940 IFML=1THEN970
950 GOSUB1730
960 IFBK=1THENGOSUB2280:BK=0:GOTO990
970 GOSUB2840
980 GOSUB2390
990 PRINT@TA+450,"HIT (P)LAY";
1000 K$=INKEY$:IFK$<>"P"THEN1000
1010 GOSUB1260
1020 K$=INKEY$:GOTO50
1030 CLS:PRINT@C(TE,6),STRING$(32,CHR$(140));
1040 PRINT@C(TE,6)+320,STRING$(32,CHR$(140));
1050 PRINT@1,STRING$(63,"-");
1060 PRINT@961,STRING$(62,"-");
1070 IFTE=1THENPRINT@0,CHR$(176);:GOTO1090
1080 PRINT@63,CHR$(176);
1090 FORA9=C(TE,1)TOC(TE,1)+895 STEP 64:PRINT@A9,B$;:NEXT
1100 FORA9=C(TE,2)TO C(TE,2)+320 STEP 64:PRINT@A9,B$;:NEXT
1110 PRINT@C(TE,3),CHR$(140)+CHR$(172)+CHR$(188);
1120 PRINT@C(TE,3)+67,CHR$(137)+CHR$(164);:PRINT@C(TE,3)+133,CHR$(
     169);
1130 PRINT@C(TE,3)+197,CHR$(154);:PRINT@C(TE,3)+259,CHR$(152)+CHR$
     (134);
1140 PRINT@C(TE,3)+320,CHR$(140)+CHR$(142)+CHR$(143);
1150 PRINT@C(TE,3)-4,CHR$(188)+CHR$(156)+CHR$(140);
1160 PRINT@C(TE,3)+58,CHR$(152)+CHR$(134);:PRINT@C(TE,3)+121,CHR$(
     150);
1170 PRINT@C(TE,3)+185,CHR$(165);:PRINT@C(TE,3)+250,CHR$(137)+CHR$
     (164);
1180 PRINT@C(TE,3)+316,CHR$(143)+CHR$(141)+CHR$(140);
1190 GOSUB1200:GOTO1230
1200 PRINT@C(TE,4),CHR$(152)+CHR$(134)+CHR$(140)+CHR$(144);
1210 PRINT@C(TE,4)+64,CHR$(137)+CHR$(164)+CHR$(140)+CHR$(129);
1220 RETURN
1230 PRINT@C(TE,5)," TIME  HOME VISITORS";
1240 RETURN
1250 REM CLOCK COUNTER
1260 S=S-20
1270 IFS<00THENS=40:T=T-1
1280 IFT=0ANDS=0THEN1300
1290 RETURN
1300 CLS:VV=VV+1:IFVV=2THEN1400
1310 TE=1:GOSUB1030
1320 PA=171:PB=170:GOSUB3300:FORA9=1TO3000:NEXT
1330 CLS
1340 PRINTCHR$(23):PRINT@472,"HALFTIME"
1350 FORA9=1TO3000:NEXT
1360 CLS:PRINT"**** DO YOU WANT TO QUIT NOW AND SAVE FACE";:INPUTI
     I$
1370 IFII$="NO"THENT=15:S=0:GOTO50
1380 IFII$><"YES"THEN1360
1390 PRINT"TRY AGAIN SOMETIME.":END
1400 IFH(1)<>H(2)GOTO1430
1410 T=4:S=0:VV=1:PRINTCHR$(23):PRINT@472,"OVERTIME!"
1420 FORA9=1TO3000:NEXT:GOTO50
1430 IFH(1)<H(2)THEN1480
1440 PRINTTAB(21);"****** YOU WIN ******"
```

Program continues

PMC-80 Expanded

Use all standard peripherals and existing software

When you buy PMC-80 you get hardware and software compatibility with the most popular microcomputer system in the world—that means thousands of disk and cassette based programs and all kinds of peripherals are instantly available!

PMC-80 has configurations that give the computer enthusiast a way to grow from a STARTER system in affordable increments. Begin at a low $675 for the basic 16K level II system and grow to the complete 48K memory system pictured above with two floppy disks for less than $3000.

FASTLOAD option inputs short programs as fast as "disk" from ordinary, standard format cassettes. Fast, reliable and economical!

PMC-80 COMMUNICATOR option provides interface to modems and parallel port printers. Take your pick of peripherals for communication with electronic bulletin boards and low cost timeshare services via phone lines from your home or business.

PMC-80 EXPANDER option provides the most powerful configuration with a total of 48K memory, provision for 4 mini-floppies, printer interface, RS-232C communications interface, plus a slot for the popular S-100 boards.

Sold through computer stores.

Personal Micro Computers, Inc. ✓422

475 Ellis Street, Mountain View, CA 94043 (415) 962-0220

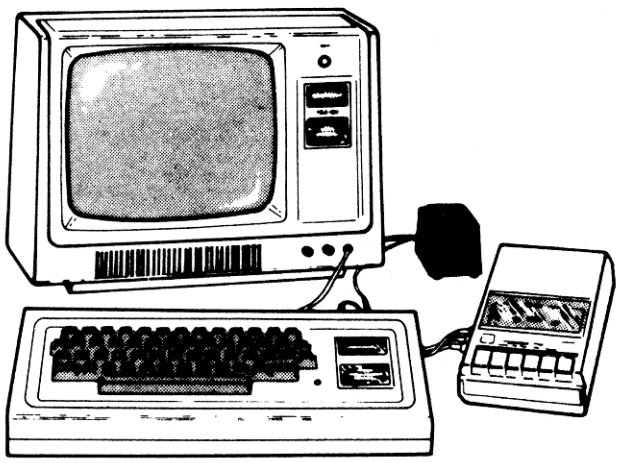
```
1450 PRINTTAB(21);"***** PURE LUCK *****"
1460 PRINTTAB(21);"HOME";H(2);:PRINTTAB(31);"VISITORS";H(1);
1470 GOTO1470
1480 PRINTTAB(21);"******* I WIN *******"
1490 GOTO1460
1500 REM ** SET UP PLAYERS ON THE SCREEN **
1510 FORA9=1TO5
1520 PRINT@OF(TE,A9),PC$;:PRINT@OF(TE,A9)+64,PD$;
1530 PRINT@DF(TE,PC,A9),PA$;:PRINT@DF(TE,PC,A9)+64,PB$;
1540 NEXT
1550 ONTEGOTO1560,1590
1560 IFSL=1THENOF$=CHR$(93):YW=-1:GOTO1610
1570 OF$=CHR$(94):YW=4
1580 GOTO1610
1590 IFSL=1THENOF$=CHR$(94):YW=4:GOTO1610
1600 OF$=CHR$(93):YW=-1
1610 ONTEGOTO1620,1650
1620 IFSC=1THENDF$=CHR$(94):YX=4:GOTO1670
1630 DF$=CHR$(93):YX=-1
1640 GOTO1670
1650 IFSC=1THENDF$=CHR$(93):YX=-1:GOTO1670
1660 DF$=CHR$(94):YX=4
1670 FORA9=1TO5
1680 PRINT@OF(TE,A9)+YW,OF$;
1690 PRINT@DF(TE,PC,A9)+YX,DF$;
1700 NEXT
1710 RETURN
1720 REM ** DRIBBLE **
1730 IFTE=1THENA4=63ELSEA4=68
1740 FORA6=1TOA5
1750 PRINT@OF(TE,ML)+A4,CHR$(131);:GOSUB1810
1760 PRINT@OF(TE,ML)+A4,CHR$(140);:GOSUB1810
1770 PRINT@OF(TE,ML)+A4,CHR$(176);:GOSUB1810
1780 PRINT@OF(TE,ML)+A4,CHR$(140);:GOSUB1810
1790 PRINT@OF(TE,ML)+A4,CHR$(131);:GOSUB1810
1800 GOTO1830
1810 FORA7=1TO20:NEXT
1820 RETURN
1830 NEXTA6
1840 PRINT@OF(TE,ML)+A4,CHR$(128);
1850 RETURN
1860 REM **LOGIC**
1870 ONPLGOTO 1880,1900,1920
1880 PRINT@TA+63,"OUTSIDE";
1890 GOTO1930
1900 PRINT@TA+64,"INSIDE";
1910 GOTO1930
1920 PRINT@TA+64,"CHOICE";
1930 PRINT@TA+1,"OFF      DEF";
1940 ONPCGOTO 1950,1970,1990
1950 PRINT@TA+72,"  3-2";
1960 GOTO2000
1970 PRINT@TA+72," 2-1-2";
1980 GOTO2000
1990 PRINT@TA+72,"MAN-MAN";
2000 PRINT@TA+194,"**SHOOTER**";
2010 ONMLGOTO 2020,2040,2060,2080,2100
2020 PRINT@TA+259,"POINT MAN";
2030 GOTO2110
2040 PRINT@TA+258,"R. FORWARD";
2050 GOTO2110
2060 PRINT@TA+258,"L. FORWARD";
2070 GOTO2110
2080 PRINT@TA+260,"CENTER";
2090 GOTO2110
2100 PRINT@TA+259,"R. GUARD";
2110 S%=50:IFPL=PCTHEN2140
2120 IFPC=PL+1ORPL=PC+2THEN2130ELSEGOTO2160
2130 S%=S%+15:GOTO2170
2140 S%=S%-15
2150 GOTO2170
2160 F%=20
2170 IFSL=SCTHENS%=S%+10ELSES%=S%-10
2180 PRINT@TA+322,"HIT (P)LAY?";
2190 K$=INKEY$:IFK$><"P"THEN2190
2200 FF=15:FF=FF+F%:IFRND(100)<FFTHEN2920
2210 F%=0
2220 S%=S%+RND(10)
```

Program continues

```
2230 PRINT@TA+322," SHOT%="S%;
2240 IF RND(100)<S%THENMM=1:H(TE)=H(TE)+2:RETURN
2250 MM=0
2260 IFRND(100)>35THENRETURNELSEBK=1
2270 RETURN
2280 FORA9=1TO10:PRINT@TA+323,"BLOCKED! ";
2290 FORA8=1TO50:NEXT:PRINT@TA+323,"        ";
2300 FORA8=1TO50:NEXT:NEXT
2310 GOSUB4820:RETURN
2320 RETURN
2330 IF RND(100)>40THENRETURN
2340 PRINT@TA+321,"OFF REBOUND!";
2350 PRINT@TA+386,"HIT (P)LAY";
2360 K$=INKEY$:IFK$<>"P"THEN2360
2370 IFTE=1THEN50ELSEGOTO590
2380 RETURN
2390 REM **-IN-**
2400 IFMM=0THENGOTO2740ELSEMM=0
2410 PRINT@TA+323,"**-IN-** ";
2420 GOSUB2430:GOTO2490
2430 GOSUB3300
2440 FORA9=1TO5
2450 PRINT@C(TE,4),"    ";:PRINT@C(TE,4)+64,"    ";
2460 GOSUB1200
2470 NEXT
2480 RETURN
2490 IFPL=1THENHD=20+RND(10):GOTO2510
2500 GOTO2550
2510 IFHD<27THENPRINT@TA+386,HD;"FOOTER";:RETURN
2520 H(TE)=H(TE)+1:GOSUB3300:FORA9=1TO10:PRINT@TA+386,"3 POINTER";
2530 FORA8=1TO50:NEXT:PRINT@TA+386,"         ";:FORA8=1TO50:NEXT:N
     EXT
2540 GOSUB4820:RETURN
2550 IF ML<>4THEN2690
2560 FORA9=1TO4
2570 PRINT@TA+387,"S";:GOSUB1810
2580 PRINT@TA+388,"L";:GOSUB1810
2590 PRINT@TA+389,"A";:GOSUB1810
2600 PRINT@TA+390,"M";:GOSUB1810
2610 PRINT@TA+391,"D";:GOSUB1810
2620 PRINT@TA+392,"U";:GOSUB1810
2630 PRINT@TA+393,"N";:GOSUB1810
2640 PRINT@TA+394,"K";:GOSUB1810
2650 PRINT@TA+387,"         ";
2660 NEXT
2670 GOSUB4820
2680 RETURN
2690 IFPL<>2THEN2720
2700 SD=RND(15):IFSD<=5GOSUB2560:RETURN
2710 PRINT@TA+386,SD;"FOOTER";:RETURN
2720 SD=RND(25)+5:IFSD<27THEN2710ELSE2520
2730 RETURN
2740 PRINT@TA+323,"* MISS * ";:FORA9=1TO500:NEXT
2750 GOSUB2330
2760 RETURN
2770 REM ** SHOOT **
2780 QQ=ML:ML=1:GOSUB1730
2790 ONQQGOTO2820,2800,2810,2820,2820
2800 ML=5:GOSUB1730:GOTO2820
2810 ML=4:GOSUB1730:GOTO2820
2820 ML=QQ
2830 RETURN
2840 FORAA=OF(TE,ML)TOC(TE,4)STEPPM(TE,ML)
2850 MO=AA
2860 GOSUB3210
2870 IFMR=1THEN2890
2880 PRINT@AA,CHR$(176);
2890 NEXT
2900 RETURN
2910 REM *** FOUL SHOTS ***
2920 IFTE=1THENOF(TE,1)=OF(TE,1)-15ELSEOF(TE,1)=OF(TE,1)+13
2930 FORD9=1TO2:GOSUB1030
2940 GOSUB3300
2950 PRINT@TA,"FOUL CALLED";
2960 IFD9=1THENPRINT@TA+64,"FIRST SHOT ";ELSEPRINT@TA+64,"SECOND S
     HOT";
2970 FORA9=FW TOFX STEP12
```

Program continues

FINE TOOLS FOR YOUR TRS-80*



UTILITY I

Here's software utility that can make programming easier for you. These powerful tools will take much of the frustration out of essential operations. You'll wonder how you managed without them. They'll stamp your work with the mark of professionalism.

RENUM—Now you can easily renumber any Level II program to make room for modification or to clean up the listing. (Only for use with 16K of memory.)

DUPLIK—This program will let you duplicate any BASIC, assembler, or machine-language program, verify the data, and even copy Level I programs on a Level II machine.

See how efficient and well-ordered your programming becomes. (T1)

Order No. 0081R $9.95.

UTILITY II

One of the most popular and useful of the utility packages, Utility II is the second in a series of programs designed to take the drudgery out of editing.

This combination of programs adds flexiblity to your BASIC programs by allowing you to combine them with other BASIC programs, or with machine-language programs or routines.

CFETCH—You'll be able to merge BASIC programs, with consecutive line numbers, into one program. CFETCH can also search through any Level II program tape and display the file name for all the programs.

CWRITE—Combine subroutines that work in different memory locations into one program. CWRITE works with BASIC and/or one or more machine-language programs. It will even give you a general checksum to verify that your program hasn't dropped any bits.

Use these two programs as your dependable "assistants" to speed merging processes and as a totally reliable retrieval means in search and edit operations.

A fine tool for your TRS-80! (T1)

Order No. 0076R $9.95.

COMPRESSION UTILITY PACK

Do you want to add sound routines to a Space Trek program that already uses 16K? Or maybe you need an extra column in that financial report program, but when you run it, you get OM errors?

With a wave of your hand, and a little help from either of the COMPRESSION programs in this package, your problems are over.

COMPRESS-80—fits in 265 bytes, deletes spaces, and offers the choice of leaving REMark line numbers in the program, or deleting them altogether.

SUPERCOMPRESS—uses 767 bytes and can do everything COMPRESS-80 can do PLUS, it packs the program into the smallest possible number of multiple statement lines.

With the Compression Utility Pack and your own programming skill, you can add all those little extras to your BASIC programs. (T1)

Order No. 0246R $19.95.

ULTRA-MON

ULTRA-MON is a unique and powerful machine-language monitor. It is ROM independent and will function in Level II or DOS BASIC. With ULTRA-MON, you will be able to write, modify, study and debug machine-language programs. Plus, you'll be able to avoid the frustrations and "bomb-outs" usually associated with machine-language programming. ULTRA-MON displays, disassembles, traces (hardcopy trace disassembly, too!), modifies, relocates memory, prints and even relocates itself with simple commands. Using interpretive execution, ULTRA-MON allows you to put breakpoints in ROM. This powerful monitor can even fetch, decode, disassemble and analyze each instruction individually so that your program cannot bomb out.

ULTRA-MON is designed for the beginning machine-language programmer as well as the professional. The documentation contains a Simple Demonstration section geared to the novice programmer. Consequently, the program is a learning device as well as an extremely useful programming tool.

If you are serious about programming, you need to add this powerful utility to your library today. (T1)

Order No. 5003R $24.95.

TLDIS & DLDIS

You've bought a super machine-code program, but now wonder how it works. Maybe you even used a quick PEEK routine to glance through it when it was in memory. If so, you definitely noticed the complete lack of comments in the code, making it almost impossible for you to decipher and understand it.

Well, Instant Software's Labeling Disassemblers are the answer to your problem.

TLDIS (Tape-based Labeling DISassembler) and DLDIS (Disk-based Labeling DISassembler) are three-pass, label-assigning disassemblers which assign labels (where appropriate) to the routines in a machine-language program. Their output is almost identical to that of a hand-assembled source code.

You can send the disassembly to a line-printer (Radio Shack parallel port) for either TLDIS or DLDIS. (The difference between these utilities is the storage mode of the disassembly.)

TLDIS can send the disassembly to cassette tape, DLDIS can send it to disk; both send it to the video monitor. The stored disassembly from TLDIS may be reassembled with Radio Shack's EDTASM™—the disassembly from DLDIS, with Apparat's extension of EDTASM™. Because of the use of labels, it is a simple matter to change any object code program by disassembling it and then making changes to the resultant source code, without losing track of jump/load addresses. Labels start with "AA00" and increment up, in even numbered steps (AA02, AA04, etc.). The odd numbers (AA01, AA03, etc.) are left for you to use for the source code during reassembly.

The printing of the disassembly may be temporarily halted by using [SHIFT] @ (just as in BASIC) or it may be ended by pressing the [BREAK] key. It also has a comments column to display ASCII characters used in a LD or CP opcode.

TLDIS and DLDIS may be relocated in memory to avoid conflict with the program you disassembler.

The next time you need to "climb inside" a machine-code program, take DLDIS or TLDIS with you. We promise that it will be an easier journey.

Order No. 0230R (TLDIS) $14.95 (T1)
Order No. 0231RD (DLDIS) $19.95 (T2)

TO ORDER:

See Your Local
Instant Software Dealer
or
Order Directly:
CALL TOLL FREE
1-800-258-5473

*A trademark of Tandy Corporation

Instant Software™ ✓2

PETERBOROUGH, N.H. 03458
603-924-7296

```
2980 PRINT@A9,PA$;:PRINT@A9+64,PB$;
2990 NEXT
3000 FORA9=FYTOFZSTEP12
3010 PRINT@A9,PA$;:PRINT@A9+64,PB$;
3020 NEXT
3030 FORA9=FW+6TOFX-6STEP12
3040 PRINT@A9,PC$;:PRINT@A9+64,PD$;
3050 NEXT
3060 FORA9=FY+6*KKTOFZ+6*KKSTEP12
3070 PRINT@A9,PC$;:PRINT@A9+64,PD$;
3080 NEXT
3090 IFTE=1THENDC=482ELSEDC=476
3100 PRINT@DC,PC$;:PRINT@DC+64,PD$;
3110 PRINT@TA+192,"HIT (P)LAY?";
3120 K$=INKEY$:IFK$><"P"THEN3120
3130 ML=1:GOSUB2840
3140 IFRND(100)<80THENGOSUB2430:H(TE)=H(TE)+1
3150 GOSUB3300
3160 NEXT
3170 IFTE=1THENOF(TE,1)=OF(TE,1)+15ELSEOF(TE,1)=OF(TE,1)-13
3180 FORRR=1TO700:NEXT
3190 TM=1
3200 F%=0:RETURN
3210 REM ** 'SET' CHECK **
3220 MP=INT(MO/64)
3230 MQ=MO-(MP*64)
3240 FORA7=0TO1
3250 FORA6=0TO2
3260 IFPOINT(MQ*2+A7,MP*3+A6)THENMR=1:GOTO3290
3270 NEXT:NEXT
3280 MR=0
3290 RETURN
3300 PRINT@PA,T;:PRINT@PB+3,S;:PRINT@PB+8,H(2);:PRINT@PB+15,H(1);
3310 PRINT@PB+3,":";:IFS=0THENPRINT@PB+5,"0";
3320 RETURN
3330 PRINTCHR$(23)
3340 PRINT@462,"MICRO  BASKETBALL"
3350 FORA9=1TO3000:NEXT
3360 CLS
3370 INPUT"***** WOULD YOU LIKE DIRECTIONS? (Y)ES OR (N)O";ZZ$
3380 IFZZ$="N"THENRETURN
3390 IFZZ$<>"Y"THENCLS:GOTO3370
3400 CLS:PRINT"**** OBJECT OF THE GAME: TO OUT-SCORE THE COMPUTER"
     ;
3410 PRINT" WITHIN THE                                ALLOTTED TIME BY
     ";
3420 PRINT" CHOOSING THE CORRECT                                COMBINA
     ";
3430 PRINT"TION OF OFFENSIVE AND DEFENSIVE
       ";
3440 PRINT"PLAYS."
3450 PRINT
3460 PRINT"DESCRIPTION OF THE GAME: THE GAME IS PLAYED ON TWO HALF
     -";
3470 PRINT"COURTS,                              ONE FOR YOUR OFFENSE A
     ND";
3480 PRINT" ONE FOR YOUR                              DEFENSE. A SCOR
     E ";
3490 PRINT"BOARD IS IN THE CORNER                               OF THE "
     ;
3500 PRINT" SCREEN, UNDER WHICH A PLAY BY
       ";
3510 PRINT"PLAY DESCRIPTION IS GIVEN."
3520 PRINT
3530 PRINT"********** THE PLAYERS: YOUR PLAYERS:";P2$;
3540 PRINT"THE COMPUTER'S:";P1$;"."
3550 PRINTTAB(38);P4$;:PRINTTAB(57);P3$;
3560 PRINT
3570 PRINT@896,"ANY KEY TO CONTINUE?";
3580 K$=INKEY$:IF K$=""THEN3580
3590 CLS
3600 PRINT"*************** LIMITS: THE GAME IS LIMITED TO TWO 15 "
     ;
3610 PRINT"MINUTE                                HALVES. EACH OFFENSIV
     E ";
3620 PRINT"PLAY EQUALS 20                                SECONDS. THE "
     ;
```

Program continues

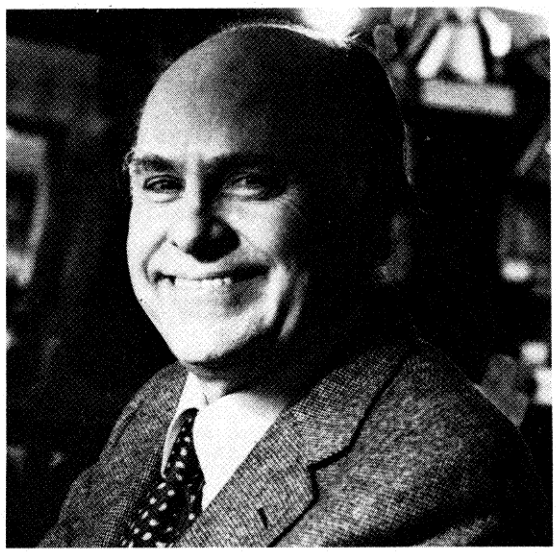
```
3630 PRINT"OFFENSIVE RESTRICTIONS ON                                EACH
     ";
3640 PRINT"PLAYER ARE";
3650 PRINT" DESCRIBED IN THE
3660 PRINT"'SCOREBOOK' SECTION."
3670 PRINT"ANY KEY?"
3680 K$=INKEY$:IFK$=""THEN3680
3690 CLS:TE=1:GOSUB1030
3700 PRINT@132,"YOUR OFFENSIVE COURT.";
3710 FORA9=1TO2500:NEXT
3720 CLS:TE=2:GOSUB1030
3730 PRINT@165,"YOUR DEFENSIVE COURT.";
3740 FORA9=1TO2500:NEXT
3750 CLS
3760 PRINTCHR$(23):PRINT@468,"SCOREBOOK":FORA9=1TO1000:NEXT
3770 CLS
3780 PRINT"THE PLAYERS ARE":PRINT"----------------------------"
3790 PRINT"THE RIGHT GUARD (G)":PRINT"THE RIGHT FORWARD (R)"
3800 PRINT"THE POINT MAN (P)":PRINT"THE LEFT FORWARD (L)"
3810 PRINT"THE CENTER (C)"
3820 PRINT@168,"G";
3830 PRINT@222,"R";:PRINT@306,"P";:PRINT@350,"L";:PRINT@424,"C";
3840 PRINT@448,"*** THE SET-UP ON THE OFFENSIVE COURT IS ";
3850 PRINT"PORTRAYED TO THE RIGHT.";
3860 PRINT"HIT 'C' TO CONTINUE."
3870 K$=INKEY$:IFK$<>"C"THEN3870
3880 GOSUB3890:GOTO3940
3890 CLS:PRINT"OFFENSIVE CHOICES                    DEFENSIVE CHOIC
     ES"
3900 PRINT"OUTSIDE SHOT (1)                    3-2 DEFENSE     (1)"
3910 PRINT"INSIDE SHOT  (2)                    2-1-2 DEFENSE   (2)"
3920 PRINT"CHOICE SHOT  (3)                    MAN-MAN DEFENSE (3)"
3930 RETURN
3940 PRINT
3950 PRINT"**** EACH PLAYER HAS A SHOOTING % OF 50 AT THE START OF
     ";
3960 PRINT"EACH         PLAY, BUT IT CAN BE AFFECTED BY THE ";
3970 PRINT"DEFENSIVE CHOICE."
3980 PRINT:PRINTTAB(15);"THE FAVORED OFFENSIVE SET-UPS."
3990 PRINTSTRING$(63,"-")
4000 PRINT"(1)OFF VS (2)DEF ** (2)OFF VS (3)DEF ** (3)OFF VS (1)DE
     F"
4010 PRINT"**** THIS WILL CAUSE THE SHOOTER'S % TO INCREASE 15%."
4020 PRINT
4030 PRINT"ANY KEY?"
4040 K$=INKEY$:IFK$=""THEN4040
4050 GOSUB3890
4060 PRINT
4070 PRINTTAB(15);"THE FAVORED DEFENSIVE SET-UPS."
4080 PRINTSTRING$(63,"-")
4090 PRINT"(1)OFF VS (1)DEF ** (2)OFF VS (2)DEF ** (3)OFF VS (3)DE
     F"
4100 PRINT"**** THIS WILL CAUSE THE SHOOTER'S % TO DECREASE 15%."
4110 PRINT:PRINTTAB(25);"THE OTHERS"
4120 PRINTSTRING$(63,"-")
4130 PRINT"(1)OFF VS (3)DEF ** (2)OFF VS (1)DEF ** (3)OFF VS (2)DE
     F"
4140 PRINT"**** THIS WILL NOT CHANGE THE SHOOTER'S %, BUT IT ";
4150 PRINT"INCREASES THE      CHANCE HE WILL BE FOULED."
4160 PRINT"ANY KEY?";
4170 K$=INKEY$:IFK$=""THEN4170
4180 CLS
4190 PRINTTAB(22);"SPECIAL OPTIONS."
4200 PRINTSTRING$(63,"-")
4210 PRINT"**** BOTH THE OFFENSE AND DEFENSE ARE REQUIRED TO PLAY
     ";
4220 PRINT"EITHER        'AGGRESSIVE' OR 'SAFE'. THIS ALSO AFFECTS
      THE ";
4230 PRINT"SHOOTING %.        IF BOTH THE DEFENSE AND OFFENSE PLAY
     ";
4240 PRINT"THE SAME, THE              SHOOTING % INCREASES 10%. IF
     ";
4250 PRINT"THEY PLAY DIFFERENT, ";
4260 PRINT"THE %         DECREASES 10%."
4270 PRINT
4280 PRINT"**** THE TYPE OF PLAY (AGGRES OR SAFE) IS SHOWN BY ";
4290 PRINTCHR$(93);" OR ";CHR$(94);"."
4300 PRINT:PRINT"**** EXAMPLE. IF AN ARROW (ON AN OFFENSIVE PLAYER
```

Program continues

Why should you read 80 Microcomputing?

80 MICROCOMPUTING is the best thing that ever happened to the TRS-80.* Through the articles and ads in *80* TRS-80* owners are able to keep up to date on the latest in everything to do with the system. Indeed, if there is any one factor which may carry Radio Shack through the coming blitz from Japan it will be their superiority of support information and programs. . . provided by *80.*

Be sure that you have a subscription to *80* and that every TRS-80* owner you know has one too. The more readers *80* has, the more material can be published each month so you have a vested interest in helping *80* to grow. Send in your subscription. . . and one for a good friend.



What is the one thing which makes your TRS-80* many times more valuable than any other computer? It's the wealth of information which is available for it. Through *80 MICROCOMPUTING* you have an incredible resource: far more information than is available for any other computer in the world. . . information that is available to you on a monthly basis. This information is priceless.

And what is the second thing which makes the TRS-80* more valuable than any other computer ever sold? It's the growing number of increasingly better programs written for it. You'll find *80* a continuing source of programs for your system. . . and through the ads, a key to the unfolding world of microcomputers.

*TRS-80 is a trademark of Tandy Corporation.

☐ YES, bill me for one year of 80 Microcomputing—$18.00

Name _____

Address _____

City _____ State _____ Zip _____

Canadian $20.00 US funds, 1 year only Foreign $28.00 US funds, 1 year only
Please allow 4 to 6 weeks for delivery
80 Microcomputing • PO Box 981 • Farmingdale NY 11737

311B8

FREE with software purchase — choice of: 1. One year subscription to InfoWorld 2. CP/M Summary ($4.95 value)

DISCOUNT Ad #11 SOFTWARE

✓ new items/new prices.

FANTASTIC PRICE PROTECTION POLICY

We'll match any advertised price on any item that we carry. And if you find a lower price on what you bought within 30 days of buying it, just show us the ad and we'll refund the difference.

It's that simple.

Combine our price protection with the availability of full professional support and our automatic update service and you have the Ultimate Software Plan.

It's a convenient, uncomplicated, logical way to get your software.

CP/M users: specify disk systems and formats. Most formats available.

	DISK WITH MANUAL	MANUAL ONLY
CP/M*		
OSBORNE ¶		
General Ledger#	$ 59	$20
Acct Rec/Acct Pay#	$ 59	$20
Payroll w/Cost#	$ 59	$20
Buy 2 get 1 free	$118	$57
All 3 & CBASIC-2	$199	$71
DIGITAL RESEARCH		
CP/M 2.2 Northstar	$149	$25
✓ CP/M 2.2 Micropolis	$169	$25
✓ CP/M 2.2 Durango F-85	$169	$25
CP/M 2.2 Cromemco	$189	$25
CP/M (other versions)	Call	
PL/I-80	$459	$35
Mac	$ 85	$15
Sid	$ 65	$15
Z-Sid	$ 95	$15
Tex	$ 70	$15
DeSpool	$ 50	$10
MICROSOFT		
Basic-80	$294	$30
Basic Compiler	$334	$30
Fortran-80	$384	$30
Cobol-80	$574	$30
Macro-80	$144	$20
Edit-80	$ 84	$20
MuSimp/MuMath	$224	$25
MuLisp-80	$174	$20
MICRO DATA BASE SYSTEMS		
HDBS	$250	$40
✓ MDBS	$795	$40
Other	Call	
S.O.F.T.W.A.R.E.		
MicroTax*‡		
Federal individual	$749	$50
Federal corporate	$249	$25
State individual	$249	$25
TCS‡		
✓ General Ledger	$ 79	$25
✓ Acct Receivable	$ 79	$25
✓ Acct Payable	$ 79	$25
✓ Payroll	$ 79	$25
✓ All 4	$269	$99
SUPERSOFT		
Forth (8080 or Z80)	$129	$25
Diagnostic I	$ 49	$20
Other disk software	less 10%	
SOFTWARE WORKS		
Adapt	$ 69	na
Ratfor	$ 86	na
COMPUTER PATHWAYS		
Pearl (level 1)#	$ 99	$25
✓ Pearl (level 2)#	$299	$40
✓ Pearl (level 3)#	$549	$65
COMPLETE BUSINESS SYSTEMS‡		
✓ Creator	$269	$25
✓ Reportor	$169	$20
✓ Both	$399	$45
MICROPRO		
✓ WordStar	$324	$60
Mail/Merge	$114	$25
✓ WordStar/Mail-Merge	$434	$85
✓ DataStar	$279	$60
✓ Word-Master	$119	$40
✓ SuperSort I	$199	$40
✓ SuperSort II	$169	$40
✓ SuperSort III	$119	$40
PEACHTREE*¶‡		
General Ledger	$399	$40
Acct Receivable	$399	$40
Acct Payable	$399	$40
Payroll	$399	$40
Inventory	$399	$40
Property Mgt	$799	$40
C.P.A. Client Write-up	$799	$40
Mailing Address	$349	$40
STRUCTURED SYSTEMS		
GL or AR or AP#	$747	$25
Payroll#	$747	$25
Inventory Control#	$447	$40
Analyst#	$197	$20
Letteright#	$167	$20
NAD#	$ 87	$20
QSORT	$ 87	$20
GRAHAM-DORIAN¶		
General Ledger#	$693	$40
Acct Receivable#	$693	$40
Acct Payable#	$693	$40
Job Costing#	$693	$40
Payroll#	$493	$40
Inventory#	$493	$40
Cash Register#	$493	$40
Apartment Mgt#	$493	$40
MICRO-AP		
Selector III-C2#	$269	$20
Selector IV#	$469	$35
✓ S-Basic	$269	$25
WHITESMITHS		
"C" Compiler★	$600	$30
Pascal (incl "C")★	$750	$45
EIDOS SYSTEMS		
Kiss	$299	$25
K-Basic	$529	$50
ORGANIC SOFTWARE		
TextWriter III	$111	$20
DateBook	$269	$25
SoHo Group		
✓ MatchMaker	$ 97	$20
✓ WorkSheet	$177	$20
"OTHER GOODIES"		
Tiny "C"	$ 89	$50
✓ Tiny "C" Compiler	$229	$50
CBASIC-2	$ 89	$15
Pascal/Z	$369	$30
Pascal/UCSD	$299	$30
Pascal/MT+	$224	$30
Pascal/M	$149	$20
Nevada Cobol	$129	$25
✓ Raid	$229	$25
✓ MAGSAM III	$129	$25
✓ MAGSAM IV	$259	$25
✓ BSTAM	$129	$10
FMS-80	$649	$45
dBASE II DBMS	$629	$50
Condor DBMS	$599	$30
Vulcan DBMS	$469	$30
T.I.M. DBMS‡	$329	$35
CBS	$369	$45
Whatsit?	$149	$25
✓ Ultra-Sort II	$159	$25
✓ MicroStat	$224	$15
String/80	$ 84	$20
✓ Vedit	$ 99	$15
Postmaster	$149	$20
WordSearch	$179	$25
SpellGuard	$269	$25
Spell Binder	$349	$45
✓ VTS/80	$489	$65
Magic Wand	$299	$45
Electric Pencil II	less 15%	
CPAids	less 12%	
APPLE II*		
MICROSOFT		
Softcard (CP/M)	$292	
Cobol	$574	
PERSONAL SOFTWARE		
Visicalc*	$122	
CCA Data Mgr	$ 84	
Desktop/Plan	$ 84	
✓ Zork	$ 34	
PEACHTREE *¶‡		
General Ledger	$224	$40
Acct Receivable	$224	$40
Acct Payable	$224	$40
Payroll	$224	$40
Inventory	$224	$40
MUSE		
Super-Text II	$127	
Other disk software	less 10%	
STC (Software Tech.)		
✓ Prof. Time & Billing	$279	
Other	less 15%	
"OTHER GOODIES"		
✓ Data Factory	$129	
Whatsit?*	$129	
✓ Creator	$229	
✓ LedgerPlus (GL A/R & A/P)	$549	
TRS-80® MODEL II		
CP/M 2.2 (P&T)	$159	$35
Electric Pencil II	less 15%	
NORTHSTAR		
✓ NorthWord	$299	
✓ Mail Manager	$239	
✓ Info Manager	$369	
✓ General Ledger	$749	
✓ Acct Receivable	$449	
✓ Acct Payable	$449	

THIS MONTH'S SPECIAL: T.I.M. DBMS JUST $299.
Terrific for inventory, mailings, financial, you-name-it! Menu-driven, auto-sort, 32000 records per file, any number of files, and dynamite documentation!

★—Special Bonus with order ‡—Requires microsoft BASIC ¶—Supplied in source code #—Requires CBASIC-2 ®—Mfgs Trademark

ORDERS ONLY—CALL TOLL FREE VISA • MASTERCHARGE

1-800-854-2003 ext. **823** • **Calif. 1-800-522-1500** ext. **823**

Overseas—add $10 plus additional postage • Add $2.50 postage and handling per each item • California residents add 6% sales tax • Allow 2 weeks on checks, C.O.D. ok • Prices subject to change without notice All items subject to availability •

THE DISCOUNT SOFTWARE GROUP ✓440
1610 Argyle Ave., Bldg. 102 • Los Angeles, CA 90028 • (213) 666-7677

ADD lowercase with our PLUG-IN piggyback board!

CENTRONICS

NOW! Add lowercase and optional second character sets to **ALL MODELS*** Centronics printers — including the popular 101A and 779: 96 Character ASCII; Optional character sets: APL, TRS-80/H-19 Graphics, Scientific, Customer-defined. *Except 730, 737 and 6000-series

Available now in two versions:

5W x 7H (all models*)—$95.00
5 x 7 optional sets—add $15.00 each

9W x 7H (101/101A/779)—$135.00
INTRO SPECIAL—FREE optional set on 9 x 7 order placed before 3/31/81

ALSO: Refurbished Printers!
Limited quantity now in stock:

730-1: $450 **306: $595**
101/101A: $795 **103: $995**
102: $1195 Shipped postage collect

DSE Digital Systems Engineering ✓368

12503 King's Lake Drive
Reston, Virginia 22091
(703) 620-2994

All products waranteed 90 days
MasterCard, VISA, Check, MO, PO

The Magic Cursor allows you to easily create screens (including graphics) on your video. A powerful command then generates the BASIC instructions to recreate the screen.

The Magic Cursor makes sophisticated Data Entry and Display easy. With **the Magic Cursor**, you define the data entry or display fields directly on your screen. The definition commands will generate the BASIC instructions to implement data entry and display.

The Magic Cursor will also
- create graphical characters simply on the screen
- duplicate characters on the screen easily
- store screens for later use in the same session
- reload screens created in previous sessions
- center blocks of text on the screen
- move blocks of text on the screen
- right justify text within a block on the screen

and more . . . it has 27 powerful commands!"

MAGIC CURSOR (32K Disk) Mod I $99.95

Trace-80™

TRACE-80 lets you observe the inner working of a machine language program. It allows you to run a machine language program in slow motion and watch the screen. You can stop execution at any time and examine the contents. You can execute your program and watch each instruction mnemonic and register contents list to the screen in place of normal screen display.

If you have a printer, **TRACE-80** allows you to execute your program in slow motion and watch the screen while your printer simultaneously prints the machine code being executed, the memory location and the instruction mnemonic, along with the current register contents.

TRACE-80 allows you to trace ROM as well as RAM because instructions are simulated in a special execution buffer.

TRACE-80/Mod I (16K Tape or Disk) $29.95
TRACE-80/Mod II $49.95

The Restauranteur's Consultant™

This food and beverage management tool dramatically reduces the human factors in food cost analysis. Those tasks required to effectively operate any restaurant or food service business. Over a half dozen reports give uniform, accurate and up to the minute information for profitability. And handling of daily menu changes requires only minutes per week instead of hours because the Consultant makes all the necessary conversions from your case prices.

Mod I, 32K, 2 Disk, Printer or Mod II $750.00

CUSTOM COMPUTER CENTER, INC.
For Ordering or Information Write
P.O. Box 58042 Houston, TX 77058
Attn: Jim Martens
or Call (713) 474-2428
✓7

WRITE FOR OUR COMPLETE SOFTWARE CATALOG!!

"Micro-Basketball proceeds much like a regular basketball game. Each offensive play has dribbling, passing, and shooting. . ."

Since the shooting percentage starts at 50, the final shooting percentage equals 50 + 0 + 10 + 3 = 63%.

Subroutines

The fun part of Micro-Basketball is its graphics, particularly seeing your men pass, dribble, and shoot (at least it's fun while you're winning). Even the computerized cheering sections get into the action. For organization's sake, I separated the program into 11 major subroutines, each designed to handle a specific part of the graphics. Dribbling (1720), shooting (2770), fouls (2910), player positioning (1500), and court drawing (1030) are called from the main body of the program (0-1020). The combination of these subroutines simulates the deployment of offenses and defenses in an actual basketball game.

All of the graphics are done using PRINT @ statements. The position of an object is found by each individual subroutine from a set of arrays using tables of numbers as starting points. From these, the computer draws the figures. The arrays C (court design), OF (offensive players), DF (defensive players) and PM (the moving basketball) are in two subdivisions, one for each half-court. The use of the subroutines and the arrays permits the program to execute the graphics quickly, enhancing the realism of the game.

The subroutine that prevents the basketball from erasing the players and the court could be useful in many programs. The @-to-POINT subroutine (lines 3210-3290) converts a PRINT @ number (MO) and

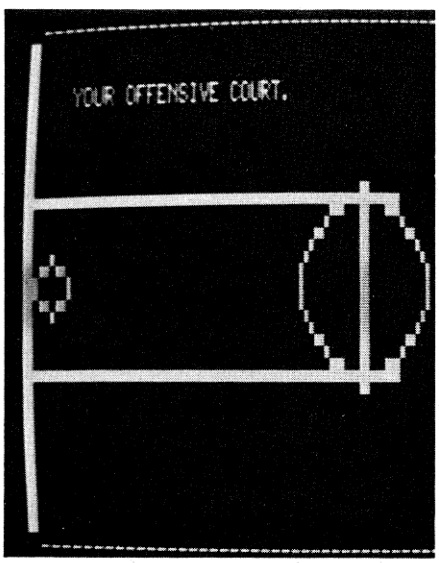


Photo 1. The player's offensive half court. Character string combinations are used to draw the basket and foul-circle.

```
     )";
 4310 PRINT" IS POINTED AT              THE BASKET, THE TEAM IS PLA
     Y";
 4320 PRINT"ING AGGRESSIVELY. IF                THE DEFENSE HAS ARR
     OWS";
 4330 PRINT" POINTED AT THE BASKET, THEY              ARE PLAYING";
 4340 PRINT" SAFELY. THE ";
 4350 PRINT"RESULT IS THAT THE % IS                DECREASED BY ";
 4360 PRINT"10%  (SEE RULE ABOVE)."
 4370 PRINT"ANY KEY?";
 4380 K$=INKEY$:IFK$=""THEN4380
 4390 CLS
 4400 PRINT"********* SHOT % RANGE: THE SHOT % OF A PLAYER IS ALSO
     ";
 4410 PRINT"ASSISTED                          BY A RANDOM NUMBER LES
     S ";
 4420 PRINT"THAN 10."
 4430 PRINT
 4440 PRINT"**** LIMITS ON PLAYERS: (1)OUTSIDE SHOT-POINT MAN, CENT
     ER,";
 4450 PRINT" AND                                          R. GUARD.
     "
 4460 PRINT"                        (2)INSIDE SHOT -CENTER, L. ";
 4470 PRINT"FORWARD, AND                                    R.
     ";
 4480 PRINT"FORWARD."
 4490 PRINT"                        (3)CHOICE SHOT -ALL PLAYERS."
 4500 PRINT"ANY KEY?"
 4510 K$=INKEY$:IFK$=""THEN4510
 4520 IFHE$="D"ORHE$="S"THEN4540
 4530 RETURN
 4540 GOSUB 1030
 4550 RETURN
 4560 DATA 64,352,353,449,107,320
 4570 DATA 127,352,353,507,65,352
 4580 DATA 493,138,778,860,93,464,179,819,865,97
 4590 DATA 476,196,708,728,216,456,194,706,736,224,487,132,772,854,
     87
 4600 DATA 482,248,760,741,230,501,250,762,733,221,470,185,825,871,
     103
 4610 DATA -6,62,-66,-68,60,6,66,-62,-60,68
 4620 P1$=CHR$(176)+CHR$(187)+CHR$(177)+CHR$(144)
 4630 P3$=CHR$(32)+CHR$(151)+CHR$(149)+" "
 4640 P4$=CHR$(32)+CHR$(150)+CHR$(148)+" "
 4650 B$=CHR$(191)
 4660 P2$=CHR$(176)+CHR$(155)+CHR$(177)+CHR$(144)
 4670 FORA9=1TO2:FORA8=1TO6
 4680 READ C(A9,A8)
 4690 NEXT:NEXT
 4700 FORA9=1TO2
 4710 FORA8=1TO5
 4720 READOF(A9,A8)
 4730 NEXT:NEXT
 4740 FORA9=1TO2:FORA8=1TO3:FORA7=1TO5
 4750 READDF(A9,A8,A7)
 4760 NEXT:NEXT:NEXT
 4770 FORA9=1TO2:FORA8=1TO5
 4780 READPM(A9,A8)
 4790 NEXT:NEXT
 4800 A5=2
 4810 GOTO30
 4820 REM ***** CROWD CHEERING
 4830 CLS
 4840 PRINTCHR$(23)
 4850 PRINT@456,"THE CROWD GOES ";
 4860 XL=RND(3):ONXLGOTO4870,4880,4890
 4870 PRINT"WILD":GOTO4900
 4880 PRINT"CRAZY":GOTO4900
 4890 PRINT"INSANE":GOTO4900
 4900 FORA9=1TO700:NEXT:CLS
 4910 FORA9=384TO896STEP256
 4920 PRINT@A9,STRING$(64,CHR$(131));
 4930 NEXT
 4940 FORA9=31TO991STEP64
 4950 PRINT@A9,CHR$(191)+CHR$(191);
 4960 NEXT
 4970 IFBK=0THEN5000
```

Program continues

"The fun part of Micro-Basketball is its graphics,. . . . Even the computerized cheering sections get into the action."

changes it into the number of lines down (MP) and the number of spaces over (MQ). It then checks all the blocks within a three-by-two space using MQ and MP to determine the POINT coordinates. The formula for the x coordinate is MQ * 2 + A7 and the y is MP * 3 + A6 (where A7 and A6 are the parameters of the loops). If there are any blocks SET within the space, the subroutine returns a 1 in MR. The shooting subroutine, in turn, skips printing the ball at that space.

```
4980 IFTE=1THENTE=2ELSETE=1
4990 HO$=PA$:PA$=PC$:PC$=HO$:HO$=PB$:PB$=PD$:PD$=HO$
5000 IFTE=1THENCW=257:CR=291:GOTO5020
5010 CW=291:CR=257
5020 FORA9=CRTOCR+512STEP256
5030 PRINT@A9,"";
5040 FORA8=1TO5:PRINTPA$"  ";:NEXT
5050 PRINT@A9+64,"";
5060 FORA8=1TO5:PRINTPB$"  ";:NEXT
5070 NEXT
5080 FORA9=CWTOCW+512STEP256
5090 PRINT@A9,"";
5100 FORA8=1TO5:PRINTPC$"  ";:NEXT
5110 PRINT@A9+64,"";
5120 FORA8=1TO5:PRINTPD$"  ";:NEXT
5130 NEXT
5140 FORA9=1TO5
5150 FORA8=CW-64TOCW+448STEP256
5160 FORA7=A8TOA8+24STEP6
5170 IFCC=1GOTO5210
5180 CC=1:PRINT@A7,PC$;:PRINT@A7+64,PD$;:PRINT@A7+128,"    ";
5190 NEXT
5200 GOTO5230
5210 CC=2:PRINT@A7,"    ";:PRINT@A7+64,PC$;:PRINT@A7+128,PD$;
5220 NEXT
5230 NEXT:NEXT
5240 RETURN
```

Sequence of Play

Micro-Basketball proceeds much like a regular basketball game. Each offensive play has dribbling, passing, and shooting, with a possibility of fouls, blocks, three-pointers, and even "slamdunks." There are three major segments that make up each offensive play. The set-up, the action, and the transition sections simulate most of the play of a regular game.

In the set-up section, the player and the computer pick offensive and defensive strategies. The offensive player picks the type of shot, a shooter, and a status, while the defensive player chooses a defense and a status. The two teams are then placed on the court. Since there are three types of defense, the defensive positioning will vary, but the offense will always be placed on the court as shown in Table 3. The scoreboard (the time clock and space directly beneath it) is then set up. It shows the offensive/defensive choices and the shooter. The play is ready to begin.

The action sequence continues until the offensive play is completed. As play begins there will be either a foul called, or the ball will be given to the shooter. If there is a foul called, the men will be lined up as in a real game and two shots taken. If play continues, the ball is dribbled and passed by the point man to the shooter.

In the next action sequence, the shooter will either take the shot or have it blocked. If the ball is blocked, play goes to the transition section. The shot, once away from the shooter, will either go in or be missed. If the shot goes in, there is a chance that it will be a "slamdunk" or a three-pointer. An inside shot of five feet or less, or any inside or choice shot by the center is counted as a "slamdunk." A shot of 27 feet or greater is counted as a three-pointer. If the shot misses, there is a 40 percent chance of an offensive rebound. An offensive rebound sends the program back to the set-up section. A made or a missed shot sends the program into the transition section.

The transition section handles the post-action play. If there is a blocked shot, "slamdunk", or three-pointer in the action section, a cheering section (one for each team) jumps up and down for the good play. After this is done, the other team is given control of the ball and the program goes back to the set-up section.

Well, armchair basketball fans, our time has come. Micro-Basketball lets us dribble, pass, shoot, "slamdunk," block, and cheer to our hearts' content. Ah, heaven. ■

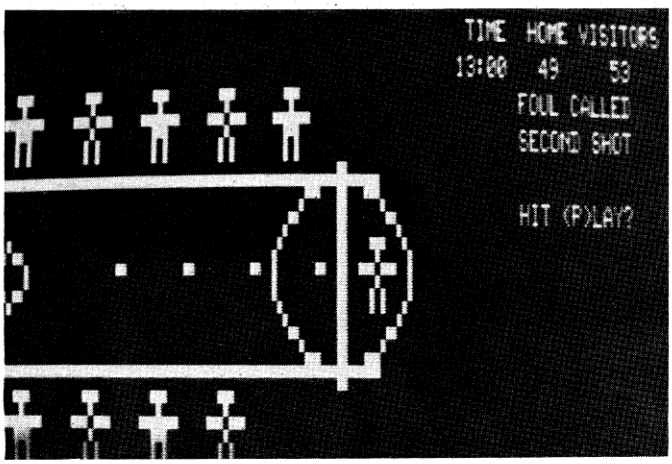


Photo 2. The two teams lined up for the second foul shot.

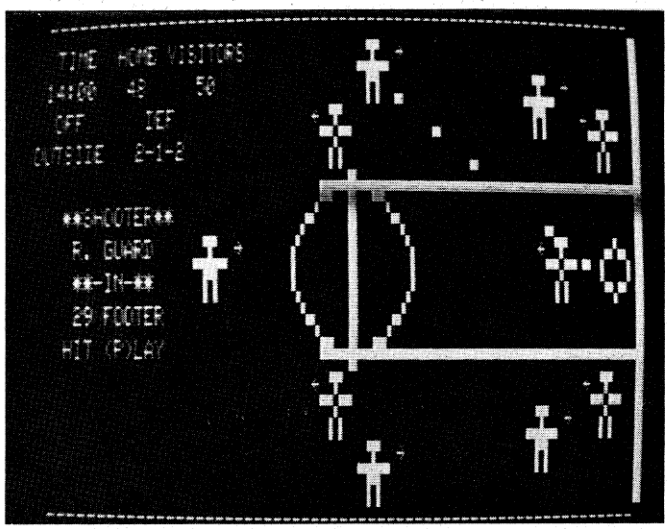


Photo 3. Action on the player's defensive court prior to the shot. The scoreboard to the left gives the facts of the play. (Photographs by Thomas Cwalina)

✓ Reader Service—see page 290


DRIVE 0 11/14/80 00 02 51
ATTRB FILE REC NMBR NMBR SPACE EOF
TYPE LEN RECS EXTS ALLOC USED BYTE
P-WKB F 256 91 3 95 91 0
D-WKB F 63 1241 13 310 306 48
D-WKB F 256 3 1 5 3 0
D-WKB F 63 301 4 75 74 212
D-WKB F 1 18229 1 40 40 244
D-WKB F 1 10937 1 45 43 154
D-WKB F 1 20 1 5 1 19
D-WKB F 256 35 2 35 35 0
D-WKB F 256 27 5 30 27 0
D-WKB F 256 6 1 10 6 0
D-WKB F 256 5 1 5 5 0
D-WKB F 1 17 1 5 1 16
D-WKB F 256 3 1
Taranto
& ASSOCIATES, INC.
The Total System Store

I started by selling programs, and a year later they said I was "the standard of the industry."

Now I'm selling the whole computer.

I'm Irwin Taranto, the one who changed the TRS-80* into a serious business computer.

Thousands of businesses tried my programs in the last year and a half, and sometimes it seems like every one of them has called me on the phone. With every call, I get another idea. I polish, alter, upgrade and correct these programs constantly.

By now I know how they work best, and exactly what they need in the way of peripherals. It's only logical that I should sell the whole computer system, not just the program diskettes.

So if you look at the computer in the picture, you'll see it says "Taranto" on it, not "TRS-80." The keyboard and CRT unit are a Tandy II* (that's what the manufacturer calls TRS-80 Model II when it's not sold through the Radio Shack). If it fits your needs better, though, we'll get the disk drive or the line printer somewhere else.

When you buy one of these Taranto computers, you get some serious advantages.

Some serious advantages.

You get hardware that's absolutely tailored to my programs. This means you'll be able to use every bit of the capability that's built into these systems.

You get my backup, down the line. If something goes wrong, we tell you how to fix it over the phone. If the problem's tough enough, I get on the phone myself. If we find out it's a hardware problem, you can get it fixed at any Radio Shack Service Center, even though it says "Taranto" on the machine. If you want to be completely covered, we'll arrange a service contract that takes care of everything.

In a lot of cases, we can help you set it up, too. I'm putting a group of authorized dealers together. Before long, they'll be all over the country, able to bring the equipment and programs right to your business. They'll spend a day or so with you helping you shake it down. It'll cost a little more, but it's good insurance.

The programs.

When you buy a Taranto computer, you're also buying these systems — any or all — each custom-tailored to your own needs, all interacting with each other, all integrated with the General Ledger.

General Ledger/Cash Journal
Accounts Payable/Purchase Order
Open Items Accounts Receivable/Invoicing
Balance Forward Accounts Receivable (new)
Payroll/Job Costing
Inventory Control (new)

Of course, if you already own a TRS-80 (any model), all our programs are available without the hardware.

Put it all together, and you have a truly serious, truly supported computer, software and hardware included — for as little as $8000.

I think they just might decide I've moved that "standard of the industry" up a notch or two.

Taranto
& ASSOCIATES, INC.

The Total System Store™

121 Paul Drive, San Rafael CA 94903.
Outside California, phone toll free (800) 227-2868.
In California (415) 472-2670.
Authorized dealers throughout America.

*Trademarks of the Tandy Corporation

A citizen's software alternative with a taxing program listing.

Income Averaging—1980

Margaret M. Grothman, CPA
Madison Area Technical College
Madison, WI 53703

Income averaging is an alternate method of federal tax computation which provides tax relief for persons whose incomes have increased sharply over a five year period. Federal income tax rates are progressive: They rise as income level rises. Because of this, a person who earns income in an uneven pattern over a period of years will pay more tax than one who earns income in a steady pattern. (See Table 1.) Income averaging reduces this additional tax for those who qualify. It treats a portion of the current year's taxable income as though it had been received evenly over five years instead of all in one year.

Requirements for Income Averaging

There are three requirements for income averaging. First, you must be a citizen or resident of the United States throughout the current tax year (called the computation year) and the four immediately preceding years (called the base period years). Secondly, you must have provided at least half of your own support in each of the base period years. On a joint return, both husband and wife must meet this requirement. The purpose of the support test is to deny the benefits of income averaging to persons first entering the job market. There are some exceptions to this support requirement; that is, certain conditions under which you may average income even though you do not meet the requirement. (See Table 2 for explanation and examples of the exceptions.)

The third requirement for income averaging is that averageable income must exceed $3,000. Averageable income is the amount by which taxable income of the computation year exceeds 120 percent of the average taxable income of the four base period years. The effect of this requirement is to eliminate normal increases in income from the special income averaging process.

Schedule G is used to calculate the tax by the income averaging method. It is divided into three sections, the first of which is used to determine base period income for each of the four preceding years. In the second part, the taxpayer computes averageable income—to determine eligibility and to use for the tax computation procedure which follows in the third section.

Base Period Income and Adjustments

Taxable income for the four preceding years must be averaged as a step toward determining averageable income. This sounds easy enough. Get out your tax returns for the last four years and look up taxable income on each one. Unfortunately, because of changes in the tax law and because of differences in the way the tax was determined, taxable income may not appear anywhere on the tax returns of those years. Schedule G provides the means for converting the numbers that do appear on those tax returns to the taxable income necessary for income averaging.

An important change in the tax law took place in 1977. Before that time, taxpayers deducted their choice of itemized deductions or a standard deduction from their adjusted gross incomes. They were also allowed a personal exemption deduction for each exemption claimed. The resulting figure was called taxable income and was the base for determining the tax. In an attempt to eliminate some of the arithmetic required of taxpayers, Congress changed the tax

(Without Income Averaging)

	Year 1		Year 2		Year 3		Year 4		Year 5		Total Tax
Taxpayer	Income	Tax	Income	Tax	Income	Tax	Income	Tax	Income	Tax	
1	$20,000	$4,177	$20,000	$4,177	$20,000	$4,177	$20,000	$4,177	$20,000	$4,177	$20,885
2	14,000	2,345	17,000	3,205	20,000	4,177	23,000	5,197	26,000	6,342	21,266
3	10,000	1,387	10,000	1,387	15,000	2,605	30,000	7,962	35,000	10,207	23,548
4	5,000	422	5,000	422	20,000	4,177	30,000	7,962	40,000	12,657	25,640
5	40,000	12,657	30,000	7,962	20,000	4,177	5,000	422	5,000	422	25,640

Total tax over the five year period is largest where the income pattern is most uneven. Income averaging can help taxpayers 2, 3, and 4, but not taxpayer 5, whose income is declining.

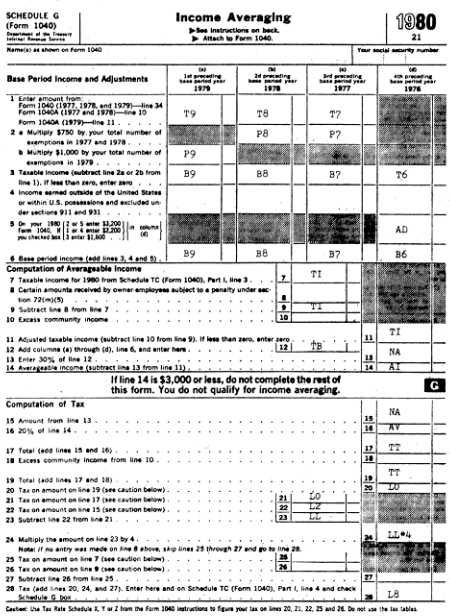
All taxes were computed using 1980 rates and exemptions. The table assumes a single taxpayer, under 65, with one exemption.

Table 1. Comparison of Taxpayers With Total Taxable Income of $100,000 Over a Five Year Period

computation procedure and redefined taxable income for tax years beginning in 1977.

Under the new law, taxpayers no longer deduct the standard deduction (renamed the zero bracket amount). They still receive the benefit of it, because tax rate schedules have been shifted to produce the same tax as before the change. In the new system, if a taxpayer wishes to itemize deductions, he or she deducts only the excess of total itemized deductions over the zero bracket amount. Taxable income is now defined as adjusted gross income less excess itemized deductions, less personal exemption deductions.

Because taxable income before 1977 and taxable income since 1977 do not have the same meaning, an adjustment is made to the taxable incomes of pre-1977 base period years to make them approximately equivalent to taxable income as defined in later years. The adjustment required is to add $2,200 if unmarried, and $3,200 if married, to taxable income. Why $2,200 and $3,200? Those were the zero bracket amounts in 1977, the year of the tax law change. This solution to the problem was quite arbitrary—other ways could have been chosen to correct for the redefinition of taxable income.



SCHEDULE G (Form 1040) — Income Averaging — 1980

Base Period Income and Adjustments

	(a) 1st preceding base period year 1979	(b) 2d preceding base period year 1978	(c) 3rd preceding base period year 1977	(d) 4th preceding base period year 1976
1 Enter amount from: Form 1040 (1977, 1978, and 1979)—line 34 Form 1040A (1977 and 1978)—line 10 Form 1040A (1979)—line 11	T9	T8	T7	
2 a Multiply $750 by your total number of exemptions in 1977 and 1978	P8	P8	P7	
b Multiply $1,000 by your total number of exemptions in 1979	P9			
3 Taxable income (subtract line 2a or 2b from line 1). If less than zero, enter zero	B9	B8	B7	T6
4 Income earned outside of the United States or within U.S. possessions and excluded under sections 911 and 931				
5 Or, your 1980 Form 1040, if you checked box	B9	B8	B7	AD
6 Base period income (add lines 3, 4 and 5)	B9	B8	B7	B6

Computation of Averageable Income

7 Taxable income for 1980 from Schedule TC (Form 1040), Part I, line 3	7	TI
8 Certain amounts received by owner-employees subject to a penalty under section 72(m)(5)	8	
9 Subtract line 8 from line 7	9	TI
10 Excess community income	10	
11 Adjusted taxable income (subtract line 10 from line 9). If less than zero, enter zero	11	TI
12 Add columns (a) through (d), line 6, and enter here	12	TB
13 Enter 30% of line 12	13	NA
14 Averageable income (subtract line 13 from line 11)	14	AI

If line 14 is $3,000 or less, do not complete the rest of this form. You do not qualify for income averaging. G

Computation of Tax

15 Amount from line 13	15	NA
16 20% of line 14	16	AV
17 Total (add lines 15 and 16)	17	TT
18 Excess community income from line 10	18	
19 Total (add lines 17 and 18)	19	TT
20 Tax on amount on line 19 (see caution below)	20	LO
21 Tax on amount on line 17 (see caution below)	21	LO
22 Tax on amount on line 15 (see caution below)	22	LZ
23 Subtract line 22 from line 21	23	LL
24 Multiply the amount on line 23 by 4. Note: if no entry was made on line 8 above, skip lines 25 through 27 and go to line 28.	24	LL*4
25 Tax on amount on line 7 (see caution below)	25	
26 Tax on amount on line 9 (see caution below)	26	
27 Subtract line 26 from line 25	27	
28 Tax (add lines 20, 24, and 27). Enter here and on Schedule TC (Form 1040), Part I, line 4 and check Schedule G box	28	L8

Caution: Use Tax Rate Schedule X, Y or Z from the Form 1040 instructions to figure your tax on lines 20, 21, 22, 25 and 26. Do not use the tax tables.

Fig. 2. Schedule G

On the 1980 Schedule G, only 1976 requires this adjustment. Next year, no such adjustment will need to be made, since all base period years will use the new definition of taxable income.

Another change which took place beginning in 1977 was the introduction of a new concept: tax table income. After subtracting excess itemized deductions (if any) from adjusted gross income, most taxpayers can look up the resulting tax table income directly in a table. The tax tables have columns corresponding to the number of exemptions claimed so that taxpayers no longer have to subtract personal exemption deductions. Taxpayers whose incomes are larger than the tables allow, or who have more exemptions than the tables provide still have to subtract their own exemption deductions and compute their tax from rate schedules.

Since the majority of taxpayers use the tables, they do not compute taxable income on their returns. Schedule G contains lines to convert tax table income to taxable income for the base period years. For 1977 and 1978, this requires subtracting $750 times the number of exemptions claimed. In 1979, the deduction was increased to $1,000 per exemption.

For most taxpayers, taxable income is the base period income used in the income averaging computation. However, if a person excluded some foreign earned income in a base period year, that amount would have to be added back to taxable income before proceeding with income averaging.

Computing Averageable Income

For the purpose of income averaging, taxable income of the computation year is divided in two parts, averageable income and non-averageable income. Averageable income has been defined as the amount by which taxable income of the computation year exceeds 120 percent of the average taxable income of the four base period years. Non-averageable income is therefore 120 percent of the average taxable income of the four base period years. Schedule G is designed to use a shortcut method to compute non-averageable income. Average base period income equals 1/4 of total base period income, and averageable income equals 120 percent or 6/5 of average base period income. Therefore, the arithmetic can be reduced to a one-step procedure by multiplying total base period income by 30 percent. (1/4 × 6/5 = 3/10). Line 13 on Schedule G is non-averageable income, although it is not labeled as such. Once non-averageable income is known, it is subtracted from total taxable income to obtain averageable income.

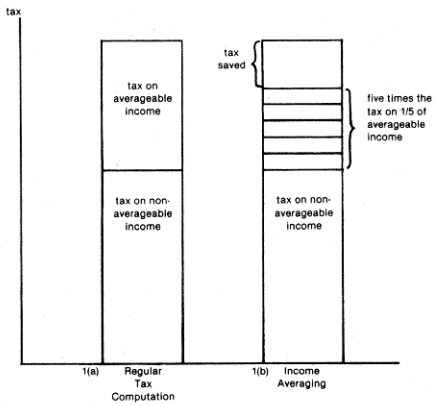


Fig. 1.

There are lines on Schedule G for two adjustments to the computation year taxable income, but they affect so few taxpayers that it isn't practical to include them in this income averaging program. On the great majority of Schedule Gs filed, these lines contain zeros. Lines 8, 10, 18, 25, 26, and 27 may be ignored when these adjustments are not required.

Computation of Tax

Fig. 1 illustrates the difference between the regular tax computation and the income averaging procedure. In 1(b), the top part of the bar (which in 1(a) represented the tax on averageable income) has been replaced by five times the tax on 1/5 of averageable income. Tax dollars are saved by avoiding the highest tax rates that apply to the taxpayer's highest levels of income.

1) There have been at least four years since the taxpayer turned 21 in which he or she was not a full-time student.	Example: A 32 year old teacher recently returned to work after having been unemployed and supported by his father for the last two years. He has not been a student for ten years. He may income average because of exception number 1.
2) Over half of the taxpayer's current year income is the result of work performed in two or more base period years.	Example: A writer has been working on a novel for the past three years while being supported by his girlfriend. This year, his only income consists of royalties from the book. He may income average because of exception number 2.
3) A person who meets the support test marries someone who does not. Income averaging on a joint return will be allowed if the spouse who does not meet the support requirement has 25% or less of their combined adjusted gross income.	Example 1: A recently married man and woman have incomes of $4,000 and $28,000. The husband does not meet the support requirement, but the wife does. They may income average on a joint return by exception number 3. Example 2: If the incomes of the couple above were each $16,000, they could not income average on a joint return, but they could file separate returns, the wife income averaging on hers. This may or may not result in lower tax than the joint return. Try both ways; separate returns with income averaging or a joint return without.

Table 2. Exceptions to the Support Requirement (Allowing persons to income average even though they have not provided half or more of their own support during the four preceding years)

On line 20, 21, and 22, the tax is computed using the 1980 tax rate schedules, not the tax tables. The tax tables are designed only for use with tax table income. Since all Schedule G computations are done with taxable income, the tax tables may not be used.

The Program

The program is written for a 16K Level II TRS-80. There is a subroutine for video display of Schedule G, and one for output to a line printer. Either or both may be used. The format for the printer option has been designed for use with a 1980 Schedule G. Printout could be on the form itself (if your printer accepts single sheets), or on plain paper to be copied along with a Schedule G transparency. This subroutine was written for use with a Radio Shack Line Printer IV (or Centronics 737). It may be necessary to modify the routine for use with other printers.

In this version, it is assumed that there is no foreign earned income in the base period years, so line 4 will always be zero. A subroutine for the foreign earned income adjustment could easily be added. It is also assumed that neither of the adjustments to averageable income mentioned before will be used.

The entire program could be used as a subroutine in a larger tax program. To ease program modification, all subroutines are labeled. Also, a Schedule G with variable names entered is shown in Fig. 2.■

Program Listing. 1980 Income Averaging

```
10 CLEAR 100: DIM BR(4,16), R(4,16): CLS
20 FOR N = 1 TO 4: FOR NN = 1 TO 16
30 READ BR(N,NN): READ R(N,NN): NEXT NN: NEXT N
40 PRINT: PRINT: PRINT: A$ = STRING$(50,140): GOSUB 90
50 PRINT: PRINT: A$ = "1980 INCOME AVERAGING": GOSUB 90
60 PRINT: PRINT: A$ = "BY MARGARET M. GROTHMAN, CPA": GOSUB 90
70 PRINT: PRINT: A$ = STRING$(50,140): GOSUB 90
80 FOR N = 1 TO 1200: NEXT N: GOTO 110
90 C$ = STRING$(32," "): W = 64 - LEN(A$): W = INT(W/2)
100 C$ = LEFT$(C$,W): PRINT C$; A$: RETURN
110 '****** INPUT FOR 1980 ******
120 CLS: PRINT: PRINT: A$ = "INDICATE FILING STATUS:": GOSUB 90
130 PRINT: PRINT: PRINT TAB(10) "1 - SINGLE"
140 PRINT TAB(10) "2 - MARRIED FILING JOINT RETURN"
150 PRINT TAB(10) "3 - MARRIED FILING SEPARATE RETURN"
160 PRINT TAB(10) "4 - HEAD OF HOUSEHOLD"
170 PRINT TAB(10) "5 - QUALIFYING WIDOW/WIDOWER"
180 PRINT: PRINT: INPUT "ENTER NUMBER"; S
190 IF S < 1 OR S > 5  GOTO 180
200 IF S <> INT(S) GOTO 180
210 INPUT "ENTER TAXABLE INCOME FROM SCHEDULE TC, LINE 3"; TI
220 IF TI <= 0 GOTO 210
230 '****** INPUT FOR BASE PERIOD YEARS ******
240 CLS: PRINT: PRINT TAB(5) "FOR 1977, 1978, AND 1979, ENTER TAX
     TABLE INCOME."
250 PRINT: PRINT TAB(7) "FOR EACH OF THE THREE YEARS, TAX TABLE IN
    COME"
260 PRINT TAB(7) "CAN BE FOUND ON LINE 34 OF FORM 1040."
270 PRINT: PRINT: PRINT TAB(10): INPUT "1977"; T7
280 PRINT TAB(10): INPUT "1978"; T8
290 PRINT TAB(10): INPUT "1979"; T9
300 PRINT: PRINT TAB(5);: INPUT "HOW MANY EXEMPTIONS WERE CLAIMED
     IN 1977"; E7
310 IF E7 < 1 GOTO 300
320 IF E7 <> INT(E7) GOTO 300
330 PRINT TAB(5);: INPUT "HOW MANY EXEMPTIONS WERE CLAIMED IN 1978
    "; E8
340 IF E8 < 1 GOTO 330
350 IF E8 <> INT(E8) GOTO 330
360 PRINT TAB(5);: INPUT "HOW MANY EXEMPTIONS WERE CLAIMED IN 1979
    "; E9
370 IF E9 < 1 GOTO 360
380 IF E9 <> INT(E9) GOTO 360
390 CLS: PRINT: PRINT: PRINT TAB(15) "FOR 1976 ENTER TAXABLE INCOM
    E."
400 PRINT: PRINT TAB(7) "TAXABLE INCOME FOR 1976 CAN BE FOUND ON F
    ORM 1040,"
410 PRINT TAB(7) "LINE 47 OR ON FORM 1040A, LINE 15."
420 PRINT: PRINT: PRINT TAB(15): INPUT "1976 TAXABLE INCOME"; T6
430 '****** COMPUTATION OF AVERAGEABLE INCOME ******
440 IF S = 2 OR S = 5 THEN AD = 3200: B6 = T6 + AD: GOTO 470
450 IF S = 1 OR S = 4 THEN AD = 2200: B6 = T6 + AD: GOTO 470
460 AD = 1600: B6 = T6 + AD
470 IF B6 < 0 THEN B6 = 0
480 P7 = 750 * E7: P8 = 750 * E8: P9 = 1000 * E9
490 B7 = T7 - P7: IF B7 < 0 THEN B7 = 0
500 B8 = T8 - P8: IF B8 <0 THEN B8 = 0
510 B9 = T9 - P9: IF B9 <0 THEN B9 = 0
520 TB = B6 + B7 + B8 + B9: NA = TB * .3
```

Program continues

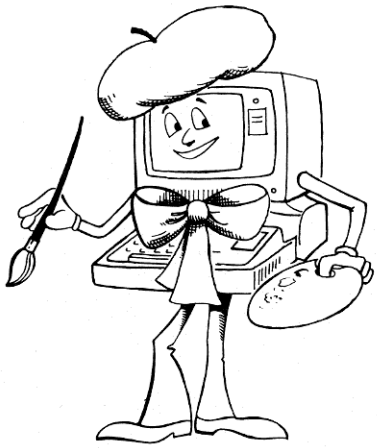
↙ Reader Service—see page 290

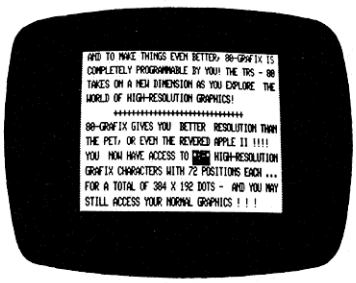
```
530 AI = TI - NA: IF AI <= 3000 GOTO 1510
540 '****** COMPUTATION OF TAX BY INCOME AVERAGING ******
550 AV = .2 * AI: TT = NA + AV: T = TT: GOSUB 1430: L0 = TAX
560 T = NA: GOSUB 1430: L2 = TAX
570 LL = L0 - L2: L8 = LL * 4 + L0
580 PRINT: PRINT TAB(11) "TAX BY INCOME AVERAGING IS" L8
590 T = TI: GOSUB 1430
600 PRINT TAB(11) "TAX BY REGULAR COMPUTATION IS" TAX
610 PRINT: PRINT: PRINT "PRESS ENTER TO CONTINUE......";
620 R$ = INKEY$: IF R$ = "" GOTO 620
630 CLS: PRINT: PRINT: PRINT TAB(3);: INPUT "DO YOU WANT A VIDEO D
    ISPLAY OF SCHEDULE G"; R$
640 IF LEFT$(R$,1) <> "Y" GOTO 1040
650 '****** VIDEO DISPLAY OF SCHEDULE G ******
660 PRINT TAB(10) "1979";: PRINT TAB(24) "1978";: PRINT TAB(38) "1
    977";: PRINT TAB(52) "1976"
670 F$ = "##,###.##"
680 PRINT "1";: PRINT TAB(7);: PRINT USING F$; T9;: PRINT TAB(21);
    : PRINT USING F$; T8;: PRINT TAB(35);: PRINT USING F$; T7;: G
    OSUB 1020
690 PRINT "2A & B";: PRINT TAB(7);: PRINT USING F$; P9;: PRINT TAB
    (21);: PRINT USING F$; P8;: PRINT TAB(35);: PRINT USING F$; P
    7;: GOSUB 1030
700 PRINT "3";: PRINT TAB(7);: PRINT USING F$; B9;: PRINT TAB(21);
    : PRINT USING F$; B8;: PRINT TAB(35);: PRINT USING F$; B7;: G
    OSUB 1010;: PRINT USING F$; T6
710 PRINT "4";: PRINT TAB(12) "0";: PRINT TAB(26) "0";: PRINT TAB(
    40) "0";: PRINT TAB(54) "0"
720 PRINT "5";: PRINT TAB(8);: PRINT STRING$(36,143);: GOSUB 1010;
    : PRINT USING F$; AD
730 PRINT "6";: PRINT TAB(7);: PRINT USING F$; B9;: PRINT TAB(21);
    : PRINT USING F$; B8;: PRINT TAB(35);: PRINT USING F$; B7;: G
    OSUB 1010;: PRINT USING F$; B6
740 PRINT STRING$(60,140)
750 PRINT TAB(28) "7";: PRINT TAB(33);: PRINT USING F$; TI;: GOSUB
     1020
760 PRINT TAB(28) "8";: PRINT TAB(38) "0";: GOSUB 1020
770 PRINT TAB(28) "9";: PRINT TAB(33);: PRINT USING F$; TI;: GOSUB
     1020
780 PRINT TAB(27) "10";: PRINT TAB(38) "0";: GOSUB 1030
790 PRINT TAB(45) "11";: GOSUB 1010;: PRINT USING F$; TI
800 PRINT TAB(27) "12";: PRINT TAB(33);: PRINT USING F$; TB
810 PRINT TAB(45) "13";: GOSUB 1010;: PRINT USING F$; NA
820 PRINT "PRESS ENTER TO CONTINUE DISPLAY...";
830 PRINT TAB(45) "14";: GOSUB 1010;: PRINT USING F$; AI;
840 IF INKEY$ = "" GOTO 840 ELSE GOTO 850
850 PRINT: PRINT STRING$(60,140)
860 PRINT TAB(45) "15";: GOSUB 1010;: PRINT USING F$; NA
870 PRINT TAB(45) "16";: GOSUB 1010;: PRINT USING F$; AV
880 PRINT TAB(45) "17";: GOSUB 1010;: PRINT USING F$; TT
890 PRINT TAB(45) "18";: PRINT TAB(54) "0"
900 PRINT TAB(45) "19";: GOSUB 1010;: PRINT USING F$; TT
910 PRINT TAB(45) "20";: GOSUB 1010;: PRINT USING F$; L0
920 PRINT TAB(27) "21";: PRINT TAB(33);: PRINT USING F$; L0;: GOSU
    B 1020
930 PRINT TAB(27) "22";: PRINT TAB(33);: PRINT USING F$; L2;: GOSU
    B 1020
940 PRINT TAB(27) "23";: PRINT TAB(33);: PRINT USING F$; LL;: GOSU
    B 1030
950 PRINT TAB(45) "24";: GOSUB 1010;: PRINT USING F$; LL * 4
960 PRINT TAB(27) "25";: PRINT TAB(37) "OMIT";: GOSUB 1020
970 PRINT TAB(27) "26";: PRINT TAB(37) "OMIT";: GOSUB 1030
980 PRINT TAB(45) "27";: PRINT TAB(54) "0"
990 PRINT TAB(45) "28";: GOSUB 1010;: PRINT USING F$; L8
1000 GOSUB 1010;: PRINT STRING$(9,"="): GOTO 1040
1010 PRINT TAB(49): RETURN
1020 PRINT TAB(47);: PRINT STRING$(11,191): RETURN
1030 PRINT TAB(47);: PRINT STRING$(11,143): RETURN
1040 '****** LINE PRINTER SUBROUTINE ******
1050 INPUT "WOULD YOU LIKE A SCHEDULE G PRINTOUT"; R$
1060 IF LEFT$(R$,1) <> "Y" GOTO 1400
1070 INPUT "WHAT IS THE TAXPAYER'S NAME"; N$
1080 INPUT "WHAT IS THE TAXPAYER'S SOCIAL SECURITY NUMBER"; S$
1090 FOR N = 1 TO 4: GOSUB 1410: NEXT N
1100 LPRINT TAB(5) N$;: LPRINT TAB(60) S$
1110 FOR N = 1 TO 4: GOSUB 1410: NEXT N
1120 LPRINT TAB(28);: LPRINT USING F$; T9;: LPRINT TAB(40);: LPRIN
     T USING F$; T8;: LPRINT TAB(51);: LPRINT USING F$; T7:
                                '** LINE 1 **
1130 GOSUB 1410: GOSUB 1410: LPRINT TAB(40);: LPRINT USING F$; P8;
     : LPRINT TAB(51);: LPRINT USING F$; P7:
                                '** LINE 2A **
1140 GOSUB 1410: LPRINT TAB(28);: LPRINT USING F$; P9:
     '** LINE 2B **
1150 GOSUB 1410: LPRINT TAB(28);: LPRINT USING F$; B9;: LPRINT TAB
     (40);: LPRINT USING F$; B8;: LPRINT TAB(51);: LPRINT USING F$
```

Program continues

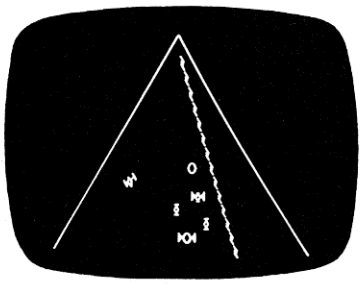
FROM PROGRAMMA
HI-RESOLUTION GRAPHICS FOR THE TRS-80®

LOWER CASE
The 80-GRAFIX board includes two sets of lower case characters at no additional cost.





INVERSE VIDEO
The 80-GRAFIX board allows you to do inverse video to high-light your screen displays.



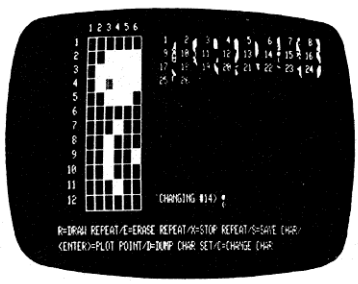
DEMONSTRATION PROGRAMS
The 80-GRAFIX board is supplied with a Character Generator software and several demonstration programs.

FINALLY, AT LAST. . .

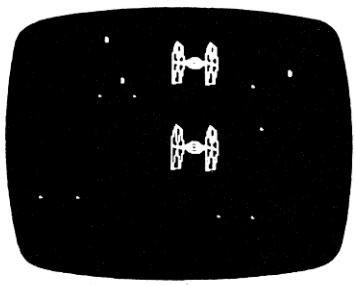
HI- RESOLUTION GRAPHICS is available for your TRS-80 computer system. The 80-GRAFIX board from PROGRAMMA International, Inc. gives your TRS-80 high resolution capability that is greater than the Commodore CBM/PET or even the revered APPLE II.

80-GRAFIX gives the TRS-80 an effective screen of 384X192 pixels, versus the normal 127X192 for the TRS-80, 80X50 for the CBM/PET, or the 280X192 of an APPLE II. As an added feature, 80-GRAFIX offers you lower case characters at no additional cost. Of course, you can also create your own set of up to 64 original characters using the supplied Character Generator software.

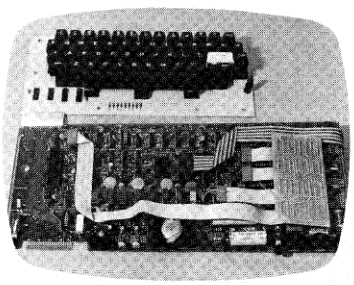
The 80-GRAFIX board is simple to install (note that this voids your Radio Shack warranty), and programming is done through BASIC. 80-GRAFIX opens up a whole new realm of software development and excitement never dreamed of for the TRS-80!



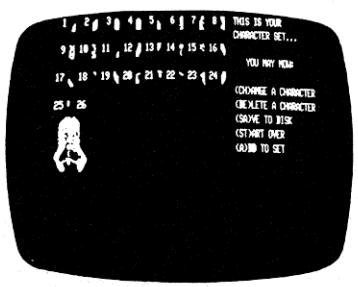
CHARACTER GENERATOR
The supplied character generator software allows you to create your own character set of up to 64 original characters.



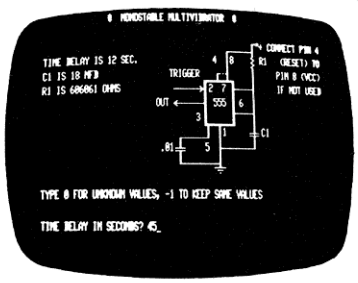
REAL-TIME GRAPHIC GAMES
With the 80-GRAFIX board you can write exciting real-time games using BASIC.



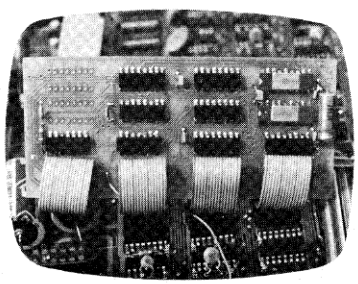
EASY INSTALLATION
The 80-GRAFIX board is simple to install and fits inside the TRS-80 case.



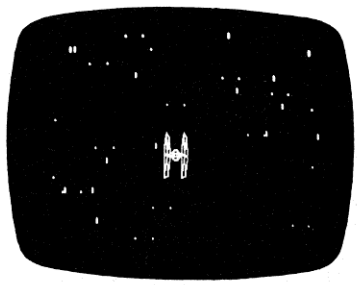
GRAPHICS GALORE
The 80-GRAFIX board and the supplied Character Generator allow you to become an artist.



ELECTRONIC DESIGN
The 80-GRAFIX board has unlimited application in Electronic design and Education.



80-GRAFIX HI-RESOLUTION
Finally, the only means to protect your computer investment is to order an 80-GRAFIX board TODAY!



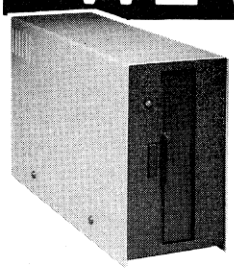
EXCITEMENT & FUN
Open up a new realm of software development with the 80-GRAFIX board.

Available exclusively through PROGRAMMA at the cost of $149.95
Please check with us for availability prior to ordering
VISA and MASTERCHARGE accepted
TRS-80 is a registered trademark of the Tandy Corp.

PROGRAMMA INTERNATIONAL, INC.
3400 Wilshire Blvd.
Los Angeles, CA 90010
(213) 384-0579 · 384-1116 · 384-1117

```
     ; B7;: GOSUB 1420;: LPRINT USING F$; T6:
                                              '** LINE 3 **
1160 GOSUB 1410: LPRINT TAB(33) "0";: LPRINT TAB(45) "0";: LPRINT
     TAB(56) "0";: LPRINT TAB(63) "     0":
                              '** LINE 4 **
1170 GOSUB 1410: GOSUB 1410: GOSUB 1420;: LPRINT USING F$; AD:
      '** LINE 5 **
1180 GOSUB 1410: GOSUB 1410: LPRINT TAB(28);: LPRINT USING F$; B9;
     : LPRINT TAB(40);: LPRINT USING F$; B8;: LPRINT TAB(51);: LPR
     INT USING F$; B7;: GOSUB 1420;: LPRINT USING F$; B6:
                                              '** LINE 6 **
1190 GOSUB 1410: LPRINT TAB(48);: LPRINT USING F$; TI:
      '** LINE 7 **
1200 GOSUB 1410: LPRINT TAB(53) "0":
      '** LINE 8 **
1210 LPRINT TAB(48);: LPRINT USING F$; TI:
      '** LINE 9 **
1220 LPRINT TAB(53) "0":
     '** LINE 10 **
1230 GOSUB 1410: GOSUB 1420;: LPRINT USING F$; TI:
     '** LINE 11 **
1240 LPRINT TAB(48);: LPRINT USING F$; TB:
     '** LINE 12 **
1250 GOSUB 1420;: LPRINT USING F$; NA:
     '** LINE 13 **
1260 GOSUB 1420;: LPRINT USING F$; AI:
     '** LINE 14 **
1270 FOR N = 1 TO 4: GOSUB 1410: NEXT N: GOSUB 1420;: LPRINT USING
      F$; NA:
                              '** LINE 15 **
1280 GOSUB 1420;: LPRINT USING F$; AV:
     '** LINE 16 **
1290 GOSUB 1410: GOSUB 1420;: LPRINT USING F$; TT:
     '** LINE 17 **
1300 LPRINT TAB(63) "     0";:
     '** LINE 18 **
1310 GOSUB 1410: GOSUB 1420;: LPRINT USING F$; TT:
     '** LINE 19 **
1320 GOSUB 1420;: LPRINT USING F$; L0:
     '** LINE 20 **
1330 LPRINT TAB(48);: LPRINT USING F$; L0:
     '** LINE 21 **
1340 LPRINT TAB(48);: LPRINT USING F$; L2:
     '** LINE 22 **
1350 LPRINT TAB(48);: LPRINT USING F$; LL:
     '** LINE 23 **
1360 GOSUB 1410: LPRINT TAB(63);: LPRINT USING F$; LL * 4:
     '** LINE 24 **
1370 GOSUB 1410: FOR N = 1 TO 2: LPRINT TAB(51) "OMIT": NEXT N:
                    '** LINES 25 AND 26 **
1380 LPRINT TAB(63) "     0":
     '** LINE 27 **
1390 GOSUB 1410: GOSUB 1420;: LPRINT USING F$; L8:
     '** LINE 28 **
1400 END
1410 LPRINT CHR$(138): RETURN
1420 LPRINT TAB(63): RETURN
1430 '****** TAX COMPUTATION SUBROUTINE ******
1440 IF S = 5 THEN S = 2
1450 TAX = 0: FOR N = 1 TO 15
1460 IF T <= BR(S,N+1) GOTO 1480
1470 TAX = (BR(S,N+1) - BR(S,N)) * R(S,N) / 100 + TAX: NEXT N
1480 TAX = (T - BR(S,N)) * R(S,N) / 100 + TAX
1490 IF TAX < 0 THEN TAX = 0
1500 RETURN
1510 '****** NOT QUALIFIED FOR INCOME AVERAGING ******
1520 CLS: PRINT: PRINT: PRINT TAB(12) "YOU DO NOT QUALIFY FOR INCO
     ME AVERAGING."
1530 PRINT: PRINT: PRINT TAB(5) "AVERAGEABLE INCOME MUST EXCEED $3
     ,000.  YOURS IS" AI "."
1540 PRINT: PRINT: PRINT: A$ = "SORRY!!!": GOSUB 90
1550 END
1560 DATA 2300,14,3400,16,4400,18,6500,19,8500,21,10800,24,12900,2
     6,15000,30
1570 DATA 18200,34,23500,39,28800,44,34100,49,41500,55,55300,63,81
     800,68
1580 DATA 108300,70,3400,14,5500,16,7600,18,11900,21,16000,24,2020
     0,28,24600,32
1590 DATA 29900,37,35200,43,45800,49,60000,54,85600,59,109400,64,1
     62400,68
1600 DATA 215400,70,300000,70,1700,14,2750,16,3800,18,5950,21,8000
     ,24,10100,28
1610 DATA 12300,32,14950,37,17600,43,22900,49,30000,54,42800,59,54
     700,64
1620 DATA 81200,68,107700,70,200000,70,2300,14,4400,16,6500,18,870
     0,22,11800,24
1630 DATA 15000,26,18200,31,23500,36,28800,42,34100,46,44700,54
1640 DATA 60600,59,81800,63,108300,68,161300,70,200000,70
```

WE WILL NOT BE UNDERSOLD



DISK DRIVES $314

40 track, 102K Bytes. Includes power supply and TRS-80* compatible silver enclosure. Ready to plug-in and run the moment you receive it. Can be intermixed with each other and Radio Shack drive on same cable. 90 day warranty. One year on power supply. Available for 220 Vac (50 Hz) operation. **External card edge included.**

FOR TRS-80*

CCI-100 TEAC	5¼", 40 Track (102K Bytes) for Model I	**$314**
CCI-100 MPI	5¼", 40 Track (102K Bytes) for Model I	$319
CCI-280	5¼", 80 Track (204K Bytes) for Model I	$429
For Zenith Z89		
CCI-189	5¼", 40 Track (102K Bytes) add-on drive	$394
Z-87	Dual 5¼" add-on drive system	$995

NEW • S-100 CCS CARDS

MAINFRAME, Z-80 CPU, CONTROLLER, RAM, and 2P + 2S CARDS $CALL

8" SHUGART SA801R DISK DRIVES $425

DISK OPERATING SYSTEMS

PATCHPAK #4 by Percom Data $ 8.95
CP/M® for Model I, Zenith **$145** • for Model II, Altos **$169**
NEWDOS Plus 40track $79 • **NEWDOS 80 $135**

DISKETTES — Box of 10 with plastic library case

5¼"	Scotch $35	Maxell $40	BASF/Verbatim $24
8"	Scotch $50	Maxell $55	BASF/Verbatim $36

CLEAR PLASTIC CASE-Holds 50 5¼"diskettes $19

COMPLETE SYSTEMS

ALTOS ACS8000 Computers $CALL
APPLE II-16K $1075 • **APPLE III-96K $3749**
Call for other Apple products
TRS-80* Model II-64K $3499 • **Model III-16K $899**
Used TRS-80* Model I Computers, tested and guaranteed $CALL
ZENITH Z89, 48K all-in-one computer $2395
ZENITH Z19 $735
TELEVIDEO 920C $748
ATARI 400 $479 **ATARI 800 $769**
APF Game Only $95 **Complete System $489**
MATTEL INTELLIVISION $229

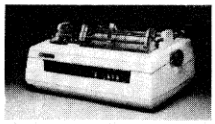
MONITORS

LEEDEX	12" B & W Video 100	$129
ZENITH	13" Color	$379
SANYO	9" B & W VM4509	$155
SANYO	12" B & W DM5012	$226
SANYO	12" Green Screen DM5112	$238
SANYO	13" Color DMC6013	$416
APF	9" B & W TVM-10	$120

TELECOMMUNICATIONS

LIVERMORE STAR MODEM 2-year guarantee $145
UNIVERSAL DATA SYSTEMS UDS-103 $179
D-CAT HARD WIRED DIRECT MODEM $189
AUTO-CAT Auto Answer, Direct Connect Modem $229

PRINTERS



NEC Spinwriter

Letter Quality High Speed Printer
Includes TRS-80* interface software, quick change print fonts, 55 cps, bidirectional, high resolution plotting, graphing, proportional spacing: R.O. $2395
R.O. with Tractor Feed **$2595** KSR with Tractor Feed **$2895**

C.ITOH Starwriter, 25 CPS, daisy wheel printer $1795
C.ITOH Starwriter II, 45 CPS, daisy wheel printer $1995
Letter quality printers. Use up to 15" paper. 1 year warranty on parts. 3 months on labor. Proportional spacing and bidirectional printing. Same as VISTA V300.

EPSON MX-80 $CALL
PAPER TIGER **IDS 445** Graphics and 2K buffer $699
IDS 460 Bidirectional, 160 cps, graphics and 2K buffer $1050
IDS 560 132 Columns, graphics $1599
ANADEX **DP-9500/01 $1345** **DP-8000 $849**
OKIDATA Microline 80 Friction and pin feed $499
Tractor Feed, friction, and pin feed $625
Microline 82 Bidirectional, friction and pin feed $719
Microline 83 Bidirectional, 120 cps, uses up to 15" paper $995
CENTRONICS
737 Friction & pin feed $780
730 Friction & pin feed $595
779 Tractor feed $969
EATON LRC 7000 + 64 columns, plain paper $269
TI-810 Includes TRS-80* software and Compressed print, vertical form control $1865

16K RAM KITS 2 for $56 $30

200 ns for TRS-80*, Apple II, (specify): **Jumpers** $2.50

ACCESSORIES

Z-80 SOFTCARD: Your key to software expansion. The plug-in Z-80 Softcard transforms your Apple into a Z-80 while keeping all the benefits of the 6502. Comes with CP/M in two disk format, MBASIC and GBASIC, full documentation and utility programs. $339.00

SCOTCH HEAD CLEANING DISKETTE: Cleans drive Read/Write head in 30 seconds; specify 5¼" or 8". $25.00

FLOPPY SAVER: Protection for center holes of 5¼" floppy disks. Installation tools and rings for 25 diskettes. $ 11.95
Re-orders of rings only $ 6.95

VIDEX BOARD 80 Column, U/L case conversion card $279.00
CRT FILM: Helps eliminate external glare, 9" $ 29.00
RF MODULATOR: Adapts video to TV $ 29.00
TRS-80 & OTHER MYSTERIES $ 18.95
NEC SPINWRITER THIMBLE $11.95 **RIBBON** $ 6.00
CCS CARDS: Parallel or serial printer interface cards $115.00
RS232: For Radio Shack Interface. $ 84.00
DISK-DRIVE EXTENDER CABLES: Fits all mini-disk drives. $ 16.95
SIX (6) PRONG ISOLATOR: ISO-2 $ 54.00
AC FILTER/6 PRONG POWER STRIP $ 39.00
DISK DRIVE CABLES: 2 drive **$29.00** 4 drive $ 35.00
DUST COVERS: TRS-80/Apple $ 7.95
PLASTIC DISKETTE HOLDER: For ring binder, holds 20 $ 8.00

COMMUNICATIONS SOFTWARE

CCI-TELNET COMMUNICATIONS PACKAGE: Communications with mainframes and micros. CP/M compatible $149.00

For fast delivery, send certified checks, money orders or call to arrange direct bank wire transfers. Personal or company checks require two to three weeks to clear. All prices are mail order only and are subject to change without notice.

DEALER (NATIONAL/INTERNATIONAL) INQUIRIES INVITED **Send for FREE Catalogue**

The CPU SHOP ✓298

TO ORDER CALL TOLL FREE 1-800-343-6522
TWX: 710-348-1796 Massachusetts Residents call 617/242-3361

5 Dexter Row, Dept. MC03M
Charlestown, Massachusetts 02129
Hours 10AM-6PM (EST) Mon.-Fri. (Sat. till 5)

Technical Information call 617/242-3361
Massachusetts Residents add 5% Sales Tax
Tandy Corporation Trademark/* Digital Research

master charge THE INTERBANK CARD VISA

✓ Reader Service—see page 290

H & E COMPUTRONICS INC.

•••EVERYTHING FOR YOUR TRS-80™•••

* TRS-80™ is a trademark of the Radio Shack Division of Tandy Corporation

Currently Available

MOD-II PROGRAMS

★ *All orders processed within 24-Hours*
★ *30-Day money back guarantee on all TRSDOS Software*
★ *Add $2.00 for shipping in UPS Areas*
★ *Add $3.00 for C.O.D. or NON-UPS Areas*
★ *Add $4.00 outside U.S.A., Canada & Mexico*
★ *We will match any bonafide advertised price in any of the Major Computer Magazines*

ALL SOFTWARE LISTED HERE WORKS WITH TRSDOS*

(1) ELECTRIC PENCIL (Michael Shrayer Software)... Complete word processor with extensive editing and printer formatting features....$325 (STANDARD TRSDOS VERSION)...$350 (DIABLO, NEC OR QUME TRSDOS VERSION).

(2) GENERAL LEDGER, ACCOUNTS RECEIVABLE, ACCOUNTS PAYABLE, INVENTORY CONTROL, INVOICING AND PAYROLL (Small Business Systems Group)...an extensive business system for the serious user...can be used one module at a time or as a co-ordinated system...$225...per module...$1299 for the complete system.

(3) GENERAL LEDGER, ACCOUNTS RECEIVABLE, ACCOUNTS PAYABLE, INVENTORY CONTROL AND PAYROLL (Compumax)...a complete user oriented business system...can be used one module at a time or as a coordinated system...$140 per module... $995 for the complete system.

(4) MOD-II UTILITY PACKAGE (Racet Computes)... adds important utilities to TRSDOS...copy files selectively...faster and more accurate file copying... repair bad directories...displays sorted directory of all files on 1 to 4 disk drives...SUPERZAP...change disk ID...and more...$150.

(5) ADVENTURE #1-#9 (Scott Adams - Adventure International)...a series of games formally only available on the large computers...your goal is to work your way through a maze of obstacles in order to recover a secret treasure or complete a mission...the package includes all 9 Adventures written by Scott Adams...$99.95.

(6) GSF (Racet Computers)...Generalized Subroutine Facility...a series of super fast machine language utilities that can be called from a BASIC program (no machine language knowledge required)...sorts 1000 items in under 5 seconds...allows PEEK and POKE statements...move data blocks...compress and un-compress data...works under TRSDOS...$50.

(7) DSM (Racet Computes)...Disk Sort Merge...sorts and merges large multiple diskette files on a 1 to 4 drive system...NOT AN IN MEMORY SORT...can actually alphabetize (or any other type of sort) 4 disk drives worth of data...sorts one complete disk of information in 10 minutes...information is provided to use DSM with the RS MAILING PROGRAM...works under TRSDOS...$150.

(8) RSM (Small Systems Software)...a machine language monitor and disassembler...can be used to see and modify memory or disk sectors...contains all the commands found on the Model-I version plus some additional commands for the MOD-II...works under TRSDOS...$39.95.

(9) BLINK BASIC LINK FACILITY (Racet Computes)... Link from one BASIC program to another saving all variables...chain programs without losing variables ...$50.

(10) BASIC CROSS REFERENCE UTILITY (Racet Computes)...lists all variables and strings used in a program (with the line numbers in which they appear) ...lists all GOTO's and GOSUB's (with the line numbers in which they appear)...searches for any specific variables or strings (with the line number in which they appear)...$50.

(11) DEVELOPMENT PACKAGE (Racet Computes)... SUPERZAP (to see, print or change any byte on a diskette)...Disassembler and MOD-II interface to the **MICROSOFT EDITOR ASSEMBLER PLUS** including uploading services and patches for Disk I/O..assemble directly into memory...save all or portions of source to disk...dynamic debug facility (ZBUG)...entended editor commands...$125.

(12) HARD/SOFT DISK SYSTEM (Racet Computes)... The software essential to interface any of the popular large hard disk drives...completely compatible with your existing software and files...allows up to 20 megabytes of storage (and larger)...directory expandable to handle thousands of files...$400.

(13) CAMEO HARD DISK DRIVE CONTROLLER... coming soon (November 1?)

(14) HARD DISK DRIVES...coming soon (Nov. 1?).

(15) H & E COMPUTRONICS, INC. SHARE-A-PROGRAM DISKETTE #1...works under TRSDOS...a collection of programs written by MOD-II owners... programs include data base management...a word processor...mail system...mortgage calculations... checkbook register...and many others...$8 (add $3 postage outside of the United States, Canada and Mexico)...FREE if you send us a diskette containing a program that can be added to the SHARE-A-PROGRAM DISKETTE.

(16) WABASH CERTIFIED DISKETTES...$39.95 (per box of 10).

(17) FLIP SORT DISKETTE STORAGE TRAY...Stores 50 diskettes...comes complete with index-dividers, tilt plates and adjustable spacing...$44.95.

(18) MASTER PAC 100...100 essential programs... BUSINESS...PERSONAL FINANCE...STATISTICS... MATH...GAMBLING...GAMES...includes 125 page manual and 2 diskettes...$99.95.

(19) BUSINESS PAC 100...100 essential business programs...INVENTORY CONTROL...PAYROLL... BOOKKEEPING SYSTEM...STOCK CALCULATIONS...CHECKBOOK MAINTENANCE...ACCOUNTS RECEIVABLE...ACCOUNTS PAYABLE... includes 125 page manual and two diskettes...$149.95.

(20) EDITOR ASSEMBLER (Galactic Software Ltd.)... the first user oriented Editor Assembler for the MODEL II and was designed to utilize all the features of the MODEL II. It includes innovative features for ease of coding and debugging and complete documentation (over 120 pages)...works under TRSDOS ...$229.00.

(21) BASIC COMPILER (Microsoft)...changes your source programs into machine language...increases program execution by 3-10 times...$395.

(22) MAIL/FILE SYSTEM from Galactic Software Ltd. stores 2,500 names per disk. No sorting time is required since the file is automatically sorted by first and last name plus Zip Code on input. Retrieve by any combination of 19 user codes. Supports an 11 digit alphanumerica Zip. Supports a message line. Comes complete with user-oriented documentation (100-page manual). Allows for company name and individual of a company and complete phone number (and extension)...works under TRSDOS...$199.00

(23) INCOME TAX PAC...Professional income tax package...most forms and schedules...output to video or line printer...automatic memory storage of all information...data can be loaded from diskette, changed and edited...built in error checking...$199.95.

(24) COMPUTER GAMES (SBSG)...Mean Checker Machine, Star-Trek III, Concentration, Treasure Hunt, Banco, Dog Star Adventure...$74.95.

ALL PROGRAMS LISTED HERE REQUIRE CP/M*

(1) CP/M (Lifeboat Associates).....an alternative operating system for the **MOD-II** that allows **MOD-II** owners to use any of the hundreds of programs available under **CP/M**...$170.

(2) CP/M HANDBOOK...(Sybex)...a step-by-step guide to **CP/M**...takes the reader through each of the **CP/M** commands...numberous sample programs... practical hints...reference tables...$13.95.

(3) GENERAL LEDGER, ACCOUNTS RECEIVABLE, ACCOUNTS PAYABLE, INVENTORY CONTROL, AND PAYROLL (Peachtree Software)...requires CP/M and MICROSOFT BASIC...professional business systems...turn key operation...can be used as single modules or as a coordinated system...$500 per module...$2500 for the complete system.

(4) WORD-STAR...The ultimate word processor...a menu driven word processing system that can be used with any printer. All standard word processing commands are included...plus many unique commands only found on WORD STAR...requires CP/M ...$495.

(5) MAIL LIST MERGE...An add on package that allows the user to send form letters (created on WORD-STAR) to any compiled mailing list (using any CP/M based MAIL program such as the PEACHTREE MAIL PROGRAM)...requires CP/M, WORD STAR and andy CP/M based mail program...$150.

(6) SELECTOR III (Micro-Ap)...complete data management system...user defined fields and codes... manages any list defined by the user...includes additional modules for simplified inventory control, accounts receivable and accounts payable...requires CBASIC-2...$295.

(7) SELECTOR IV (Micro-Ap)...the ultimate data management system...all features use the **SELECTOR III** plus...data file format conversions...full page report formatter...computations...global search and replace ...hard disk compatible...data/text merging...$550.

(8) GLECTOR (Micro-Ap)...add on package to the **SELECTOR**...general ledger that allows the user to define a customized chart of accounts...$350.

(9) CBASIC-2...a non-interactive BASIC used for many programs that run under **CP/M**...allows user to make more efficient use of disk files...eliminates the use of most line number references...require on such programs as the **SELECTOR**...$120.

(10) MICROSOFT BASIC...an enhanced version of the **MICROSOFT BASIC** found on **TRSDOS**...adds commands such as chaining (allows the user to LOAD and RUN a new program without losing the variables currently in memory)...long variable length file records, WHILE/WEND and others...can be used with the BASIC COMPILER to speed up programs (3-10 times faster execution)...$325.

(11) MASTER TAX (CPAids)...professional tax preparation program...prepares schedules, A, B, C, D, E, F, G, R/RP, SE, TC, ES and forms 2106, 2119, 2210, 3468, 3903, 2441, 4625, 4726, 4797, 4972, 5695 and 6521. Printing can be on readily available pre-printed continuous forms, on overlays, or on computer generated IRS approved forms. Maintains clint history files...interactive with CP/AIds General Ledger...$995.

(12) GENERAL LEDGER II (CPAids)...designed for CPA's...stores complete 12 month detailed history of transactions...generates financial statements, depreciation, loan amortizations, journals, trial balances, statements of changes in financial position, and compilation letters...includes payroll system with automating posting to general ledgers...prints payroll register, W2's and payroll checks...$450.

(13) ELECTRIC PENCIL (Michael Shrayer Software) ...Complete word processor with extensive editing and printer formatting features...$275 (Standard printer version)...$300 (DIABLO, NEC or QUME version).

(14) BASIC COMPILER (Microsoft)...changes your source programs into machine language...increases program execution by 3-10 times...$395.

* (CP/M IS A REGISTERED TRADEMARK OF DIGITAL RESEARCH)

H & E COMPUTRONICS INC.
MATHEMATICAL APPLICATIONS SERVICE

✓9

50 N. PASCACK ROAD
SPRING VALLEY, NEW YORK 10977

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

master charge

24 HOUR ORDER LINE
(914) 425-1535

VISA

NEW!!!
MOD-II NEWSLETTER
$12/year (or 12 issues)

✓ Reader Service—see page 290

H&E COMPUTRONICS INC.

...EVERYTHING FOR YOUR TRS-80™...

TRS-80 is a trademark of the Radio Shack Division of Tandy Corporation

BUSINESS PAC 100

100 Ready-To-Run Business Programs

★ All orders processed within 24-Hours
★ 30-Day money back guarantee on all Software (less a $3 penalty for handling)

(ON CASSETTE OR DISKETTE).....Includes 110 Page Users Manual.....5 Cassettes (Or Diskettes)
Inventory Control.....Payroll.....Bookkeeping System.....Stock Calculations.....
Checkbook Maintenance.....Accounts Receivable.....Accounts Payable.....

BUSINESS 100 PROGRAM LIST

NAME	DESCRIPTION
1 RULE78	Interest Apportionment by Rule of the 78's
2 ANNU1	Annuity computation program
3 DATE	Time between dates
4 DAYYEAR	Day of year a particular date falls on
5 LEASEINT	Interest rate on lease
6 BREAKEVN	Breakeven analysis
7 DEPRSL	Straightline depreciation
8 DEPRSY	Sum of the digits depreciation
9 DEPRDB	Declining balance depreciation
10 DEPRDDB	Double declining balance depreciation
11 TAXDEP	Cash flow vs. depreciation tables
12 CHECK2	Prints NEBS checks along with daily register
13 CHECKBK1	Checkbook maintenance program
14 MORTGAGE/A	Mortgage amortization table
15 MULTMON	Computes time needed for money to double, triple, etc.
16 SALVAGE	Determines salvage value of an investment
17 RRVARIN	Rate of return on investment with variable inflows
18 RRCONST	Rate of return on investment with constant inflows
19 EFFECT	Effective interest rate of a loan
20 FVAL	Future value of an investment (compound interest)
21 PVAL	Present value of a future amount
22 LOANPAY	Amount of payment on a loan
23 REGWITH	Equal withdrawals from investment to leave 0 over
24 SIMPDISK	Simple discount analysis
25 DATEVAL	Equivalent & nonequivalent dated values for oblig.
26 ANNUDEF	Present value of deferred annuities
27 MARKUP	% Markup analysis for items
28 SINKFUND	Sinking fund amortization program
29 BONDVAL	Value of a bond
30 DEPLETE	Depletion analysis
31 BLACKSH	Black Scholes options analysis
32 STOCVAL1	Expected return on stock via discounts dividends
33 WARVAL	Value of a warrant
34 BONDVAL2	Value of a bond
35 EPSEST	Estimate of future earnings per share for company
36 BETAALPH	Computes alpha and beta variables for stock
37 SHARPE1	Portfolio selection model-i.e. what stocks to hold
38 OPTWRITE	Option writing computations
39 RTVAL	Value of a right
40 EXPVAL	Expected value analysis
41 BAYES	Bayesian decisions
42 VALPRINF	Value of perfect information
43 VALADINF	Value of additional information
44 UTILITY	Derives utility function
45 SIMPLEX	Linear programming solution by simplex method
46 TRANS	Transportation method for linear programming
47 EOQ	Economic order quantity inventory model
48 QUEUE1	Single server queueing (waiting line) model
49 CVP	Cost-volume-profit analysis
50 CONDPROF	Conditional profit tables
51 OPTLOSS	Opportunity loss tables
52 FQUOQ	Fixed quantity economic order quantity model
53 FQEOWSH	As above but with shortages permitted
54 FQEOQPB	As above but with quantity price breaks
55 QUEUECB	Cost-benefit waiting line analysis
56 NCFANAL	Net cash-flow analysis for simple investment
57 PROFIND	Profitability index of a project
58 CAP1	Cap. Asset Pr. Model analysis of project
59 WACC	Weighted average cost of capital
60 COMPBAL	True rate on loan with compensating bal. required
61 DISCBAL	True rate on discounted loan
62 MERGANAL	Merger analysis computations
63 FINRAT	Financial ratios for a firm
64 NPV	Net present value of project
65 PRINDLAS	Laspeyres price index
66 PRINDPA	Paasche price index
67 SEASIND	Constructs seasonal quantity indices for company
68 TIMETR	Time series analysis linear trend
69 TIMEMOV	Time series analysis moving average trend
70 FUPRINF	Future price estimation with inflation
71 MAILPAC	Mailing list system
72 LETWRT	Letter writing system-links with MAILPAC
73 SORT3	Sorts list of names
74 LABEL1	Shipping label maker
75 LABEL2	Name label maker
76 BUSBUD	DOME business bookkeeping system
77 TIMECLCK	Computes weeks total hours from timeclock info.
78 ACCTPAY	In memory accounts payable system-storage permitted
79 INVOICE	Generate invoice on screen and print on printer
80 INVENT2	In memory inventory control system
81 TELDIR	Computerized telephone directory
82 TIMUSAN	Time use analysis
83 ASSIGN	Use of assignment algorithm for optimal job assign.
84 ACCTREC	In memory accounts receivable system-storage ok
85 TERMSPAY	Compares 3 methods of repayment of loans
86 PAYNET	Computes gross pay required for given net
87 SELLPR	Computes selling price for given after tax amount
88 ARBCOMP	Arbitrage computations
89 DEPRSF	Sinking fund depreciation
90 UPSZONE	Finds UPS zones from zip code
91 ENVELOPE	Types envelope including return address
92 AUTOEXP	Automobile expense analysis
93 INSFILE	Insurance policy file
94 PAYROLL2	In memory payroll system
95 DILANAL	Dilution analysis
96 LOANAFFD	Loan amount a borrower can afford
97 RENTPRCH	Purchase price for rental property
98 SALELEAS	Sale-leaseback analysis
99 RRCONVBD	Investor's rate of return on convertable bond
100 PORTVAL9	Stock market portfolio storage-valuation program

☐ **CASSETTE VERSION $ 99.95**
☐ **DISKETTE VERSION $ 99.95**
☐ **MODEL II VERSION $149.95**

ADD $2.00 FOR SHIPPING IN UPS AREAS
ADD $3.00 FOR C.O.D. OR NON-UPS AREAS
ADD $4.00 OUTSIDE U.S.A, CANADA & MEXICO

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

H&E COMPUTRONICS INC.
MATHEMATICAL APPLICATIONS SERVICE™

✓9

50 N. PASCACK ROAD
SPRING VALLEY, NEW YORK 10977

VISA

master charge

24 HOUR ORDER LINE
(914) 425-1535

✓ Reader Service—see page 290

M&E COMPUTRONICS INC.

•••EVERYTHING FOR YOUR TRS-80™•••

TRS-80 is a trademark of the Radio Shack Division of Tandy Corporation

PAYROLL

Payroll invoices many complex calculations and the production of reports and documents, many of which are required by government agencies. It is an ideal candidate for the computer. With this Payroll system in-house, you can promptly and accurately pay your employees and generate accruate documents/reports to management, employees, and appropriate government agencies concerning earnings, taxes, and other deductions. The package has been converted to the TRS-80™ and is now a well documented, op-line, interactive, micro-computer system with the capabilities of (or exceeding) many larger systems.

CAPABILITIES:

- ★ performs all necessary payroll tasks including:
  - file maintenance, pay data entry and verification
  - computation of pay and deduction amounts
  - printing of reports and checks
- ★ can handle salaried and hourly employees
- ★ employees can receive:
  - hourly or salary wage
  - vacation pay
  - holiday pay
  - piecework pay
  - overtime pay
- ★ employees can be paid using any combination of pay types (except, hourly cannot receive salary and salary cannot receive hourly)
- ★ special non-taxable or taxable lump sums can be paid regularly or one time (bonus, reimbursements, etc)
- ★ health and welfare deductions can be automatically calculated for each employee
- ★ earnings-to-date are accumulated and added to permanent records; taxes are computed and deducted: US income tax, Social Security tax, state income tax, other deductions (regular or one time)
- ★ paychecks are printed; computer checks with your company letterhead can be purchased from SBSG
- ★ calculations are accumulated for; employee pay history, 941A report, W-2 report, insurance report, absentee report
- ★ fully linked to General Ledger. Each employee's payroll information can be distributed to as many as (12) twelve different GL accounts; system automatically posts to cash account

INVENTORY CONTROL/INVOICING

- ★ **ISAM** (Indexed Sequential Access Method) eliminates the necessity for time consuming sort.
- ★ Pre-Allocated Files for IMMEDIATE update and inquiry capabilities.
- ★ Fast Disk storage and retrieval.
- ★ Inventory Master Record includes...class...SKU...Division...Retail...Cost...Beginning Balance...Period Sale Units...Period Receipts...On Order...On Hand...Minimum Reorder Point...Recommended Reorder Amount...Vendor Number...Period Sale Dollars...YTD Sale Units...YTD Sale Dollars.
- ★ Calculated and Displayed Formulas include...Gross Margin ($)...Gross Margin (%)...Gross Margin ROI (%)...Average Inventory Retail ($)...Average Inventory Cost ($)...Turn-Over (%).
- ★ Reports Generated include...Master File Listing...Class Description Listing...Transaction Audit Trail...Minimum Reorder Point by Vendor...Retail Price List...Retail & Cost Price List...Period Sales Report ...Year to Date Sales Report...Stock Status (Screen or printer output) ...Commission Report (for salesmen and buyers).
- ★ Transaction Types include...Sales, Vendor Receipts...Vendor Orders...Customer Returns...Vendor Returns...Transfer Stock.

GENERAL LEDGER

The General Ledger accounting system consolidates financial data from other accounting subsystems (A/R, A/P, Payroll, direct posting) in an accurate and timely manner. Major reports include the Income Statement and Balance Sheet and a "special" report designed by management. The beauty of this General Ledger system is that it is completely user formatted. You "customize" the account numbers, descriptions, and report formats to suit particular business requirements. These programs were developed 5 years ago for the Wang micro-computer and have been tested in many environments since then. The package has been converted to the TRS-80™ and is now a well documented, on-line, interactive micro-computer system with the capabilities of (or exceeding) many larger systems.

CAPABILITIES:

- ★ more than 200 chart of accounts can be handled
- ★ account number structure is user defined and controlled
- ★ more than 1,750 transactions may be entered via:
  - direct posting; done by hand; validated against the account file before acceptance
  - external posting; generated by A/R, A/P, Payroll or any other user source
- ★ data is maintained and reported by:
  - month
  - quarter
  - year
  - previous three quarters
- ★ reports (samples on back) include:
  - trial balances
  - income statement
  - balance sheet
  - special accounts reports and more.....
- ★ user formats reports with the following designated as you wish:
  - titles
  - headings
  - account numbers
  - descriptions
  - subtotals
  - totals
  - skip lines
  - skip pages
- ★ up to eight levels of totals - fully user designated
- ★ menu driven; easy to use; full screen prompting and cursor control

M&E COMPUTRONICS INC. ✓9

MATHEMATICAL APPLICATIONS SERVICE™

50 N. PASCACK ROAD
SPRING VALLEY, NEW YORK 10977

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

master charge

VISA

24 HOUR ORDER LINE
(914) 425-1535

PRICING	MOD-1 VERSION	MOD-II VERSION	MOD-III VERSION
ACCOUNTS RECEIVABLE	$125	$225	$199.95
ACCOUNTS PAYABLE	$125	$225	$199.95
GENERAL LEDGER	$125	$225	$199.95
PAYROLL	$125	$225	$199.95
INVENTORY	$175	$275	$199.95
INVOICING	$150	$250	$199.95
COORDINATED INVENTORY/INVOICING ACCOUNTS RECEIVABLE	$449	$749	$599.95
COORDINATED AR-AP-GL	$375	$675	$599.95
COORDINATED AR-AP-GL with PAYROLL	$495	$899	$799.95
EXTENDED COORDINATED AR-AP-GL INVOICING/INVENTORY without PAYROLL	$799	$1299	$1199.95

H&E COMPUTRONICS INC.

•••EVERYTHING FOR YOUR TRS-80™•••

TRS-80 is a trademark of the Radio Shack Division of Tandy Corporation

1980 INCOME TAX PAC

Completely Revised ★ Latest Tax Tables ★ Fully Tested ★ Complete Manual and Documentation

★ ★ The New Version Of The Income Tax Pacs Are Full Of Error Catching Codes ★ ★

★ ★ Making It Impossible To Make An Error ★ ★

— Follow The Simple Step By Step Procedure That Makes Tax Preparation Simple —

★ INCOME TAX PAC A

FOR LEVEL II 16K

- DOES FORM 1040 and 1040A
- SCHEDULE A ITEMIZED DEDUCTIONS
- SCHEDULE B INTEREST and DIVIDENDS
- OUTPUT TO VIDEO DISPLAY
- SCHEDULE C TAX COMPUTATION

★ INCOME TAX PAC B

FOR LEVEL II with or without Printer, Cassette or Disk. Has all features of Income Tax A **PLUS,**

- WORKS WITH LINE PRINTER
- FORMATS FORM 1040 and 1040A FOR TRACTOR FEED FORMS
- SCHEDULE C INCOME FROM A PERSONALLY OWNED BUSINESS
- FORM 2106 EMPLOYEE BUSINESS EXPENSE

★ ★ PROFESSIONAL ★ ★ INCOME TAX PAC C

- FORM 1040 (LONG FORM)
- FORM 1040A (SHORT FORM)
- FORM 2106 EMPLOYEE BUSINESS EXPENSE
- FORM 2440 DISABILITY INCOME EXCLUSION
- FORM 2441 CREDIT FOR CHILD AND DEPENDENT CARE EXPENSES
- FORMS 3903 MOVING EXPENSE ADJUSTMENT
- FORM 4797 SUPPLEMENTAL SCHEDULE OF GAINS AND LOSSES
- SCHEDULE A ITEMIZED DEDUCTIONS
- SCHEDULE B INTEREST AND DIVIDENDS
- SCHEDULE C PROFIT (OR LOSS) FROM BUSINESS OR PROFESSION
- SCHEDULE D CAPITAL GAINS AND LOSSES
- SCHEDULE E SUPPLEMENTAL INCOME SCHEDULE
- SCHEDULE G INCOME AVERAGING
- SCHEDULES R & RP-CREDIT FOR THE ELDERLY
- SCHEDULE SE-COMPUTATION OF SOCIAL SECURITY SELF-EMPLOYMENT TAX
- SCHEDULE TC TAX COMPUTATION
- OUTPUT TO VIDEO OR LINE PRINTER
- FORMATS FOR TRACTOR FEED OR INDIVIDUAL FORM FEED PRINTERS
- AUTOMATIC MEMORY STORAGE FOR INCOME TAX PREPARERS
- INSTANT LINE CHANGE
- BUILT IN ERROR CHECKING

FOR MODEL I (32K) or MODEL II (64K) WITH 1 OR MORE DISK DRIVES

ALL SPECIFICATIONS SUBJECT TO CHANGE

★ A COMPLETE LINE OF **NELCO TAX FORMS** ARE AVAILABLE

- INDIVIDUAL FEDERAL and STATE FORMS
- 2 OR MORE PART FORMS
- TRACTOR FEED FORMS
- PLASTIC OVERLAYS

H&E COMPUTRONICS INC.
MATHEMATICAL APPLICATIONS SERVICE™

50 N. PASCACK ROAD
SPRING VALLEY, NEW YORK 10977

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

PLEASE SEND ME:

☐ INCOME TAX PAC A ($19.95)
☐ INCOME TAX PAC B ($49.95)
☐ PROFESSIONAL INCOME TAX PAC C ($99.95)
☐ MODEL II PROFESSIONAL INCOME TAX PAC C ($199.95)

★ All orders processed within 24-Hours
★ 30-Day money back guarantee on all Software
★ Add $2.00 for shipping in UPS Areas
★ Add $3.00 for C.O.D. or NON-UPS Areas
★ Add $4.00 outside U.S.A., Canada & Mexico

CREDIT CARD NUMBER _____ EXP. DATE _____

SIGNATURE _____

NAME _____

STREET _____

CITY _____ STATE _____ ZIP _____

VISA

master charge

24 HOUR ORDER LINE
(914) 425-1535

Be there on schedule with this secretarial software.

By Appointment Only

David D. Busch
515 E. Highland Ave.
Ravenna, OH 44266

Keeping track of appointments has ranged from tying a string round one's finger to making notes in a bulky, personal appointment calendar.

The TRS-80 not only can remember an entire year's dates, but can retrieve and display appointments in a few seconds. Thus, the Appointment Secretary is a disk-based date-management program that can put your personal calendar onto a 5 1/4-inch diskette. It will run on any TRS-80 Model I with 32K and at least one disk drive.

Appointment Secretary keeps track of one year's appointments. I've left leap years and a fourth retrieval option up to you.

The data is stored in a sequential disk file of 365 elements which I called APPTS. The first major routine, in lines 160-200 is to load the existing appointment file from disk, except on the first run, when an error trap in lines 70-140 creates the file. This step avoids a FILE NOT FOUND error, and the lines should be deleted after the first run of the program.

The data in the APPTS file is loaded into string array APP$(N). When the file is first created, the information in each of the 365 positions is null (line 120), and these "non-data" are then loaded into APP$(N) and filled in as you enter appointment data.

Lines 220-430 set up the menu for the program along with a fancy border, which can be deleted. Three options are offered: entering appointments, accessing, or saving data.

Options

All user input of dates, as when entering new appointments, is done in MM/DD form, i.e., 06/05 for June 5. This information is converted to consecutive days of the year in a subroutine, lines 1510-1590. The date, stored in the string variable DA$, is parsed into its MM and DD components, then converted to numerical value. It is produced by adding the DD value to the number of days in the year that elapsed prior to the month (M). Number of days is obtained from an array, DA(N).

The process is reversed in lines 1610-1630 to change the consecutive day of a retrieved appointment into MM/YY form.

Storage of the appointment data is simple. Entry takes place in lines 520-620. Once the program has determined the consecutive day number of the entry (DA), it inserts the appointment information into an element of the array APP$(DA). The new appointment is always concatenated onto the end of the existing contents of the array. Additional entries can be made, up to the 255-character limit for each position in the array.

One trick I learned was to add a CHR$(26) and a CHR$(29) onto the end of each entry. The first generates a line feed when the string is printed, and the second returns the cursor to the beginning of the line. The result: each entry is listed on its own line on the CRT.

Appointments can be accessed in three ways. The program searches for a specific day and input is converted to consecutive day (DA) and that element of the array printed. Or, a range of days can be displayed. The MM/DD conversion is done twice, and the beginning and ending days of the range are used as limits for a FOR-NEXT loop. The loop prints out all the non-null entries in between the limits. The "all appointments for now until..." option works much the same, using the number of days requested as the upper limit of the loop.

An interesting possibility is that of searching through data for keywords. For example, you may remember that you have an appointment with Bob Fisher, but can't recall when. Try a subroutine using Disk BASIC's INSTR feature. To do this input the keyword string. Then send the program through a search of the 365 elements of APP$. Here is an example:

```
2000 FOR N = 1 TO 365
2010 IF INSTR(APP$(N),KEYWORD$))<>0
     GOSUB 2030
2020 NEXT N
2030 Convert N to MM/YY here, and print
     APP$(N)...
2040 .............
2050 RETURN
```

Because INSTR returns the position where the located string begins, you can print out only that portion of the string. Use MID$, beginning with the value returned by INSTR and all of the characters up to the next line feed.

To keep track of appointments for more than a year, drop the APPTS file name I used and apply your own, with a name for each year: A1980, A1981, etc. By substituting a variable for the file name, you can input the year desired each time the program is run. There are even ways to allow the program to overlap two years, so that retrieval of appointments from December 21 to January 21, for example, can be accomplished in a single program run.

This program is debugged until 1984. Then, the leap year will throw in a monkey wrench. You might add a few lines that will test the year to see if it is divisible by 400, and increment DA(3) by one.■

Program Listing 1

```
20
     CLEAR 5000
30 DIM APP$(365),DA(12)
40 FOR M=1 TO 12:READ DA(M):NEXT M
50 DATA 0,31,59,90,120,151,181,212,243,273,304,334
60 CLS:PRINT:PRINT
70 '****** ERROR TRAP -- REMOVE LINES 70-140 AFTER FIRS
    T RUN ******

80 INPUT"HAS THIS PROGRAM BEEN RUN B
    EFORE";A$
90 IF LEFT$(A$,1)="y" GOTO 160
100 OPEN "O",1,"APPTS"
110 FOR N=1 TO 365
120 PRINT #1,"";",";
130 NEXT N
140 CLOSE 1
150 ' ****** LOAD APPOINTMENT FILE FROM DISK ******

160 OPEN "I",1,"APPTS"
170 FOR N=1 TO 365
180 INPUT #1, APP$(N)
190 NEXT N
200 CLOSE 1
210 '****** PRINT BORDER AND MENU ******

220 CLS
230 FOR N=1 TO 63
240 PRINT @ N, CHR$(159)
250 NEXT N
260 PRINT
270 PRINT "     DO YOU WANT TO :"
280 PRINT
290 PRINT
300 PRINT "            1.) ENTER APPOINTMENTS"
310 PRINT "            2.) ACCESS APPOINTMENTS"
320 PRINT "            3.) SAVE DATA"
330 PRINT
340 PRINT "        ENTER CHOICE :"
350 PRINT
360 PRINT
370 FOR N=833 TO 895
380 PRINT @ N, CHR$(190)
390 NEXT N
400 FOR X=0 TO 41
410 SET(1, X)
420 SET(127, X)
430 NEXT X
440 ' ****** INPUT CHOICE FROM MENU ******

450 A$=INKE
    Y$
460 IF A$="" GOTO 450
470 A=VAL(A$)
480 IF A<1 GOTO 450
490 IF A>3 GOTO 450
500 ON A GOTO 520, 640, 1440
510 ' ****** ENTER APPOINTMENTS ******

520 CLS
530 PRINT
540 PRINT
550 GOSUB 1520
560 CLS
570 PRINT
580 PRINT
590 LINE INPUT "ENTER APPOINTMENT :";A$
600 ' --- ADD NEW APPOINTMENT TO EXISTING, PLUS LINEFEE
    D
        AND CURSOR RETURN ---

610 APP$(DA)=A
    PP$(DA)+"  " +A$+CHR$(26)+CHR$(29)
620 GOTO 220
630 '****** ACCESS APPOINTMENTS ******

640 CLS
650 PRINT
660 PRINT
670 PRINT "DO YOU WANT APPOINTMENTS FOR:"
680 PRINT "          1.) A SPECIFIC DAY"
690 PRINT "          2.) A RANGE OF DAYS"
700 PRINT "          3.) ALL APPOINTMENTS FROM NOW UNTI
    L..?"
710 PRINT
720 PRINT "        ENTER CHOICE :"
730 A$=INKEY$
740 IF A$="" GOTO 730
750 A=VAL(A$)
760 IF A<1 GOTO 730
770 IF A>3 GOTO 730
780 ON A GOTO 800, 910, 1180
790 '--- APPOINTMENTS FOR SPECIFIC DAY ---

800 CLS
810 PRINT
820 PRINT
830 GOSUB 1520
840 PRINT APP$(DA)
850 PRINT
860 PRINT
870 PRINT"PRESS ANY KEY TO CONTINUE"
880 IF INKEY$ ="" GOTO 880
900 '--- APPOINTMENTS FOR A RANGE OF DAYS ---

910 CLS
920 PRINT
930 PRINT"ENTER LOWER DATE BOUNDARY"
940 GOSUB 1520
950 D1=DA
960 PRINT "ENTER UPPER DATE BOUNDARY"
970 GOSUB 1520
980 IF D2<D1 THEN PRINT "SORRY. THAT'S NEXT YEAR.":GOTO
    920
990 D2=DA
1000 CLS
1010 PRINT
1020 PRINT
1030 FOR N=D1 TO D2
1040 IF APP$(N)="" GOTO 1120
1050 DA=N
1060 GOSUB 1610
1070 M$=STR$(M)
1080 D=DA-F1
1090 D$=STR$(D)
1100 DA$=M$+"/" +D$
1110 PRINT DA$;" :";APP$(N)
1120 NEXT N
1130 PRINT
1140 PRINT "PRESS ANY KEY TO CONTINUE"
1150 IF INKEY$ ="" GOTO 1150
1160 GOTO 220
1170 '---APPOINTMENTS FROM PRESENT DAY UNTIL... ---
1180 CLS
1190 PRINT
1200 PRINT
1210 PRINT "ENTER TODAY'S DATE :"
1220 GOSUB 1520
1230 PRINT "HOW MANY DAYS' APPOINTMENTS DO YOU WISH TO
    SEE";
1240 INPUT D2
1250 D2=DA+D2
1260 IF D2>365 THEN D2=365
1270 CLS
1280 PRINT
1290 FOR N=DA TO D2
1300 IF APP$(N)="" GOTO 1380
1310 DA=N
1320 GOSUB 1610
1330 M$=STR$(M)
1340 D=DA-F1
1350 D$=STR$(D)
1360 DA$=M$+"/" +D$
1370 PRINT DA$;"  : ";APP$(N)
1380 NEXT N
1390 PRINT
1400 PRINT"PRESS ANY KEY TO CONTINUE"
1410 IF INKEY$ ="" GOTO 1410
1420 GOTO 220
1430 '****** SAVE APPOINTMENTS TO DISK ******

1440 OPE
    N "O",2,"APPTS"
1450 FOR N=1 TO 365
1460 PRINT #2, CHR$(34); APP$(N); CHR$(34);",";
1470 NEXT N
1480 CLOSE 2
1490 GOTO 220
1500 ' ****** CONVERT MM/DD TO CONSECUTIVE DAY *******


1510 CLS
1520 DA=0
1530 INPUT "ENTER DATE  (MM/DD)";DA$
1540 M$=LEFT$(DA$, 2)
1550 D$=RIGHT$(DA$, 2)
1560 M=VAL(M$)
1570 D=VAL(D$)
1580 DA=DA(M)+D
1590 RETURN
1600 ' ****** CONVERT CONSECUTIVE DAY BACK TO MM/DD ***
    ***

1610 FOR G=12 TO 1 STEP -1
1620 IF DA>DA(G) THEN M=G:F1=DA(G):RETURN
1630 NEXT G
```

"Appointment Secretary keeps track of one year's appointments. I've left leap years and a fourth retrieval option up to you."

Some thoughts on writing poetry in BASIC and playing tennis without nets.

Computer Cantos

Ward D. Griffiths III
4131 Calimesa St.
Las Vegas, NV 89110

Almost every literate person dreams of writing poetry at one time or another. My talent for poetry is non-existent, but I have a fair amount of skill at BASIC programming. I decided to see what I could do with the ability I have to make up for the one I don't.

My first step was to investigate existent poetry programs. Those I found lacked either flexibility or produced the wrong style: haiku is interesting and fun to read, but I prefer stanzas and lines. (I use my program as a random idea generator because it often puts words into original and thought-provoking combinations.)

Armed with a 16K TRS-80 with Level II BASIC, I went to work. With an initial version of line patterns, no verb tenses, no plural nouns, and a serious problem recognizing transitive and intransitive verbs, I hacked, rewrote, renumbered and modified the program into the memory-devouring monster it is now.

SPOONS

Smoldering paperboys engender cheap spectre.
What humans behind spoons berate the thumb?
Gaps improve red vision.
Do analagous fedoras breathe?

Pornographic virtues were not breathing,
Romans hallucinate amateurishly ahead of them.
Souls could not be soft smoldering gloves,
Whose perceptive bones push?

The spoon eavesdropped.
Horrendous schoolmarms may not be persisting.
Whose galaxy eavesdropped?
Precise hells would not be foods,

Her writer cried idiotically.
Pushing fairs fun lamentable speck,
Their arduous acknowledging of spoon and branch.
Whose mandibles alongside spoons punt the science?
Spoons alongside the mandibles are not splendorous.

TEARS

Impotent spectres will have been impotent, but
How do rankled under hands rankle?
He stroked a foolish stem,
Spines fondle vital spine.

Walking drawing and raindrop.
Whose chasm upchucked?
Units of tear split.
Improving thunders scratch strange prophecy.

This wonderful relieving of tear and studio.
Plagiaristic doors may be plagiaristic, but
mandibles of tear stopped.
Raking saucer and document,

Nosy ripples may be boxes.
Do idiotic vices eavesdrop?
Running space and bracelet,
Courteous sheaths may not be running.

Our obedient misery embraced a xylophone.
Tears from the miseries should not have been blue.

How the Program Works

All data is read into string arrays for the different parts of speech. A noun, a verb and an adjective are chosen for use in the title and saved for occasional use in the text of the poem. Stanza length is set between three and six lines.

A loop is then entered to produce a poem length of 11 to 19 lines plus a final line at the loop's exit. For each line, words are selected from all parts of speech and a punctuation mark is assigned. Not all the words are used in every line pattern. Unused words are ignored.

A line pattern is chosen and executed. If a pattern requires a word be changed, carrier variables are assigned and subroutines executed. At the loop's exit, the final line uses words from the previous line and the title in an attempt to tie the poem together.

After the last line, there is no linefeed, and everything stops until any key is struck. The screen is then cleared and a new poem is written.

Limitations

The program is a tight fit in 16K of memory. While running,

Continued to page 158

Program Listing

```
100 'POETRY PROGRAM VERSION 3.0
200 'COPYRIGHT 1980 BY WARD D. GRIFFITHS III
300 'PERMISSION TO USE, NOT TO SELL
1000 'INITIALIZATION
1050 CLEAR 300:DEFINT A-Z:RANDOM
1100 NOUN=300:HVERB=90:OVERB=40:IVERB=30:TVERB=HVERB+OV
     ERB:IVERB=IVERB+OVERB:ADJECT=210:PREPO=30:ARTIC=10
     :QUEST=10:BVERB=30
1150 DIM N$(NOUN),TV$(TVERB),IV$(IVERB),AJ$(ADJECT),PP$
     (PREPO),AR$(ARTIC),IN$(QUEST),SV$(BVERB)
1200 FOR I=1 TO NOUN:READ N$(I):NEXT I:FOR I=1 TO TVERB
     :READ TV$(I):NEXT I:RESTORE:FOR I=1 TO NOUN+HVERB:
     READ W$:NEXT I:FOR I=1 TO IVERB:READ IV$(I):NEXT I
     :FOR I=1 TO ADJECT:READ AJ$(I):NEXT I:FOR I=1 TO P
     REPO:READ PP$(I):NEXT I
1250 FOR I=1 TO ARTIC:READ AR$(I):NEXT I:FOR I=1 TO QUE
     ST:READ IN$(I):NEXT I:FOR I=1 TO BVERB:READ SV$(I)
     :NEXT I
1300 'PROGRAM START
1350 CLS:PRINT
1400 'TITLE SELECTION
1450 'ASSIGN AND DECODE WORDS
1500 W$=N$(RND(NOUN)):GOSUB 3950:NT$=W$:NT=W:'NOUN
1550 W$=TV$(RND(TVERB)):GOSUB 3950:V1=W:GOSUB 3950:V2=W
     :VT$=W$:'VERB
1600 W$=AJ$(RND(ADJECT)):GOSUB 3950:TA$=W$:TA=W:'ADJECT
     IVE
1650 PP$=PP$(RND(PREPO)):'PREPOSITION
1700 ON RND(5) GOTO 1750,1800,1850,1900,1950:'CHOOSE TI
     TLE PATTERN
1750 S$=TA$:GOSUB 4050:PRINT ,A$;TA$;" ";NT$:GOTO 2000:
     '#1
1800 S$=NT$:GOSUB 4050:PRINT ,"TO ";VT$;" ";A$;NT$:GOTO
      2000:'#2
1850 PRINT ,"THE ";NT$:GOTO 2000:'#3
1900 N$=NT$:N=NT:GOSUB 4500:PRINT ,N$:GOTO 2000:'#4
1950 S$=NT$:GOSUB 4050:AJ$=TA$:AJ=TA:GOSUB 4200:PRINT ,
     PP$;" ";A$;NT$;" ";AD$:GOTO 2000:'#5
2000 'TEXT OF POEM
2050 S=RND(4)+2:'STANZA LENGTH(3-6 LINE)
2100 FOR L=1 TO RND(9)+10:'POEM LENGTH(11-19 LINE PLUS
     END)
2150 IF (L-1)/S=INT((L-1)/S) THEN PRINT:'SPLIT STANZA
2200 'ASSIGN AND DECODE STRINGS FOR POEM LINES
2250 W$=N$(RND(NOUN)):GOSUB 3950:N1=W:N1$=W$:'FIRST NOU
     N
2300 W$=N$(RND(NOUN)):GOSUB 3950:N2=W:N2$=W$:'SECOND NO
     UN
2350 W$=TV$(RND(TVERB)):GOSUB 3950:T1=W:GOSUB 3950:T2=W
     :TV$=W$:'TRANSITIVE VERB
2400 W$=IV$(RND(IVERB)):GOSUB 3950:I1=W:GOSUB 3950:I2=W
     :IV$=W$:'INTRANSITIVE VERB
2450 W$=AJ$(RND(ADJECT)):GOSUB 3950:AJ=W:AJ$=W$:'ADJECT
     IVE
2500 PR$=PP$(RND(PREPO)):'PREPOSITION
2550 AR$=AR$(RND(ARTIC)):'ARTICLE
2600 Q$=IN$(RND(QUEST)):'INTERROGATIVE
2650 BV$=SV$(RND(BVERB)):'STATE-OF-BEING VERB PHRASE
2700 P$=MID$(".....,,;;?",RND(20),1):'PUNCTUA
     TION
2750 ON RND(19) GOTO 2800,2850,2900,2950,3000,3050,3100
     ,3150,3200,3250,3300,3350,3400,3450,3500,3550,3600
     ,3650,3700:'SELECT SENTENCE PATTERN
2800 N$=N1$:N=N1:GOSUB 4500:PRINT N$;" ";TV$;" ";AJ$;"
     ";N2$;P$:GOTO 3750:'#1
2850 N$=N1$:N=N1:GOSUB 4500:V$=IV$:V=I2:GOSUB 5400:PRIN
     T AJ$;" ";N$;" ";BV$;" ";V$;P$:GOTO 3750:'#2
2900 N$=N1$:N=N1:GOSUB 4500:PRINT AJ$;" ";N$;" ";IV$;P$
     :GOTO 3750:'#3
2950 N$=N1$:N=N1:GOSUB 4500:PRINT Q$;" ";AJ$;" ";N$;" "
     ;IV$;"?":GOTO 3750:'#4
3000 V$=TV$:V=T1:GOSUB 5000:S$=N2$:GOSUB 4050:PRINT AR$
     ;" ";AJ$;" ";N1$;" ";V$;" ";A$;N2$;P$:GOTO 3750:'#
     5
3050 V$=IV$:V=I1:GOSUB 5000:N$=N1$:N=N1:GOSUB 4500:PRIN
     T Q$;" ";V$;" ";PR$;" ";N$;" ";IV$;"?":GOTO 3750:'
     #6
3100 V$=IV$:V=I2:GOSUB 5400:N$=N1$:N=N1:GOSUB 4500:PRIN
     T V$;" ";N$;" ";TV$;" ";AJ$;" ";N2$;P$:GOTO 3750:'
     #7
3150 N$=N1$:N=N1:GOSUB 4500:GOSUB 4200:PN$=MID$("THEM.Y
     OU. HIM. HER. US.  ME.  IT.  ",RND(7)*5-4,5):PRINT
      N$;" ";IV$;" ";AD$;" ";PR$;" ";PN$:GOTO 3750:'#8
3200 N$=N1$:N=N1:GOSUB 4500:PRINT AJ$;" ";N$;" ";BV$;"
     ";AJ$;", BUT":GOTO 3750:'#9
3250 N$=N1$:N=N1:GOSUB 4500:NH$=N$:N$=NT$:N=NT:GOSUB 45
     00:PRINT Q$;" ";NH$;" ";PR$;" ";N$;" ";TV$;" THE "
     ;N2$;"?":GOTO 3750:'#10
3300 V$=IV$:V=I1:GOSUB 5000:GOSUB 4200:PRINT AR$;" ";N1
     $;" ";V$;" ";AD$;P$:GOTO 3750:'#11
3350 V$=IV$:V=I1:GOSUB 5000:PRINT "WHOSE ";N1$;" ";V$;"
     ?":GOTO 3750:'#12
3400 N$=N1$:N=N1:GOSUB 4500:NH$=N$:N$=N2$:N=N2:GOSUB 45
     00:V$=IV$:V=I2:GOSUB 5400:PRINT NH$;" ";BV$;" ";AJ
     $;" ";V$;" ";N$;P$:GOTO 3750:'#13
3450 V$=VT$:V=V2:GOSUB 5400:PRINT AR$;" ";AJ$;" ";V$;"
     OF ";NT$;" AND ";N1$;P$:GOTO 3750:'#14
3500 N$=N1$:N=N1:GOSUB 4500:V$=IV$:V=I1:GOSUB 5000:PRIN
     T N$;" OF ";NT$;" ";V$;P$:GOTO 3750:'#15
3550 V$=TV$:V=T2:GOSUB 5400:PRINT V$;" ";N1$;" AND ";N2
     $;P$:GOTO 3750:'#16
3600 V$=IV$:V=I1:GOSUB 5000:PRINT "THE ";NT$;" ";V$;P$:
     GOTO 3750:'#17
3650 V$=TV$:V=T1:GOSUB 5000:S$=AJ$:GOSUB 4050:PN$=MID$(
     "THEY YOU SHE  HE  WE  IT   I",RND(7)*4-3,4):PRINT
      PN$;" ";V$;" ";A$;AJ$;" ";N1$;P$:GOTO 3750:'#18
3700 N$=N1$:N=N1:GOSUB 4500:NH$=N$:N$=N2$:N=N2:GOSUB 45
     00:PRINT AJ$;" ";N$;" ";BV$;" ";NH$;P$:GOTO 3750:'
     #19
3750 NEXT L:'NEXT LINE
3800 'PATTERN FOR LAST LINE
3850 N$=NT$:N=NT:GOSUB 4500:NH$=N$:N$=N1$:N=N1:GOSUB 45
     00:PRINT NH$;" ";PR$;" THE ";N$;" ";BV$;" ";TA$;".
     ";
3900 Z$=INKEY$:IF LEN(Z$)>0 THEN 1300ELSE 3900
3950 'TRAILER DECODING
4000 W=VAL(RIGHT$(W$,1)):W$=LEFT$(W$,LEN(W$)-1):RETURN
4050 '"A"-"AN"AGREEMENT
4100 A$="A ":FOR I=1 TO 5:IF LEFT$(S$,1)=MID$("AEIOU",I
     ,1) THEN A$="AN "
4150 NEXT I:RETURN
4200 'ADJECTIVE TO ADVERB CONVERSION
4250 ON AJ GOTO 4300,4350,4400,4450:RETURN
4300 AD$=AJ$+"LY":RETURN:'ADD"LY"
4350 AD$=LEFT$(AJ$,LEN(AJ$)-1)+"ILY":RETURN:'ADD"ILY"
4400 AD$=LEFT$(AJ$,LEN(AJ$)-1)+"Y":RETURN:'"BLE"TO"BLY"
4450 AD$=AJ$+"ALLY":RETURN:'ADD"ALLY"
4500 'NOUN PLURALIZATION
4550 ON N GOTO 4600,4650,4700,4750,4850,4900,4950:RETUR
     N
4600 N$=N$+"S":RETURN:'ADD"S"
4650 N$=N$+"ES":RETURN:'ADD"ES"
4700 N$=LEFT$(N$,LEN(N$)-1)+"IES":RETURN:'"Y"TO"IES"
4750 IF RIGHT$(N$,1)="E" THEN N$=LEFT$(N$,LEN(N$)-1)
4800 N$=LEFT$(N$,LEN(N$)-1)+"VES":RETURN:'"F"TO"VES"
4850 N$=LEFT$(N$,LEN(N$)-2)+"E"+RIGHT$(N$,1):RETURN:'"M
     AN"TO"MEN"
4900 N$=N$+"E":RETURN:'ADD"E"
4950 N$=LEFT$(N$,LEN(N$)-2)+"I":RETURN:'"US"TO"I"
5000 'VERB TENSOR-SIMPLE PAST
5050 ON V GOTO 5100,5150,5200,5250,5300,5350:RETURN
5100 V$=V$+"D":RETURN:'ADD"D"
5150 V$=V$+"ED":RETURN:'ADD"ED"
5200 V$=LEFT$(V$,LEN(V$)-1)+"IED":RETURN:'"Y"TO"IED"
5250 V$=V$+RIGHT$(V$,1)+"ED":RETURN:'DOUBLE CONSONANT+"
     ED"
5300 V$=LEFT$(V$,LEN(V$)-1)+"T":RETURN:'"D"TO"T"
5350 V$=LEFT$(V$,LEN(V$)-2)+"A"+RIGHT$(V$,1):RETURN:'VO
     WEL NEXT TO END TO"A"
5400 'VERB TENSOR-PRESENT PARTICIPLE
5450 ON V GOTO 5500,5550,5600:RETURN
5500 V$=V$+"ING":RETURN:'ADD"ING"
5550 V$=LEFT$(V$,LEN(V$)-1)+"ING":RETURN:'REMOVE"E",ADD
     "ING"
5600 V$=V$+RIGHT$(V$,1)+"ING":RETURN:'DOUBLE CONSONANT+
     "ING"
5650 'VOCABULARY DATA
5700 'NOUNS
5750 DATA CANOE1,BATTLESHIP1,GHOST1,SOLDIER1,MONSTER1,R
     OBOT1,PAPER1,PAPERBOY1,RAPIST1,PHOTOGRAPHER1
5800 DATA AUTOMOBILE1,FORK1,CARNATION1,FINGERNAIL1,LEVE
     R1,BLOB1,WAR1,AARDVARK1,CLOCK1,CRAVAT1
5850 DATA LAW1,AXE1,FROG1,BAND1,RIPPLE1,SPARROW1,FALL1,
     DOG1,GLADE1,BELL1
5900 DATA MANDIBLE1,MUSIC1,WINE1,SOUL1,WORD1,TITLE1,ROO
     T1,STONE1,ROMAN1,HUMAN1
5950 DATA MACHINE1,SPOON1,APE1,HILL1,CITY3,UNIFORM1,WAT
     ER1,RING1,HAND1,LEG1
6000 DATA CHAIN1,PARADISE1,BEAST1,SPACE1,EYE1,WOMAN5,BE
     AUTY3,PLACE1,BUTTON1,KNOWLEDGE1
6050 DATA PYRAMID1,OPINION1,BOX2,GOAT1,FIRE1,SPIDER1,FO
     OD1,TIGER1,MOUNTAIN1,LIFE4
6100 DATA WISDOM1,MISERY3,MARE1,TIME1,SCIENCE1,KING1,SE
     A1,ASHTRAY1,BREEZE1,MORAL1
6150 DATA BALL1,CHOCOLATE1,VERSE1,CRAMP1,CROSS2,WRITER1
     ,PENTACLE1,DISH2,MIND1,MATTER1
6200 DATA HELMET1,LOVER1,SCHOONER1,MORNING1,EVENING1,CU
     P1,SAUCER1,STAR1,NOVA1,KANGAROO1
6250 DATA SLAB1,OCEAN1,CRATER1,SPECK1,RADIO1,SUN1,GATE1
     ,PICTURE1,HEAVEN1,HELL1
6300 DATA LETTER1,BASKET1,TAPE1,COMPUTER1,PERSONA6,STUD
     IO1,PROPHECY3,MISTRESS2,MASTER1,FEDORA1
6350 DATA COAT1,SLEEVE1,GLOVE1,INTRUDER1,HOLIDAY1,HORSE
     1,MULE1,MATRON1,BENCH2,TEAM1
6400 DATA COUNTRY3,WALRUS2,POOL1,STUDY3,CONSERVATORY3,R
     EVOLVER1,CARBINE1,FILM1,WIND1,STICK1
6450 DATA CHALK1,GULL1,CANYON1,ALLEY1,GORGE1,CONTINENT1
     ,STREAM1,CHASM1,GAP1,CRUST1
6500 DATA SURFACE1,BIRTH1,FILAMENT1,DISC1,THUMB1,WIRE1,
     PREMIER1,SHOE1,BOOT1,MAST1
6550 DATA TRACK1,LINE1,FIBER1,THEATRE1,SPECTRE1,TREE1,S
     TUMP1,LEAF4,BRANCH2,CELL1
6600 DATA UNIT1,SPINE1,FLOWER1,STEM1,PETAL1,WALL1,DRAWI
     NG1,WINDMILL1,ARM1,EAR1
6650 DATA INSECT1,KEY1,MARK1,MEDAL1,EGYPTIAN1,IDEOLOGY3
     ,TERM1,CROWD1,GRAPE1,FRUIT1
6700 DATA PAPYRUS7,OCTOPUS7,RAINDROP1,IVY3,BRICK1,THUND
```

Program continues

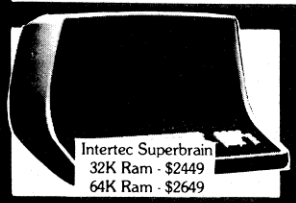
```
ER1,LIGHTNING1,SPARK1,DUCT1,CORRIDOR1
6750 DATA TUNNEL1,VENT1,ROOM1,BUCKLE1,PUMP1,HOOKER1,MOO
     SE1,GUTTER1,SYMPHONY3,WATCH2
6800 DATA BRACELET1,SWAN1,MOMENT1,DOOR1,FALL1,EXIT1,SHE
     ATH1,GODDESS2,BLASPHEMY3,MARRIAGE1
6850 DATA GROUP1,CLAN1,HATCH2,GENERATOR1,COVER1,BONE1,F
     ATHER1,MOTHER1,DAUGHTER1,SON1
6900 DATA MATTRESS2,MAGNATE1,GALAXY3,ASTEROID1,METEORIT
     E1,COMET1,NIGHT1,FLATCAR1,FLINTLOCK1,FLUID1
6950 DATA JINX2,NIGHTCAP1,OCTAVE1,CENSOR1,SEQUIN1,SUPER
     MAN5,TEAR1,TAX2,TEACHER1,METAL1
7000 DATA VEGATABLE1,VENOM1,VICE1,VIOLET1,VIRGIN1,VISIO
     N1,VIRTUE1,WALTZ2,WIG1,FAIR1
7050 DATA FORMULA6,CARDIGAN1,CARAVAN1,CAMEL1,ZEBRA1,QUE
     EN1,DRAG QUEEN1,XYLOPHONE1,YETI1,FROGMAN5
7100 DATA TRAIN1,LESBIAN1,HOMOSEXUAL1,PERVERT1,FLASHER1
     ,MOVIE1,TRUCKDRIVER1,TAXICAB1,ELEPHANT1,PHONOGRAPH
     1
7150 DATA PAPERBACK1,BOOK1,SPACESHIP1,VULCAN1,HARRIDAN1
     ,FANATIC1,INVADER1,HEADMASTER1,SCHOOLMARM1,ABORIGI
     NE1
7200 DATA PRISON1,PRISONER1,SPY3,DOCUMENT1,WOLF4,ELF4,D
     WARF4,DINOSAUR1,CANNIBAL1,MISSIONARY3
7250 'TRANSITIVE VERBS
7300 DATA POMMEL34,TRUNCATE21,ASK12,DESTROY12,PRINT12,P
     URIFY13,HATE21,LIKE21,PLACATE21,ACKNOWLEDGE21
7350 DATA MOLLIFY13,BUILD15,SIEZE21,ENGENDER12,VACATE21
     ,GRAB34,DISTILL12,CONDEMN12,EMBRACE21,MURDER12
7400 DATA DOMINATE21,DREAD12,ENGAGE21,ESPOUSE21,ENGROSS
     12,BRUSH12,BERATE21,PRESENT12,REMOVE21,SEVER12
7450 DATA SHOCK12,PENETRATE21,PERCEIVE21,RAKE21,RELIEVE
     21,WELCOME21,HIT39,WRECK12,THROTTLE21,STAB34
7500 DATA SPEND15,BEND15,ENJOY12,STAPLE21,FOLD12,MUTILA
     TE21,IMPALE21,HELP12,PACIFY13,RAPE21
7550 DATA INVEST12,RUIN12,NAIL12,IMPRESS12,REACH12,CRUS
     H12,IMPEDE21,SCRATCH12,SLAP34,CHASE21
7600 DATA DECIMATE21,PLUG34,DETECT12,FONDLE21,PICK12,HU
     RT19,AID12,HEAR11,TOUCH12,LICK12
7650 DATA SQUASH12,BURY13,BLOCK12,PLEASE21,CLEAN12,CLEA
     NSE21,PAT34,PET39,INDICATE21,RUB34
7700 DATA STROKE21,PURGE21,MANGLE21,BAR34,BARE21,HARM12
     ,DIVE BOMB12,NURSE21,DOCTOR12,WELD12
7750 'VERBS TRANSITIVE OR INTRANSITIVE
7800 DATA STOMP12,TYPE21,STOP34,KILL12,RUN36,WALK12,ECH
     O12,PUNT12,HALLUCINATE21,GUESS12
7850 DATA FINISH12,FERMENT12,VAPORIZE21,RETURN12,READ19
     ,LAMENT12,CHANGE21,SERENADE21,WRIGGLE21,SPLIT39
7900 DATA RUSH12,QUESTION12,DARE21,LOVE21,SLOW12,INDULG
     E21,ADD12,PUSH12,IMPROVE21,FLIP34
7950 DATA WASH12,SOAK12,SHOWER12,CHEW12,STRIP34,HURRY13
     ,WORRY13,PONDER12,CONSIDER12,RAISE21
8000 'INTRANSITIVE VERBS
8050 DATA JUMP12,SWIM36,BREATHE21,SKI12,ABSCOND12,JOKE2
     1,FORNICATE21,SIT36,SCREAM12,TALK12
8100 DATA SUICIDE21,SOAR12,EAVESDROP34,SIGH12,GRUMBLE21
     ,SWOOP12,ROMP12,CRY13,UPCHUCK12,HUM34
8150 DATA CREEP12,INSIST12,PERSINKEY$ST12,DESIST12,SLOB
     BER12,DIVERSIFY13,RANKLE21,SMOLDER12,PROCEED12,DOZ
     E21
8200 'ADJECTIVES
8250 DATA SAD1,HOPELESS1,HORRENDOUS1,EXUBERANT1,ANALAGO
     US1,MAGNIFICENT1,AFFABLE3,TYPICAL1,STUPID1,DUMB1
8300 DATA PREVARICATIVE1,EVIL1,GORGEOUS1,WONDERFUL1,HUM
     ANE1,TIRED1,MOROSE1,BOLD1,WISE1,JUBILANT1
8350 DATA BLUE1,BLACK1,CREATIVE1,SPLENDOROUS1,DISAGREEA
     BLE3,EXTRAORDINARY2,ABSURD1,RELUCTANT1,LOQUACIOUS1
     ,BAD1
8400 DATA HARD1,SOFT1,SMOOTH1,TORTUROUS1,MEANINGFUL1,VI
     TAL1,SHORT1,UBIQUITOUS1,IDIOTIC4,IMPRESSIVE1
8450 DATA CAUSTIC4,RED1,FORBIDDING1,MURDEROUS1,ARGUMENT
     ATIVE1,EASY2,MASTERFUL1,DESTRUCTIVE1,OMNIPOTENT1,O
     MNIPRESENT1
8500 DATA ROUGH1,UNRELENTING1,ACTIVE1,WRETCHED1,HATEFUL
     1,JOYOUS1,BEAUTIFUL1,BENEFICIENT1,AGREEABLE3,NAIVE
     1
8550 DATA KNOWLEDGEABLE3,MALEVOLENT1,VENGEFUL1,VIBRANT1
     ,VIOLENT1,ALARMING1,REPROACHFUL1,FOOLISH1,LUSCIOUS
     1,LASCIVIOUS1
8600 DATA TEPID1,VAPID1,PRETTY2,GRAND1,PERCEPTIVE1,IGNO
     RANT1,HAPPY2,SIMPERING1,TEMPESTUOUS1,ARDUOUS1
8650 DATA INEXHORABLE3,QUICK1,GLORIOUS1,NOISOME1,TANGIB
     LE3,INTANGIBLE3,ABRASIVE1,DERISIVE1,PRECISE1,IMPRE
     CISE1
8700 DATA BORING1,EXCITABLE3,ENERGETIC4,TERSE1,CONCISE1
     ,VERBOSE1,FLUID1,WHOLESOME1,MANIACAL1,MAGIC4
8750 DATA DISTANT1,COLD1,HOT1,MARKED1,LIQUID1,FRESH1,DU
     STY2,CRUDDY2,DROUSY2,WEAK1
8800 DATA HORRIBLE3,TRAGIC4,MONSTROUS1,GUILTY2,CANNIBAL
     ISTIC4,HUNGRY2,CARNIVOROUS1,OMNIVOROUS1,PLAGIARIST
     IC4,SELF-RIGHTEOUS1
8850 DATA CORRECT1,PROPER1,RELIGIOUS1,DEVOUT1,AESTHETIC
     4,PAINLESS1,DRY1,WET1,DAMP1,NOSY2
8900 DATA AMATEURISH1,FANATICAL1,FANTASTIC4,STRANGE1,OD
     D1,EVEN1,SAVAGE1,EDUCATED1,LONGITUDINAL1,LOVING1
8950 DATA BISEXUAL1,NAKED1,HEATED1,ALERT1,PHALLIC4,HUMO
     ROUS1,BASIC4,DIRTY2,PERVERTED1,SHARP1
9000 DATA IMPRESSIVE1,PERSISTENT1,HELPFUL1,LAMENTABLE3,
     PLEASANT1,CONSIDERATE1,IMPRESSIONABLE3,KIND1,NAKED
     1,OBVIOUS1
```

Program continues

ΩMEGA SALES CO. ✓389

"WHOLESALE COMPUTER PRICES" DIRECT TO THE PUBLIC

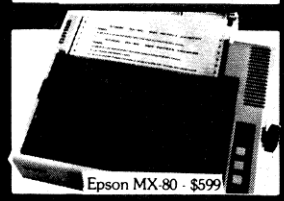
East Coast: 12 Meeting St., Cumberland, RI 02864
West Coast: 3533 Old Conejo Rd., #102, Thousand Oaks, CA 91320



Intertec Superbrain
32K Ram - $2449
64K Ram - $2649

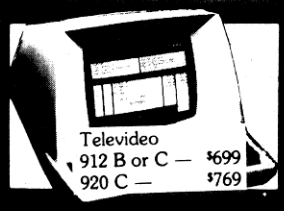
NEC spinwriter
5510-5530 - $2449

Apple II - 16K, 48K
Call for Price



Epson MX-80 - $599

Diablo 630 - $1995
(with tractor feed) $2195



Televideo
912 B or C — $699
920 C — $769

PRODUCT SPECIAL of the MONTH!!

Products are NOW IN STOCK AT ΩMEGA Sales Co.

Atari 800 16K - $749
32K - $849

Atari 850 Interface Module $159
Atari 825 Printer $695

Okidata Microline 80 - 525

Centronics - Ltd. quan.	
730-3	$ 525
737-1	$ 625
704-9	$1250
730-1	$ 525
753	$1750
703-9	$1550

ΩMEGA OFFERS THE BEST DELIVERY AND PRICE ON:
APPLE • ATARI • TRS-80 Model II • INTERTEC • DIABLO • EPSON • HEWLETT—PACKARD • SOROC • COMMODORE • NEC • QUME • CENTRONICS

CALL TOLL FREE FOR ΩMEGA'S PRICE!
EAST COAST: 1-800-556-7586

ΩMEGA sells only quality merchandise to our customers.
ΩMEGA will try to match any current advertised price with similar purchase conditions.
Before you buy anywhere else — be sure to call ΩMEGA Sales Co.
WEST COAST: 1-805-499-3678 EAST COAST: 1-401-722-1027

ΩMEGA ships via UPS, truck, or air. COD's.
Visa, Mastercharge accepted, with no service charge.

VISA **Telex: 952106**

master charge

```
9050 DATA INDULGENT1,LOVABLE3,LAZY2,SLOTHFUL1,MUNDANE1,
     URBANE1,ASININE1,LEGIBLE3,CULPABLE3,PATHETIC4
9100 DATA COURTEOUS1,REVERENT1,OBEDIENT1,THRIFTY2,CHEAP
     1,MISERABLE3,APATHETIC4,BRAVE1,LOGICAL1,HEAVY2
9150 DATA FAIR1,GULLIBLE3,INSISTANT1,INSTANT1,MOUNTAINO
     US1,PONDEROUS1,METEORIC4,MORAL1,METALLIC1,NEAR1
9200 DATA DEAR1,OPINIONATED1,OBTUSE1,IMPOTENT1,PORNOGRA
     PHIC4,PHOTOGRAPHIC4,PHOTOGENIC4,PERVERSE1,PROVOCAT
     IVE1,QUESTIONABLE3
9250 DATA ROMANTIC4,SOULFUL1,STUDIOUS1,SCHOLASTIC4,SUIC
     IDAL1,UNIFORM1,VENOMOUS1,VACANT1,VIRTUOUS1,VIRGINA
     L1
9300 'PREPOSITIONS
9350 DATA UNDER,OVER,THROUGH,AROUND,AHEAD OF,BEHIND,WIT
     HIN,WITHOUT,BEFORE,AFTER
9400 DATA BENEATH,ABOVE,INSIDE,OUTSIDE,NEXT TO,AWAY FRO
     M,WITH,BEYOND,ALONGSIDE,BELOW
9450 DATA IN,OUT OF,ADJACENT TO,FROM,NEAR,BY,UNDERNEATH
     ,FOR,AGAINST,ALONGSIDE
9500 'ARTICLES
9550 DATA THIS,THAT,HIS,HER,ITS,SOME,OUR,YOUR,THEIR,MY
9600 'INTEROGATIVE WORDS, PHRASES
9650 DATA HOW DO,WHERE DO,WHY DO,WHEN DO,WHAT,WHICH,WHO
     SE,WITH WHOM DO,DO,FOR WHAT DO
9700 'STATE-OF-BEING VERBS, PHRASES
9750 DATA ARE,ARE NOT,HAVE BEEN,HAVE NOT BEEN,WILL BE,W
     ILL NOT BE,MAY BE,MAY NOT BE,WERE,WERE NOT
9800 DATA CAN BE,CANNOT BE,MAY HAVE BEEN,MAY NOT HAVE B
     EEN,COULD BE,COULD NOT BE,COULD HAVE BEEN,COULD NO
     T HAVE BEEN,WOULD BE,WOULD NOT BE
9850 DATA WOULD HAVE BEEN,WOULD NOT HAVE BEEN,SHOULD BE
     ,SHOULD NOT BE,SHOULD HAVE BEEN,SHOULD NOT HAVE BE
     EN,WILL HAVE BEEN,WILL NOT HAVE BEEN,MIGHT HAVE BE
     EN,MIGHT BE
```

there are less than 200 bytes left. More string space is cleared than is actually required because the program runs faster that way.

Here is a detailed explanation.

Lines 1000–1250: Initialization. String space cleared, variables set to amounts of data, vocabulary arrays dimensioned and filled.

1300–1350: Clear screen for new poem. Start of main program.

1400–1650: Assign and decode words for title.

1700–1950: Five title patterns chosen by random ON-GOTO.

2000–2050: Start poem text. Set stanza length at 3–6 lines.

2100–2150: Set poem length, loop for separate lines. Linefeed to separate title from text and split stanzas.

2200–2700: Assign and decode words and punctuation for one line.

2750–3700: 19-line patterns chosen by random ON-GOTO.

3750: Loop to next line.

3800–3850: Special pattern for final line.

3900: Hit any key to start new poem. Back to line 1300.

3950–4000: Remove trailing digit from coded word, return decoded word and code number.

4050–4150: Make A or AN agree with following word.

4200–4450: Adjective to adverb conversion. Method chosen by code number.

4500–4950: Noun pluralization. Structured as previously. If code number is high (9), no change is made, as in "FISH".

5000–5350: Verb to past tense. Structured as nouns. Uses second code number.

5400–5600: Verb to participle. Structured as others. Uses first code number.

5650–9850: Data. Nouns, verbs and adjectives are coded for the subroutines. The others are used as they are. There are 10 items per line to make counting easier.

The program is modular and easy to change. Here are some things I've used or considered:

DATA is easily added or deleted. The variables in line 1050 must be changed to fit. Code numbers can be determined by comparison with similar data, or by walking through the subroutines.

The string arrays can be replaced with routines to search data, reducing the memory requirement, but slowing execution. Using disk files rather than data statements will reduce the memory requirement.

Additions to the subroutines can be done by adding a line to make the change, expanding the applicable ON-GOTO, and adding data with the new code number as desired.

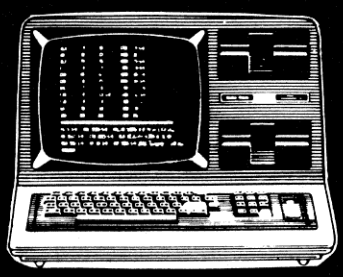
Title and line patterns may be added or deleted. A good book of free verse can supply patterns. Change the ON-GOTO involved.

Stanza and poem length are a matter of preference. If desired, there need be no splitting of stanzas, or poems of epic length. I've set the line loop to go on indefinitely writing the same poem. It's also a way to troubleshoot the subroutines and line patterns.

If the output warrants it, all the data could be in lowercase, with a subroutine to capitalize the first letter in a line and every word in the title. Floral graphics could be added here and there. The program is designed to be improved, modified, expanded or played with.

For those who are not enthusiastic about the typing necessary to get this program on line, copies on cassette tape for TRS-80 Level II 16K are available from the author for $8.00. Comments, inquiries, conversion problems and other ideas for modifications will be answered. ■

"My talent for poetry is non-existent, but I have a fair amount of skill at BASIC programming."



NATIONAL TRS-80®
MICROCOMPUTER SHOW

New York Statler
Exposition Hall
(opposite Pennsylvania Railroad Station
and Madison Square Garden)
7th Ave. & 33rd Street

May 21, 22, 23, 1981
Thursday Noon to 6 PM
Friday 11 AM to 6 PM
Saturday 10 AM to 4 PM

For the Businessman, Educator, Professional and Hobbyist.

- Commercial Exhibits and Sales of Microcomputers, Software, Books, Magazines, Supplies, Parts, Printers, Etc.
- Free Seminars
- Famous Name Guest Speakers.
- User Groups. (Mod I, II, III, Color and Pocket Computers).
- Door Prizes - A TRS-80® Computer will be given away free each day, plus other prizes.

Avoid standing on line - Send in your registration today!

Radio Shack and TRS-80 are registered trademarks of Tandy Corporation, which has no relationship to Kengore Corporation or The National TRS-80 Microcomputer Show.

REGISTRATION FOR TRS-80® MICROCOMPUTER SHOW

May 21, 22, 23, 1981 — New York Statler Hotel

Name _____ Title _____

Company Name _____

Address _____

City, State, Zip _____

Please send _____ registrations at $10.00 each.

(Registration Badge will be sent to you on May 1st.)

Send To:
Kengore Corporation, Dept. 80
3001 Route 27
Franklin Park, N.J. 08823 ✓230

(Be sure to enclose check or money order.)

Don't eschew obfuscation.
Get sneaky about protecting your software. Here's how.

No Forwarding Address

Bertram A. Thiel
159 W. Main St.
Frostburg, MD 21532

Occasionally you may need to make your programs safe from tampering. Methods range from the use of a DELETE command as the last statement, to more complex ones.

The following routines will help prevent tampering.

First, see *80 Microcomputing*, January 1980, p. 93, for background on program storage in memory. Note: It isn't the line numbers that are used when the BASIC interpreter is operating (except for GOTO and GOSUB statements), but the addresses before the line numbers which point to the next pointer.

Once your program is written and debugged, start at the first pointer, addresses 17129 and 17130 (Level II only). Store the

```
65300 P1=17129
65310 P2=PEEK(P1)+PEEK(P1+1)*256: REM POINTER TO NEXT P
     OINTER
65320 IF PEEK(P1+2)+PEEK(P1+3)*256=65280 THEN 65350: RE
     M TO ROUTINE DELETION COMMAND
65330 POKE P1+2,0:POKE P1+3,0
65340 P1=P2: GOTO 65310
65350 DELETE 65280-65360
65360 END
```

Program Listing 1

```
65300 P1=17129
65310 P2=PEEK(P1)+PEEK(P1+1)*256: REM POINTER TO NEXT P
     OINTER
65320 IF PEEK(P1+2)+PEEK(P1+3)*256=65280 THEN 65370: RE
     M TO ROUTINE DELETION COMMAND
65330 PRINT"TO SAVE LINE NUMBER";PEEK(P1+2)+PEEK(P1+3)*
     256;"THEN ENTER THE LETTER   Y   ELSE JUST PRESS E
     NTER";
65340 INPUT P3$: IF P3$="Y" THEN P3$="": P1=P2: GOTO 65
     310
65350 POKE P1+2,0:POKE P1+3,0
65360 P1=P2: GOTO 65310
65370 DELETE 65280-65380
65380 END
```

Program Listing 2

```
10 PRINT"MULTIPLICATION TABLE"
20 POKE17191,205
30 FORX=1TO12
40 T=X*X
50 PRINTX;"X";X;"=";T
60 NEXTX
70 POKE17191,207
80 END
```

Program Listing 3

```
65400 REM      ROUTINE TO FIND RAM ADDRESS
65410 REM      OF A BASIC LINE NUMBER
65420 REM      B THIEL 1/12/80
65430 INPUT "PROGRAM LINE NUMBER";L
65440 P1=17129
65450 P2=PEEK(P1)+PEEK(P1+1)*256: REM POINTS TO NEXT PO
     INTER
65460 L1=PEEK(P1+2)+PEEK(P1+3)*256: REM LINE NUMBER
65470 IF P2=0 THEN PRINT "LINE";L;" NOT FOUND, END OF P
     ROGRAM IS AT ADDRESS";P1+1:STOP
65480 IF L1=L THEN PRINT "ADDRESS OF LINE";L1;" IS";P1+
     2;" AND";P1+3: STOP
65490 P1=P2: GOTO 65450
```

Program Listing 4

ASYLUM!

You are sitting alone. It is 2:00 AM. Your eyes are bloodshot. As you peer into your computer screen, you suddenly scream, "I must be crazy!" If this has ever happened to you, or the men in white coats from Deathmaze 5000 have hauled you away, it is time for you to enter the most ambitious 3-D graphics adventure yet offered by Med Systems: **ASYLUM!**

3-D PERSPECTIVE GRAPHICS

Asylum features the full screen 3-D perspective graphic displays that have made Deathmaze and Labyrinth best sellers. You can actually see what you are doing and where you are going! The mazes and buildings are bit-coded. This allows us to store **gigantic** mazes in small amounts of memory. These programs are **not** just a series of stored pictures. Our mazes typically contain **over 600 locations**. Further, machine-language programming gives instantaneous graphics generation and game response!

ADVANCED LANGUAGE INTERPRETER

Asylum also features one of the most advanced input routines available. Players are no longer limited to one and two word commands. Entire sentences may be entered from a vocabulary of over 200 words!

ASYLUM places you on a cot in a small room. Periodically, a janitor lobs a hand-grenade through the window of your locked door. What you do next could mean survival and escape! It could also mean permanent residence in the home for Deathmaze survivors! To leave, you will have to deal with guards, fellow survivors, doctors, the infamous Crazed Carpenter, and much, much more. Don't expect to get out any time soon!

TRS-80 Level II 16K or Model III 16K $14.95

DEATHMAZE 5000 places you in a gigantic five-story building. There is only one goal. **ESCAPE ALIVE!** Monsters, dogs, vampires, and other vile horrors will plague your every step as you struggle to survive one of the most challenging adventures ever written. As of December 20, only two people outside the Med Systems' staff were known to have escaped!

TRS-80 Level II 16K or Model III 16K $12.95

LABYRINTH places you in a huge maze of tunnels inhabited by gnomes, ghosts, witches, and an evil minotaur. You must find the weapons and treasures needed to destroy the minotaur before he destroys you! There is food enough in the maze to hold out for months!

TRS-80 Level II 16K or Model III 16K $12.95

REWARD!

This man escaped Deathmaze only to be hauled off to Asylum! From his condition you would never realize that he designed both **Deathmaze** and **Asylum**. Those few others who escape either nightmare may send their correct solution to us. On May 30, a drawing will be held. Six intrepid adventurers will win their choice of three programs from Med Systems' catalog and a shirt with the Deathmaze or Asylum logo. Only the correct solutions are eligible. All judgements final. Please enclose a SASE for return of solutions or notification of correctness. All winners will be contacted directly.

SATISFACTION GUARANTEED!

Asylum, Deathmaze 5000, and Labyrinth are guaranteed to be the most incredible 16K 3-D graphic adventures you can buy. If for any reason you are not satisfied with these products, return your order within 14 days for a prompt and cheerful refund.

ORDERING INFORMATION

Orders are processed within five working days. We pay all postage and handling within the U.S., Canada, and U.S. territories. European orders please include $2.00 for air post.

☐ Asylum ($14.95) $ _____

☐ Deathmaze 5000 ($12.95) $ _____

☐ Labyrinth ($12.95) $ _____

☐ Programs on Disk (add $4.00) $ _____

☐ Programs on Cassette (N/C)

TOTAL $ _____

Name _____

Street _____

City _____ State _____ Zip _____

Computer:

☐ TRS-80 LII 16K ☐ Model III 16K

☐ Mastercard ☐ VISA ☐ check

MC or VISA # _____

Expiration Date _____

Med Systems Software ✓128

P.O. Box 2674-W Chapel Hill, NC 27514

(919) 933-1990

✓Reader Service—see page 290

RSGLMOD1 - $50 - Enhancements to Radio Shack General Ledger 1.1 Special feature - Copy ONLY ONE disk for back-up security.

Print features - General ledger with beginning balances; current month activity; ending balances ** Classified balance sheet.

Print Options - Omit current column on income statement ** Omit account numbers on balance sheet and/or income statement.

Input Display - Previous document name, number of entries/total entries permitted for current session, and dollar total of current document are always visible. These are nice anytime and just plain excellent after you return from an interruption.

Input Options - Set up an "automatic" account number. Touch ENTER to use. Change to another account anytime ** Re-do a document after it is balanced (Before starting next document) ** Change document name and entry date for each entry. One possibility is a check register with only one cash entry ** Automatic credits to liability, capital, and income accounts ** Others.

Supplied as program lines which must be MERGED into the original programs, or, send a disk copy of the original programs and the modifications will be installed on your diskette — no additional charge.

DDIR81 - $23 - Organize your disks. Creates program lines of directory contents using TRSDOS or NEWDOS. 32K/one drive.

CAT - $12 - You have to plan ahead to beat the computer or a friend. X's and O's advance with each human move. Fast graphics.

FTDEMO80 - $12 - Displays and EXECUTES the NEWDOS/80 Appendix A programs and keyboard entries. A real time saver. 32K. Disk only. Deduct $3 each for 2nd-3rd-4th program. Add 4% for MasterCard/Visa

NEWDOS/80 - $135 **DOUBLER - $215** **DOUBLE-ZAP II - $47** **DOUBLE-ZAP (2.1)- $37** **BASE 2 MODEL 850 - $760 w/Cable**
MODEL III DISK DRIVES **FIRST - $760** **SECOND - $350** **EXTERNAL - $430**

✓445 **AT-80** **3827 Dismount Dallas, Texas 75211 (214) 339-0498**

ADD-ON MEMORY? GET THE FACTS!
Japanese 16K RAM chips have a one-to-ten in-service failure ratio to U.S. chips -- from a 1980 Hewlett Packard Study.

Do not buy carelessly specified chips or chips of unknown manufacture. We offer 4116-compatible chips from the two top Japanese manufacturers, NEC and Fujitsu, for most popular computers and expansion memory boards, including:

***APPLE *TRS-80 *NEW PET *HEATH H-89 *SUPERBRAIN *EXPANDORAM *many others**

NEC UPD 416-C 200 NSEC Plastics..
~~$44.95/16K~~ SPECIAL $39.95/16K

FUJITSU 8116E 200 NSEC Ceramics..
~~$51.95/16K~~ SPECIAL - $46.95/16K

*Price at ad copy deadline ... We'll beat any legitimate price for comparable chips. Hi-volume users, dealers, or clubs, ask about further quantity discounts.

Guaranteed good. Send check or money order to:

MINIS & MICROS, INC.
19754 VICTORY BOULEVARD
WOODLAND HILLS, CA 91367

(213) 342-4535 ✓98

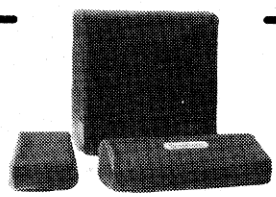
NO Shipping Charge
CA residents add 6% sales tax

✓100

CompuCover®

COVER YOUR INVESTMENT

- Cloth Backed Naugahyde Vinyl
- Improved Reliability
- Longer Life
- Waterproof & Dustproof
- Two Decorator Colors Saddle Tan • Black



TRS-80 MODEL I		TRS-80 MODEL III	$14.95
Keyboard	$ 7.95	TRS-80 COLOR COMPUTER	9.95
Cassette	4.95	Line Printer I	$16.95
Video Display	9.95*	Line Printer II	9.95
Package Offer	19.95*	Line Printer III	15.95
*NOTE-Add $3.00 for Expansion Interface		Line Printer IV	9.95
TRS-80 5¼" Disk	$ 4.95	Line Printer VI	14.95
Two Disk Cover (side by side)	7.95	Daisy Wheel Printer II	16.95
TRS-80 MODEL II		Quick Printer I	9.95
Entire Unit	$22.95	Quick Printer II	5.95
Keyboard Only	7.95		
Three Disk Unit (8" Drives)	18.95		

Over 200 covers, write for complete list.

Send check or money order to
CompuCover
P.O. Box 324 (Dept A)
Mary Esther, FL 32569
Phone (904) 243-5793

Include $1.50 for postage and handling.
Overseas orders include $4.00 postage.

DEALER INQUIRES INVITED

MAXELL® OR **Dysan** CORPORATION

Some computorists pay less - but may not receive Shuggart™ or IBM® approved disks.

8" SINGLE SIDE DOUBLE DENSITY.....Box of **10** for **$60**
8" DOUBLE SIDE DOUBLE DENSITY.....Box of **10** for **$70**
5¼" MINI.....Box of **10** for **$50**
DYSAN® DISKS
5¼" MINI.....Box of **5** for **$25**
(Specify - 8" Soft or Hard Sector/5" Soft or Hard Sector)

VISA

C.O.D. - $1.00 Additional

✓121

Custom Electronics Inc.

238 EXCHANGE STREET
CHICOPEE, MA. 01013

413 · 592 · 4761

established 1960 · closed mondays

APPLE TRS-80 IBM WANG ALTAIR DATA GENERAL MATTEL CROMENCO
ATARI TI/99·4 PET

EDUCATIONAL SOFTWARE

For TRS-80* Color Computer, PET, Apple II.

ELEMENTARY	**MATH**
SCIENCE	**BIOLOGY**
GEOGRAPHY	**HISTORY**
ECONOMICS	**ACCOUNTING**
FOREIGN LANG.	**BUSINESS ED.**
GRAMMAR	**FARM RECORDS**

COIN INVENTORY

Write for FREE Catalogue:

MICRO LEARNINGWARE ✓89
BOX 2134, N. MANKATO MN 56001
507-625-2205

VISA & MASTER CHARGE ACCEPTED

We pay 15% royalty for Educational Programs listed with us.

Apple is a trademark of Apple Computer Co.
TRS-80 is a registered trademark of TANDY CORP.
Pet is a trademark of Commodore Bus. Machines.

HOMES for the TRS-80

Featuring

High Quality, Commercial and Economy Models.

Computer Consoles for $129.95.
Printer Stands $39.95.
Complete TRS-80 Business System, built-in.

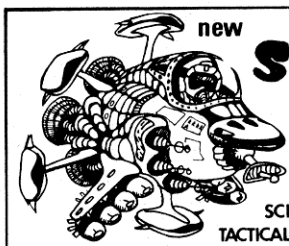
Custom furniture for the TRS-80 office or home decor.
—FOR 24 HOUR INFORMATION—
PHONE 408-946-1265

AVS AUDIO-VIDEO SYSTEMS 2485 AUTUMNVALE AVE. SAN JOSE, CA. 95132

Dealer inquiries invited.

---

new



SPACE ACE 21

SCIFI GAME OF TACTICAL SPACE COMBAT

Design your own space fighter. You choose the weapons, equipment, engines, and more. Then engage in a one on one sudden death fly-off. Human opponent or 10 different computer opponents. Fight in 2 or 3 dimensions. Action Sounds and Graphics. The most imaginative scifi game we offer. $19.95
(written by former USAF phantom pilot)

More Super Games with Action Sounds and Graphics:

!!! ANTS !!!	Amazing battle between ant colonies. Easy to play, Challenging to master!	$14.95
PARSECTOR V	The ultimate space war. Unique split video, each player has a private display.	$19.95
PARSECTOR 8	Tournament version of Parsector V. Larger galaxy to conquer.	$20.95
HIGH SPEED LIFE	The fastest, most advanced life game. Up to 2000 gen./min possible	$14.95
NAME THAT STATE QUIZ	Educational fun. Draws State and asks questions.	$14.95

TRS 80 Level II 16k, Model I or III required. ✓231
For cassette and instructions send CHECK or MONEY ORDER.

Synergistic Solar Inc. PO Box 560595 Miami FL 33156

Please write for more info. Dealer inquiries invited.

✓146

Memory Dump of Listing 3

```
17129    5      Next pointer
17130    67     is 17157
17131    10
                Line 10
17132    0
17133    178    PRINT
17134    34     "
17135    77     M
17136    85     U
17137    76     L
17138    84     T
17139    73     I
17140    80     P
17141    76     L
17142    73     I
17143    67     C
17144    65     A
17145    84     T
17146    73     I
17147    79     O
17148    78     N
17149    32     space
17150    84     T
17151    65     A
17152    66     B
17153    76     L
17154    69     E
17155    34     "
17156    0      eol
17157    20     Next pointer
17158    67     is 17172
17159    20     Line 20
17160    0
17161    177    POKE
17162    49     1
17163    55     7
17164    49     1
17165    57     9
17166    49     1
17167    44     ,
17168    50     2
17169    48     0
17170    53     5
17171    0      eol
17172    32     Next pointer
17173    67     is 17184
17174    30     Line 30
17175    0
17176    129    FOR
17177    88     X
17178    213    =
17179    49     1
17180    189    TO
17181    49     1
17182    50     2
17183    0      eol
17184    42     Next pointer
```

Program continues

contents in a buffer, after multiplying the second address by 256 and adding it to the first. Then change the following two address contents to zeros, or anything you like, by POKEing in the appropriate values.

Once this is done no one can touch the program through ordinary means.

Program Listing 1 is for structured programs or programs where there are no GOTO or GOSUB statements. First, enter your program and make sure it performs. Then enter Listing 1 manually or from your subroutines using the CFETCH from Instant Software's TRS-80 Utility II. List it. Type RUN 65280, and ENTER. LIST the program to confirm all lines are zero and that the routine has disappeared. Then run the program with all its possibilities to make sure there are no hidden problems. Then CSAVE it on tape.

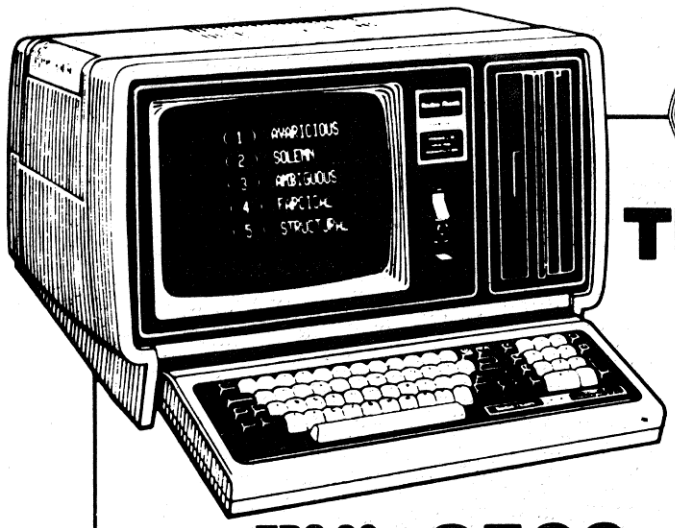
If you feel a little mischievous, change line 65340 to:

```
65340 POKE P1+2,RND(255):
      POKE P1+3,RND(254)
```

which will give random line numbers.

If you wish to un-number a program that has line references, make a list of all line numbers used in GOTO and GOSUB statements and enter Program Listing 2. This changes each line one at a time as you press ENTER; a Y means "yes, save that line number." The routine will erase itself as before. Do not try to use the RND functions here, since the interpreter requires sequential numbers in the line positions to work properly.

Now to make your TRS-80 do something that no programmable calculator can do; change its own program. The illustration is Program Listing 3. Enter it exactly as shown, delete spaces except for the one between the line numbers and the statement; and the one in the first print statement. If you have installed it properly, the arithmetic will be wrong. The POKE statements change the multiply function in line 40 to an add, and then change it back in line 70. Refer to the chart on page 100 of

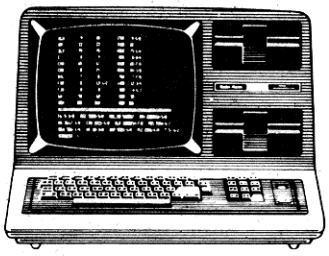


Radio Shack®
DEALER

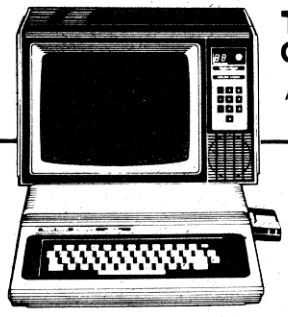
TRS-80™ DISCOUNT

☐ NO OUT-OF-STATE TAX
☐ NO SHIPPING COSTS

TRS-80 MODEL II 64K $3500
PACKS ENOUGH DATA HANDLING POWER FOR MANY SMALL BUSINESSES.



TRS-80 MODEL III 32K-2 DISKS
$2100
NEW PERSONAL COMPUTER... REAL-TIME CLOCK, SHARPER CRT IMAGES AND FASTER LOADING CASSETTES



TRS-80 COLOR COMPUTER OR VIDEO $339 each
A LOW COST, COLOR COMPUTER FOR PERSONAL BUSINESS OR ENTERTAINMENT

CERTIFIED CHECKS
CASHIERS CHECKS
OR CREDIT CARDS

MasterCard VISA

PERRY OIL & GAS INC. ✓51

137 NORTH MAIN STREET, PERRY, MICH. 48872
PHONE (517) 625-4161

WARRANTIES HONORED BY ALL RADIO SHACKS • *T.M. TANDY CORP.

PURE FIG GOODNESS!

Now you can use Forth on your TRS-80 Model I with 16K and a single disk. This is a stand-alone version of the widely accepted Forth Interest Group Model 1.1 with the standard Forth editor. Extensions include disk format/backup, upper and lower case, output to parallel or RS-232 printers, and an 8080 assembler/disassembler. Documentation includes an introduction, glossary, and every single line of the system, including disk drivers and boot up code, all written in Forth!

TRS-80 fig-FORTH for $49.95 Add 2.50 for shipping. Ca. res. add 6% for tax. Order from:

NAUTILUS SYSTEMS ✓55

P.O. Box 1098, Santa Cruz Ca. 95061
TRS-80 is a trademark of Tandy Corp.

FINANCIAL ANALYSIS

Our Financial Analysis System is menu driven for easy user control. The Case Method approach to documentation shows how each program in the system can be applied to solving typical financial problems including:

- Net Present Value Analysis
- Present value of single deposit
- Present value of annuity
- Future value of single deposit
- Future value of annuity
- Break-Even Analysis
- Growth Stock Valuation
- Bond Analysis
- Days Between Dates
- Amortization Schedules
- Sum-of-the-years depreciation
- Straight line depreciation
- Declining balance depreciation
- Monthly payment to amortize a loan
- Number of payments to amortize a loan
- Balloon Payment
- IRR

Minimum hardware requirements:
TRS-80* 16K, 1 Disk, Level II Basic
Please specify Model I or Model II.

Send check or M.O. for $59.95 to CYBERWARE, 3608 Wildgrove, Arlington, TX 76017

Credit card customers order
TOLL FREE 800-227-1617 ext. 403
California Residents call 800-772-3545 ext. 403
✓83

Texas residents add 5% sales tax
***TRS-80 is a trademark of the Radio Shack Division of Tandy Corporation.**

cyberware

```
17185    67     is 17194
17186    40
17187    0      Line 40
17188    84     T
17189    213    =
17190    88     X
17191    207    *
17192    88     X
17193    0      eol
17194    61     Next pointer
17195    67     is 17213
17196    50
17197    0      Line 50
17198    178    PRINT
17199    88     X
17200    59     ;
17201    34     "
17202    88     X
17203    34     "
17204    59     ;
17205    88     X
17206    59     ;
17207    34     "
17208    61     =
17209    34     "
17210    59     ;
17211    84     T
17212    0      eol
17213    68     Next pointer
17214    67     is 17220
17215    60
17216    0      Line 60
17217    135    NEXT
17218    88     X
17219    0      eol
17220    83     Next pointer
17221    67     is 17235
17222    70
17223    0      Line 70
17224    177    POKE
17225    49     1
17226    55     7
17227    49     1
17228    57     9
17229    49     1
17230    44     ,
17231    50     2
17232    48     0
17233    55     7
17234    0      eol
17235    89     Next pointer
17236    67     is 17241
17237    80
17238    0      Line 80
17239    128    END
17240    0      eol
17241    0      Next pointer
17242    0      is 0 (end of
                program)
```

the January 1980 issue and you will see that the codes in the POKE statements correspond to * and +. The trick is to find out where in memory to POKE. Don't worry, it's simpler than learning BASIC.

First, decide what you want to do with POKE, and write your program using POKE00000,000 inserts for proper spacing. If you use more than 16K memory, refer to the Level II manual about the method to addressing above the 16K boundary. Load in Program Listing 4 either manually or using the CFETCH, and run it by typing RUN 65400 ENTER. It will ask you which line you wish the RAM address for. Mark this down and all the others you will be POKEing into. (All numbers we're using are decimal.)

Use the immediate mode of the computer to examine blocks of memory, for instance:

```
FOR X = 17129 TO 17168:
PRINT X;PEEK(X),:NEXT
```

will list the first 40 memory contents of a program on the screen. The listing shown is of the memory contents of Listing 3.

To find the value of a pointer or a line number, the second half must be multiplied by 256 and added to the first half for a decimal equivalent. All other characters and numbers are stored in their ASCII equivalent, one to a memory byte, and all operators and reserved words are coded according to the chart in the article. Find the addresses you have to POKE into and edit your program. Do not increase the size of the program or you may POKE into the wrong character. If you want to POKE in a number of sequential bytes, use FOR-NEXT loops reading from data statements in order to save space.

You could conceivably POKE in your program, but that's what the computer does during a CLOAD.

When you get a self-changing program up and running, use one of the un-number routines to protect it further, and challenge a friend to explain how you did it.■

SAVE
on add-ons for
TRS-80®
Software & Hardware

NEW TOLL FREE ORDER NUMBER

THE MICROCONDUCTOR™
the ultimate data base manager for your TRS-80®

The MICROCONDUCTOR™ is not just a file manager but a true Data Base Management System suitable for both the novice and professional users.

DATA FILES—No limit on the number of records a file can have.
FIELDS—Any type (string, interger, single, double). Eight entry modes (including defaults, counting, and suppress).
REPORTS—Four ways to generate reports. Total numeric column(s). Print on any paper in any format (statements, labels, etc.).
SORT—Any field(s) (i.e. multiple-key sort). Any size file, numeric or ASCII. Ascending or descending.
MAINT.—Command anticipation. Record duplication, Direct access and sequential search.
UPDATING/MERGING—Add, subtract, multiply, divide fields. Combine results from previous calculations. Test for any condition and take action.

TRS-80® MODEL I $249
TRS-80® MODEL II $399

MONTHLY SPECIAL
Shugart SA801 Bare Drive **$459**

DISK DRIVE SALE
SHUGART SA400
TF-3
with power supply and chassis
full warranty (35TK)

ONLY **$299** LIMITED QUANTITIES

NEW FOR TRS-80®

TF-8 80 TRACK DISK DRIVE by MPI
Double your capacity. Single head mini floppy. More than 200K bytes of storage. Complete with power supply and chassis.
$629

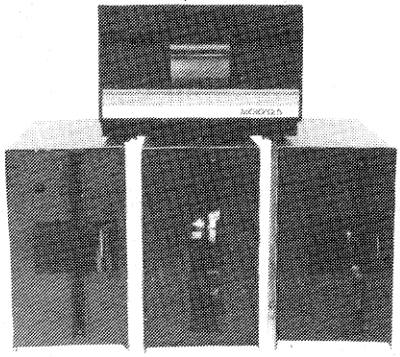
TF-9 DUAL 80 TRACK DISK DRIVE by MPI
Quadruple your capacity. 400K bytes of storage (like having 4-40 track drives in one unit). Complete with power supply and chassis.
$779

Disk Drive Sale!
Complete with power supply and chassis.

Shugart SA400L, 40 trk.	$339
Pertec FD200, 40 trk.	$379
TF-5 MPI B51, 40 trk.	$359
TF-7 Micropolis 77 track	$569
TDH-1 Dual sided drive, 35 track	$469
TF-3 Drive sys. 2 Shugart 40 trk.	$658

SOFTWARE SALE

NEWDOS 80	$149
NEWDOS+ 40 trk. $110 30 trk.	$99
AJA Business Pkg.	$359
The Source	$100
Basic Compiler	$195



® Registered trademark of Radio Shack

Disk Expansion System

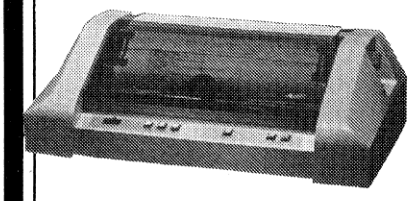
- 2 Shugart SA400 TF-3 ..... $658
- 1 Two-Drive Cable $26
- 1 Expansion Interface 32K $459
- 1 35-track DOS+ $99

TOTAL LIST PRICE $1242

SPECIAL PRICE ONLY **$1,149**

MOD II 8" Disk System

- 1 Drive System $949
- 2 Drive Expansion System $1,445



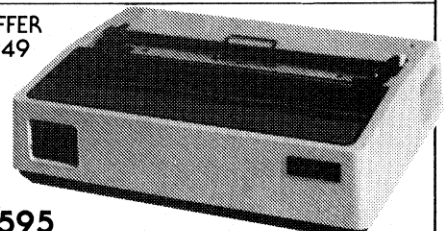
Anadex 9500 .... $1,449

Printers

Microline 83	$1,060
Centronics 779	$1,069
Centronics 700	$1,129
Base 2	$649
Centronics 737	$839
Centronics 702-9	$1,995
Malibu	$2,093
Paper Tiger	$895
Daisy Wheel	$1,799

INTRODUCTORY OFFER
SAVE $300 LIST $949

Okidata Microline 80
$629



EPSON PRINTER **$595**

★BARE DRIVES FOR ANY MICROCOMPUTER

Pertec FD200	$282	FD250	$359
Shugart SA400 35 track	$269	SA400L 40 track	$279
MPI B51	$279	B52	$349
MPI B91	$399	B92	$525

Memory Kit (16K)	$45.00
AC Isolator (6 socket)	$49.95
Disk Head Cleaner	$19.95
Diskettes Verbatim (10) hard ring	$30.00

MICROCOMPUTER TECHNOLOGY INCORPORATED

3304 W. MacArthur
Santa Ana, CA 92704
(714) 979-9923

Order Desk Only 800-854-7222
Telex #678401TAB IRIN

VISA MasterCard

ALL PRICES CASH DISCOUNTED
FREIGHT FOB FACTORY
ASK FOR FREE CATALOG

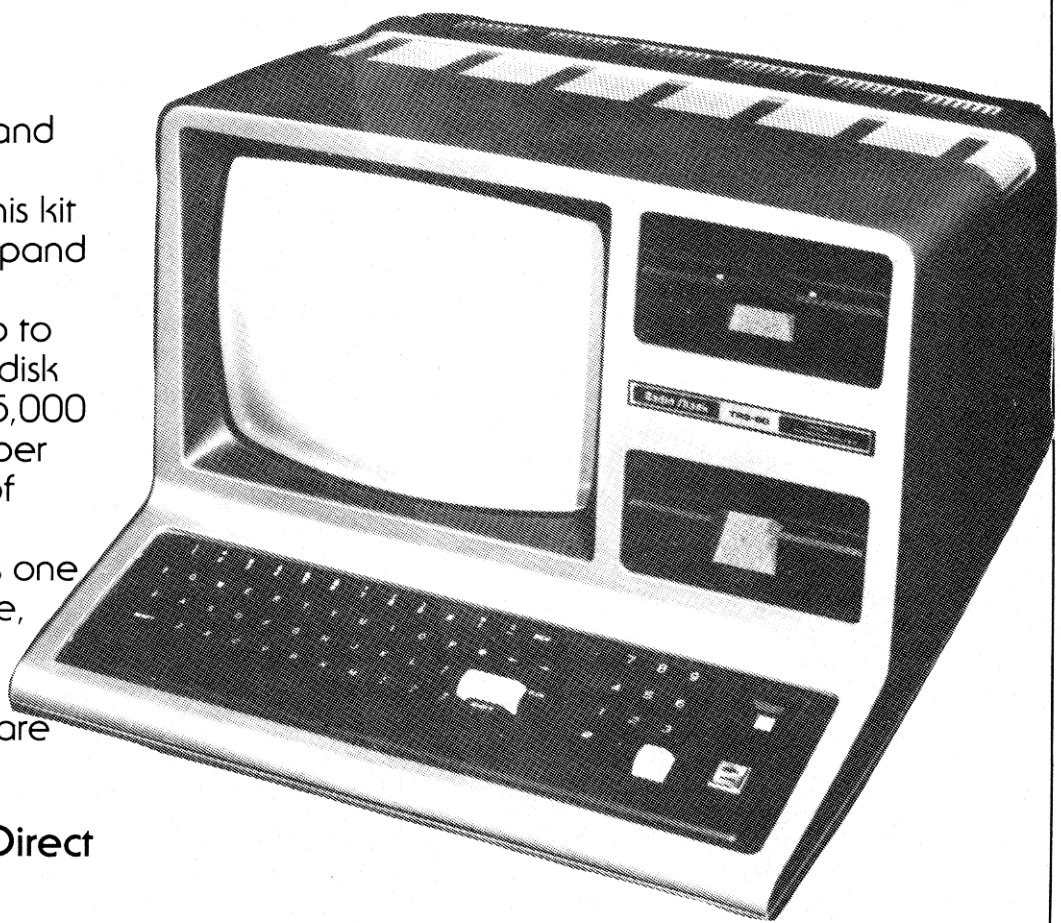
TRS-80® Model III Owners . . .

Why Wait?

Expand Your System Now

New low cost internal disk kit available to expand your storage capability. With this kit now you may expand your Model III computer with up to four MTI 40 track disk drives, giving 175,000 bytes of storage per drive for a total of 700K.

The kit includes one 40 track disk drive, controller, power supply, cables, mounting hardware and installation documentation.



Manufacturer Direct Price

ONLY $699

COMPLETE SYSTEMS AND COMPONENTS AVAILABLE*

MTI MODEL III SYSTEMS

16K RAM, Model III Basic	$899
2-Disk, 48K	$1949
System Cover	$6.95

*Uses MTI Disk Drives and Memory.

MTI DISK DRIVES FOR MODEL III

Internal Kits	
Disk Drive 1	$699
Disk Drive 2	$279
External Add-on Kits	
Disk Drive 3	$379
Disk Drive 4	$359

Model III DOS & Manual	$21.95
16K RAM Kit	$59
RS-232 Serial Interface	$95

MTI

MICROCOMPUTER TECHNOLOGY, INC.

✓28

3304 W. MACARTHUR, SANTA ANA, CA 92704

★ PHONE (714) 979-9923 ★ TELEX #678401TAB IRIN

Order Desk Only 800-854-7222

® TRS-80® IS A REGISTERED TRADEMARK OF TANDY CORPORATION

MasterCard VISA

ALL PRICES CASH DISCOUNTED
FREIGHT FOB FACTORY
ASK FOR FREE CATALOG

Jamming and cramming with an eye on economy.

El Cheapo Packer

Greg Browne
1515 Jefferson
Alamogordo, NM 88310

Ever needed to branch to a statement in the middle of a multi-statement line? Ever wanted to shorten the loading or execution time? Ever wished you could get *rid* of all those unnecessary line numbers?

After using my TRS-80 for about six months and running into all these situations, I decided that I had to do something.

I could EDIT program lines. I could even renumber them. But, somehow, I couldn't eliminate extra line numbers. I couldn't create new lines using the same program code without a lot of retyping.

Although the EDIT routine in the TRS-80 BASIC interpreter was good at editing a single line, it wouldn't do anything else.

Some Alternatives

You can buy some very good software such as Cottage Software's Packer to pack and unpack your programs, that is, combine lines or separate statements to different lines. There are two problems with this. The first is cost. Most of the available programs cost about $30. The second is a bit subjective. I don't mind using someone else's program, but I prefer to control the logic and flow of my program. Commercial packer/unpacker programs massage your code into neat packages whether you want them to or not.

My routine simply packs two lines together, or breaks one line into two. It doesn't check your logic or change your code, but it saves much retyping time.

If your programming volume is small, and you just want to pack a few lines together or break out a statement from the middle of a line without retyping, then my routine may be ideal for you.

To understand my routine, you should know how a BASIC program is stored in memory. Program Listing 1 is a short program which POKEs solid white graphics characters into every position on the screen, a rather slow white-out program. The same program is shown in Table 1 as it is stored in memory. Packing and unpacking is done by manipulating the "overhead" which the interpreter uses to access the program.

All BASIC programs are stored in the same format, starting at memory location 17129 (decimal). Every line uses five extra bytes of memory to identify and sequence it into the program. Those five bytes store two integers and a line separator. The integers are stored as two decimal bytes in standard (backward) Z-80 format. That is, the least significant byte is first and the most significant byte is second. To obtain the integer equivalent of a two-byte decimal stored number, you add the first byte and 256 times the second. The result is the integer value.

Table 1 illustrates the arrangement of the five extra bytes, four in front and one behind. The first two-byte integer is a pointer to direct the interpreter to the first memory location of the next line. The second two-byte integer is the line number.

The fifth byte, at the end of the line, is the terminator, always zero, which the interpreter uses during execution and to fix the line pointers after the use of the EDIT routine changes the length of a program.

Essentially, the beginning of each line points to the beginning of the next, and so on, until the end of the program. The last line points to a two-byte location which contains the value zero, and that is a signal to the interpreter that it has reached the end of the program. If you want to check this on your TRS-80, enter the sample program as it is shown, spaces and all, and PEEK to see how it corresponds to Table 1.

Packing

Knowing the format and starting point of a program stored in memory, it is simple to alter the arrangement of its lines. To eliminate a line, you use the pointers to jump forward line-by-line, examining each line number (stored in the third and fourth bytes of each line) until the desired line is found.

If the line is not found, the search will eventually point to a next-line pointer with that zero value, signaling that we have searched unsuccessfully to the end of the program. If this happens in my routine, it will say

```
100 FOR I = 15360 TO 16383
200 POKEI,191
300 NEXT I
400 STOP
```

Program Listing 1.

CONTENTS

LINE	MEMORY LOCATION	DECIMAL	INTEGER	EXPLANATION
1	17129-30	(255/66)	17151	Points to next line
	17131-32	(100/0)	100	Line number
	17133-49	var		Program code
	17150	(0)		End-of-line marker
2	17151-52	(11/67)	17163	Points to next line
	17153-54	(200/0)	200	Line number
	17155-61	var		Program code
	17162	(0)		End-of-line marker
3	17163-64	(19/67)	17171	Points to next line
	17165-66	(44/1)	300	Line number
	17167-69	var		Program code
	17170	(0)		End-of-line marker
4	17171-72	(25/67)	17177	Points to next line
	17173-74	(144/1)	400	Line number
	17175	(148)		Program code (for Stop)
	17176	(0)		End-of-line marker
x	17177-78	(0/0)	0	End-of-program marker

Table 1. Program Listing 1 in Memory

LINE? and stop.

However, if the routine finds the line, it has to bridge over the existing pointer, line number and previous line terminator. It does this by POKEing the pointer at the beginning of the disappearing line back to the location of the pointer in the previous line. This causes the interpreter to point right past the packed line to the succeeding line.

To keep the interpreter from ending the new line prematurely (at the old terminator), the routine uses that memory location to store a statement separator by POKEing the value for a colon. To clean things up a bit, the routine converts the four unneeded bytes to blanks.

To illustrate this, assume we are packing line 200 of Program Listing 1. Referring to Table 1 again, we see that to bridge over line 200, we have to POKE the contents of memory locations 17151-52 back into locations 17129-30. Then we convert location 17150 to a colon and change locations 17151-54 into blanks.

You now have a combined line, but you are still wasting space with four useless blanks. If you now EDIT the new line, you can remove those blanks. Or, in a few cases, correct the logic. Merging may require adding a statement (such as an ELSE) or removing one (such as a DATA or colon) in order to retain the logic and syntax of the program.

If no additions are necessary, you can save four bytes of memory with every line you pack. That may not seem like much, but you will be surprised at the number of unneeded lines in most programs.

Unpacking

Having packed two lines together, you run your program only to discover a ?UL ERROR. That line number is a needed branch point! What do you do? After this happened to me, I wrote a routine to unpack my mistakes.

When you discover that you must branch to a statement in the middle of a line, you can only do it by beginning a numbered line at that statement. You can retype the line beginning at that statement, or, with a little editing and my routine, you can unpack it.

The editing makes room for those five new bytes of memory needed by the interpreter to access the line. Only four bytes need to be added, however, since the colon preceding the statement will be used as the fifth byte. To help the routine find the point of separation, the colon needs to be changed to a marker which is any keyboard character (except a comma or a colon) not used in the line up to the separation point.

Insert four blank spaces after the marker, and let the routine do the rest. It searches for the line to be separated; then searches through the line for the marker.

If it fails to find the marker, it says MARKER? and stops. If it does find the marker, it changes it to a zero (terminator) and puts the value stored at the beginning of the unseparated line (which points at the next line) into the next two added bytes. It

CONTENTS

LINE	MEMORY LOCATION	DECIMAL	INTEGER	EXPLANATION
1	17129-30	(11/67)	17163	Points to next line
	17131-32	(100/0)	100	Line number
	17133-49	var		Program code (part)
	17150	(38)		Separation marker (&)
	17151-52	(32/32)		First two blanks (Pointer)
	17153-54	(32/32)		Second two blanks (line #)
	17155-61	var		Program code (to separate)
	17162	(0)		End-of-line marker
2	17163-64	(19/67)	17171	Points to next line
	17165-66	(44/1)	300	Line number
	etc.			

Table 2. Portion of Program Listing 2 in Memory

✓ Reader Service—see page 290

```
100 FOR I = 15360 TO 16383&    POKEI, 191
300 NEXT I
400 STOP
```

Program Listing 2

also stores the new line beginning point back in the old pointer location, and finally stores its line number in the last two new bytes.

To illustrate, look at Program Listing 2, our sample program with line 200 packed onto the end of line 100. It has the marker and four blanks inserted at the separation point. Table 2 shows the memory representation of the first part of this program.

The unpacking is done by POKEing the values in locations 17129-30 into locations 17151-52 (immediately behind the marker) and loading locations 17129-30 with location 17151 in two-byte decimal form. The marker is made to a terminator (zero). The new line number is loaded into 17153-54 in two-byte decimal form, and it's done.

Cautions

To keep the routine small, no fancy expansion or compacting features were included. I did little logic checking. The routine is, after all, designed for me to use in conjuction with my RENUM and EDIT functions of the TRS-80 interpreter, and I do logic checking.

You can blow up a program with this routine, so I suggest a few cautions:

- The routine doesn't know what is being packed and will blindly stuff incompatible statements together on a line.
- The routine will pack two very long lines together. This is fine until you list or run a program with a line of code longer than the buffer.
- The routine deletes referenced lines, creating ?UL errors.
- The routine cannot pack the first line of a program.
- When unpacking lines, the routine will accept an out-of-sequence line number.
- If your marker for unpacking is not unique, the line will be unpacked in the wrong place.
- When unpacking, the routine won't check the four bytes behind the marker. It will destroy whatever is there, if you forget to insert blanks.

Running the Routine

The routine prompts for input, and, when packing, will loop continuously until stopped. Since unpacking requires some advance preparation of the line, the routine must be rerun for each line to be unpacked.

How about applying this routine to a program already written? I keep a copy of this routine on tape, numbered (as the listing is) beginning with line 32760. These numbers are high enough to allow me to use that over discussed PEEK-POKE method of merging programs.

This is done by fooling the interpreter about the beginning location of a program during CLOAD. For anyone who is still unfamiliar with this method, Table 3 reviews the mechanics.

I attach this cheap packer to all my programs during development and load my copy of RENUM at the same time. For those of you with a 16K tape system who hate to go through the

1) CLOAD the program to be worked on. (Make sure all line numbers are lower than the packer routine.)
2) PEEK(16633) —if 2 or greater, follow steps 3 and 5.
—if 1 or zero, follow steps 4 and 5.
3) Type POKE 16548,PEEK(16633) – 2:POKE 16549,PEEK(16634):CLOAD
4) Type POKE 16548,PEEK(16633) + 254:POKE 16549,PEEK(16634) – 1:CLOAD
5) Immediately after CLOAD finishes, type POKE 16548,233:POKE 16549,66

Table 3. Merging Tape Programs

commands needed to call RENUM, try the following line of code:

```
I = 16803:POKE I, 195 POKE I + 1,76:
POKE I + 2:124
```

After this, you can get directly to RENUM by typing LINE. I leave it to you to discover why.

The program is formatted to make it easy to study and copy. Every word of code is necessary, but spaces and line feeds are extra. They can be left out. Without them the routine uses 474 bytes of memory, numbered as in the listing.

The routine command number is RUN 32760. You can bypass this by inserting a temporary first line in the code GO TO 32760. Then, if you renumber the program, the routine is still easy to run. The command becomes, simply, RUN. However, to run the program, the command would be RUNnn, where nn is the first line after your temporary line.

You can find other uses, too for the Cheap Packer. I was able to use the routine and some variable table assignments to reduce the computers's response time in my Othello game from 32 seconds to less than 21 seconds per move! ■

```
32000 '          *************************
                 *   'CHEAP'  PACKER     *
                 *         BY            *
                 *      GREG BROWNE      *
                 *************************

32090 '* * * * * * * * * * * * * * * * * * * * * * * * * * *
       *
       *         PROGRAM OPERATION NOTES
       *
32100 '*
       * 32760   DEFINE VARIABLES : SET PRIOR LINE POINTER
       *         LOC'N TO 0 (P) : SET CURRENT LOC'N TO
       *         17129 (C) : PACK OR UNPACK? (A):
32110 '*        PACK = GOTO LINE 32762
       *         UNPACK = CONTINUE ON THIS LINE
32120 '*        GET LINE TO SPLIT (O) : GET NEW LINE NUMBER
       *         TO ADD (N) : GET POSITION MARKER (P$) :
       *         GET ASCII VALUE OF P$ (M) : CONVERT NEW LINE
       *         NUMBER TO TWO-BYTE FORM (B1 = MSB - B2 = LSB)
32130 '*        GO TO THE SEARCH SUBROUTINE AT LINE 32764
32140 '*
       * 32761   SEARCH LINE FOR MARKER (M) :
       *         NO MARKER = END
       *         MARKER    = MOVE UP, POKE POINTER TO NEXT LINE
32150 '*                     HERE, POKE THIS LOCATION BACK TO
       *                     START OF LINE, POKE LINE NUMBER
32160 '*                     INTO THIRD AND FOURTH BYTES BEYOND
       *                     MARKER, CONVERT MARKER TO ZERO,
       *                     END.
32170 '*
       * 32762   GET LINE TO PACK (O) : END IF 0 : OTHERWISE
       *         GO TO THE SEARCH SUBROUTINE AT LINE 32764
32180 '*
       * 32763   STOP IF FOUND LINE IS FIRST LINE : OTHERWISE
       *         POKE TO PREVIOUS POINTER CURRENT POINTER
       *         CONTENTS - (C) TO (P) AND (C+1) TO (P+1) :
32190 '*        POKE A COLON TO PREVIOUS LINE TERMINATOR :
       *         POKE BLANKS OVER FORMER POINTER AND LINE
       *         NUMBER : REINITIALIZE VARIABLES AND DO IT AGAIN
32200 '*
       * 32764   SEARCH SUBROUTINE
       *         GET CURRENT LINE NUMBER (L) : FIND LOCATION
       *         OF NEXT LINE (F) : IF F = 0 (END OF PROGRAM)
       *         BEG OFF AND STOP
32210 '*        IF LINE NUMBER IS ONE SEARCHED FOR - RETURN
       *         OTHERWISE MOVE UP AND TRY NEXT LINE.
       * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
32760 DEFINT A-Z:P=0:C=17129:
      INPUT"(0=PACK ,1=UNPACK)";A:
      IF A=0 THEN 32762
         ELSE INPUT"SPLIT LINE #";O:INPUT"NEW LINE #";N:
              INPUT"POSITION MARKER";P$:
              M=ASC(P$):B1=N/256:B2=N-B1*256:
              GOSUB 32764

32761 FOR I=C+4 TO F:
      IF PEEK(I)<>M NEXT:PRINT"MARKER?":END
         ELSE I=I+1:POKE I,PEEK(C):POKE I+1,PEEK(C+1):
              J=I/256:POKE C+1,J:POKE C,I-J*256:
              POKE I+2,B2:POKE I+3,B1:POKE I-1,0:END

32762 INPUT"LINE TO DELETE (0=END)";O:
      IF O=0 END
         ELSE GOSUB 32764

32763 IF P=0 STOP
         ELSE POKE P,PEEK(C):POKE P+1,PEEK(C+1):POKE C-1,58:
              FOR I=C TO C+3:POKE I,32:NEXT:
              P=0:C=17129:GOTO 32762

32764 L=PEEK(C+2)+256*PEEK(C+3):F=PEEK(C)+256*PEEK(C+1):
      IF F=0 PRINT"LINE?":STOP
         ELSE IF L=O THEN RETURN
                 ELSE P=C:C=F:GOTO 32764
```

Program Listing 3. Cheap Packer

✓ Reader Service—see page 290

POSTMAN DATA HANDLER

Ver. 1.0 – by Fred LaForest

A **machine language** mailing list program that will do:

- 650 Labels on a 35 track disk drive
- 1534 lables on an 80 track drive
- 10 fields (2 user defined)
- FAST SORTS 500 records in 30 seconds (use any or all keys in any order)
- Fully usable on a one (1) drive system (capacities shown are for a single drive system)
- Any label stock ½" thru 1½" vertical (single label horizontal)
- Print one label or a sequence of labels
- **Purge duplicates** with or without user assistance
- 9 digit zip code
- Fast search on any field - random access - 3 second average
- Easy screen for fast editing
- REQUIRES MIN. 1 DRIVE and 32K OF MEMORY, TRS-80*, MOD I

This program is now available in 2 different packages.

1) A **sample package** that does all the functions of the full system (except the purge) and sells for $25 and is to be used as a sales tool only. This is a fully operational package but can not be enlarged or modified in any way. Comes with the complete documentation and credit can be issued to the real package if returned to its place of purchase within 20 days.

2) The **full program** that includes the PURGE function with full documentation. This package will be updated as time goes on with new ideas so it includes a registration card.

Now it even works on Newdos-80

List Price $125.00

Send $25 for Sample Package – if not everything you expected, return sample disk for full refund (less shipping). You can't lose!!

THE CREATOR

The CREATOR is a new type of program for the microcomputer operator. Yes operator!! Easy enough for the person just getting into the market. Use and create a program that is very sophisticated that programmers will comment highly about. The program will create error free basic programming code. Not almost ready to run BUT READY TO RUN WHEN YOU ARE FINISHED. YES, gives birth to a program. Just answer simple questions and have a simple background in the disk system of your computer (if you read your basic manual when you have questions you will have no problems). THIS PROGRAM IS NOT A DATA BASE!!

Now in the package comes the report generator that is in the same concept as the CREATOR. It is called REPORTER. This program creates report output for the CREATOR for either screen or printer.

These 2 programs are on one diskette and are available for only $295 complete. The system requirements are one of the below:

TRS-80 MOD I, 32K DISK
TRS-80 MOD II, 64K DISK
APPLE II, 32K DISK

This is the most outstanding programming package available from anywhere. Now you can create INVENTORY SYSTEMS, PAYABLES and RECEIVABLES, CHECK REGISTER and EXPENSE REGISTER, and MUCH MORE!

This package is ready for delivery only $295 for any one of the systems above (PLEASE STATE SYSTEM WHEN ORDERING).

10% OFF Save 10% this month Software Only!!

SUPER-UTILITY

by K. Watt

— MAIN PROGRAM LIST —

ZAP UTILITY
Display Sector (Disk, File)
Display Memory
Compare Disk Sectors
Copy Disk Sectors
Verify Disk Sectors
Zero Disk Sectors
String Search
Sector Search

PURGE UTILITY
Kill Selected Files
Get Disk Directory
Zero Unused Directory Entries
Zero Unused Granules
Remove System Files
Kill By Category
Change Name, Date, Password, Auto Command
Change File Paramaters
Remove Passwords

DISK FORMAT UTILITY
Standard Format
Format Without Erase
Special Format
Read Address Marks

DISK COPY UTILITY
Standard Copy With Format
Standard Copy Without Format
Special Copy (to back up any protected disks) Purchaser Use - Only for his own personal disks.

TAPE COPY UTILITY
This program is to make back-up of any TRS-80 tape, no matter how it is recorded (note again this program is for the use of the original purchaser for his own programs only)

DISK REPAIR UTILITY
Repair Gat Table
Repair Hit Table
Repair Boot
Read Protect Directory Track
Recover Killed Files
Check Directory

MEMORY UTILITY
Move Memory
Exchange Memory
Compare Memory
Zero Memory
Test Memory
Input Byte From Port
Output Byte To Port
Memory To Disk
Disk To Memory

— For TRS-80, MOD I —

For a more complete overview, send a self addressed stamped envelope. This program is sold on disk only and retails for $49.95.

OTHER THINGS OF INTEREST

For The TRS-80* MOD I

ST-80 III the top of the line in communication packages, MOD I.

List	Cash Discount
$150.00	$137.00

LEX-II MODEM answer/orig.

List	Cash Discount
$179.00	$139.00

SINGLE 5¼ DISK DRIVES 40 Track with Power Supply and Case.

List	Cash Discount
$399.00	$299.00

DUAL 5¼ DISK DRIVES 40 Track with Power Supply and Case.

List	Cash Discount
$799.00	$579.00

80 TRACK DRIVES with Power Supply and Case.

List	Cash Discount
$650.00	$410.00

DUAL DRIVES 80 Track with Power Supply and Case.

List	Cash Discount
$1100.00	$800.00

All hardware must be pre-paid to receive cash discount or Use your Master Card and Visa

Dealer inquiries invited. **SSM SOFT SECTOR MARKETING, INCORPORATED** ✓434 **Dealer inquiries invited.**

6250 Middlebelt • Garden City, MI 48135 • 1 (313) 425-4020

C.O.D. – Certified Check, M.O. or Cash only. Sorry no C.O.D. over $150.00! Most orders shipped next day. All orders must have shipping included. Please add 2% or $2.50, which ever is higher for shipping. Michigan residents, please add 4% tax. Add extra $1.50 for C.O.D. Personal checks take 3 weeks to clear. Send $1.00 for catalog - get $2.00 credit on next order.

An innovative word processing system for TRS-80* MOD I

©1980 by David Welsh

It is time to put your word processing program away and use a Word Processing System

*Lazy Writer is the product of ABC Sales

LAZY WRITER Takes on Scripsit© by Radio Shack® and Electric Pencil©®

Has all the things that other word processing programs should have. Easy to use, written all in machine code. / It permits the inserting and deleting by characters, words, sentences, and paragraphs / Page scrolling up and down / Search ahead of the cursor or behind the cursor for any character / The cursor can be moved up, down, left and right / You can seek top of file and bottom of file / Block move of text, block delete of text/ Search and replace or search delete / Unlimited insert (to the limit of your machines memory) / Permits use with lower case /

Has things that other programs should have, but don't. Upper and lower case output to your printer (if your printer accepts lower case) without having your computer modified. ON UPPER CASE ONLY MACHINES: This program marks the capital letters so you can see which letters are CAPITALS and which are not. / Will change all upper characters text to lower case or all lower case to upper, A SINGLE COMMAND / Will capitalize the first letter of all sentences and all proper noun I's, WITH A SNGLE COMMAND / LOADS ANY ELECTRIC PENCIL / FILE, ASCII SAVED FILES, EDTASM FILES or BASIC PROGRAMS SAVED ASCII / Permits installing special control characters in your text for your printers special features, like double wide or condensed print / Definable screen length and definable print length to 255 characters wide / Screen editing that is not final till your command. This means that you can edit your file on the screen and if you don't like how it reads you can cancel and leave it the way it was / You can append files (which means that you can put one file to the end of another file) / No lost characters at the end of the line, even for the fastest typist / A directory of all your files is available to the user without leaving the program / Saving programs to disk easy enough for the non-computer user / To save memory, not all the program modules are in memory at one time but are called from the disk as needed / You can set tab positions like on a typewriter / 10 CUSTOM COMMAND KEYS for the experienced user there is a command file that permits many special functions that are all user defined (not enough space for better explanation in ad, send for complete overview) / Program has HELP file that is a short review of the commands that are available /

Standard Printer Module. This printer module is provided for the user as a standard feature. Optional special printer routines for custom printer will be available in the near future. In this original release, it has the following printer drivers and will support the following printing devices: RS232, TRS232 and PARALLEL printer ports. You have the following format commands: Justifies Text, Centers Text, Centers Title, Line Spacing, Line Length from 3-255 characters and Set Margins / Also send any ASCII code to any printer from the text / Save formatted text to the disk for spooling later / Information for customer to load his own special printer driver / Printing can be stopped and started by the user at any time and then restarted where you left off / You can print entire file or just print to bottom of the page /

Communication Package. RS232 COMMUNICATION TERMINAL PROGRAM permits you to communicate with other computers. Transfer files from one machine to another. Permits dumping memory across the phone lines. Receive files from other TRS-80's and "Shake Hands" with larger computers. This is the complete system called LAZY WRITER. There is no package written for the TRS-80* that is as comprehensive. This package is available for the TRS-80* MOD I, 32K or larger with at least a single disk drive. List price is from

$125.00

Dealer Inquiries Invited

VISA master charge

✓434 **SSM** SOFT SECTOR MARKETING, INCORPORATED
6250 Middlebelt • Garden City, MI 48135
(313) 425-4020

C.O.D - Certified Check, M.O. or Cash only. Sorry, no C.O.D. over $150.00! Most orders shipped next day. All orders must have shipping included. Please add 2% or $2.50, which ever is higher for shipping. Michigan residents, please add 4% tax. Add extra $1.50 for C.O.D. Personal checks take 3 weeks to clear. Send $1.00 for catalog - get $2.00 credit on next order.

Future Plans

Some additions will be for no additional charge and others will be at optional cost.

Complete support for Spinwriter, Diablo, Qume & 737 Centronics Printers, to use all of the features these printers offer.

Hanging Indents

Newsletters for users of this system to keep them informed of any questions, problems and special products made for use with this system.

Modify this package for TRS MOD III

Program to do form letters.

This is the queueing simulator you've been patiently waiting for.

Line Up!

John M. Anderson
149 Cliffside Drive
Wilmington, NC 28403

Have you ever chosen the wrong checkout line in the grocery store? Or had your patience strained while waiting in line for gas?

How does the proprietor decide on the number of check-out stands with all the variables and considerations?

One approach would be to simulate checkout activities using a computer and observe the results of the simulation.

This article describes a program in Level II BASIC for the TRS-80 with 16K. It demonstrates the dynamics of queueing using the mechanics of simulation.

The Models

Two queueing models, each with variations, can be simulated. Model I allows you to choose up to three channels, each with a single server. Arrivals enter the system at random intervals, followed by exponential distribution. You must specify average arrival time of the variables. Fig. 1 shows the general form of Model I. Each of the servers processes its customers at random service times. Service times follow the exponential distribution which is specified when you enter an average time.

With this model, you can observe the decisions of the system as to how many servers to provide and how the queues could be controlled.

Model II demonstrates a single channel, multi-server system. You can choose two or three servers in that channel. As with Model I, you can specify the particular exponential distributions. For example, interarrival times and service times are specified by entering an average arrival interval for the system and average service times for the servers. An arrival will not leave the system until it has passed through each of the specified service stations in sequence. By varying the number of servers and the service time distributions, you can observe the response of the system to a particular arrival interval distribution. Fig. 2 shows the general form of Model II.

Clock Control

Both models synchronize the arrivals and departures using a fixed-time incrementing approach. With this method, the clock is advanced in one-time-unit increments.

Two options are available for controlling the clock: user control, and computer control. Under user control, the clock is increased by one unit each time you press a non-numeric key on the keyboard. This mode is particularly useful since it allows the activities of the system to be viewed in a stop-frame attitude, showing the incremental changes in all points of the system.

Clock incrementing can also be left to the computer, in which case the simulation will operate automatically. If you begin with manual incrementing and later want automatic incrementing, the mode can be changed by simply depressing a numeric key. Getting from the automatic incrementing mode to the manual incrementing mode is not allowed by the program. Since the computer is under your complete control, the program can be stopped at any time and resumed at the touch of a key.

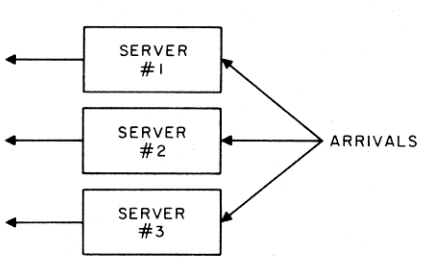


Figure 1

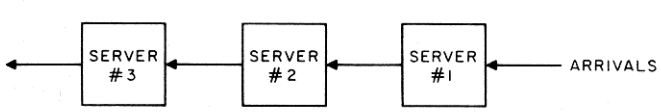


Figure 2

Queue Control

As arrivals enter the system, they are assigned to servers or queues. This is done by using one of three methods: (1) By user control; (2) randomly by the computer; and (3) using the shortest queue rule. Using method (1) has an advantage, it allows you to experiment. You have total control over the assignment of arrivals. Method (2) has the computer assign the arrivals at random, with each queue having the same chance of being chosen. For those who like to keep things balanced, the computer will assign arrivals to the shortest queue under method (3).

Service Rule

All arrivals are served on a first-come-first-serve basis. No provision is made for other service rules.

Video Display

The key feature of this program is the video display unit which displays the simulation's dynamics. Photo 1 shows the screen layout after the clock has run for 20 time units on a Model I system configuration. The system defined for this simulation had an average interarrival time of two time units. Three channels were specified, each having a single server. The servers had average service times specified at five, six, and eight time units, respectively. The clock was incremented automatically by the computer and halted after 20 periods.

Notice the unbalanced queues

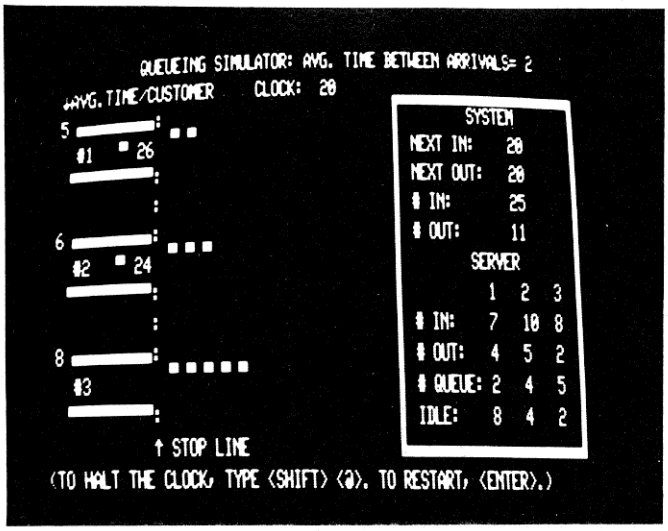


Photo 1. Queueing Simulator

in Photo 1 where the arrivals are assigned to queues randomly by the computer. In this case you expect the line to be longer at service station three because server three averages longer service times than the other two servers.

A scorebox displays the changing statistics during the simulation. Data is provided both for the system and for each channel. At the bottom of the scorebox, the entries labeled idle represent the time units which a service station stood empty. There will always be a few idle periods at the beginning of the simulation while the system waits for the first few customers to arrive. After running for a while, those entries will depend upon relationships between the service times and the queue assignment method chosen.

One note on the queue length may be of interest. When a particular queue reaches 12 customers in length, a second queue is started for that service station. Any channel will handle up to 24 customers in its queue. The 25th customer overloads the channel and the program stops after printing an overload message. This feature exists because of limited screen space. However, it is not a problem since you will be aware of the status of the system before a queue reaches that overload value.

With this program, you can try different system configurations and observe how the system responds for a variety of interarrival and service time distributions. That will help you understand why systems behave as they do. However, there is no guarantee that understanding the problem will lead to a solution. ■

Program Listing

```
10 CLS:PRINT@22,"QUEUEING SIMULATOR
                          BY J.M. ANDERSON":PRINT
   " "
20 PRINT"   THE FOLLOWING PROGRAM DEMONSTRATES THE DYNA
   MICS OF QUEUEING  USING THE MECHANICS OF SIMULATIO
   N.  THE MODEL ALLOWS YOU TO     CHOOSE THE AVERAGE
    ARRIVAL INTERVAL AND THE AVERAGE SERVICE     TIME
   S FOR UP TO THREE CHANNELS.";
30 PRINT"  PRESS <ENTER> TO CONTINUE.":GOSUB990
40 PRINT" ":PRINT"   yOU CAN ALSO CHOOSE THE METHOD FOR
    CONTROLLING THE CLOCK:    EITHER YOU  CONTROL ITS
    ADVANCE, OR THE COMPUTER WILL ADVANCE   IT AUTOMA
   TICALLY.":GOSUB990
50 PRINT" ":PRINT"   FINALLY, YOU CAN CHOOSE THE METHOD
    FOR DIRECTING THE ARRIVALSTO THEIR QUEUES: AT YOU
   R COMMAND, AT RANDOM, OR USING THE       'SHORTEST
    QUEUE' RULE.":GOSUB990
60 CLS:PRINT:PRINT"   INTERARRIVAL TIMES AND SERVICE TI
   MES ARE GENERATED AT RANDOM FROM EXPONENTIAL DISTR
   IBUTIONS.  YOU SPECIFY THE AVERAGE NUMBER OF ARRIV
```

Program continues

✔ Reader Service—see page 290

```
     ALS IN A GIVEN TIME PERIOD AND THE AVERAGE NUMBER
     OF    ARRIVALS SERVICED IN THAT TIME PERIOD ";
 70 PRINT"FOR EACH SERVER.":GOSUB990
 80 PRINT:PRINT"    IF YOU CHOOSE A SINGLE-CHANNEL MODEL,
     YOU HAVE ANOTHER       OPTION: YOU CAN CHOOSE TO
     SEQUENCE MULTIPLE SERVERS IN THAT     CHANNEL (UP
     TO 3), OR YOU CAN LOOK ONLY AT THE SIMPLE SINGLE
      SERVER MODEL.":GOSUB990
 90 DIMT(3,24)
100 CLS:PRINT"HOW MANY CHANNELS (UP TO 3)";:INPUTN:IF(N
    >3)OR(N<1)THEN100
110 IFN<>1THEN140  ELSEN=1:INPUT"YOU HAVE CHOSEN A SING
    LE-CHANNEL MODEL. DO YOU WANT MULTIPLE    SERVERS
    IN SEQUENCE (Y OR N)";A$:PRINT" "
120 IFA$="Y"THENINPUT"HOW MANY SERVERS (2 OR 3)";Z1:IF(
    Z1<2)OR(Z1>3)THEN120  ELSEPRINT"YOU HAVE CHOSEN TO
     SEQUENCE"Z1"SERVERS IN A SINGLE CHANNEL. THE  DIS
    PLAY WILL SHOW THE SERVERS ONE BELOW THE OTHER.  T
    HE OUTPUT  FROM #1 WILL BE INPUT TO #2";
130 IFA$<>"Y"THEN140  ELSEPRINT", THE OUTPUT FROM #2 WI
    LL BE THE     INPUT TO #3, AND THE OUTPUT FROM #3
    WILL LEAVE THE SYSTEM."
140 PRINT"WHAT IS THE AVERAGE TIME BETWEEN ARRIVALS";:I
    NPUTA1:IFA1<=0THEN140
150 IFA$="Y"THENN=Z1
160 FORI=1TON
170 PRINT"WHAT IS THE AVERAGE TIME TO SERVICE A CUSTOME
    R FOR SERVER #"I;:INPUTR(I):IFR(I)<=0THEN170
180 NEXT
190 CLS:PRINT"<ENTER> YOUR CHOICE FOR CONTROLLING THE C
    LOCK:                          1 = AT YOUR COMMAND
                                       2 = A
    UTOMATICALLY."
200 INPUTC1:IF(C1<1)OR(C1>2)THEN190
210 IFC1=1THENPRINT"PRESS <ENTER> TO ADVANCE THE CLOCK.
                                SHIFT TO AUTOMATIC BY
     PRESSING ANY NONZERO NUMBER.":GOSUB990
220 IFC1=2THENPRINT"THE CLOCK WILL ADVANCE AUTOMATICALL
    Y."
230 IFN=Z1THEN270  ELSECLS:PRINT"<ENTER> YOUR CHOICE FOR
 ASSIGNING ARRIVALS TO QUEUES:  1 = YOU CHOOSE THE QUEUE
                                2 = RANDOM ASSIGNMENT
                                3 = SHORTEST QUEUE."@
240 INPUTC2:IF(C2<1)OR(C2>3)THEN240  ELSEC2=C2:IFC2=1TH
    ENPRINT"YOU WILL BE ASKED 'WHICH QUEUE?' UPON EACH
     ARRIVAL.            PRESS <1>, <2>, OR <3>  TO I
    NDICATE YOUR CHOICE OF QUEUE.":GOSUB990
250 IFC2=2THENPRINT"QUEUE ASSIGNMENTS WILL BE MADE RAND
    OMLY BY THE COMPUTER.":GOSUB990
260 IFC2=3THENPRINT"ARRIVALS WILL BE ASSIGNED TO THE SH
    ORTEST QUEUE.  IN CASE OF A  TIE, THE FIRST OF THE
     SHORTER QUEUES WILL BE ASSIGNED.":GOSUB990
270 CLS:PRINT@10,"QUEUEING SIMULATOR: AVG. TIME BETWEEN
     ARRIVALS=";A1;:PRINT@88,"CLOCK:";:PRINT@65,CHR$(9
    2);"AVG.TIME/CUSTOMER";:ONNGOTO300  ,290  ,280
280 FORI=1TO20:SET(I+4,31):SET(I+4,36):NEXT:PRINT@707,"
    #3";
290 FORI=1TO20:SET(I+4,19):SET(I+4,24):NEXT:PRINT@451,"
    #2";
300 FORI=1TO20:SET(I+4,7):SET(I+4,12):NEXT:PRINT@195,"#
    1";
310 PRINT@844,"[ STOP LINE";:FORI=140TO780STEP64:PRINT@
    I,":";:NEXT
320 FORI=82TO126:SET(I,5):SET(I,40):NEXT:FORI=5TO40:SET
    (82,I):SET(126,I):NEXTI
330 PRINT@178,"SYSTEM";:PRINT@235,"NEXT IN:";:PRINT@299
    ,"NEXT OUT:";:PRINT@363,"# IN:";:PRINT@427,"# OUT:
    ";:PRINT@498,"SERVER";:PRINT@564,"1   2   3";
340 PRINT@619,"# IN:";:PRINT@683,"# OUT:";:PRINT@747,"#
     QUEUE:";:PRINT@811,"IDLE:";
350 PRINT@896,"(TO HALT THE CLOCK, TYPE <SHIFT> <@>. TO
     RESTART, <ENTER>.)";
360 FORI=1TON:PRINT@128+256*(I-1),R(I);:NEXT
370 GOSUB1040 :PRINT@246,A;:IFA=0THEN370
380 N1=0:N2=0:B1=0:Q1=0:PRINT@0,"";:FORI=1TON:IFS(I)=0T
    HENL(I)=L(I)+1
390 NEXT
400 IFC1=1THENGOSUB990
410 C=C+1:PRINT@95,C;
420 IFC=ATHENGOSUB760
430 IFC<>ATHEN550
440 AA=A:GOSUB1040 :A=A+AA
450 FORI=1TO24:IFT(Q1,I)=0THEN470
460 NEXT:GOTO980
470 GOSUB1100
480 IFI=1THENT(Q1,I)=A+X8:GOTO500
490 T(Q1,I)=T(Q1,I-1)+X8
500 S(Q1)=1
510 Q(Q1)=Q(Q1)+1
520 J1=J1+1:J(Q1)=J(Q1)+1
530 GOSUB850  :IFC=ATHENQ1=0:N1=0:N2=0:B1=0:GOTO420
540 Q1=0:N1=0:N2=0
550 FORI=1TON:IFT(I,1)=CTHEN570
560 NEXT:GOTO640
570 Q(I)=Q(I)-1
580 IFQ(I)=0THENS(I)=0
590 PRINT@200+256*(I-1),CHR$(128);"     ";
600 FORI1=1TO23:T(I,I1)=T(I,I1+1):NEXT:T(I,24)=0
610 B1=I:  :J2=J2+1:K(B1)=K(B1)+1:GOSUB850  :Q1=0:N1=0:N
    2=0
620 IFZ1<>0THENQ1=B1+1:IFQ1>NTHEN550  ELSEGOTO450
630 GOTO550
640 PRINT@246,A;:M=999:FORI=1TON:IF(T(I,1)<M)AND(T(I,1)
    <>0)THENM=T(I,1)
650 NEXT:IFM<>999THENPRINT@310,M;
660 IFN=Z1THENPRINT@374,J(1);:GOTO680
670 PRINT@374,J1;
680 IFN=Z1THENPRINT@438,K(N);:GOTO700
690 PRINT@438,J2;
700 B=1:FORI=627TO635STEP4:PRINT@I,J(B);:B=B+1:NEXT
710 B=1:FORI=691TO699STEP4:PRINT@I,K(B);:B=B+1:NEXT
720 B=1:FORI=755TO763STEP4:IFQ(B)>0THENPRINT@I,Q(B)-1;
730 B=B+1:NEXT
```

Program continues

```
740 B=1:FORI=819TO827STEP4:PRINT@I,L(B);:B=B+1:NEXT
750 GOTO380
760 Q1=0:N1=0:N2=0:IF(N=1)OR(N=Z1)THEN790  ELSEONC2GOTO
     770  ,790  ,810
770 FORV1=1TO5:FORV=1TO30:PRINT@334,"           ";:NEXT
     :PRINT@334,"WHICH QUEUE?";:FORV=1TO30:NEXT:NEXT
780 Z$=INKEY$:IFZ$=""THEN780  ELSEQ1=VAL(Z$):IF(Q1<1)OR
     (Q1>N)THEN780  ELSEPRINT@334,"                ";:R
     ETURN
790 Q1=INT(RND(N)):IFN=Z1THENQ1=1
800 RETURN
810 W =9999:FORU=1TON:IFQ(U)>=WTHEN830
820 W=Q(U):Q1=U
830 NEXT:RETURN
840 RETURN
850 IFQ1=0THENQ1=B1
860 IFQ1=1THENN1=Q(1)-1:GOTO890
870 IFQ1=2THENN1=Q(2)-1:GOTO890
880 N1=Q(3)-1
890 IFN1>=12THENN2=N1-12:N1=12
900 IFN2>12THEN980
910 IFN1<0THENPRINT@200+256*(Q1-1),"    ";:RETURN
920 PRINT@200+256*(Q1-1),CHR$(131);T(Q1,1);
930 IFN1=0THENPRINT@142+256*(Q1-1),"          ";:RETURN
940 FORI=142+256*(Q1-1)TO142+256*(Q1-1)+2*N1-1STEP2:PRI
     NT@I,CHR$(176);
950 NEXT:PRINT@I,"   ";
960 IFN2=0THENPRINT@206+256*(Q1-1),"  ";:RETURN
970 FORI=206+256*(Q1-1)TO206+256*(Q1-1)+2*N2-1STEP2:PRI
     NT@I,CHR$(140);:NEXT:PRINT@I,"    ";:RETURN
980 FORI=1TO10:PRINT@579,"
          ";:FORV=1TO50:NEXT:PRINT@579,"******** QUE
     UE OUT OF BOUNDS ********";:FORV=1TO100:NEXTV:FORV
     =1TO50:NEXT:NEXT:PRINT@832,"";:END
990 Y$=INKEY$:IFY$=""THEN990
1000 IFVAL(Y$)<>0THENC1=2
1010 RETURN
1020 Z$=INKEY$:IFZ$=""THEN1020
1030 RETURN
1040 A6=1/A1
1050 X6=0:S6=0:K6=RND(0)
1060 S6=S6+A6*EXP(-A6*X6)
1070 IFS6>K6THEN1090
1080 X6=X6+1:GOTO1060
1090 A=X6:RETURN
1100 A8=1/R(Q1)
1110 X8=0:S8=0:K8=RND(0)
1120 S8=S8+A8*EXP(-A8*X8)
1130 IFS8>K8THEN1150
1140 X8=X8+1:GOTO1120
1150 RETURN
```

How to use your interrupts.

Doing Two Things at Once

Buzz Gorsky
712 Hillside Dr.
Carlisle, PA 17013

I have two programs which I use with amateur satellite operation. One derives antenna headings for satellite passes which can be displayed on the screen, printed out or used to control the antenna directly. The other is for a QSO file, to provide me with names, locations, time and satellites of previous contacts. Unfortunately, my computer has been unable to perform the antenna tracking at the same time it runs the QSO file program.

I realize that there are other time-dependent tasks, such as manipulating electric lights and intrusion alarms, during which the computer would not be available for other work. There should be a better way—and there is.

This article will describe how I solved this problem. The technique should work well for a variety of similar tasks. I will discuss the TRS-80 with expansion interface, which has resident "heartbeat," disk and a real time clock program. Those without a disk system may use the clock with their own software.

Resident Clock Program

My first task was to learn about the resident clock program, get into it, and add routines. It works on interrupts to tell when the clock display is to be updated. If interrupt routines were entered, I could do other tasks in addition to updating a display. Then, I would check the time to begin the antenna tracking task and turn the antenna rotor off or on.

I planned to have my BASIC program POKE values into a machine language program to specify the timing, while the machine program worked on interrupts. I could then run the QSO file program while tracking.

I unravelled the real time clock using information from the expansion interface and TRSDOS manuals from Radio Shack. Address 4012H would jump or call to an interrupt routine, and show that the clock routine begins at 4518H. I found a RETURN instruction in the machine program, and tried to modify the program just before any returns.

I wrote a routine to update the clock "backwards" so that the hours are updated instead of the seconds (and vice versa). Thus, I could quickly see if my modifications worked. None did!

I had one other bit of information—address 4040H has the heartbeat count; 4041H, the seconds; 4042H, the minutes; and 4043H, the hours. I looked through the machine instructions after 4018H to see if any of the values were shown and found that at 45BCH there are: 21 40 41, which is LD HL, 4041H.

I then wrote a program to substitute a JUMP to my program in these three addresses (45BCH-45BEH), and ended my routine by loading HL with 4041H and then jumping to 45BFH, the next instruction in the clock routine. That worked!

I then arranged my BASIC program to POKE the hour and minute for antenna tracking and then the number of seconds the rotor is to run during each successive minute while tracking. A zero value indicates that the tracking is accomplished and the clock is restored to normal. Program Listing 1 shows the pertinent parts of the BASIC program.

This program involves POKEing values derived in BASIC. After the machine is called, the CMD'R' restarts the clock by enabling the interrupts (which the machine language program had disabled). This re-enabling could be done by a machine code, but I did it in BASIC so that during debugging I could do some PEEKing with the clock off.

Program Listing 2 shows the assembly language program which does the work in sequential modules. When first executed from DOS, the entire program loads and execution begins at line 220. There, the location 4049H gets an address one lower than the address where it starts. That will protect it from BASIC and avoid the MEMORY SIZE question.

After that, we go back to DOS; execution from DOS does little except load the program and protect it. I then call BASIC and execute the BASIC program.

In line 750, the machine language program is called, beginning at location FE49H, or line 250. Here the interrupts are dis-

```
700  DEFUSR0 = &HFE49:REM ADR OF MACHINE LANG ROUTINE
710  POKE &HFE00,H: REM HR WHEN TRACKING BEGINS
720  POKE &HFE01,M: REM MIN WHEN TRACKING BEGINS
730  FOR I = 1 to N: POKE &HFE01 + I,D(I): NEXT: REM FE02 AND ADJACENT
     HOLD THE D(I) OR VALUES FOR # OF SECS ROTOR IS TO BE ON EACH
     SUBSEQUENT MINUTE WHEN TRACKING BEGINS
740  POKE &FE01 + I, 0 :REM 0 IN NEXT ADR INDICATES END OF TASK
750  X = USR(0): CMD"R"
```

Program Listing 1

abled, and we substitute a JUMP to FE56H. This is the address labeled TIME in line 310, for the instruction which had existed in the clock routine at 45BCH. When the clock is re-enabled and gets to 45BCH, it will go to this program starting at line 310. After that substitution the RET in line 300 returns to the BASIC program where the interrupts are re-enabled, starting the clock again.

As the clock runs, the routine starting at 310 is entered every second. AF is saved and the current hour and minute are compared with the values stored in HR and MIN by the BASIC program. If there is no match, then EXIT is begun (line 670). HL is loaded here with 4041H, the step which belongs in 45BCH, and there is a JUMP to 45BFH to resume the clock.

When there is a match, it is time to begin the antenna tracking task. In NOW, line 410, we substitute the address for RUN, FE6EH, into the clock routine where we had previously stored FE45H. Each time the clock runs it will jump to RUN instead of to TIME. It is time to RUN this part of the program and to continue right along. The POP and PUSH of AF in 430 and 440 are done because RUN must save AF each time it is entered from the clock.

We must then POP out the value previously saved so that it can be saved on the first time through the RUN routine. Once that is done we look to see if this is the 0 second of a minute. If not, we return to the clock via CONT. If it is second 0, then we turn the antenna rotor on (the hardware is controlled by the IN commands) and put the correct TIME into HL from the sequence of values stored by the BASIC POKEs.

This value is now stored in TIMOF and we look to see if it is 0. If so, the task is done and we jump back where the clock is restored to normal functioning. Otherwise we load HL with 4041H, the current second count, and continue with CONT to see if this second matches that stored in TIMOF, which would indicate that the rotor should be turned off. If not, we EXIT, otherwise TIMUP will turn the rotor off with the IN command; increment NEXT, and go to EXIT to return to the clock.

Operation of the system is easy. When I know that I will want antenna tracking done, I use the TIME and CLOCK instructions from DOS to set the correct time and start the clock. I then execute the assembly language program, go to BASIC, execute the satellite tracking program, and continue with any other work I'm doing. The program will continue monitoring the time, and antenna tracking will begin on time. For example, I can load and execute my BASIC QSO file program and working the satellite, while the computer is controlling the antenna.

Since the clock is inaccurate when disk functions are used, extensive use of disk presents the only problem with this scheme. Aside from that, this is the perfect way to get the machine to do two things at once. ■

```
                00100 ;ROUTINE TO ENTER CLOCK ROUTINE
                00110 ;AT SPECIFIED TIME AND THEN ROTATE ANTENNA
                00120 ;CORRECT AMOUNT EACH MINUTE
                00130 ;VALUES ASSOCIATED WITH BASIC PROGRAM
                00140 ;BY BUZZ GORSKY, K8BG
FE00            00150           ORG     0FE00H
0001            00160 HR        DEFS    1       ;STORES HR TO START
0001            00170 MIN       DEFS    1       ;STORES MIN TO START
003C            00180 TIMES     DEFS    60      ;STORES # SECS FOR ROTOR ON
                00190                           ;EACH MIN
FE3E 00         00200 NEXT      DEFB    0
FE3F 00         00210 TIMOF     DEFB    0
FE40 21FFFD     00220 BEGIN     LD      HL,HR-1
FE43 224940     00230           LD      (4049H),HL      ;PROTECT FROM BASIC
FE46 C32D40     00240           JP      402DH           ;BACK TO DOS
FE49 F3         00250 START     DI
FE4A 21BC45     00260           LD      HL,45BCH
FE4D 36C3       00270           LD      (HL),0C3H       ;JUMP INTO CLOCK ROUTINE
FE4F 2156FE     00280           LD      HL,TIME
FE52 22BD45     00290           LD      (45BDH),HL
FE55 C9         00300           RET     ;BACK TO BASIC PROGRAM
FE56 214340     00310 TIME      LD      HL,4043H        ;CURRENT HR
FE59 F5         00320           PUSH    AF
FE5A 3A00FE     00330           LD      A,(HR)
FE5D BE         00340           CP      (HL)
FE5E 2039       00350           JR      NZ,EXIT
FE60 2B         00360           DEC     HL              ;CURRENT MIN
FE61 3A01FE     00370           LD      A,(MIN)
FE64 BE         00380           CP      (HL)
FE65 2032       00390           JR      NZ,EXIT
                00400
FE67 216EFE     00410 NOW       LD      HL,RUN
FE6A 22BD45     00420           LD      (45BDH),HL
FE6D F1         00430           POP     AF
FE6E F5         00440 RUN       PUSH    AF
FE6F 214140     00450           LD      HL,4041H        ;CURRENT SECONDS
FE72 3E00       00460           LD      A,0
FE74 BE         00470           CP      (HL)
FE75 2014       00480           JR      NZ,CONT
FE77 DB04       00490           IN      A,(4)           ;ROTOR ON
FE79 3A3EFE     00500           LD      A,(NEXT)
FE7C 2102FE     00510           LD      HL,TIMES
FE7F 85         00520           ADD     A,L
FE80 6F         00530           LD      L,A
FE81 7E         00540           LD      A,(HL)
FE82 323FFE     00550           LD      (TIMOF),A
FE85 B7         00560           OR      A               ;DONE IF A IS 0
FE86 2818       00570           JR      Z,BACK
FE88 214140     00580           LD      HL,4041H        ;CURRENT SECONDS
FE8B 3A3FFE     00590 CONT      LD      A,(TIMOF)
FE8E BE         00600           CP      (HL)
FE8F 2802       00610           JR      Z,TIMUP
FE91 1806       00620           JR      EXIT
                00630
FE93 DB03       00640 TIMUP     IN      A,(3)   ;TURN ROTOR OFF
FE95 213EFE     00650           LD      HL,NEXT
FE98 34         00660           INC     (HL)
FE99 214140     00670 EXIT      LD      HL,4041H        ;BACK TO CLOCK
FE9C F1         00680           POP     AF
FE9D C3BF45     00690           JP      45BFH
                00700
FEA0 21BC45     00710 BACK      LD      HL,45BCH
FEA3 3621       00720           LD      (HL),21H
FEA5 DB03       00730           IN      A,(3)   ;TURN ROTOR OFF
FEA7 214140     00740           LD      HL,4041H
FEAA 22BD45     00750           LD      (45BDH),HL
FEAD F1         00760           POP     AF
FEAE C3BC45     00770           JP      45BCH   ;BACK TO RESTORED CLOCK
                00780
FE40            00790           END     BEGIN
00000 TOTAL ERRORS
```

Program Listing 2

Searching for Strings in your Basic program.

Find it Fast!

James Yowell Yelvington
811 East Will Rogers Drive
Stillwater, OK 74074

When you write a BASIC program is it always carefully planned in advance, neatly structured and flowcharted, and documented completely?

Seated at the computer one day, I was weary and my fingers wandered over the bouncing keys. Oops! Sorry, Adelaide Procter, but "A Lost Chord" (poem published in *Legends and Lyrics*, 1858) was not my problem. It was, instead, lost variable references, instructions, text strings, and other important information, mislaid somewhere in my BASIC program. What I needed was a machine language program which would quickly scan an entire BASIC program and find whatever I lost.

Searching for character strings is something computers do naturally, though the process may be time-consuming. BASIC statements are encoded before being placed in the BASIC program storage area, making it necessary to encode the string being searched for, before scanning the BASIC program.

I hoped I wouldn't have to write a routine to encode the search string, since there clearly had to be such a routine already in ROM. I looked through *Supermap* (Fuller Software, 1979) and *Disassembled Handbook for the TRS-80* (Richcraft Engineering, 1980), identified some likely spots, and checked them out. I found that by loading the HL register pair with the starting address of the string and calling 1BC0H in ROM, I ended up with the string encoded, the HL register pair pointing to one less than the starting address of the encoded string, and a terminating byte of zero following the encoded string.

I decided to load the A register with the first character of the search string and to run through the BASIC program area comparing each character with the A register until I found a match. I then loaded the second character of the string into A, advanced one location in the BASIC program, and looked for another match. I continued until I reached the end of the string, whereupon I knew I had found the entire string. Then I displayed the line number corresponding to the find, went back and loaded A again with the first character of the string, backed up in the BASIC program to one location past the start of the find, and repeated the whole process. If I encountered a mismatch before reaching the end of the string, I would not display the line number.

To get the line number, I had to know the format used to store BASIC programs in memory. Each line of BASIC begins with a two-byte pointer to the beginning of the next line, followed by a two-byte line number, the statements themselves, and a single byte of zero. In the search routine, I skipped over the pointers and stored the line numbers in a temporary holding place while the text of the line was being searched. The end of a program was signaled by two bytes of zero where the pointer to the next line should be.

The task of displaying the line number, with the necessary conversion to ASCII, is another I hoped to avoid. So, back to *Supermap* to find a ROM routine. The likeliest spot was at 0FAFH, but I didn't know how to use it. Prior experience with other ROM routines suggested that the HL register pair would be the key one to prepare. Should HL point to the line number or contain it? Since line numbers only require two bytes, I tried loading HL with the line number and then calling 0FAFH. It worked.

After further experimentation I found that a string embedded within a keyword would not be recognized by the search routine. For example, the string IN would not be recognized where it occurred in the keyword PRINT. The reason was that the word PRINT did not exist in the encoded program, having been replaced by the corresponding code symbol or token. Also, the string JOE AND MARY would

not be recognized in a line such as PRINT "HELLO, JOE AND MARY." In the BASIC line the word AND, enclosed in quotes, would not be encoded. The same principle applies to shifted (capital) letters. When enclosed in quotes they remain shifted. Otherwise they are unshifted, whether in the string or in the BASIC program. BASIC comments, too, are left unencoded, whether preceded by REM or the single quote (apostrophe).

I began to feel that some revision of the program would be desirable. Suppose I wanted to search for the word OR where it might occur between quotes?

Two possibilities came to mind. I could enclose search strings in quotes, or I could somehow signal my intentions in the first byte or two of the string. The latter choice seemed most practical, so I looked around the keyboard for a character I would be willing to eliminate as a possibility for the first character of a search string. No character seemed to be dispensable. I decided to use two characters together to signal that the search string was not to be encoded.

I chose the double ampersand (&&). I modified the search string input/encoding routine, so that if the string began with two ampersands, they would not be considered part of the string, but would cause the string not to be encoded. The only limitation was that it would not be possible to search for an encoded string which began with two ampersands.

I turned to consider how best to call the routine into execution. Using the SYSTEM command followed by a slash (/) and the entry address would be the usual way, but a more convenient method would be a DOS command. I placed an instruction at the beginning of the program which loaded the program's entry address into the vector for the FIELD command.

The vector, located at 417CH, is three bytes long and normally contains C3 2D 01 in non-disk systems. (C3 is the jump instruction and the next two bytes form the inverted address of the L3 error routine.) By substituting the entry address of the search routine for that of the L3 error routine, I could call up the search routine simply by entering the direct BASIC command FIELD.

It was necessary to enter the routine the first time through the SYSTEM and slash method. The FIELD method worked reliably as long as the system was not reset and no disk or Level III system was operational. Then I figured out a better way.

A Better Way

Remembering that the ROM keyboard input routine could be intercepted for special processing, I set up a single keystroke call to the search program. While I was at it, why not debounce the keyboard? I added instructions at the head of the program to load the keyboard vector (4016H-4017H) with the entry address of a special keyboard processing routine, which I inserted in front of the search routine.

In the keyboard routine I loaded the BC register pair with a timing constant (48) and then called the ROM delay routine at 60H. I next had to test the keyboard input (in the A register) for whatever character I chose as a signal to jump to the search routine.

The program was now complete. After final testing and polishing I saved it on tape and added it to my library of utilities on BASIC programming.

Commentary

The following is a running commentary on the program, from start to finish. The numbers refer to line numbers in the Program Listing.

2-24	These EQU instructions identify special addresses, such as ROM routines and reserved RAM locations.
26	The ORG statement may be changed to locate the program wherever you wish.
28-32	This intercepts the ROM keyboard input routine and causes a call to the special keyboard processing routine at line 34.
34-54	Special keyboard processing routine
34	Call ROM keyboard scan routine.
36	Save A register, containing keyboard input character, from damage by delay routine.
38	Load timing constant. Value 48 may be increased or decreased as necessary to obtain desired debounce effect.
40	Call ROM delay routine.
42	Restore A register.
44	Check for SHIFT plus down arrow.
46	If so, jump to defeat special keyboard routine.
48	Check for SHIFT plus right arrow.
50	If not, return to ROM from special keyboard routine.
52	Keep stack straight. Allows for call from ROM without corresponding return statement.
54	Jump to search routine.
56-60	Restore original keyboard vector, thus ending operation of special keyboard routine (and debounce). Return to BASIC.
62-68	Set up jump to search routine by use of FIELD command. Not functional unless entry is made directly to this point.
70-72	Start of search routine proper. Initialize flag for the NOT FOUND message to be displayed at end of search.
74	Set up to display prompt message (STRING=).
76	Display it.
78-116	Routine to input and check search string.
78	Call ROM string input routine. Gets string into input buffer, ends it with a byte of zero, returns with HL containing address of location one byte before start of input string.
80-82	Point to first byte of string, then load it into A.
84-86	Test A for zero (null string) and jump if so. (Note: OR A takes one byte less than CP 0.)
88-98	Test for double ampersand at beginning of string; take proper action.
100-114	Test for zero (null string) following double ampersand. If so, go to null string routine; otherwise bypass encoding process.
116	Save address of first byte of search string (bypassing double ampersand) at location labeled STR.
118-132	Get address of start of BASIC program from reserved RAM location. Load contents of that address into A and test for zero. If so, advance one location and test its contents for zero. Two bytes of zero at start of BASIC program indicates that no program exists, so jump to display NO BASIC PGM and return to BASIC.
134-136	Back up to one byte before start of program to set up for following search loop.
138-154	Beginning of line search loop. Get first two bytes of line (pointer to next line) and test each for zero. If both zeroes, then end of program is reached, so jump to end search routine. (Note: XOR A sets A register to zero. OR

E and OR D statements test these registers for zero by comparing them with A.)

156-164 Get next two bytes of line (line number) into DE register pair and save at location labeled LIN for display in case search string is found in this line.

166 Beginning of scanning loop (within line search loop) to search text of BASIC line for match with search string. Advance to next byte of BASIC line text. (Note: HL points to position in BASIC line; DE points to position in search string. Register A will hold current character from search string; C will hold character from BASIC line; B will hold displacement of current character from beginning of search string.)

168-170 Initialize B register to zero displacement. Initialize DE to first byte of search string.

172 Beginning of innermost loop to compare BASIC line text, character by character, with search string. Loop is executed as long as consecutive matches are found.

174-176 Test for end of search string (zero byte). If so, a complete match has been found. If not, jump.

178 Save HL and B registers from damage by line number display routine. (Note: EXX statement is faster and more compact than PUSH HL and PUSH BC statements. The C register and the DE pair are also saved, but without need.)

180-182 Load zero into NOT FOUND flag byte, so it won't be displayed.

184-192 Check current line cursor position of video display to avoid breaking line number between two consecutive display lines. If no room, force line feed before displaying line number.

194-200 Display current line number, indicating search string found in this line. Follow by blank to separate displayed line numbers.

202-204 Restore HL and B registers and jump.

206-224 Continuation of loop begun at line 172 where complete match has not yet been found. Load current BASIC character into C.

208-210 Test C register for zero, indicating end of BASIC line. (Note: INC C then DEC C is an efficient way to test C for zero.)

212 If end of BASIC line, jump to start of line search loop (outer loop) for new line.

214-224 If not, compare current BASIC character with current search string character (in C and A registers, respectively). If they are not the same, then jump. If same, then move to next character in BASIC line and in search string. Increment displacement value in B. Repeat innermost loop.

226-234 Continuation of loop begun at line 166 (mid level loop) where mismatch has occurred between string and BASIC characters. Back up to location in BASIC text where successive matches began and repeat mid level loop to continue scanning line.

226-232 An efficient way to subtract displacement in B register from HL pair. Since C register is no longer needed, load B into C, then zero out A register and load into B. Now 16-bit subtraction can be done by SBC HL, BC statement. Carry flag will not interfere because it was reset by XOR A statement.

236-240 Load HL with start of NULL STRING message and call display routine. Then end program by jump back to BASIC.

242-246 Display NO BASIC PGM message and return to BASIC.

248-258 Test NOT FOUND flag. If zero, return to BASIC. If equal to one, display message and then return to BASIC.

260-276 Messages for display on screen. Each is terminated by a byte of zero, as needed by ROM routine at 28A7H.

260 Control codes back up the cursor to start of display line, and then erase to end of line.

278 Defines storage space for start address of search string (DEFS 2 statement could also have been used.)

280 Defines space for line number (DEFS 2 would also work.)

282 Defines space for NOT FOUND flag (DEFS 1 would also work.)

284 Permits start at key-

```
                00001 ;**FINDIT**, J. YELVINGTON, JUNE 1980
0060            00002 DELAY   EQU     60H
032A            00004 SHOCHR  EQU     32AH
0361            00006 INSTR   EQU     361H
03E3            00008 KBSCAN  EQU     3E3H
06CC            00010 RTRN    EQU     6CCH
0FAF            00012 SHOLIN  EQU     0FAFH
1BC0            00014 CODE    EQU     1BC0H
28A7            00016 SHOMSG  EQU     28A7H
4016            00018 KBPTR   EQU     4016H
40A4            00020 BSTP    EQU     40A4H
40A6            00022 CURSP   EQU     40A6H
417C            00024 FIELDP  EQU     417CH
7EE6            00026         ORG     7EE6H
7EE6 21EF7E     00028 KBD     LD      HL,KBIN
7EE9 221640     00030         LD      (KBPTR),HL
7EEC C3CC06     00032         JP      RTRN
7EEF CDE303     00034 KBIN    CALL    KBSCAN
7EF2 6F         00036         LD      L,A
7EF3 013000     00038         LD      BC,48
7EF6 CD6000     00040         CALL    DELAY
7EF9 7D         00042         LD      A,L
7EFA FE1A       00044         CP      26
7EFC 2806       00046         JR      Z,KBNULL
7EFE FE19       00048         CP      25
7F00 C0         00050         RET     NZ
7F01 E1         00052         POP     HL
7F02 1812       00054         JR      FIND2
7F04 21E303     00056 KBNULL  LD      HL,KBSCAN
7F07 221640     00058         LD      (KBPTR),HL
7F0A C9         00060         RET
7F0B 3EC3       00062 FIND1   LD      A,0C3H
7F0D 327C41     00064         LD      (FIELDP),A
7F10 21167F     00066         LD      HL,FIND2
7F13 227D41     00068         LD      (FIELDP+1),HL
7F16 3E01       00070 FIND2   LD      A,1
7F18 32FF7F     00072         LD      (NF),A
7F1B 21CE7F     00074         LD      HL,STRM
7F1E CDA728     00076         CALL    SHOMSG
7F21 CD6103     00078 STRIN   CALL    INSTR
7F24 23         00080         INC     HL
7F25 7E         00082         LD      A,(HL)
7F26 B7         00084         OR      A
7F27 CAAC7F     00086         JP      Z,NOSTR
7F2A FE26       00088         CP      '&'
```

Program continues

UNIQUE !

40 & 80 Track "FLIPPY" Drives for TRS-80*

REAL VALUE

AEROCOMP offers the best value in microcomputer disc drives on the market today! Reliability, features and cost tough to beat. We deliver...and we stand behind our products, as evidenced by the only FREE TRIAL OFFER in the industry. Examine your systems needs and order today!

MYSTERY REMOVED

There appears to be some confusion in the terminology used to describe disc drives and their features. Here's what we mean:

- **FLIPPY** Allows the use of both sides of a diskette with a single-headed drive by simply turning the diskette over (model 40-1&80-1).
- **TRACK DENSITY** Specified in tracks per inch (TPI). Refers to the number of tracks per radial inch on the diskette. Typically 48 TPI=40 usable tracks and 96 TPI=80 usable tracks.
- **DOUBLE DENSITY** Refers to recording density in bits per inch (bpi). Typically single density means data can be recorded up to 2,938 bpi; double density means data can be recourded up to 5,876 bpi.
- **DOUBLE SIDED** Refers to number of read/write heads. Single-sided is one head, read/write one side only; double-sided is dual heads allowing read/write operations on both sides of the diskette. A double sided drive appears as two seperate drives to the controller.
- **ACCESS TIME** The time required for the head to move from one track to the next. Typacilly 5 to 40 milliseconds (ms).

SELECT YOURS!

- **40-Track "FLIPPY" $349.95**
(Model 40-1) 48 TPI, single dens. 125K bytes/side, double dens. 250K/side.
- **80-TRACK "FLIPPY" $459.95**
(Mod. 80-1) 96 TPI, single dens. 250K bytes/side, double dens. 500K/side.
- **40 TRACK DUAL HEAD $459.95**
(Mod. 80-2) 48 TPI, double-sided, 40 tracks/side; single dens. 250K bytes, double dens. 500K.
- **80 TRACK DUAL HEAD $599.95**
(Mod. 160-2) 96 TPI, double-sided, 80 tracks/side; single dens. 500K bytes, double dens. 1 megabytes.

NOTE: All capacity values are unformated. All models capable of single or double density. All with power supply/silver enclosure. 115 VAC, 50-60 Hz. 115/230 VAC available.

SPECIAL PACKAGES

STARTER 40-Track FLIPPY drive,2-Drive cable,TRSDOS 2.3 Disk & Manual,Freight & Ins. $369.95 (reg. $401.00)

#1 40-Track FLIPPY drive
2-Drive cable
Newdos/80
Freight & Ins.$459.00(reg.$528.00)

#2 80-Track FLIPPY drive
2-Drive cable
Newdos/80
Freight & Ins. $569.00 (reg.$638.00)

#3 TWO (2) 40-Track FLIPPY drives
4-Drive cable
Newdos/80
Freight & Ins. $785.00 (REG. $893.00)

#4 TWO (2) 80-Track FLIPPY drives
4-Drive cable
Newdos/80
Freight & Ins. $999.00 (reg.$1113.85)

- **DISK OPERATING SYSTEMS**
Newdos/80 $149.00
VTOS 4.0 $125.00
DOSPLUS 3.1 $99.95

- **DISKETTES, SOFT SECTOR,5 1/4"(box of 10)**
Single-sided, single density $29.95
Double-sided, double density $39.95
- **CABLES**
2-drive $24.95
4-drive $34.95
Extender $16.95

FREE TRIAL OFFER

Order your AEROCOMP Disc Drive and use it with your system for up to 14 days. If you are not satisfied for ANY REASON (except misuse or improper handling), return it, packed in the original shipping container, for a full refund. We have complete confidence in our products and we know you will be satisfied! ORDER TODAY!

WARRANTY

We offer you a 120 day unconditional warrenty on parts and labor against any defect in materials and workmanship. In the event service, for any reason, becomes nescessary, our service department is fast, friendly and cooperative.

100% TESTED

AEROCOMP Disc Drives are completely assembled at the factory and ready to plug in when you receive them. Each drive is 100% bench tested prior to shipment. We even enclose a copy of the test checklist, signed by the test technician, with every drive. AEROCOMP MEANS RELIABILITY!

ORDER NOW!!

To order by mail, specify Model Number(s) of Drive, cable, etc. (above), enclose check, money order, VISA or MASTERCHARGE card number and expiration date, or request C.O.D. shipment. Texas residents add 5% sales tax. Add $5.00 per drive for shipping and handling. Please allow 2 weeks for personel checks to clear our bank. No personel checks will be accepted on C.O.D. shipments-cash, money orders or certified checks only. You will receive a card showing the exact C.O.D. amount before your shipment arrives. Be sure to include your name and shipping address. WE SHIP PROMPTLY! In the event there is a slight delay, you will be notified of the shipping date and we will NOT charge your bankcard until the day we ship!

COMPARE AND BUY AEROCOMP!

	"FLIPPY"	ACCESS TIME (track to track)	HEAD LOAD SOLENOID	DISC EJECTOR	CAPACITY (unformated single density)	EASY-ENTRY DOOR	FREE TRIAL
AEROCOMP	YES	5ms.	YES	YES	250K bytes (both sides)	YES	YES
RADIO SHACK*	NO	40ms.	YES	NO	109K bytes	NO	NO
PERCOM	YES	25ms.	YES	NO	250K bytes (both sides)	YES	NO
MPI	NO	5ms.	YES	YES	125K bytes	YES	NO
SHUGART	NO	40ms.	YES	NO	109K bytes	NO	NO
TANDON	NO	5ms.	NO	NO	125K bytes	NO	NO

Factual material from current manufacturer's data sheets is believed reliable but cannot be guaranteed, comparing Aerocomp Model 40-1 to similar models.

The TRS-80* expansion interface limits the track to track access time to 12ms. *Trademark of Tandy/Radio Shack.

WRITE AEROCOMP TODAY FOR MORE VALUES !!!

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 Applications and Technical information, call (214) 337-4346 or drop us a card.

Dealers inqiries invited

AEROCOMP
Redbird Airport, Bldg. 8
P.O. Box 24829
Dallas, TX 75224 ✓387

✓Reader Service—see page 290

```
7F2C 200F      00090          JR      NZ,ST2
7F2E 23        00092          INC     HL
7F2F 7E        00094          LD      A,(HL)
7F30 FE26      00096          CP      '&'
7F32 2008      00098          JR      NZ,ST1
7F34 23        00100          INC     HL
7F35 7E        00102          LD      A,(HL)
7F36 B7        00104          OR      A
7F37 CAAC7F    00106          JP      Z,NOSTR
7F3A 1805      00108          JR      ST3
7F3C 2B        00110 ST1      DEC     HL
7F3D CDC01B    00112 ST2      CALL    CODE
7F40 23        00114          INC     HL
7F41 22FB7F    00116 ST3      LD      (STR),HL
7F44 2AA440    00118          LD      HL,(BSTP)
7F47 7E        00120          LD      A,(HL)
7F48 B7        00122          OR      A
7F49 2006      00124          JR      NZ,SU
7F4B 23        00126          INC     HL
7F4C 7E        00128          LD      A,(HL)
7F4D B7        00130          OR      A
7F4E 2865      00132          JR      Z,NOPGM
7F50 2B        00134          DEC     HL
7F51 2B        00136 SU       DEC     HL
7F52 23        00138 GETPTR   INC     HL
7F53 5E        00140          LD      E,(HL)
7F54 23        00142          INC     HL
7F55 56        00144          LD      D,(HL)
7F56 AF        00146          XOR     A
7F57 B3        00148          OR      E
7F58 C25F7F    00150          JP      NZ,GETLIN
7F5B B2        00152          OR      D
7F5C CABE7F    00154          JP      Z,ENDBAS
7F5F 23        00156 GETLIN   INC     HL
7F60 5E        00158          LD      E,(HL)
7F61 23        00160          INC     HL
7F62 56        00162          LD      D,(HL)
7F63 ED53FD7F  00164          LD      (LIN),DE
7F67 23        00166 SCAN     INC     HL
7F68 0600      00168          LD      B,0
7F6A ED5BFB7F  00170          LD      DE,(STR)
7F6E 1A        00172 SC1      LD      A,(DE)
7F6F B7        00174          OR      A
7F70 C2947F    00176          JP      NZ,SC3
7F73 D9        00178          EXX
7F74 AF        00180          XOR     A
7F75 32FF7F    00182          LD      (NF),A
7F78 3AA640    00184          LD      A,(CURSP)
7F7B FE39      00186          CP      57
7F7D FA857F    00188          JP      M,SC2
7F80 3E0D      00190          LD      A,13
7F82 CD2A03    00192          CALL    SHOCHR
7F85 2AFD7F    00194 SC2      LD      HL,(LIN)
7F88 CDAF0F    00196          CALL    SHOLIN
7F8B 3E20      00198          LD      A,20H
7F8D CD2A03    00200          CALL    SHOCHR
7F90 D9        00202          EXX
7F91 C3A47F    00204          JP      SC4
7F94 4E        00206 SC3      LD      C,(HL)
7F95 0C        00208          INC     C
7F96 0D        00210          DEC     C
7F97 CA527F    00212          JP      Z,GETPTR
7F9A B9        00214          CP      C
7F9B C2A47F    00216          JP      NZ,SC4
7F9E 04        00218          INC     B
7F9F 23        00220          INC     HL
7FA0 13        00222          INC     DE
7FA1 C36E7F    00224          JP      SC1
7FA4 48        00226 SC4      LD      C,B
7FA5 AF        00228          XOR     A
7FA6 47        00230          LD      B,A
7FA7 ED42      00232          SBC     HL,BC
7FA9 C3677F    00234          JP      SCAN
7FAC 21EF7F    00236 NOSTR    LD      HL,NSTRM
7FAF CDA728    00238          CALL    SHOMSG
7FB2 C3CC06    00240          JP      RTRN
7FB5 21E27F    00242 NOPGM    LD      HL,NPGM
7FB8 CDA728    00244          CALL    SHOMSG
7FBB C3CC06    00246          JP      RTRN
7FBE 3AFF7F    00248 ENDBAS   LD      A,(NF)
7FC1 B7        00250          OR      A
7FC2 CACC06    00252          JP      Z,RTRN
7FC5 21D87F    00254          LD      HL,NFM
7FC8 CDA728    00256          CALL    SHOMSG
7FCB C3CC06    00258          JP      RTRN
7FCE 1D1E      00260 STRM     DEFW    1E1DH
7FD0 53        00262          DEFM    'STRING='
7FD7 00        00264          DEFB    0
7FD8 4E        00266 NFM      DEFM    'NOT FOUND'
7FE1 00        00268          DEFB    0
7FE2 4E        00270 NPGM     DEFM    'NO BASIC PGM'
7FEE 00        00272          DEFB    0
7FEF 4E        00274 NSTRM    DEFM    'NULL STRING'
7FFA 00        00276          DEFB    0
7FFB 0000      00278 STR      DEFW    0
7FFD 0000      00280 LIN      DEFW    0
7FFF 01        00282 NF       DEFB    1
7EE6           00284          END     KBD
00000 TOTAL ERRORS
```

Program Listing

board routine by entering slash (/), without starting address immediately after loading program.

The Search

Keep the following points in mind when using the search program.

1. The program will be assembled to load from 7EE6H (32486 decimal) to 7FFFH (32767), occupying the top 282 bytes of memory in a 16K system. By changing the ORG statement, you may assemble it to load at any location you choose.

2. The keyboard routine at lines 28 to 60 may be removed. (No keyboard debounce or single keystroke access to the routine would then be possible.)

3. Entry into another machine language program which sets up its own special keyboard processing routine will terminate operation of this keyboard routine.

4. The FIELD command access routine at lines 62 to 68 may be removed if you do not wish to access the search routine in this manner. This is advisable if you are also using DOS, Level III, or some other program which uses the DOS command vectors.

5. There are three possible entry points to the program. The first entry, at 7EE6H (32486), sets the keyboard routine to work and returns directly to BASIC. The second, at 7F0BH (32523), sets up the FIELD command access method, then goes into the search routine. The third, at 7F16H (32534), simply executes the search routine.

6. You can have both the keyboard and the FIELD access methods simultaneously operational.

7. If the keyboard routine is in effect, you can call the search routine simply by pressing SHIFT and the right arrow. You should not have problems with bouncing keys. If the keys still bounce, you can increase the value 48 (decimal) in line 38 to a larger value. If the keys seem too sluggish, then you can decrease the value.

8. To defeat the keyboard routine instantly, press SHIFT and the down arrow. To test whether the keyboard routine is in effect, press SHIFT and the right arrow. If the display shifts to large characters, then the keyboard routine is not in effect. Use the CLEAR key to shift back to small characters or, to preserve the screen content, use POKE 16445, 0. If the keyboard routine is in effect, enter the search routine, getting the prompt STRING = . To leave the search routine without conducting a search, press ENTER. You will get the NULL STRING message and be returned to BASIC.

9. The search routine will list a line number as many times as the search string is found in the line. Occurrences of the string may overlap: the string ANA will be found twice in the word BANANA and the string ** will be found three times in ****.

10. When you enter the search string, all punctuation marks, blanks, and special characters (except two leading ampersands) will be considered part of the string.

11. The sequence number of a line cannot be searched for, since the search routine scans only line texts. Any numbers, or parts of numbers, in the text of a line may be searched for, however, since they are maintained there in their ASCII form. This permits tracking down line references in GOTO and GOSUB statements.

12. To search for an unencoded string, enter two ampersands (&&) followed immediately by the string. Do this to search for shifted letters or keywords occurring between quotes in the BASIC program.

13. You can have the entire search string encoded or not, but you cannot have it partially encoded. Two or more separate searches may then be needed.

14. A search string may be as short as a single character or as long as 240 characters. You can count the number of blanks in your program by tallying the number of line references displayed after a search for blanks.

Your personal bloodhound, FINDIT, stands ready to track down almost anything you want to find in your BASIC program. ■

Dragonquest!

In a desperate race against the sun you search for SMAEGOR Monarch of Dragonfolk, who has kidnapped the Princess of the Realm and holds her in a distant and unknown place. In a quest for Honor and glory, you must search the land, seeking out the tools needed for the ultimate confrontation. On The River Delta, in the abandoned Temple of Baathteski, Goddess of the Blade, everywhere, clues abound. But WHERE is the Princess?

Now, as never before, the genius of CHARLES FORSYTHE shines in this new machine language ADVENTURE. DRAGONQUEST! Can YOU save M'lady from the iron clutches of SMAEGOR?

TAPE $15.95 *Dealer Inquiries Invited* **DISK $21.95**

THE PROGRAMMER'S GUILD ✓364
BOX 66, PETERBOROUGH, N.H. 03458
(603) 924-6065 AFTER 6 PM EST
MASTERCHARGE AND VISA ACCEPTED

✓Reader Service—see page 290

Bugs are easier to find than jackalopes, if you use these techniques.

Wild Glitch Hunting

John Warren
East Carolina University
Greenville, NC 27834

Your program finally ran all the way through. You've fixed all the errors—SYNTAX, UL, RETURN WITHOUT GOSUB, TYPE MISMATCH. Now you're ready to wrap it up, send it off to a software house and wait for the royalties to roll in.

No, not yet! Although you've dealt with the more serious errors (they are the easiest to deal with), there are other, more subtle errors. These are what make programmers lose their hair.

The first step in detecting and dealing with hidden bugs is to work the whole thing out on paper, step by step. Use a pocket calculator, but remember to write out all the intermediate results, too, for later reference. Then input identical data into the program and see if you get the same answer.

If you do, try different data. Run through any loops in the program. More likely, on at least one of the tries, you were wildly off. Recheck the math you did by hand. If it's correct, then you're off on a bug hunt.

Bug Hunting

Break the program into a series of boxes like the flowchart—you did write one, didn't you? At points in the program, insert a print statement to display out the values of the variables at that point. This is called an audit trail. Don't skimp on the information you put in the statement. Something like 100 PRINT A,B,D,E may seem straightforward until you try to remember what 23 −45 4 0 stands for in the middle of the printout.

```
100 PRINT"LINE 100 A = ";A;"
B = ";B;"D = ";D;"E = ";E
```

provides a clearer result:

```
LINE 100 A = 23 B = −45 D = 4 E = 0
```

This is why you saved the intermediate results.

Since I write my program lines at intervals of 10, inserting the audit trail on lines that end with nine makes them easier to find when the debugging is complete and they must then be removed. In computers like the TRS-80, where multi-statement lines are permitted, ending the lines with a REM statement like ***** also makes them easy to find:

```
109 PRINT"LINE 100 A = ";A;"B = ";B;"
D = ";D;"E = ";E:'*****
```

You can let the program run normally after the audit trail, or you can let it pause at some of the audit lines by inserting one of two commands. INPUT A$ (make sure that the string variable A$ is not being used in the program) will cause the program to halt until you press ENTER. On the other hand, STOP will stop the program and print out the message, BREAK ON (the line number). To continue, type CONT and press ENTER.

Now compare your results with the audit trail. Hopefully, at some point, you'll disagree with the computer. When that disagreement occurs go to the program and look for an error between that printout and the one before it. Some of the common errors include using the wrong operational sign (+ instead of *, etc.), the wrong variable name or a gosub to the wrong subroutine. A hard-to-detect error is an improperly initialized increment. For example, if N = N + 1 is used to count the number of times control passes through a particular line, N must be initialized at zero. Generally, all variables are initialized at zero when the program is RUN, but this action is sometimes erratic. Also running through a program repeatedly (where it ends with a GOTO to the beginning) will cause an error unless increment is reinitialized.

Immediate Response Mode

The TRS-80 also has a handy immediate response mode that allows you to examine variables and inject new values in the middle of a run. Pressing the BREAK key is equivalent to a STOP command. The program halts, and the line number is displayed. To continue, type CONT and press ENTER. However, this isn't the only option in immediate response. After hitting BREAK, type in a PRINT statement followed by some of the variables. Do not use a line number. For example, ? A,B,C, (? is Level II shorthand for PRINT. Level I users can use P.). When you hit ENTER, the computer will display the values of those variables on the next line. Remember, commas cause the values to be separated by about a quarter of a line, while semicolons allow closer spacing (usually one space between values).

There are two ways to display arrays when debugging. The longer way is to treat each variable independently.

```
? A(1);A(2);A(3)
```

This is feasible if the array is short, or you only want to examine certain segments.

A better approach is to use the TRS-80 multiple statement line capability and build a FOR-NEXT loop.

```
FORX = 1TO10:?X,A(X):NEXT
```

which results in:

```
1 4
2 5
3 8
etc.
```

The first figure is the number of the array element, and the second, the value. Note that since the numbers are printed out on separate lines, only 13 can be seen at once. This can be overcome by using a carriage return suppressor (a comma or a semicolon) or, if you have quick reflexes, by pressing SHIFT @ in Level II which will freeze the display for study. Press any key to continue. Pressing the space bar and holding it freezes the display in Level I.

If you are planning to continue the program after printing out these values, do not use a variable in the FOR-NEXT loop in the program. This will change the value of that variable. My habit is to restrict the variables X, Y and Z to FOR-NEXT loops. Since a loop is self-initializing, the variable's previous value is unimportant.

The immediate response mode can also be used to inject values during a run. Perhaps after BREAKing and examining variable values you find that A is five when it should be three. However, it would be best to check if the rest of the program is okay instead of debugging right away. If you just CONTinued, the incorrect value would mess up the rest of the audit trail.

Instead, type A = 5, press ENTER, and CONTinue. This reinitializes the variable A with the value five. If the rest of the program is OK, the audit trail will be correct.

You can also jump to other areas of the program by directly inputting the command GOTO and the line number. Subroutines can be tested the same way. For example, you have a subroutine at line 1000 that converts a six-character string (A$) into an array of one-character strings (A$(1) to A$(6)). The correct functioning of the subroutine can easily be verified with this immediate response command:

```
A$ = "ABCDEF":GOSUB1000:
FORX = 1TO6:?A$(X):NEXT
```

These procedures operate only if you have not edited, which automatically sets all the variable values to zero.

Inputting

One of the most frustrating parts of debugging is having to input the same data over and over, particularly in the early stages of debugging where it seems that every line has a syntax error. You repair the error, input the data again and the program blows up three lines further on. If there are only two or three INPUTs, this isn't a problem, but when there are dozens, the temptation to skip debugging can get strong.

One way to deal with this problem is to replace all INPUTS with direct assignments. A good practice is to retain the INPUTS, but to insert an apostrophe in front so that the computer will treat them as REM statements. Inserting a flag on the same line as the direct assignment will make it easier to delete, when debugging is finished. Always insert the REM statement flag after the assignment, since the computer will ignore everything on the line following an apostrophe. Multiple statement lines do not work in this context. For example,

```
100 'INPUT A$
109 A$ = "GOOD":'*****
```

Direct assignment does not work when control loops through a line several times. For example, when an array is being loaded. A solution is to replace the INPUT command with a READ command, and load from a series of DATA statements.

Level II's editing function is effective here. Move the cursor to the INPUT, type 5C, then space once and type READ. Do not delete the space. The run isn't affected, and when the time comes to replace READ with INPUT, that space is welcome.

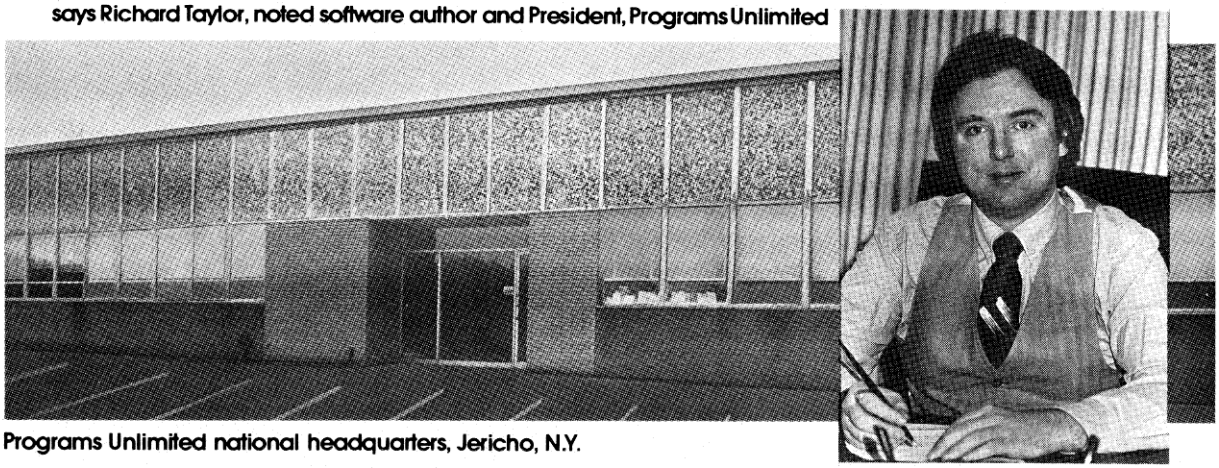
Good programming may be genius, but debugging is just sweat. With the right approach and a little planning, your debugging can be kept down. ■

✓ Reader Service—see page 290

"The ideal software source...

"We couldn't find it.

says Richard Taylor, noted software author and President, Programs Unlimited



Programs Unlimited national headquarters, Jericho, N.Y.

We know what it's like out there. Because that's where we came from.

Before we decided to become software entrepreneurs, we were just like you—micro-computer enthusiasts searching and sifting through myriad magazines for the ideal mail order software source.

What we hoped to find was a single entity that offered an ultra-wide selection of software (and peripherals) for our particular micro-computers. That pre-selected only the best of the many similar-sounding programs that reached the market each month. That could give us personal assistance with the purchase-decision process. And that stood behind its products with a no-holds-barred guarantee.

Isn't that what you've been looking for too?

When we couldn't find it, we decided to become it. And PROGRAMS UNLIMITED was born.

An Information Services Department: To Solve, Not Sell

We're professionally staffed and solidly financed. We set high standards for the programs we carry, whether developed by ourselves or others; if they don't meet our standards, we know they won't meet yours.

We have a computer Bulletin Board to keep you apprised of our latest offerings, and a toll-free number for telephone orders. In addition, we've established an Information Services Department; its primary function is not to sell merchandise, but to provide expert answers—from Richard Taylor and others–to questions you may have about your system.

Our Guarantee is *Really* a Guarantee

We know who we are and what we're up to, buy **you** may not—yet. That's why we give you the best guarantee we can think of: If for any reason you're not satisfied with any PROGRAMS UNLIMITED purchase, return it within ten days for a full, no-hassle refund.

Think we're going overboard? You're right.

We know that the future of our business is **repeat** business. We want you to be satisfied first time, second time, **every** time you buy.

We want to earn the right to be **your** *software source.*

the software source™

PROGRAMS

So we founded it."

Mention This Ad and Save $5.00 on These Introductory Specials!*

THE WIZARD

TRS-80 MODEL I 32K DISK
LATEST FROM AUTHOR
OF TRS-OPERA
RICHARD TAYLOR

Challenge your companions to a fast-paced question and answer game coupling sound effects and non-stop graphics. **THE WIZARD**'s a wiz at assimilating new subjects . . . its built-in utility program lets you create even more subjects to give **THE WIZARD** a mind of your own. Timed responses give each player a chance to steal from their opponents or force other players into answering. Match wits with **THE WIZARD.**

A PROGRAMS UNLIMITED EXCLUSIVE!
REGULARLY $19.95

ORCHESTRA 80

A MUSIC SYNTHESIS SYSTEM
BY JON BOKELMAN
TRS-80 16K LEVEL 2
YOU HAVE TO HEAR IT
TO BELIEVE IT!

Turn your 16K Level II into a high quality musical instrument. The software includes a digital synthesizer, a music language compiler, a full screen editor, a file manager and a voice initialization utility. The hardware is a single 1½" x 2" PC board which plugs into the expansion connector on the keyboard or the screen printer port on the expansion interface. Plugs into any amplifier. Write your own music or join scores of others in trading music.

REGULARLY $79.95
FREE DISK OF MUSIC!

SUPER HOST

TRS-80 MODEL I 32K DISK
BREAKTHROUGH! UNLIMITED
ACCESS TO YOUR TRS-80!

The finest **HOST** program on the market and a PROGRAMS UNLIMITED EXCLUSIVE! — Runs on NEWDOS, NEWDOS-80, TRSDOS — Data-in and Data-out translation tables — Correctly advances the date on all DOS's — Masks out undesirable characters (i.e. BREAK) — Remembers the date and time after reboot — Lowercase driver option — Optional password entry — Automatic disconnect upon loss of carrier —Sets Nulls and Line feeds — Plus, many more outstanding features!

REGULARLY $29.95

DIRINDEX

By Robert Alleger
TRS-80 MODEL I 32K DISK
EASIEST WAY YET TO
ORGANIZE YOUR DISKS!

DIRINDEX is a disk utility program that easily organizes all of your programs stored on disk. **DIRINDEX** runs on ANY DOS and uses machine language routines to read your directories and sort all your files either by diskette number/side or in alphabetical order. Keeping your index updated is simple and fast! Choose from 2 printout formats or use your CRT. A 48K system handles 1200 entries on up to 100 diskettes (both sides). 32K system handles 413 entries/45 diskettes. Runs on one drive.

A PROGRAMS UNLIMITED EXCLUSIVE!

REGULARLY $19.95
(ON DISK)

EPSON MX80 PRINTER

TRS-80 MODEL I/II/III
THE DREAM PRINTER FOR
THE 1980's

This is the printer you have been hearing about! 9 x 9 dot matrix with true descenders. Bi-directional, logic seeking, multiple copy, TRS Graphics, boldface. Software selectable 40, 80, 66 or 132 columns. After 50 million characters you can replace the print head for **under $30!**

All this for **under $650.** It's a printer that could only have come from the world's largest manufacturer of print mechanisms, Epson.

CALL FOR SPECIAL PRICE

TRS-OPERA AND CHALLENGE

TWO ALL-TIME FAVORITES
NOW FOR THE MODEL III

Two of the finest show pieces for the TRS-80 Model I are now available for the MODEL III! Now MODEL III owners can thrill to the sounds of THE WILLIAM TELL OVERTURE as they flow from their cassette cable into any amplifier they wish to use. You will swear you're hearing several tones at the same time.

CHALLENGE is a fun-filled game with SOUND and GRAPHICS which has become a true classic for the MODEL I. 1 or 2 players can **CHALLENGE** each other or the computer in a fast paced real-time guessing game.

REGULARLY $19.95 (BOTH)
ON CASSETTE

ZAP

By JOHN WOLFE
TRS-80 MODEL I 32K DISK
THE BEST OF ALL ZAPS!

SINGLE or DOUBLE DENSITY! AUTO REPEAT while scrolling through sectors OR TRACKS! COMPUTE HASH CODES. DOUBLE FLASHING CURSOR keeps track of byte to modify and the ASCII equivalent. BYTE COUNTER shows your current position in the sector. Compare or copy—sector to sector, memory to sector, sector to memory. All functions HARDCOPY. Rewrite a sector. Dump sectors to memory. Display memory. Toggle between drives. Display FILE. Copy data. Runs on TRSDOS, NEWDOS 2.1 and NEWDOS 80.

A PROGRAMS UNLIMITED EXCLUSIVE!
REGULARLY $29.95

MISSILE ATTACK

TRS-80 16K LEVEL II
DEFEND YOUR CITIES FROM
NUCLEAR DESTRUCTION!

Based on the famous Arcade Game. Super SOUND and REALTIME GRAPHICS make this machine language program a real winner.

. . . You command ICBM strategic site 45B. The Russians have launched an attack on the U.S. Mainland. World War III has begun! Our only hope is to defend our cities as best we can. You are charged with defending three major Metropolitan areas, can you hold up under the pressure and save these three cities?

REGULARLY $19.95
ON CASSETTE

There's Lots More in Our Free Catalog. Ask For Your Copy Now!

HOW TO ORDER:

- **PHONE TOLL FREE (800) 645-6038** (except NY State).
- NY State residents phone collect (orders only) (516) 997-8668. (Telephone hours: 9-5, Monday through Friday).
- To order by computer, phone: (516) 334-3134.
- To order by mail, write: Dept. 381M, Box 265, Jericho, N.Y. 11753.

master charge THE INTERBANK CARD

Add $1.00 for postage and handling. N.Y. residents: add sales tax. *Introductory Specials valid through 4/30/81; subsequent prices subject to change without notice.

VISA*

FULL GUARANTEE

If, within 14 days, you are for any reason dissatisfied with any purchase from PROGRAMS UNLIMITED, return the merchandise in original condition for a prompt, full refund.

UNLIMITED™

✓110

Watch for our Retail Centers opening soon from coast to coast.

✓ Reader Service—see page 290

EDAS EDAS

A sophisticated Editor & Assembler setting the standard for the '80 Model I & Model III. All EDAS commands and SOURCE text can be entered in either upper case or lower case. Direct assembly form memory or disk by means of *GET assembler directives. This gives text buffer capacity equal to your drive configuration! 30,000 bytes of symbol table.

Direct assembly to disk or memory for faster debugging operations! DOS "system" command functions KILL, DIR, FREE, and LIST are available from within the environment of EDAS.

The Editor, with renumber, maintains command syntax identical to the BASIC editor. Global change permits you to alter a string throughout a designated range of lines while block move relocates lines of text.

EDAS is priced at $79 plus $3 S&H. A 72-page manual included.

cmdfil

Now you can append two or more CMD files and/or SYSTEM tapes. Perform transfer to & from disk/tape of SYSTEM/CMD modules with offset capabilities. Read VTOS ISAM overlays. More! $20

MISOSYS SERIOUS SOFTWARE

LDOS is the latest generation of operating systems for the TRS-80 computers. LDOS is completely documented in an extensive operating manual and total on-going support from some of the most knowledgeable people in the industry is provided. In short, LDOS is THE operating system to use with the TRS-80 computers!

$139.00
+$3 S&H

LDOS 5.1

dsmblr

Complement your assembly language tools with this Z80 disassembler which produces screen, printer, cassette, or disk file output. A two-pass process provides SYMBOLS for 16-bit address and 8-bit relative references. EQUates & ORG are generated. Read SYSTEM programs & display load address range. $20 (DSMBLR I for non-disk use is $15)

Specify Model I or Model III

master charge VISA

MISOSYS - Dept MU
5904 Edgehill Drive
Alexandria, Virginia 22303
703-960-2998 MicroNET 70140,310
Dealer Inquiries Invited ✓221

diskmod

Turn your Editor Assembler into a disk package. This 32K patch modifies EDTASM for DOS operation. Features? Add full disk I/O, block move, global change, printer pagination with optional prompting, sorted symbol table, print memory utilization, correct DEFM expansion, protect memory, and recover after BOOT. From within the EDTASM you will have DIR, KILL, & FREE. Upgrade your EDTASM today! Version for EDTASM+ coming soon. $20.

THE BØØK

THE BØØK must be a part of your Z-80 language tools. Volume I gives you access to all math operations in your Level II ROM including ASCII-Binary conversions. Included is a symbol table of the entire machine noting over 500 addresses. Volume II tells you everything you wanted to know about the Level II I/O - printer, keyboard, video, and cassette routines are fully explained. Each volume has a fully-commented listing of all the routines discussed. THE BØØKs will save you hours of assembler program development time. Don't start programming without THE BØØKs. Each volume is priced at $14.95 + $1.50 S&H or buy both for $24.95.

MACHINE LANGUAGE PROGRAMS "BUGGING" YOU??

Debug them -- with C-ALL!!

- GENERAL OPERATION MODE
  machine language tape I/O, user memory and CPU register control, I/O port control, memory operations like fill, block move, and search
- DISASSEMBLY MODE
  dump to screen or screen and printer the assembly language mneumonics for instructions or data of a known format. Includes page formatting and breakpoint control.
- TRACE MODE
  large number of options: default displays four lines of CPU data, trace address, hex code and assembly language mneumonic prior to execution. Delay is 1 second or 1/16 second with shift pressed. Top of CRT displays CPU register and memory data; bottom displays last 7 instructions executed. Optional dump to printer.

C-ALL: JUST $19.95

(Level II or DOS, 32K recommended)

THE ALTERNATIVE SOURCE ✓138

1806 Ada Street
Lansing, MI 48910
517-487-3358
517-485-0344

Add: $2.00/postage, $1.50/COD, 4%/Visa.

VISA/MC Order Line Only (except Mich.) 800-253-4358 ext. 100

FINDISK-II The ultimate in automatic disk indexing with exclusive features ..... Model-I $20.00
AUTOMATICALLY create INDEX of programs or data from all your disks, print disk LABELS, print alphabetized MASTERLIST, do fast SEARCH, add DESCRIPTIONS. Also automatically: detect DATA or SYS disks, PURGE disks and index of old files, and UPDATE from revised disks only.

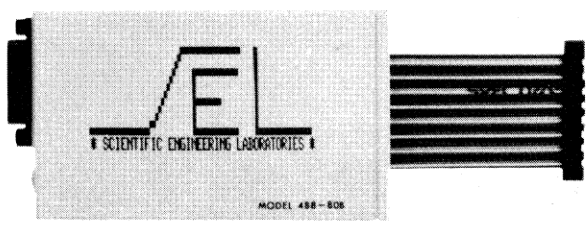
SOLAR-I The critical calculations for passive design... Model-I $30.00 ..... Model-II $45.00
INPUT: any latitude, orientation, slope, roof overhang, storage type, building loss, OUTPUT: solar angles, shading, time, heat gain/loss, percent solar, fuel use. Print report by hour, month, year in presentation format.

RIA-II Complex Real Estate Investment Analysis ..... Model-I $30.00.....Model-II $45.00
Analysis for investor or homeowner using Elwood method. INPUT: Project costs, loan and tax data, expenses, depreciation rate. OUTPUT: Cap rate/value, mortgage payments, before/after tax cash flows, return (IRR), profit/gain from sale over any time series.

DEPRECIATE-I Manage a list of depreciable items ..... Model-I $15.00.....Model-II $20.00
Tracks long list of depreciable items with varying purchase dates, depreciation rates, or per cent business use. Update any time. Print tax form. Used by many CPAs.

STRUCT-I Graphic design of steel/wood beams and moment transfer..... Model-I $15.00
INPUT: span/cantilever, uniform/point loads, beam material. OUTPUT: with screen graphics beam moment & shear diagrams. Print job report with diagram, stresses, and required beam sizes.

Min 32K. On disk (Mod-I one drive order tape). Add $1.00 postage (Mich. add 4% tax) VISA/MC
✓88 **DOCUMAN SOFTWARE BOX 387-A KALAMAZOO, MI 49005 (616) 344-0805**



Everything needed to add *powerful GPIB-488 controller capability* to TRS-80, Model 1, Level 2 or DOS

SCIENTIFIC ENGINEERING LABORATORIES
11 NEIL DRIVE • OLD BETHPAGE, NEW YORK 11804
TELEPHONE (516) 694-3205 ✓291

IEEE-488 to TRS-80* INTERFACE

Mod. 488-80B
$225.00
+ shipping, insurance & tax

SPECIFY DISK OR TAPE

For Model 3 Operation Contact Factory

*Trade Mark of Tandy Corp. There is no affiliation between Scientific Engineering Laboratories and Tandy Corporation or Radio Shack.

Copy utility for single-drive TRSDOS users.

Single-Drive File Copy

Buzz Gorsky
712 Hillside Drive
Carlisle, PA 17013

The major task I am facing with my TRS-80 is a research project which will involve entering, editing and then repeatedly searching through thousands of entries in several random access files. I recently upgraded my 16K Level II system by adding memory, the RS232C, and a disk drive.

My TRSDOS system already has a program to copy files *if* two drives are available. While this backup program permits copying the contents of an entire disk, it is not convenient for duplicating a single data file. The GETDISK and GETTAPE programs permit saving a disk file on tape and then copying from the tape to another file—but then, I got the disk to get *away* from tape and all of those problems.

File Copier

I was thus motivated to write a program to permit file copying and to facilitate keeping a duplicate set of data files for those times during the project when Murphy's Law is bound to strike.

The program is short and straightforward, and may interest others with a single disk-drive system.

It begins in line 20 with some housekeeping. A little experimentation with the CLEAR statement may be required depending on available memory. Since each file record will be stored as a 256-byte string, as one of the R(I) and R1(I), the number of records which can be moved at a time depends on the memory available. The variable Z in line 20 determines how many records will be moved at a time, and that value can be reduced easily by changing the value of Z.

In line 30 the counter Y is zeroed and the user can input the filespec. The program then prompts you to put the source file in the drive.

After opening and fielding the buffer, the program then acquires the records sequentially.

Y is the actual record number, while X keeps track of how many have been dealt with in each pass. When Y equals the last record, or the FOR-NEXT loop finishes as X exceeds Z, record acquisition ceases. If the FOR-NEXT loop has been completed (line 100), X is decreased by one, so that it bears the same relation to Y as it would if the loop was exited in line 90.

This must be done, as when the NEXT statement is hit the last time through, X is increased to exceed Z. The loop is complete only when it is larger than Z.

Receiving Disk

The files are closed in line 200 and the program prompts the user to put the receiving disk in drive. Y is then reset to equal what it did at the start of the last run of record acquisition, by setting Y = Y – X. The appropriately named file is also opened and the buffer fielded. Then the W loop begins. Here W will keep track of the number of records transferred on this run, and Y will again equal the actual record number.

This loop is exited if Y equals the end of file marker for the source file (Z1), or when W exceeds X.

If the loop completes, then the file is closed and the user is prompted to return the source disk to the file. The program returns to line 40 to begin acquiring files again, this time beginning with Y incremented from the value of the last file transferred. Had the transfer been complete, the program would indicate "file copied" after the file was closed.

This utility uses little disk space and makes copying any disk files quite painless. I would advise using Verify to be sure that all goes well.■

```
10 REM PROGRAM TO COPY FILES FROM ONE DISK TO ANOTHER U
     SING ONE DRIVE BY BUZZ GORSKY K8BG
20 CLEAR25000:DEFSTR R:DEFINT W-Z:Z=60:DIM R(Z),R1(Z)
30 Y=0:CLS:INPUT"ENTER FILE NAME";R:INPUT"PUT SOURCE DI
     SK IN DRIVE, HIT ENT WHEN READY";A$
40 OPEN "R",1,R:Z1=LOF(1)
50 FIELD 1,255 AS F$,1 AS F1$
60 FOR X=1 TO Z
70 Y=Y+1
80 GET 1,Y:R(X)=F$:LSET F$="":R1(X)=F1$:LSET F1$=""
90 IF Y=Z1 THEN 200
100 NEXT X:X=X-1
200 CLOSE:CLS:INPUT"PUT RECEIVING DISK IN DRIVE, HIT EN
     T WHEN READY";A$:OPEN "R",1,R:FIELD 1, 255 AS F$,
     1 AS F1$:Y=Y-X
210 FOR W=1 TO X:Y=Y+1
220 LSET F$=R(W): LSET F1$=R1(W)
230 PUT 1,Y
240 LSET F$="": LSET F1$=""
250 IF Y=Z1 THEN 300
260 NEXT
270 CLOSE:CLS:INPUT"PUT SOURCE DISK IN DRIVE, HIT ENT W
     HEN READY";A$:GOTO40
300 CLOSE:CLS:PRINT@470,"FILE COPIED"
```

Program Listing 1.

Text analysis by frequency of letters.

Letter Counter

Walter J. Atkins, Jr., Ph.D.
Quarters 4410A
US Air Force Academy, CO 80840

One day while searching for something to be curious about, I wondered: "What is the most used letter in the English language?"

I found that others had wondered about the same thing. For example, Samual Morse designed his famous telegraph code so that the most frequently used letters were represented by the simplest symbols.

As my curiosity became a total fascination with all of this, I decided it wasn't good enough just to take someone else's word for it. My omnipotent TRS-80 could tell me. I then sat down and designed the following program.

Letter Counter

Letter counter will analyze text that you type into a text buffer by counting the frequency of occurrence of each of the twenty-six letters of the English alphabet. Then it will display the frequency of each letter as well as its percentage of occurrence in the total number of letters.

First you must determine the best way to differentiate between the letters. Since each in a BASIC program is represented by its ASCII equivalent, I felt that the easiest way would be to count the occurrence of each ASCII code. The TRS-80's ASCII codes are on pages C/1 and C/2 of the Level II BASIC Reference Manual. You will notice that ASCII character codes 65–90 represent the uppercase letters, A–Z. Codes 96–127 represent the lowercase letters a–z. Since the standard TRS-80 does not display these, the program counts only uppercase letters. You can change it so that it will recognize both, by changing line 235 and inserting line 237:

```
235 IF(BIN< 65 OR BIN>90) THEN 237
      ELSE P = BIN – 64:GOTO 245
237 IF(BIN>95 AND BIN<128) THEN P =
      BIN – 95:GOTO 245 ELSE GOTO 250
```

Four Modules

The program is arranged into four modules. The module titles give us an idea of what the module does.

The text input module, for example, is used to accept lines of alphanumeric text from the keyboard, storing it in a working buffer. Each line of this text may contain up to 200 characters. After it is typed in, the ENTER key is pressed. This moves the line indicator to the next line. After the last line is entered, pressing ENTER again begins the letter count.

The letter count module is then used to read lines of text from the buffer. It also counts the occurrences of each letter of the alphabet. It finds the percentage of occurrence of each letter.

Next, the frequency table module produces the table of the alphabet, containing the number of times and percentage of occurrence in the text of each letter.

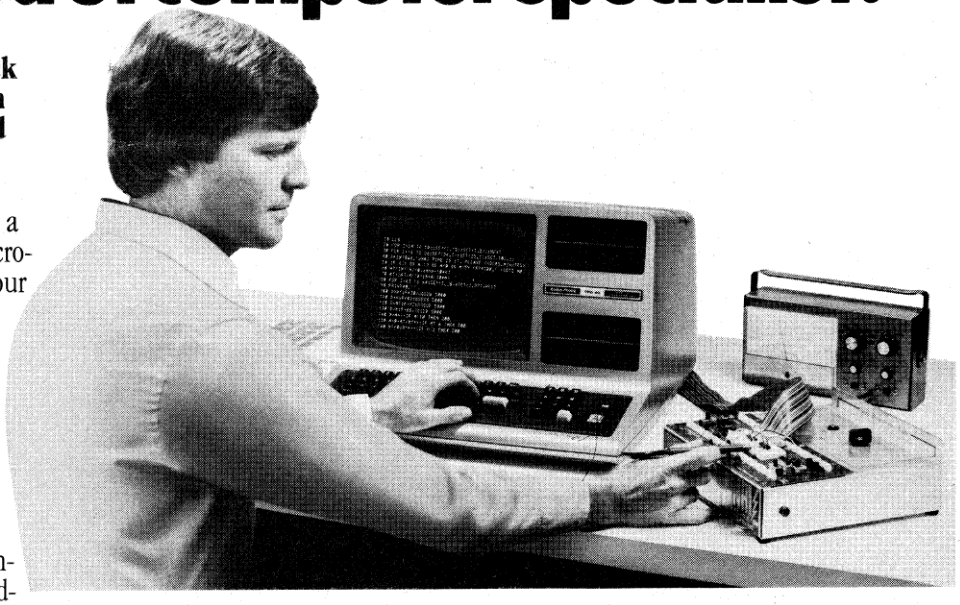
Finally, the frequency graph module produces a graph of the percentage of occurrence. This graph enables you to see which letters were used most frequently in text.

TECHNICAL DATA

Program length: 2078 bytes
Memory remaining after loading (16K system): 13454 bytes
Line Usage:
25—Set aside 10000 bytes of memory for text storage
30—Make array A( ) an array of integers
40—Dimension arrays T$( ), A( ) and B( )
—Text array T$ holds 50 lines of text
—Arrays A( ) & B( ) store counts for each letter
100–145—Enter text into storage buffer. Max line length allowed is 200 characters.
200–250—Count the number of times each letter appears in the text.
255—Counts the total number of letters in the text
300–360—Print table showing each letter, the number of times it appears, and it's percentage of the total
400–470—Graph percentages of each letter on screen
480–490—Display 'continue' prompt at right side of screen

Table 1

Now NRI takes you inside the new TRS-80 Model III microcomputer to train you at home as the new breed of computer specialist!



NRI teams up with Radio Shack advanced technology to teach you how to use, program and service state-of-the-art microcomputers...

It's no longer enough to be just a programmer or a technician. With microcomputers moving into the fabric of our lives (over 200,000 of the TRS-80™ alone have been sold), interdisciplinary skills are demanded. And NRI can prepare you with the first course of its kind, covering the complete world of the microcomputer.

Learn At Home in Your Spare Time

With NRI training, the programmer gains practical knowledge of hardware, enabling him to design simpler, more effective programs. And, with advanced programming skills, the technician can test and debug systems quickly and easily.

Only NRI gives you both kinds of training with the convenience of home study. No classroom pressures, no night school, no gasoline wasted. You learn at your convenience, at your own pace. Yet you're always backed by the NRI staff and your instructor, answering questions, giving you guidance, and available for special help if you need it.

Training includes new TRS-80 Model III microcomputer, solid state volt-ohm meter, digital frequency counter, and the NRI Discovery Lab with hundreds of tests and experiments.

(TRS-80 is a trademark of the Radio Shack division of Tandy Corp.)

You Get Your Own Computer to Learn On and Keep

NRI training is hands-on training, with practical experiments and demonstrations as the very foundation of your knowledge. You don't just program your computer, you introduce and correct faults...watch how circuits interact... interface with other systems...gain a real insight into its nature.

You also build test instruments and the NRI Discovery Lab, performing over 60 separate experiments in the process. You learn how your troubleshooting tools work, and gain greater understanding of the information they give you. Both microcomputer and equipment come as part of your training for you to use and keep.

Send for Free Catalog... No Salesman Will Call

Get all the details on this exciting course in NRI's free, 100-page catalog. It shows all equipment, lesson outlines, and facts on other electronics courses such as Complete Communications with CB, TV and Audio, Digital Electronics, and more. Send today, no salesman will ever bother you. Keep up with the latest technology as you learn on the latest model of the world's most popular computer. If coupon has been used, write to NRI Schools, 3939 Wisconsin Ave., Washington, D.C. 20016.

NRI McGraw-Hill

NRI Schools
McGraw-Hill Continuing Education Center
3939 Wisconsin Avenue
Washington, D.C. 20016
NO SALESMAN WILL CALL
Please check for one free catalog only.

All career courses approved under GI Bill.
☐ Check for details.

- ☐ Computer Electronics Including Microcomputers
- ☐ TV/Audio/Video Systems Servicing
- ☐ Complete Communications Electronics with CB • FCC Licenses • Aircraft, Mobile, Marine Electronics
- ☐ CB Specialists Course
- ☐ Digital Electronics • Electronic Technology • Basic Electronics
- ☐ Small Engine Repair
- ☐ Electrical Appliance Servicing
- ☐ Automotive Mechanics
- ☐ Auto Air Conditioning
- ☐ Air Conditioning, Refrigeration, & Heating including Solar Technology

Name (Please Print) Age

Street

City/State/Zip

Accredited by the Accrediting Commission of the National Home Study Council 179-031

✓ Reader Service—see page 290

The program counts the letter by examining each ASCII character while totaling the occurrence of each. The text to be analyzed has been stored by lines in a string array, T$. After all of it is entered, each line is scanned using the MID$ function to isolate each character. Each is then converted to its decimal ASCII equivalent, using the ASC function. A count is stored in array A and incremented each time it is encountered. There is an array element for each letter of the alphabet.

After all the letters have been tallied, the program adds up each letter to find its total number of times in text. Using this total, it calculates the percentage of occurrence of each letter.

The percentages are used to create a table of each letter and its frequency, or occurrence.

This information is also used in the frequency graph module in order to graph or chart the letter frequency of the alphabetic symbols. ■

String Variables **Variables**

Name	Use
G$	Screen title in graph mode
H$	Screen border (63 asterisks)
LE$	Used in counting module
TI$	Screen title at beginning of program

Numeric Variables

C	Array index counter for array T$( ) in the counting module
G	Length of variable G$
I	Array index counter for array T$( ) in the input module
J	Loop counter
K	Loop counter
L	Line counter for lines displayed on CRT screen
P	Array index
T	Total number of letters in text
X	Print position in table module
Y	Print position in graph module
BI	Used in counting module (numerical ASCII code)
XS	Bar graph X position
XX	Print position in graph module
YS	Bar graph Y start position

Array Variables

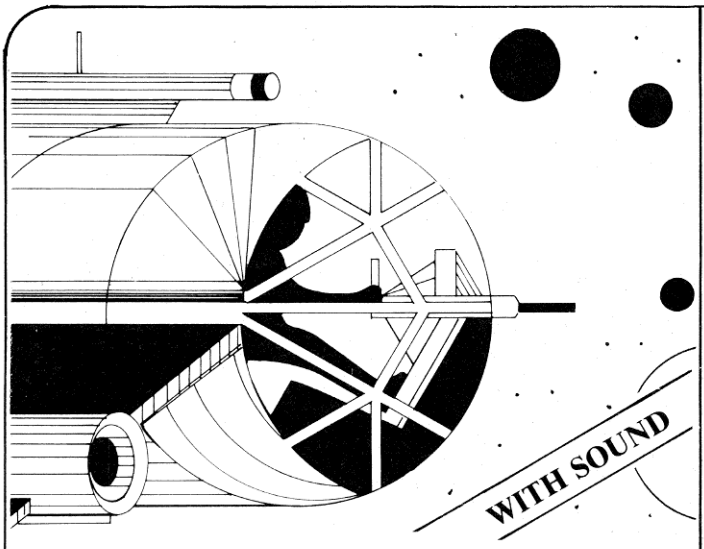
Name	Dimension	Use
A( )	27	Letter count storage
B( )	27	Bar graph length storage
T$( )	50	Text storage

Table 2


```
10 REM *  LETTER COUNTER *
15 REM *  BY WALTER J. ATKINS, JR.
20 REM *  MAR 80
25 CLEAR10000:CLS
30 DEFINT A
35 H$=STRING$(63,"*")
40 DIM T$(50),A(27),B(27)
100 '    ---- TEXT INPUT MODULE ----
105 TI$="LETTER COUNTER":TI=LEN(TI$):PRINT@32-(TI/2),TI
    $:PRINT:PRINTH$:PRINTH$
110 FOR J=1 TO 50:T$(J)="":NEXT J:FOR J=1 TO27:A(J)=0:N
    EXTJ
115 PRINT"  i WILL COUNT THE NUMBER OF TIMES EACH LETTE
    R APPEARS IN THE":PRINT"TEXT YOU ENTER.":L=13
120 PRINTH$:PRINT"BUFFER READY ..  ENTER TEXT .. PRESS
    ''ENTER'' AFTER EACH LINE":PRINTH$
125 PRINT"PRESS ''ENTER'' TWICE AFTER LAST LINE TO STAR
    T LETTER COUNTING":PRINTH$
130 IF L=6,135 ELSE I=1
135 INPUT"*";T$(I)
140 IF LEN(T$(I))>200 PRINT"LINE TOO LONG.....MAX OF 20
    0 CHARACTERS ":L=L+2:IF L>=15 FOR X=1TO1000:NEXTX:
    CLS:L=6:GOTO120GOTO135
145 IF T$(I)<>""THEN I=I+1:L=L+1:IF L>16,L=6:CLS:GOTO12
    0ELSE GOTO135ELSE GOTO150
150 PRINTH$;:PRINT I-1"LINES ENTERED ..... COUNTING ...
    .. PLEASE WAIT."
200 '    ---- LETTER COUNTING MODULE ----
205 FOR C=1 TO I-1
210 L=LEN(T$(C))
215 FOR K=1 TO L
220 LE$=MID$(T$(C),K,1)
225 BIN=ASC(LE$)
230 IF BIN=128 THEN 250
235 IF (BIN<65 OR BIN>90)THEN 250
240 P=BIN-64
245 A(P)=A(P)+1
250 NEXTK:NEXTC
255 T=0:FOR J=1TO26:T=T+A(J):NEXTJ:IF T<>0 THEN T=100/T
300 '    ---- FREQUENCY TABLE MODULE ----
305 CLS
310 PRINT"LETTER"TAB(10)"NUMBER"TAB(20)"PERCENT"TAB(30)
    "LETTER"TAB(40)"NUMBER"TAB(50)"PERCENT"
315 PRINTH$
320 X=128:FORJ=1TO26
325 GOSUB360
330 IF X<=832THEN X=X+64:J=J+1:GOTO325
335 X=158:J=J+1
340 GOSUB360
345 IFX<=926 AND J<=25 THEN X=X+64:J=J+1:GOTO340
350 PRINT@960,H$;
355 GOSUB480:CLS:GOTO405
360 PRINT@X+2,CHR$(J+64)"       "A(J)"       "INT(A(J)*
    T*100+.5)/100" %";:RETURN
400 '    ---- FREQUENCY GRAPH MODULE ----
405 CLS:G$="GRAPH OF LETTER DISTRIBUTION":G=LEN(G$)
410 PRINT@32-(G/2),G$;
415 FOR J=1TO26:B(J)=INT(A(J)*T*100+.5)/100:NEXTJ
420 FOR J=1TO26:B(J)=INT(B(J)*.4*100+.5)/100:NEXTJ
425 FOR Y=3TO43 STEP 10:SET(6,Y):NEXTY
430 FOR Y=3TO43:SET(7,Y):NEXTY
435 FOR X=8TO111:SET(X,43):NEXTX
440 FOR X=10TO110 STEP4:SET(X,44):SET(X+1,44):NEXTX
445 XX=64:FOR X=965 TO 1015 STEP2:XX=XX+1:PRINT@X,CHR$(
    XX);:NEXTX
450 PRINT@64,"100";:PRINT@256,"75";:PRINT@448,"50";:PRI
    NT@704,"25";:PRINT@897,"0";
455 FOR X=1TO500:NEXTX:PRINT@0,CHR$(30);
460 G$="PERCENT OF EACH LETTER IN TEXT":G=LEN(G$)
465 PRINT@32-(G/2),G$;
470 XS=10:FOR X=1TO26:YS=43-B(X):FOR Y=YS TO 43:SET(XS,
    Y):SET(XS+1,Y):NEXTY:XS=XS+4:NEXTX
475 GOSUB480:CLS:GOTO110
480 PRINT@570,"PRESS";:PRINT@633,"'ENTER'";:PRINT@699,"
    TO";
485 PRINT@763,"GO";:PRINT@827,"ON";
490 PRINT@891,"";:INPUTI$:RETURN
500 /15
```

Program Listing

GO BOLDLY . . .
Where No TRS-80* Program Has Gone Before!



ASTEROID

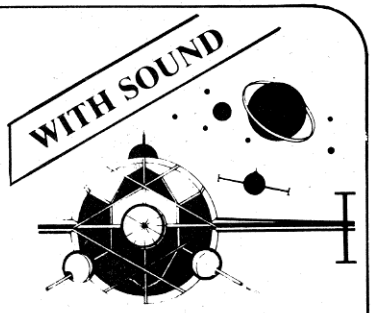
DATE: 28.02.2047
LOCATION: 270 million miles from Terra
MISSION: Maintaining Terra's Space Lanes

Briefing will follow:
1.1 Your mission is to destroy any asteroids in your sector and to prevent alien spacecraft from infiltrating the Terran Defense Network.
1.2 Your ship is armed with an anti-matter cannon. You can shoot large asteroids, but this turns them into many smaller asteroids, each capable of destroying your ship.
1.3 In addition, alien ships can make instantaneous hyperspace jumps into your area and start firing on your ship.
1.4 You'll need lightning reflexes and nerves of steel to survive Asteroid. We have no use for non-survivors!

Asteroid, a real-time, machine-language game, features variable levels of difficulty, superb high-speed graphics, sound effects and automatic score keeping. (T1) or (T2)

Order No. 0237R $14.95 Tape.
Order No. 0247RD $19.95 Disk.

BALL TURRET GUNNER

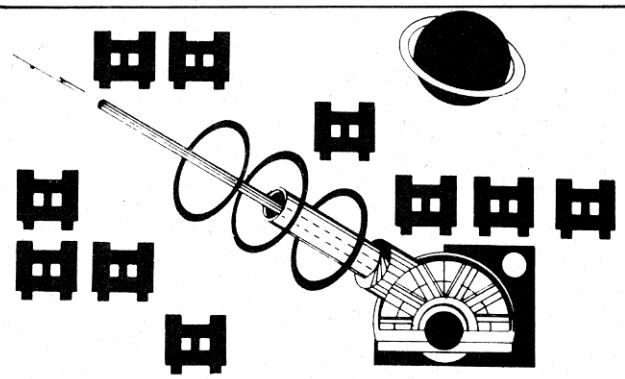


For years the Petro Resource Conglomerate has attacked our photon collection stations and strangled our deep-space trade routes. The PRC Exxonerator Class light fighters (code name: Gnat) have been their main weapon. Now you can strike back, by joining the Ball Turret Gunner Service.

Imagine yourself at the control console of an LW-1417 Stratoblazer (Type B Strategic Laser Weapon). Your Hindsight Director informs you that a Gnat fighter is coming in for an attack. You pivot your gigawatt laser turret until you can see the target on your monitor. The Range Indicator shows him coming in fast. The Targeting Computer studies his course and speed as your finger tenses over the firing key. You know you'll have only a fraction of a second in which to react. The Gnat fighter's evasive maneuvers cause him to dance in your sights. Suddenly, you see the FIRE Command and you react instinctively. Your laser beam lashes out and reduces the Gnat to an expanding ball of ionized gas. Mission accomplished!

Ball Turret Gunner, with your choice of multiple levels of difficulty, optional sound effects and superb graphics, is more than just a game. It's an adventure. Experience it! (T1)

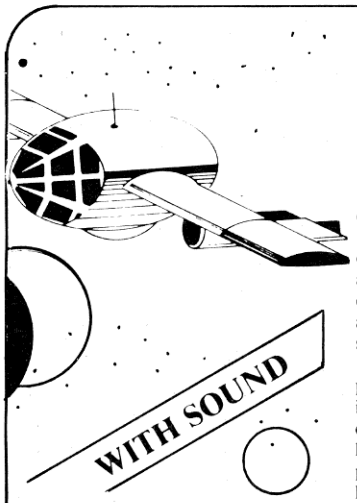
Order No. 0051R $9.95.



INVADERS

The INVADERS are coming! Earth's defenses are dead except for your Laser base. Your assignment is to destroy the approaching INVADERS before they destroy Earth. Before Earth's sensors failed, they detected 550 armed invaders in space, speeding toward us in 10 attack formations of 55 in each group. The sensors detected four different types of attack craft: Large, Medium, Small, and a short profile craft which is the most difficult to destroy. If you cannot stop these space attackers they will stop Earth. . . . . for good. (T1)

Order No. 0240R $9.95.



COSMIC PATROL

WARNING: PLAYERS OF THIS GAME SHOULD BE PREPARED FOR A STATE OF REALISM HITHERTO UNAVAILABLE ON THE TRS-80

Skilled players soon master many difficult computer games, but COSMIC PATROL is in a world all its own. The challenge intensifies! Supporting graphics and sound (optional) make each encounter an exciting new experience. It all adds up to a Super 3-S package. . . skill, sight and sound.

Scenario: The Cosmic Patrol program puts you in the command chair of a small interstellar patrol craft. Your mission is to defend Terran space and prey on the Quelon supply ships which carry essential parts and lubricants for that implacably hostile robotic force. The drone freighters are fairly easy pickings for the accomplished starship pilot, but beware of the I-Fighter escorts. They're armed, fast and piloted by intelligent robots linked to battle computers. They *never* miss.

The Cosmic Patrol program is not just another search and destroy game. With its fast, real-time action, impressive sound option and superb graphics, this machine-language program is the best of its genre.

Don't keep putting quarter after quarter into arcade games or spending big bucks for video game cartridges. Get Cosmic Patrol from Instant Software—and get the best for less! (T1)

Order No. 0223R $14.95 Tape.
Order No. 0224RD $14.95 Disk.

TO ORDER

See Your Local Instant Software Dealer or Call Toll-Free 1-800-258-5473

Instant Software™ ✓2

PETERBOROUGH, N.H. 03458 603-924-7296

(T1) = TRS-80 Model 1, Level II, 16K RAM.

(T2) = TRS-80 Model 1, Level II 16K, Expansion Interface 16K + 1 disk drive.

*A trademark of Tandy Corporation

✓ Reader Service—see page 290

An unsolicited testimonial on the benefits ($) of computing.

Just a Matter of Time

David Busch
515 E. Highland Avenue
Ravenna, OH 44266

Is your personal computer an expensive toy, or does it earn its keep?

That's probably the most common question posed by spouse, friends and the non-computer owning public.

The question is reasonable. After all, few businesses would consider purchasing a computer without extensive cost-justification studies. Many Model I's find themselves pressed into service in a variety of business applications, and a great many more are used for personal tasks. But few are purchased by owners who really have a solid idea of what they expect the computer to do. The actual savings potentials are far down on our lists of priorities.

I decided to look hard at a disk system and one user—me—to see if a multi-thousand dollar investment made sense in the long run. My results surprised me! I estimated that if my time is worth $10 per hour, I will save an average of 470 hours, nearly $5000, in an average year. Like EPA ratings, your results will differ.

The numbers are a little misleading: It appears that I've managed to save nearly a quarter to a third of a year's time. Actually, the computer hasn't enabled me to work any longer than the weird hours I already keep, but rather, I'm able to get much more done in the time span now available to me.

About 400 of the saved hours came from using Radio Shack's Scripsit in my writing. The other 70 hours came from applications common to most personal/business micro users. In five years, that $700 in yearly savings alone would pay for a $3500 computer.

My cost justification concentrated on things that I did manually before, and now rely on the computer for help. Each included estimates of the net time and/or money saved.

Personal Finance

As an example, I use Radio Shack's Budget Management package to keep track of checks written for both my personal and business use. Checks for food, utilities, home expenses, car payments, and other bills as well as business expenses awaiting reimbursement, are entered into the budget management data files, along with an account number that posts the check to an appropriate expense account.

I enter checks once a week, usually completing the job in five minutes. That's a time investment of four hours and 20 minutes over a year. When tax time rolls around, I'm able to print out the year's checks, sorted by expense account number. I estimate that keeping a ledger manually, or sorting through all the checks by hand in the spring (I've used both methods) takes anywhere from 16 to 30 hours a year. These are for personal expenses, not the more complex bookkeeping required for a business itself.

If my average yearly savings amounts to 15 hours, and my time is worth $10 an hour, I've saved $150 dollars, minus $30 paid for the program, or $120 net the first year.

About once a month I receive a bill or invoice that I think I have already paid. It used to take five minutes to search through my bill envelope for a receipt, or my checkbook for a stub. Because I'm now paying nearly everything by check, I can locate a payment record in the budget management file in a minute or less. Yearly savings: 48 minutes or $8.

Personal Activities

My TRS-80 helps me with several of my hobbies, such as music, photography, and book and film collecting.

I commonly buy four or five albums a month, and record them on cassettes for my car stereo. To choose the best tape length for a given album, it's necessary to time all the selections. A program that took me 10 minutes to write adds up the minutes and seconds separately and provides timing totals for each side and disk in an album set. Total time required: about five minutes a month vs 20 minutes, using manual methods. Net time savings: about three hours a year, or $30.

I wrote a program that uses my computer as a digital timer for various black and white and color photographic processes. It stores instructions, including time, temperature, and chemicals used in disk files, and prompts the user to dump solutions or agitate with visual and audible signals. It took three or four hours to write the program, but I sold it, so I consider that time invested to have been recompensed.

Meanwhile, I save several minutes each time I process film by not having to reset a manual timer between steps. Estimated saving: 10 minutes a month. At $10 an hour, that's $20 a year.

My modified version of PIMS keeps track of my book and film collections, as well as magazine subscriptions. I frequently take printouts of items I already have when I go scavenging for used editions to add to my collections. Keeping these lists manually took 10 hours a year. This file maintenance now amounts to an hour over 52 weeks. Net savings: nine hours a year, $90.

Business—Personal and Otherwise

Because I use my car for business, I like to keep track of maintenance costs. I wrote an Autocost program that includes data on every tank of gas I've purchased since August, 1978 and the mileage of every business trip since September, 1976.

I have always kept accurate miles-per-gallon records, and periodically figured the true costs for operating my motor vehicles. These tasks took about five hours a year, and now can be completed in a tenth that time. Net time savings: 4 1/2 hours, or $45.

One of the unpleasant tasks of taking pictures for magazine articles is the need to write up photo lab orders. With an average of 15 rolls of film a month to send off, usually one or two at a time, it takes me about an hour and 15 minutes each month—five minutes per roll—to find the order forms, fill them out, and address envelopes for mailing. Most of this information is standardized, so I wrote a short program to do it for me. For most photo projects, I shoot one roll of black-and-white film, and require one contact sheet. This data are built into the

program as default values. By entering the name of the project, client and hitting ENTER a few times, the computer and my printer now format and print the lab order automatically. For special cases, I can insert additional instructions. Net gain: 12 hours a year, $120 worth of my time.

More Programs

A similar program prepares invoices for clients. I fill in only a name, project description, and price for the approximately 10 separate jobs I do each month. The printer and computer do the rest. Forty-five minutes a month are reduced to five because I can usually prepare invoices for the upcoming month's work at a single sitting, with a yearly gain of eight hours that are worth $80 to me.

Of all the paperwork I do, my travel expense reports take the most time. I have to account for each day's hotel, car rental, food, mileage, or airfare expenses, total these, and compare them to the billing for the assignments I've covered during a trip. I spent a lot of time, approximately four hours, writing a travel expense report program to allow me to enter these data quickly, group and total relevant categories of expenses, and print them out in a readable report.

I travel three or four times a month, and have to fill out a report for each trip. In addition, I have always prepared one or more "dummy" expense reports with cost estimates before leaving on a trip, just to be certain ahead of time that my company wasn't losing money. These reports are so complicated that a travel expense report generator was one of the first projects I tackled when I got my TRS-80. I quickly discovered that the 26 single-letter variables allowed in Level I weren't sufficient to store the various categories, and I had to use the single array A(n) as additional variables. I then discovered that my first, inefficient effort wouldn't fit in the 4K I had available. Even so, the attempt was worth it to me, in order to eliminate the half-hour sessions spent five or six times a month struggling with expense reports.

My Disk BASIC version is shorter and more efficient, and asks for about five minutes of my time to input all the required information. The 30 minutes a month I now spend add up to a 30-hour savings during the course of a year, and a $300 gain in time.

The Answer to a Writer's Prayer

I began using Radio Shack's Scripsit in March to compose my articles, and hope that I never have to touch a conventional typewriter again. It is literally impossible for me to add up the ways in which I save time and effort using this word processing system, but let me cite a few.

I commonly write at least two drafts of a piece before sending it off. In recent years, I've relied on a finish typist to prepare the final version, but, rather than depend on my typist's ability to read my handwriting, I've continued to run through an article one last time as neatly as possible, incorporating all the editing changes and cut-and-paste butchery that went into the creation of the piece. The final draft of an average article took me about two hours to prepare. In a month in which I complete 10 assignments, that adds up to 20 hours wasted on typing that no one, other than my typist, will ever see.

With Scripsit, I can hack away to my heart's content, editing on the screen as I please. I can save one version of a story on disk, and completely transform the article into a second, if I like, with the original still available, should I change my mind. Then, when I'm satisfied, I can have the finished draft printed out on my dot-matrix printer in five minutes or so. I usually keep busy with other tasks, stopping to feed single sheets to my printer at intervals, so the actual time investment is negligible. Some clients and magazines, especially the personal computer books, will accept the excellent dot matrix printing of my Integral Data Systems IP125. For others, I continue to send my errorless (I hope) computer-printed drafts off to the typist.

Net gain in a year: 240 hours for an amazing $2400 savings. That figure makes me wonder why I still can't afford a daisy-wheel printer.

Scripsit saves me time in article preparation. One project I worked on involved writing one-page news releases about the visits of international executives to a trade show. There was a lot of standard information about the show and my client that had to be worked into every news release. One or more paragraphs had to be individually composed about the particular vistor discussed in the release. To complicate things, many of the internationals were from the same countries, and I wanted to mix up the releases so that all those going to Mexico and Japan wouldn't say the same thing.

With 50 releases to prepare—10,000 words—I could have spent a week or more on the project. Instead, I wrote 15 stock paragraphs and gave each a different Scripsit file name. I then pieced them together, using Scripsit's chaining capability, wrapping them around custom paragraphs tailored for each visitor. Only about 20 percent of the 10,000 words had to be written from scratch. A 40-hour job was trimmed to eight hours, and I saved $320 worth of time on a single project. If, as I expect, I complete five similar jobs in a year, I've made myself another $1600.

I didn't even have to think hard to come up with my $4700 worth of time-savers.

As my uses for the TRS-80 grow, and software becomes more readily available, I'm sure cost-justifying my personal computer will become even easier. I just wish I knew why that daisy-wheel printer I need can't be paid for out the $5000 I just saved. ■

✓ Reader Service—see page 290

THIS REMARKABLE DEVICE WILL MAKE YOUR TRS-80 WORK.

When you are ready to put your TRS-80 to work, you need a cassette system you can depend on. The TC-8.

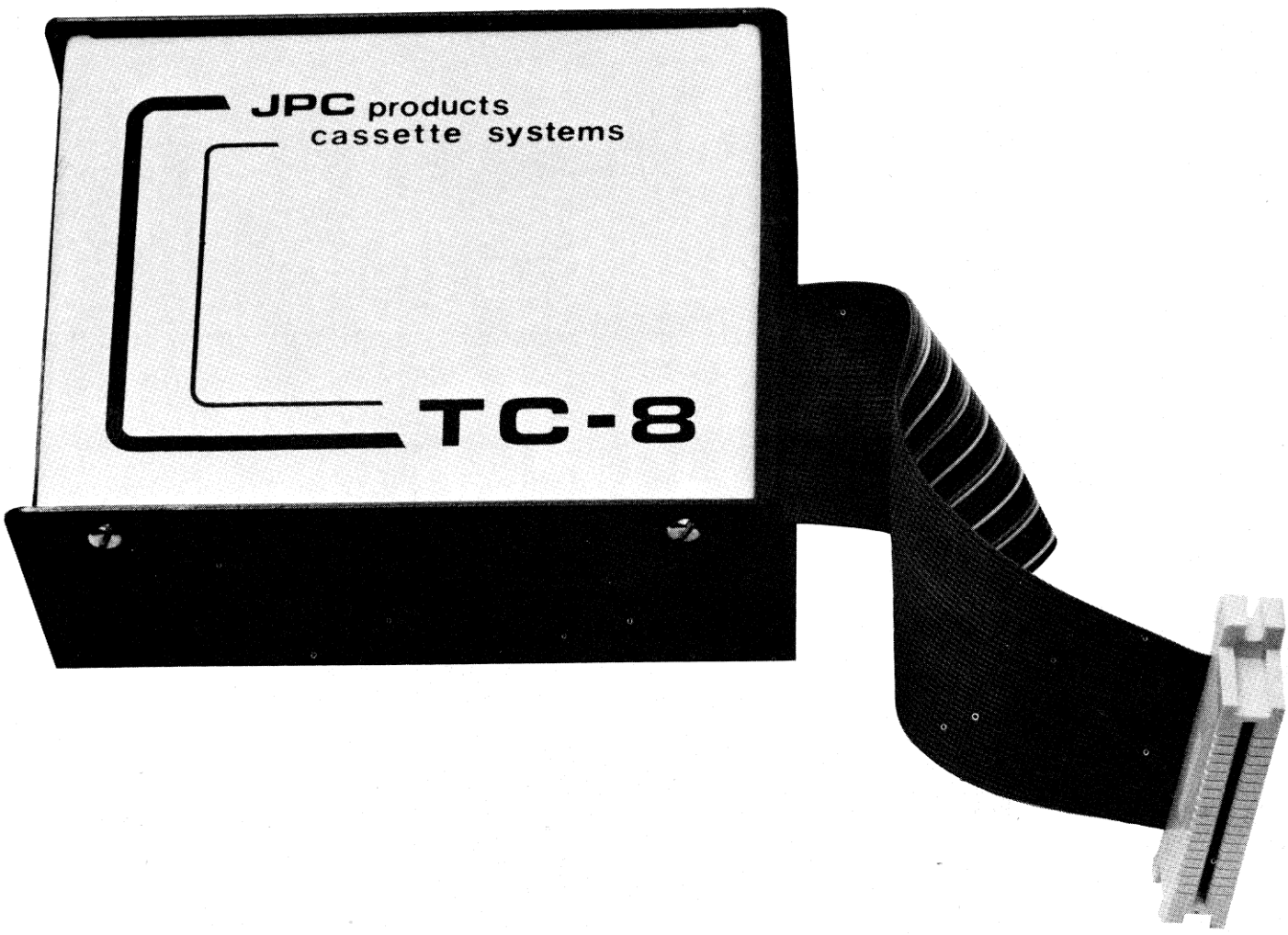
With a real file manager that supports named BASIC and system files. And searches for files by name without hanging up.

The TC-8 does all of that and loads 5 times faster than ordinary TRS-80's. At nearly 3000 BAUD.

And it loads every time. Without fiddling with the volume.

The TC-8 puts your TRS-80 to work.

Seriously.



The TC-8 loads programs and data fast.

Every time.

The TC-8 is available as a kit for $89.95 or completely assembled for $119.95. Either way, it is fully guaranteed. If it doesn't work, for any reason, simply return it to us and we will repair or replace it free. Order the TC-8 directly from us or from your local computer store. You may call our order phone at (505) 294-4623. We accept Visa and Master Card. Don't wait any longer; order your TC-8 today.

✓190

JPC PRODUCTS CO. 12021 PAISANO CT. ALBUQUERQUE, NM 87112

This program will help you:

Mind Your Own Business

Gregory R. Glau
PO Box 1627
Prescott, AZ 86302

It doesn't seem to matter what type of business you're in—retail, wholesale, contracting, service—monthly sales fluctuate and with them, your net profit and cash position.

We run a small heating/air-conditioning business and we've been amazed at what we've learned about our business by using our TRS-80 to analyze financial information, help in bidding jobs, etc. It's helped us analyze what we've done wrong in the past, but lately we've been using the computer for an even more important task: Projecting future information.

The idea of forecasting sales, overhead, profit and so on really isn't new and all businessmen know they need to do it. But the problem is the calculation involved—you have to multiply your yearly sales by the percent of business you do every month to arrive at a monthly sales figure. Then you have to find your cost of sales, again on a monthly basis and deduct it from your monthly amount, which leaves your gross profit. Gross profit less overhead leaves your net profit (or loss).

This isn't too difficult when you do it for a whole year, but then if you try to determine what will happen if your sales go up by a few percent, all the other figures will need to be changed. It's like setting up a line of dominoes—knock one down and all the rest fall. Here, if you change a figure, you have to recalculate forever!

". . . we've been amazed at what we've learned about our business by using our TRS-80 to analyze financial information, help in bidding jobs, etc."

Program Possibilities

But with a TRS-80 on your desk, you just give it the raw data and it does all the hard work. We wrote a BASIC program based on information from our own financial statement. It's written in Level II BASIC and will store and execute in 4K. All of a sudden we could visually see what was happening to our business over an entire year. We could see on our video display which months were profitable and in which ones we lost money. We learned that we have to somehow sell more during those "down" months.

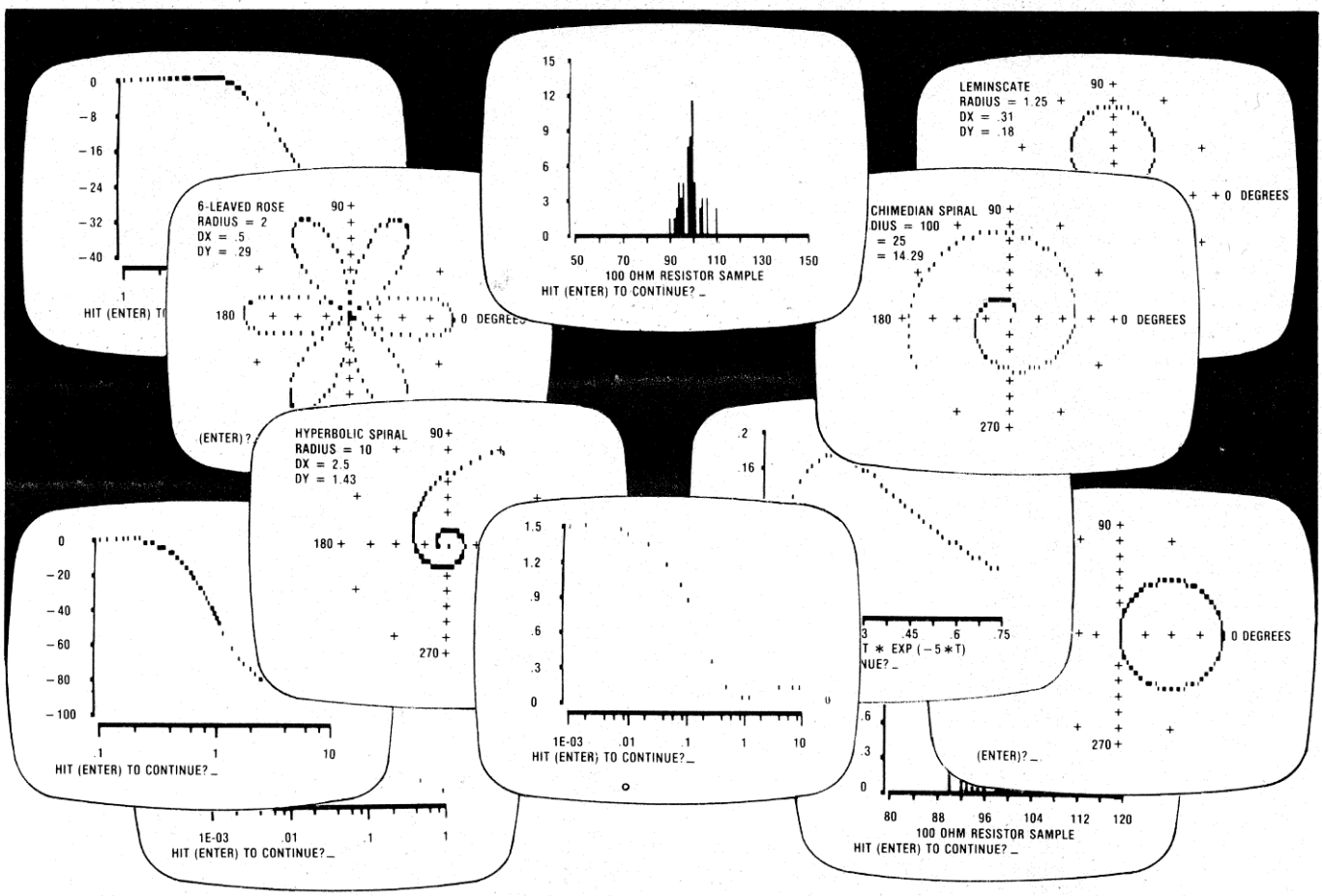
But let's carry the idea a step further. If we can display what we've done in our business for the past year, why not project what we might do during the next 12 months? Let's ask our TRS-80 what would happen if we increased sales by, say, 10 percent. By putting an input line into the program, along with the math to raise or lower that beginning sales figure, we can project any future sales increase or decrease and see what results those projections will have on all the other cost and profit figures.

How about gross cost of sales? If we can buy a product a bit cheaper, or get our workmen to work a little faster, we can lower this figure. What would happen to your business if you could lower your cost of sales by, for instance, 3 percent?

Finally, what about overhead? If you could cut down on office supplies or gasoline or electricity, you could lower your overhead. What effect would a decrease of, for example, 5 percent in your overhead have on your monthly profit figure?

The program as written will display exactly what we did last year, month by month, or, by answering any or all of the three INPUT questions, we can increase or decrease any of the figures that affect that final profit line.

This is probably the best part of this program, for we're able to ask our TRS-80 what would happen to that net profit line, on a monthly basis, if we, for example, increased sales by 10 percent, dropped the overhead by 3 percent, and lowered gross



ENGINEERING SOFTWARE FROM THE ENGINEERING PEOPLE.

For more than 30 years, engineers and technicians have relied on Sams technical, electronics, and computer books. Now, Sams has put its technical expertise together in a complete series of circuit design software programs.

7 packages encompass 37 different pretested, debugged programs. Fully documented. Easy to use. Available on tape or disk for popular microcomputers. Multi-program tapes only $24.95... disks $29.95.

Sams Software helps you optimize your circuit design by letting you quickly look at all the design parameters and possibilities. Sams Software is the fastest way to solve complicated, repetitive engineering, mathematical, and statistical problems that use up creative engineering time and rob you of engineering freedom.

Sams Software is the difference between static and dynamic designing. If you're still trying to compete in today's world with old-fashioned techniques, catch-up with the future. Find out how Sams Circuit Design Software can change your ideas into working designs faster.

Mail the coupon today for a copy of the Sams Software Catalog... or call toll-free for the name of your nearest Sams Software outlet— 1-800-428-3696.

Howard W. Sams & Co., Inc.
4300 West 62nd Street
P.O. Box 7092
Indianapolis, IN 46206

Name _____

Company _____

Address _____

City _____ State _____ Zip _____

The microcomputer I currently use is: ___TRS-80 ___APPLE II

_____OSI OTHER: _____

Interested in: Technical_____ Personal_____

Educational_____ Business_____

Sams ™ ✓457

AD067

SOME STRAIGHT TALK ABOUT DISK DRIVES

- DON'T BE CONFUSED BY ALL THE BRAND NAMES YOU SEE IN THE MARKET PLACE, THERE ARE VERY FEW MANUFACTURERS OF THE BASIC DRIVE CHASSIS, ALL THE OTHER NAMES ARE THOSE OF THE ASSEMBLERS OR THE RETAILERS.
- AS MANUFACTURED, THE DRIVE WILL NOT RUN ON A TRS-80$, IT MUST BE MODIFIED BY THE ASSEMBLER.
- THE QUALITY OF THE DRIVE DELIVERED TO YOU IS DEPENDENT ON BOTH THE MANUFACTURER AND THE ASSEMBLER, THE BEST CAN TURN TO JUNK IF THE ASSEMBLY IS IMPROPERLY DONE.
- THE POWER SUPPLY AND CASE ARE VERY IMPORTANT COMPONENTS OF THE COMPLETE DRIVE, THE CASE MUST ALLOW PROPER COOLING AIR FLOW, AND THE POWER SUPPLY MUST MAINTAIN TWO CONSTANT VOLTAGES.
- YOU MUST DEPEND ON THE COMPANY SELLING YOU THE DRIVE TO SERVICE IT AT REASONABLE COST WHEN IT FAILS YOU, THE MANUFACTURER IS NOT EQUIPPED TO DO THIS!
- THE BEST MEASURE OF QUALITY IN A DRIVE IS IT'S SPECIFICATIONS, WILL IT HANDLE DOUBLE DENSITY, WHAT IS THE TRACK TO TRACK ACCESS TIME, THE ANSWERS TO THESE TWO QUESTIONS INDICATE THE PRECISION OF IT'S COMPONENTS.
- WHAT KIND OF DRIVE SHOULD YOU BUY ? LEVEL IV HAS CHOSEN TO DISTRIBUTE EXCLUSIVELY, THE MPI LINE, ALL MODELS OF MPI ARE DOUBLE DENSITY RATED AND REQUIRE ONLY A FIVE MILLI-SECOND TRACK TO TRACK ACCESS TIME.
- WHAT DO ALL THE MODEL NUMBERS MEAN?
  B-51= 40 TRACKS SINGLE HEAD SINGLE SIDE B-52= 40/40 TRACKS DOUBLE HEAD DOUBLE SIDE
  B-91= 80 TRACKS SINGLE HEAD SINGLE SIDE B-92= 80/80 TRACKS DOUBLE HEAD DOUBLE SIDE
  (DOUBLE HEADS)-READ BOTH SIDES OF DISK (DUALS)-TWO DRIVES IN ONE CASE (RAW)-NO POWER SUPPLY OR CASE
- WHERE SHOULD YOU BUY YOUR DRIVE, LEVEL IV IS ONE OF THE OLDEST AND LARGEST DISTRIBUTORS OF TRS-80$ EQUIPMENT, LOOK AT THE ADS IN YOUR OLD MAGAZINES, MANY OF THE ADVERTISERS ARE NO LONGER IN BUSINESS, LEVEL IV HAS BEEN A LEADER SINCE THE BEGINNING, WE STAND BEHIND OUR PRODUCTS, AND WE'LL BE HERE WHEN YOU NEED HELP.
- WHERE DO THE NATIONALLY KNOWN AUTHORS BUY THEIR DRIVES ? LEVEL IV CAN SHOW COPIES OF SALES RECEIPTS FOR DRIVES TO MOST OF THEM, LANCE MICKLUS, SCOTT ADAMS, VERNON HESTER, BOB (CAPTAIN 80) LIDDIL, DICK BALCOM, KIM WATT, ETC. TO THESE MEN COMPUTING IS NOT A HOBBY, IT IS THEIR PROFESSION.
- CALL FOR OUR LOW PRICES ON NEW AND USED DRIVES, REMEMBER, WE ALSO TAKE TRADES, AND PAY THE SHIPPING!

✓14

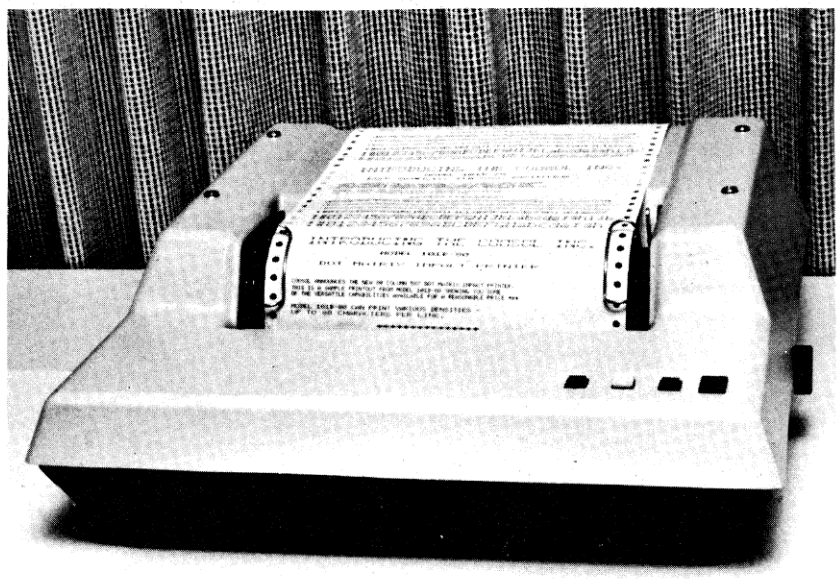
LEVEL IV PRODUCTS INC. 32238 SCHOOLCRAFT, LIVONIA, MI 48150
PHONES: MI (313) 525-6200 OTHERS 800-521-3305 (TOLL FREE)

$ a trademark of the RADIO SHACK DIV. of TANDY CORP.

COOSOL DATA LOGGER IMPACT PRINTER

FEATURES INCLUDE:

- Uses Standard Typewriter Ribbon
- Micro Processor Controlled
- Can Operate Continuously— No Thermal Problems (Has an all metal print head)
- 5 x 7 to Larger 10 x 7 and Larger 10 x 14 Dot Matrix Character Generator
- Standard 96 ASCII Character Font
- Upper and Lower Case Printing
- Up to 88 Characters Per Line
- Single Line Print Rate Is 160 CPS
- Average Print Rate Is 60 CPS For Ten Lines
- Graphics Capability With Extended Character Modes
- Programmable With 32 System Level Software Commands
- Standard Parallel and Serial Interface
- Software and Hardware Reset Interface
- Software Line Counting For Paging
- Baudrate Select From 110 to 9600
- Manual Paper Advance
- Manual Selftest and Reset
- Adjustable Tractor Width From 1 to 9½ Inches
- Interfaces with Apple, Atari, OSI, T.I., TRS-80 and the List Goes On ·



$495 Kit, 101B-80KE $545 Assembled & Tested 101B-80E

·Registered Trademarks of Apple Computer Inc., Atari Inc., Ohio Scientific Inc., Texas Instruments Inc., Tandy Corp. respectively.

FACTORY DIRECT ✓78

COOSOL, INC. P.O. BOX 743, ANAHEIM, CA 92805 (714) 545-2216 7 Days a Week

cost by 4 percent. What if our sales went down 5 percent, our overhead increased 3 percent and our gross costs . . . well, you get the idea, and the possibilities are endless.

The net profit figure for each month is also an approximate cash-flow projection, so you can get a rough idea in which months you might have a cash problem. . . so that you can do something about it before it really causes a problem!

Inputting Data

If you're just thinking about going into business, this program can really be useful, as you should have a budget set up before you open your doors, and you can use this program to see how you'll do based on your budget projections. . .and, what will happen to your new venture if your sales decrease by 10 percent, or if your overhead is 5 percent more than what you'd figured on and so forth.

The program will run as listed, so you can type it in and play with it to get an idea of how it works, before putting your own figures into it.

To custom-work it for your own business, you need to plug in your own financial information. The first thing you need is your total sales figure from last year (or, if you're just starting out, your projected sales figure). The variable S1 in line 100 holds this figure. To customize the program, change line 100 to read: 100 S1 = XXX (whatever your sales figure is).

Then, you need (your bookkeeper can figure this out) the percentage of your sales that you do each month. For instance, you might do 12 percent of your business in January, 8 percent in February, 11 percent in March and so on, with the total coming out at about 100 percent. You don't need to go to five decimal places, but the more accurate you are, the better are the final figures. In the program, this information is stored in lines 10, 20, and 30, using the variables shown in Table 1.

To customize this program, change lines 10, 20, and 30 to reflect your own monthly percentage figures.

The next figure you need, your cost of sales, is on your financial statement. These are direct costs such as what an item costs you to buy, the labor to install it, etc., as compared to indirect costs, such as your salary, advertising, etc. You put your cost into the program as line 130: 130 S3 = XXX (your cost of sales).

Finally, you need your overhead figure from last year's profit-and-loss statement. We view overhead as a total yearly expense and since this is a profit-projection type of program (as versus a cash-flow forecast), we charge 1/12 of the total overhead to each month. The program will figure the monthly overhead for you; just put your own total overhead in line 110, where S2 = XXX (your overhead).

As a quick example of how you can get your own figures, let's pull out the (imaginary) statement for Courtney's Cookie Consortium (these figures are the ones used in the sample program). See Example 1.

The other variable you'll see in the program, S4, is your gross cost percent. The program computes this for you automatically in line 140, where it divides the gross costs by the total sales. In Example 1, you're dividing 210,000 by 300,000 for a gross cost percent of 70. This means that the direct costs of Courtney's Cookie Consortium average seventy cents out of every dollar in sales. If Courtney could buy her ingredients a little cheaper, or somehow improve her operating efficiency, she could lower this percentage.

Using the Program

When you run this program, you'll note that the total sales, and the total cost of sales, will not add up to exactly what you plugged into the program as S1 and S3. The reason for this is that those figures—S1 and S3—are really just starting points for the program. For example, the program multiplies the total yearly sales (S1) by the monthly percentages of sales and then adds them up to display the yearly total. The percentages for each month shown in the sample program are from our own business and so don't add up to exactly 100 percent. We didn't carry them out enough decimal places because we don't need to be that exact.

```
MONTH     EST SALES   COST/SALES   GROSS $$    O'HEAD      NET
JAN       26550       18585        7965        6666         1299
FEB       25500       17850        7650        6666          984
MAR       20040       14028        6012        6666         -654
APR       14970       10479        4491        6666        -2175
MAY       18120       12684        5436        6666        -1230
JUNE      21720       15204        6516        6666         -150
JULY      29940       20958        8982        6666         2316
AUG       30570       21399        9171        6666         2505
SEPT      21120       14784        6336        6666         -330
OCT       29460       20622        8838        6666         2172
NOV       30809       21566        9243        6666         2577
DEC       31260       21882        9378        6666         2712

TOTALS    300059      210041       90018       80000        10026

THIS IS A SALES SUMMARY PROGRAM, IN WHICH YOU CAN CHANGE
SALES FIGURES TO NOT ONLY DISPLAY WHAT YOU HAVE DONE
BUT ALSO WHAT THE FUTURE MIGHT BRING.

TO CHANGE ANY OF THE FIGURES (SALES, OVERHEAD, GROSS COST)
ANSWER THE QUESTIONS.  REMEMBER THAT TO DECREASE A FIGURE,
JUST PUT A MINUS SIGN BEFORE YOUR ANSWER.  FOR EXAMPLE
     .05 MEANS YOU WANT TO INCREASE THE FIGURE BY 5%,
    -.07 MEANS YOU WANT TO DECREASE THE FIGURE BY 7%.

NOTE ALSO THAT YOU CAN CHANGE ANY OF THE FIGURES
IN ANY COMBINATION--RAISE SALES, LOWER OVERHEAD, ETC.

HIT ENTER TO CONTINUE. . . . .
```

Sample run

M0—January's percentage of sales
M1—February's percentage of sales
M2—March's percentage of sales
M3—April's percentage of sales
M4—May's percentage of sales
M5—June's percentage of sales
M6—July's percentage of sales
M7—August's percentage of sales
M8—September's percentage of sales
M9—October's percentage of sales
N1—November's percentage of sales
N2—December's percentage of sales

Table 1. Variables list

Courtney's Cookie Consortium		Year Ending 12/31/80
Total Sales		$300,000 (S1)
Less cost of sales:		
Materials	$ 110,000	
Labor	100,000	
		− 210,000 (S3)
Gross profit		90,000
Less overhead		★ 80,000 (S2)
NET PROFIT		$10,000

Example 1

Of course, you can get as accurate as you wish when you put your own information into the program.

Now you've taken the four steps needed to customize this program: Lines 10, 20 and 30 hold your own monthly sales percentages; line 100 holds your gross sales as S1; line 110 has your total overhead as S2; and line 130 shows your cost of sales as S3.

When you run this program your sales, costs, overhead, etc., will all be displayed for a 12-month period. This gives you a great reference point to see how you're doing in relation to last year's business, or in comparison to any projections you've made. It also gives you the capability of correcting bad decisions before they hurt too much. For instance, say that you plan to increase your sales 10 percent, and so you raise the overhead (and your salary), by, say, eight percent.

Now, using this program, you can keep an eye on actual sales and your projected sales, month by month. If you find for example, that January's sales are lower than planned, be aware of it. But if February, and maybe March, also produce lower sales, you'd better do something to get your overhead back in line. Now you don't have to wait until the end of the year to find out something like this—you can see it monthly as you run this program.

This program will not only show you where you've been over the past year, but will also give you a good idea of where you might have trouble during the upcoming 12 months.

Like anything we do with financial analysis, this type of information is useless until something positive is done with it. If you find a problem in your business from the projections you make, don't just think, "Yep, got a problem there." Instead, do something to try to correct it (advertise more wisely, work a bit harder at making sales, cut your costs, check you suppliers to try to cut direct costs). You'll soon discover this program is a useful tool that will really help your business.■

Projecting Profits Program

```
1 CLS
5 GOSUB3500
10 M0=.0885:M1=.085:M2=.0668:M3=.0499:M4=.0604
20 M5=.0724:M6=.0998:M7=.1019:M8=.0704:M9=.0982
30 N1=.1027:N2=.1042
100 S1=300000
110 S2=80000
111 REM S1= 1977 SALES; S2= 1977 OVERHEAD
130 S3=210000
140 S4=S3/S1
150 REM  S4= GROSS COST PERCENTAGE OF SALES
200 CLS:INPUT" INPUT ANY CHANGE IN SALES (5%= .05)";A
210 IF A= 0 THEN 250
220 K4 = S1 * (1+A )
230 GOTO300
250 K4=S1
300 CLS:INPUT"ANY CHANGE % IN OVERHEAD (1% = .01)";B
310 IFB=0THEN350
320 K5=S2*(1+B )
330 GOTO400
350 K5=S2
400 REM NEW SALES =K4; NEW O'HEAD = K5
410 CLS:INPUT"INPUT ANY CHANGE IN GROSS COSTS (1%= .01)";C
420 IFC=0THEN450
430 K7=S4*(1+C)
440 GOTO500
450 K7=S4
500 REM K7= COST OF SALES %
510 GOTO2000
600 CLS:PRINT"MONTH";TAB(9);"EST SALES";
601 PRINTTAB(21);"COST/SALES";TAB(33);"GROSS $$";
602 PRINTTAB(45);"O'HEAD";TAB(57);"NET"
650 PRINT"JAN";TAB(8);J1;TAB(20);P1;TAB(32);L1;TAB(44);G9;TAB(56);G1
660 PRINT"FEB";TAB(8);J2;TAB(20);P2;TAB(32);L2;TAB(44);G9;TAB(56);G2
670 PRINT"MAR";TAB(8);J3;TAB(20);P3;TAB(32);L3;TAB(44);G9;TAB(56);G3
680 PRINT"APR";TAB(8);J4;TAB(20);P4;TAB(32);L4;TAB(44);G9;TAB(56);G4
690 PRINT"MAY";TAB(8);J5;TAB(20);P5;TAB(32);L5;TAB(44);G9;TAB(56);G5
700 PRINT"JUNE";TAB(8);J6;TAB(20);P6;TAB(32);L6;TAB(44);G9;TAB(56);G6
710 PRINT"JULY";TAB(8);J7;TAB(20);P7;TAB(32);L7;TAB(44);G9;TAB(56);G7
720 PRINT"AUG";TAB(8);J8;TAB(20);P8;TAB(32);L8;TAB(44);G9;TAB(56);G8
730 PRINT"SEPT";TAB(8);J9;TAB(20);P9;TAB(32);L9;TAB(44);G9;TAB(56);H0
740 PRINT"OCT";TAB(8);K1;TAB(20);Q1;TAB(32);E1;TAB(44);G9;TAB(56);H1
750 PRINT"NOV";TAB(8);K2;TAB(20);Q2;TAB(32);E2;TAB(44);G9;TAB(56);H2
760 PRINT"DEC";TAB(8);K3;TAB(20);Q3;TAB(32);E3;TAB(44);G9;TAB(56);H3
765 FORX=1TO64:PRINT"-";:NEXTX
770 PRINT"TOTALS";TAB(8);W8;TAB(20);W7;TAB(32);W9;
780 PRINTTAB(44);K5;TAB(56);G0
1999 GOTO1999
2000 REM-TO GET MONTHLY FIGURES
2010 J1=INT(K4*M0):J2=INT(K4*M1):J3=INT(K4*M2):J4=INT(K4*M3):J5=INT(K4*M4)
2020 J6=INT(K4*M5):J7=INT(K4*M6):J8=INT(K4*M7):J9=INT(K4*M8 )
2030 K1=INT(K4*M9):K2=INT(K4*N1):K3=INT(K4*N2)
2040 W8=INT(J1+J2+J3+J4+J5+J6+J7+J8+J9+K1+K2+K3)
2050 REM G4 = SALES $$$
2200 REM COST OF SALES= SALES/MONTH (J1) * K7
2210 P1=INT(J1*K7):P2=INT(J2*K7):P3=INT(J3*K7):P4=INT(J4*K7)
2220 P5=INT(J5*K7):P6=INT(J6*K7):P7=INT(J7*K7):P8=INT(J8*K7)
2230 P9=INT(J9*K7):Q1=INT(K1*K7):Q2=INT(K2*K7):Q3=INT(K3*K7)
2240 W7=P1+P2+P3+P4+P5+P6+P7+P8+P9+Q1+Q2+Q3
2600 G9=INT(K5/12)
2700 G1=J1-P1-G9:G2=J2-P2-G9:G3=J3-P3-G9:G4=J4-P4-G9:G5=J5-P5-G9
2710 G6=J6-P6-G9:G7=J7-P7-G9:G8=J8-P8-G9
2720 H0=J9-P9--G9:H1=K1-Q1-G9:H2=K2-Q2-G9:H3=K3-Q3-G9
2750 G0=G1+G2+G3+G4+G5+G6+G7+G8+H0+H1+H2+H3
2800 REM L1= GROSS $$
2810 L1=J1-P1:L2=J2-P2:L3=J3-P3:L4=J4-P4:L5=J5-P5:L6=J6-P6:L7=J7-P7:L8=J8-P8
2820 L9=J9-P9:E1=K1-Q1:E2=K2-Q2:E3=K3-Q3
2825 L1=INT(L1):L2=INT(L2):L3=INT(L3):L4=INT(L4):L5=INT(L5):L6=INT(L6)
2826 L7=INT(L7):L8=INT(L8):L9=INT(L9):E1=INT(E1):E2=INT(E2):E3=INT(E3)
2830 W9=INT(L1+L2+L3+L4+L5+L6+L7+L8+L9+E1+E2+E3)
2900 GOTO600
3500 PRINT"THIS IS A SALES SUMMARY PROGRAM,  IN WHICH YOU CAN CHANGE"
3510 PRINT"SALES FIGURES TO NOT ONLY DISPLAY WHAT YOU HAVE DONE"
3520 PRINT"BUT ALSO WHAT THE FUTURE MIGHT BRING. "
3525 PRINT
3540 PRINT"TO CHANGE ANY OF THE FIGURES (SALES, OVERHEAD, GROSS COST)"
3550 PRINT"ANSWER THE QUESTIONS.  REMEMBER THAT TO DECREASE A FIGURE, "
3560 PRINT"JUST PUT A MINUS SIGN BEFORE YOUR ANSWER.  FOR EXAMPLE"
3570 PRINT"      .05 MEANS YOU WANT TO INCREASE THE FIGURE BY 5%, "
3580 PRINT"     -.07 MEANS YOU WANT TO DECREASE THE FIGURE BY 7%. "
3590 PRINT
3600 PRINT"NOTE ALSO THAT YOU CAN CHANGE ANY OF THE FIGURES"
3610 PRINT"IN ANY COMBINATION--RAISE SALES, LOWER OVERHEAD, ETC. "
3630 PRINT"HIT ENTER TO CONTINUE. . . . . ";:INPUT Z
3640 RETURN
```

TOTAL PERSONAL FINANCIAL MANAGEMENT

A PACKAGE OF SEVEN TRS-80 PROGRAMS THAT CAN HELP YOU CONTROL YOUR FINANCIAL DESTINY. (COST IS DEDUCTIBLE!)

1. **INCOME TAX CALCULATOR**—Use this program to figure your income tax. Screen input is easy to follow. Output to screen (or printer if you have one). Input items can be changed without entire rerun. Vary assumptions and minimize your tax! Input follows Federal 1040.

2. **REAL ESTATE TAX SHELTER**—Enter information regarding rental property and computer projects cash flow and tax consequences. Vary assumptions to maximize return. Computes amortization and depreciation.

3. **ESTATE ADEQUACY**—This program determines the adequacy of your estate to provide adequate support to your family should you die prematurely. You input family members ages, needed income, college expenses, assets in estate, insurance, etc. —Computer determines adequacy of assets. Vary assumptions and percent return on investments.

4. **ESTATE TAXES**—Use this program to figure your estate tax and minimize. Two modules: One is simple computation—The other is more detailed. Useful for both novices and experienced estate tax planners.

5. **CASH FLOW**—Project your monthly cash flow by entering items of income and outgo. Use this with Program No. 7 to plot out a savings and investment program.

6. **FINANCIAL STATEMENT**—This program prepares a personal financial statement for you. Find out how much you're really worth!

7. **GROWTH OF SAVINGS**—Input amount of money to invest in savings, interest rate, and watch the computer show you how much interest you earn and how the principal grows, for as many months or years as you wish.

COMPLETE INSTRUCTIONS INCLUDED

Printer can be used, but not necessary.
Level II 16K required.

TOTAL PACKAGE OF SEVEN PROGRAMS—$35.00

THIS ENTIRE PACKAGE OF PROGRAMS IS PRICED AT $35.00 FOR CASSETTE USERS AND $40.00 FOR DISK USERS.
INDIVIDUAL PROGRAMS ARE NOT SOLD SEPARATELY.
SEND CHECK OR CHARGE TO VISA OR MASTERCARD—ONE WEEK DELIVERY.
INSTRUCTION BOOK INCLUDED IN PACKAGE IS ALSO SOLD SEPARATELY FOR $1.00.

FINANCIAL COMPUTER SYSTEMS ✓115
Suite C-204 • 4651 Roswell Road, N.E. • Atlanta, Georgia 30342

✓Reader Service—see page 290

This software package is no bomber.

B17

Stewart E. Fason
Via La Selva
Palm Beach, FL 33480

B17
ABS Suppliers
Ann Arbor, MI
$22

Recently, I heard of an extraordinary program. I called the bright young man who wrote it and persuaded him to send me a copy to review. Within three days I received it with some comprehensive instructions. Then the fun began: The old computer mastermind closely examined the program, and, unlike many utilities I've examined, found that it is everything it's cracked up to be.

ABS claims that the B17 will allow you to save, verify and load BASIC or machine language programs almost four times faster than an ordinary cassette recorder. Data arrays (string or numeric) can be saved and loaded up to 165 times faster! All of this magic with no aggravation—nearly for free. It works with Level II, 16, 32 or 48K.

I have just spent the weekend using a B17, and it works. In fact, I am so impressed that I paid for my complimentary copy.

Load and Verify

The B17 is both fast and easy to use. Power-up and set memory size to 31600. Have the B17 in cassette ready to load. Type: SYSTEM ENTER B17 ENTER. The asterisk will flash four times, then a series of little arrows will flow showing that something is being input. In about 10 seconds, READY will appear. CLOAD your BASIC program as usual.

Assume you just loaded your Startrek 16K. It takes about four minutes. Now prepare a new tape to record. Type SAVE "STAR" and enter. Note: You must name each program saved. The name can be any six letters or numbers. While you are saving a program, the little arrows flow to your right showing that data is being output. In just over a minute, Star has been saved.

Verify the save by rewinding the tape, typing LOAD? enter. The arrows go the other way, and, if the save was good, the READY appears. I have yet to discover a bad one.

"The old computer mastermind closely examined the program, and, unlike many utilities I've examined, found that it is everything it's cracked up to be."

When you want to load this new, fast Startrek, answer MEMORY SIZE with enter. Type SYSTEM ENTER STAR ENTER. The B17 has appended itself to the BASIC program. You get four asterisks, then the arrows. When the loading is complete and you run the program, the B17 machine language program disappears leaving full memory available. If you want B17 to remain, reserve space at power up (31600). You reserve memory only if you wish to modify and/or save another copy of the BASIC program.

Data Handling

So much for the fun and games. Let's get down to serious business and see what B17 does with data handling. Your Level II manual shows you how to save and input data using the INPUT#-1 and PRINT#-1 statements. The system is easy, but slow.

Here is a sample program which creates 2000 numbers of six and seven digits and saves the whole lot on tape.

```
10 DIM A (2000)
20 FOR I = 1 TO 2000:B = RND (0): A(I) = B:
   NEXT
30 FOR I = 1 TO 2000:PRINT#-1,A(I): NEXT
```

The above does the job very well in about two hours and 20 minutes.

Now, change line 30 to read 30 PUT. Yes, that's all...just PUT. Run the new program.

```
1 REM  YOU MAY USE THIS PROGRAM WITH THE FOLLOWING:
2 REM  1. YOUR TRS-80 (NOTHING ELSE REQUIRED)
3 REM  2. STRINGY FLOPPY (HARDWARE/SOFTWARE REQUIRED)
4 REM  3. TC-8  (HARDWARE/SOFTWARE REQUIRED)
5 REM  4. B-17  (SOFTWARE REQUIRED)
10 CLS
20 CLEAR7000: REM SAVED SPACE FOR STRING VARIABLES
30 DEFSTR A,T:REM DEFINES A & T AS STRING VARIABLES
40 INPUT"STRING=1  NUMERICAL=2";QD:IFQD=1THEN 60
50 CLEAR: REM NEGATES LINE 30 IF DATA INPUT TO BE SOLEL
     Y NUMERIC
60 DIM A(500), B(500): REM LIMITS NUMBER OF DATA ITEMS
70 INPUT"HOW MANY ITEMS TO BE SORTED?";N
80 E=0
90 INPUT"CREATE LIST=1 DATA FROM B17=2 TC8=3 S/F=4 TRS=
     5";S
100 IFS=2THEN110 ELSE IFS=3THEN120 ELSE IFS=4THEN130 EL
     SE IFS=5THEN180ELSEFORI=1TON:GOTO170
110 CLS:INPUT"DATA TAPE IN CASSETTE?  PRESS 'ENTER'";EE
     :GET:FORI=1TON:E=E+1:NEXT:GOTO 530 : REM B17 INPUT
      ROUTINE
120 CLS:INPUT"DATA TAPE READY? PRESS 'ENTER'";EE:OPEN:F
     ORI=1TON:E=E+1:INPUT#1,A(I):PRINTA(I):NEXTI:CLOSE:
     GOTO 530 : REM TC-8 INPUT ROUTINE
130 INPUT"OPEN FILE # 0-9";B: REM STRINGY FLOPPY INPUT
     ROUTINE
140 @OPENB
150 FORI=1TON:@INPUTA(I):E=E+1:PRINTA(I):NEXTI
160 @CLOSE:GOTO 530
170  E=E+1:PRINT"ENTER ITEM #  ";I;:INPUTA(I):NEXTI:GOT
     O190: REM NEW LIST FROM KEYBOARD ROUTINE
180 CLS:INPUT"DATA TAPE READY?  PRESS 'ENTER'";EE:FORI=
     1TON:E=E+1:INPUT#-1,A(I):PRINTA(I):NEXT: REM TRS-8
     0 (SLOW) INPUT ROUTINE
190 CLS:OUT254,1: REM DELETE IF NO SPEEDUP KIT
200 L=1: REM SORTING ROUTINE BEGINS
210 B(L)=N+1
220 M=1
230 J=B(L):PRINT@415,"SORTING":CLS
240 I=M-1
250 IF J-M < 3 THEN 450
260 M1=INT(RND(0)*(J-M))+M
270 I=I+1
280 IFI=JTHEN 370
290 IFA(I)<=A(M1) THEN 270
300 J=J-1
310 IFI=J THEN 370
320 IFA(J)>=A(M1) THEN 300
330 T=A(I)
340 A(I)=A(J)
350 A(J)=T
360 GOTO 270
370 IF I>=M1 THENI=I-1
380 IFJ=M1THEN 430
390 T=A(I)
400 A(I)=A(M1)
410 A(M1)=T
420 L=L+1
430 B(L)=I
440 GOTO 230
450 IFJ-M<2THEN 500
460 IFA(M)<A(M+1) THEN 500
470 T=A(M)
480 A(M)=A(M+1)
490 A(M+1)=T
500 M=B(L)+1
510 L=L-1
515 REM END OF SORTING ROUTINE
520 IFL>0THEN 230
530 OUT254,1:CLS
540 FORI=1TO(N-G)
550 IFI<10THENFG=2
560 IFI>9THENFG=1
570 PRINTTAB(FG);I;"";A(I)
580 Y=Y+1:IFY=15THENPRINT"NEXT PAGE PRESS 'ENTER'";:INP
     UT:Y=0
590 NEXTI
600 M=N-G
610 G=0
615 REM SHOWS AMOUNT OF STRING DATA SPACE YOU HAVE AVAI
     LABLE
620 IFQD=1THENPRINTFRE(A)"BITS LEFT";
630 OUT254,0:Y=0:PRINT"LIST AGAIN=0 SAVE TO TC8=77 TO B
     17=88 TO S/F=99 ":INPUT"TRS80=66 DELETE=-1  TO PRI
     NTER=.9 ADD ITEMS ENTER HOW MANY";UU:IFUU=66THEN79
     0ELSEIFUU=0THEN 530 ELSEIFUU=99THEN 660 ELSEIFUU=-
     1THEN 720 ELSEIFUU=88THEN 780
640 IFUU=.9THEN 710 ELSE IF UU=77 THEN 740
650 CLS:N=E+UU:FORI=E+1TON:GOTO 170
655 REM STRINGY FLOPPY DATA SAVE ROUTINE
660 INPUT"WHICH FILE #(0-9) DO YOU WISH TO WRITE ?";C
670 @OPENC
680 FORI=1TON:@PRINTA(I):NEXTI
690 @CLOSE
700 PRINT:PRINTN;" ITEMS HAVE BEEN SAVED ON S/F":INPUT"
     PRESS ENTER TO RETURN TO MENU";VV:GOTO 630
710 FORI=1TON:LPRINTA(I):NEXTI:PRINTN;" ITEMS HAVE BEEN
      PRINTED.":CLS:GOTO 630
720 CLS:PRINT:PRINT:INPUT"HOW MANY ITEMS DO YOU WISH TO
     DELETE?";X
730 FORI=1TOX:INPUT"ENTER ITEM NO.";Q:A(Q)="[":NEXTI:G=
     G+X:E=E-X:GOTO 190
740 CLS:INPUT"READY TO RECORD?  PRESS 'ENTER'";EE:OPEN:
     FORI=1TON:PRINT#1,A(I):NEXTI:PRINTN;" ITEMS HAVE B
     EEN SAVED ON CASSETTE."
750 CLOSE
760 INPUT"RETURN TO MENU PRESS ENTER";SQ:GOTO 630
770 REM B-17 DATA SAVE ROUTINE
780 CLS:INPUT"READY TO RECORD?  PRESS 'ENTER'";EE:PUT:I
     NPUT"TO RE-INITIALIZE PROGRAM=1  RETURN TO MENU=2"
     ;SF:IFSF=1THEN10ELSE 630
785 REM ARCHAIC, SLOW, FRUSTRATING, TRS-80 DATA SAVE RO
     UTINE!
790 CLS:INPUT"READY TO RECORD?  PRESS 'ENTER'";EE:FORI=
     1TON:PRINT#-1,A(I):NEXT:PRINTN;" ITEMS HAVE BEEN S
     AVED TO TAPE":GOTO630
```

Program Listing 1

(Have your cassette ready to record.) I assume you have B17 in memory. The little arrows dance across the top right corner of your screen for about 20 seconds, and the cassette stops.

There is no way to verify a data save but, using good quality tape, I have had only one bad save in 500. Now, let's try to get the saved data back into memory with this program:

```
10 DIM A (2000)
20 GET
30 FOR I = 1 TO 2000: PRINT A (I): NEXT
```

Your tape has been rewound and set for play. Run the above and watch the arrows flow for about 20 seconds. Then that data will be printed. The B17 GETs and PUTs data faster than the TRS-80 can print it. String data is handled as easily as the above numeric data, however, you must clear the space you need.

The B17 will never replace disks as it will not search for a particular piece of data, however, it has many useful applications.

Program Listing 1 is a natural for the B17. You can use it with a Stringy Floppy or TC-8 as well. This program lets you create a new file, input a previously saved file, add or delete, save or output to your printer. It also sorts—darn fast for BASIC—50 items in about 15 seconds.

In my business I must keep track of over 500 clients. This program does so alphabetically and numerically (by account number). In a matter of minutes my secretary can add, delete, sort and present me with an updated list.

Many of my friends have used this program sucessfully. A group of attorneys use it to keep track of court appearance, filing dates, etc. A series of entries might look like Table 1.

Saving 60-75 percent of the time you spend loading programs makes the B17 well worth the money. Its data handling speed makes it a steal. The introductory price is $22, but by the time you read this, it may be $50 and well worth it.

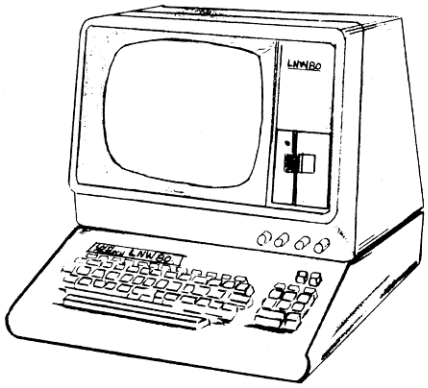
One note: If you have the cassette mod from Radio Shack, you must add one switch in order to bypass the mod. The B17 won't work with any added audio processing devices.■

00	Judge	Client	Remarks	Attorney
80-09-12	Jones	Smith	File Answer	Hookum
80-09-12	Gray	Ajax Const.	Pre Trial Hearing	Skinnem

Note: The above will printout with the heading on top no matter where it is placed in the program. When the computer sees that first zero and finds no other items beginning with zero, it places that item first. The dates will arrange numerically.

Table 1. Sample Entries for Sorting Program

THE FIRST TRS-80® COMPATIBLE COMPUTER WITH HIGH DENSITY COLOR GRAPHICS!



LNW RESEARCH introduces the LNW80, a high performance color computer, compatible with the TRS-80™ Model I. The fully integrated LNW80 is a sophisticated and versatile microcomputer with the following powerful features.

COMPATIBILITY

Hardware and software compatible to the Radio Shack TRS-80™ Model I computer, provides the widest software base of any microcomputer. Cassette interface; expansion bus.

DISPLAY

Quality upper and lower case display.

Two modes of color graphics, high resolution graphics, 384 x 192 in eight colors—higher density than the Apple II.* Low density color graphics of 128 x 192 are also available in eight colors.

High resolution—black and white graphics—of 384 x 192 mixed with text and TRS-80™ standard graphics.

Reverse video, composite video RF output.

PERFORMANCE

The LNW80 utilizes the fast Z-80A microprocessor which executes at a speed of 4 MHZ—over twice the speed of the TRS-80™ Model I.

LNW80

PC BOARD **$89.95**

NEW — EXTERNAL DATA SEPARATOR
ASSEMBLED AND FULLY TESTED
$14.95
SOME SOLDERING REQUIRED

SYSTEM EXPANSION

AT $69.95 [PC BOARD & USER MANUAL]

- SERIAL RS232C/20mA I/O
- FLOPPY CONTROLLER
- 32K BYTES MEMORY
- PARALLEL PRINTER PORT
- DUAL CASSETTE PORT
- REAL-TIME CLOCK
- SCREEN PRINTER BUS
- ONBOARD POWER SUPPLY
- SOFTWARE COMPATIBLE
- SOLDER MASK, SILK SCREEN

Ask about our : Keyboard cabinet Leedex VIDEO 100-80

LNW RESEARCH

✓53

LNW RESEARCH 3183-E AIRWAY AVE COSTA MESA CA 92626 714-641-8850

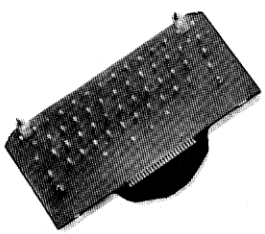
ORDERING INFORMATION

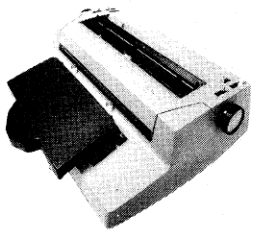
Add $3 for postage and handling. CA residents add 6% sales tax

VISA master charge

*Apple II is a TM of Apple Computer, Inc
TRS-80 is a TM of Tandy Corp

At last --- the DYNATYPER TYPEWRITER INTERFACE!





Turn your electric typewriter into a low cost, high quality hard copy printer. 1 year warranty

DYNATYPER – Rochester Data's patented* Computer/Typewriter Interface is the industry standard for typewriter output.

- 2 minutes to initially install and 5 seconds to remove or replace.
- You do not have to modify your typewriter. All factory warranties and maintenance agreements on your typewriter will be honored.
- Compatible with all power carriage return typewriters having standard U.S. keyboard. The Dynatyper works with Selectrics (model 1) and most non-Selectrics (model 2). Please specify. Typewriter conversion between models takes 2 minutes and the kit (17 plungers) is available for a nominal fee.
- The Dynatyper is compatible with all major word processing software. (Scripsit, Pencil, Applewriter, Easywriter, Magic Window, Visi-calc CCADBM, Supertext, Write On)
- Interfaces available for TRS-80, APPLE, PET/CBM, OSI, Northstar, HP-85, H-89. Weighs only 3 lbs. Extremely portable.
- Delivery: Stock to two weeks. Price $499 for complete system. F.O.B. Rochester, Domestic. VISA and Master Card accepted. Call Ken Yanicky at 716-244-7804

*Pat. Pending

✓468

ROCHESTER DATA incorporated

3000 Winton Road South, Rochester, N.Y. 14623

More CFR Spectaculars!!!

ASCII Parallel Input *SELECTRIC Typewriter/Printer

TRS-80 COMPATIBLE

TYPE ELEMENT NOT INCLUDED

Only $699.00

- Compatible w/Printer Cable or Expansion Interface
- Refurbished Selectric Printer
- Parallel Input, 15" Frame, 132 Cols.
- Upper & Lower Case Typewriter/Printer
- Complete & Ready For Use
- Corresp. (Office) Code
- Data Included
- Fully Formed Characters, Approx 15 CPS

Add $20.00 for Handling & Packaging. Pay Shipping When Delivered.

This Month's Specials

SHUGART 901 8" Floppy Disk Drive
Used, whole untested. Data & Schematics included. ..... 2/$499.00 Plus Shipping

SHUGART SA400 5¼" Mini Floppy
Used, whole, untested. Data & Schematics included ..... 2/$299.00 Plus Shipping

CDC9400 (MPI BR-803) 8" Floppy Drive
Soft Sectored. Used, whole, untested. Data & Schematics included ..... 2/$399.00 Plus Shipping

12" VIDEO MONITOR Motorola #XM-351
80x24 or 16x64 Display. Requires only Composite Video input and 110/230 VAC, 50/60 Hz. Tested, functional & guaranteed. Data and schematics included. ..... $95.00ea. Plus Shipping

9" VIDEO MONITOR Motorola #XM-227
Same characteristics as above 12" video. Tested, functional and guaranteed. Data & Schematics included. ..... $79.00ea. Plus Shipping

Call For Our Bargain-Packed PERIPHERAL FLYER!!!
Prices Are Subject To Change On All Items

CFR Associates, Inc.

✓316

MAIL ORDERS: Box 144, Newton, N.H. 03858

PHONE ORDERS: 617/372-8536 (Sorry, No Collect Calls) Mastercharge & VISA Welcome

WAREHOUSE: 18 Granite St. Haverhill, Mass. 01830

Think of our users: supply a trainer module with your next program.

Better Documentation

George C. Ferber
106 E. Court Street
Fairmount, IL 61841

One of programming's major problems is that of providing concise, yet effective, instructions. Currently, three forms are used—flowcharts, printed instructions, and monitor prompts.

Monitor prompts must be painfully brief because of the monitor's space limitations. These instructions are built into the program's code.

Additional user assistance can be provided in a trainer module, which can offer more elaborate instructions.

Developing a Trainer

When developing a trainer, a number of line messages are necessary. Thus, a training program requires a short auxiliary program that will perform such a task. The auxiliary program's code does not accompany the main program, but rather is a utility used in its production.

The disk file containing the message becomes a part of, and must accompany, the trainer module.

One useful utility is a message generator (Program Listing 1.) This utility accomplishes the following:

1) Identifies message with its disk file name.
2) Enters 63 character lines of text.
3) Establishes a capacity of 132 lines.
4) Enters "eof" to terminate.
5) Writes file to disk where trainer can access it.

Each 63 character line becomes a record. N records occupy the disk file, plus record N + 1, the end of file marker. The total message may contain blocks of lines needed at different points in the trainer module. These blocks of adjacent lines may be sequentially accessed as required.

The trainer obtains the total message from its disk file, counting the number of lines in the process, storing the count in integer variable R. A block locator and processor routine designed to select a block and display it where needed is included.

The locator employs an integer variable, for example C, which will be used to hold the line numbers in each sequential block of lines. For want of a better description, call variable C "the stepper." An integer variable, for example I, that will serve as an incrementer, is also used.

The routine must sequentially scan R records. Just prior to the NEXT RECORD statement, increment I by C. I = I + C. The FOR statement must include FOR X = I + 1 TO I + C. The message processing code is included between the FOR and the NEXT statements.

Now, assume at line 42 of the trainer module that you need the first four lines of the message displayed.

1) Place the value 4 in the stepper variable C.
2) Insert a program line that effects a branch to the block finder and processor routine and returns.

```
42 C = 4:L = 128:GOSUB ROUTINE
```

The routine will access lines 1, 2, 3 and 4, process them, including some form of display, leaving the routine's incrementer C with a value of 4 in it.

Next, assume line 62 of the trainer module requires the next three lines of the file's message. Repeating the procedure:

1) Place the value 3 in the stepper variable C.
2) Insert program line:

```
62 C = 3:L = 128:GOSUB ROUTINE
```

The routine will access lines 5, 6, and 7, process them, leaving the routine's incrementer I with a value of 7 in it. In this manner, all of the blocks can be accessed sequentially to the end of the file.■

```
PAGE # 1 OF TEMP @ 55  CHARACTERS/LINE

10 CLEAR9000:DIMA$(132)
12 CLS PRINT@210,"MESSAGE GENERATOR
                 (AN AUXILIARY PROGRAM)
        1. ASSEMBLES 63 CHARACTER LINES
        2. ACCEPTS SPACES, QUOTES, COMMAS, COLONS, ETC"
14 PRINT@518,"3. ACCEPTS UP TP 132 LINES
      4. ENTER
        EOF  TO TERMINATE":FORD=1TO2500:NEXTD
16 CLS:PRINT@276,:INPUT"ENTER FILENAME ";NAM$:CLS
18 PRINT@768,CHR$(31):PRINT@960,CHR$(31):I=I+1
20 PRINT@768,"LINE";I:PRINT@808,"(ENTER  E  TO REDO)"
22 PRINT@832,STRING$(63,"#"):LINEINPUTA$(I):IFA$(I)="E"

    THEN24ELSE26
24 I=I-2:GOTO18
26 IFA$(I)="EOF"THEN30
28 PRINT@640,A$(I):GOTO18
30 OPEN"O",1,NAM$:FORX=1TOI:PRINT#1,A$(X):NEXTX:CLOSE1
32 CLS:PRINT@276,"!  !   DONE   ! !"
```

Program Listing

If your setting records dubbing around with tape. . . have we got a fix for you!

Babydub

Dennis Bathory Kitsz
Roxbury, VT 05669

The mail arrives just in time for the computer club meeting, and in it is the long awaited Clobberbrains I/Q Program. You recall the ad; 'Sorts your files even before you enter them! Only $49.95!' As club members arrive, you power up the TRS-80, pop the cassette into the player, and. . .

You've missed most of the meeting trying to load the tape. C* errors appear again and again, and when you finally do get it loaded, it crashes immediately. And it hasn't loaded since. Now what?

Babydub is a short program which copies most cassette programs up to 15.5K bytes long. The machine language version loads in five seconds and stays in place, allowing any number of programs to be backed up.

Babydub improves the cassette-load process used by earlier TRS-80's, allowing a somewhat greater variation in the volume setting on the tape player. The resulting copy will be identical in content with the original, but copied on your tape player with the volume setting your TRS-80 responds to best.

Some other uses of Babydub are: preparation of condensed versions of editor/assembler object codes, eliminating silent breaks during assembly; quick examination of tapes being loaded; and recovering lost programs or program names.

How It Works

This program does not load tape as a program, but as a series of bytes. The leader 'buzz' is searched, and the synchronization byte is found. The program bytes are then input in order (including the bytes of code used by the TRS-80 to guide its own program loading sequence). An on-screen copy of the program is presented. If a break in the data greater than 1.2 seconds is present, Babydub assumes the program is complete. Babydub does not verify the validity of a program tape.

The bytes marked ** in the following listings represent the most significant bytes of the origin address; you should use 4300 hex for the lowest RAM position that will not interfere with ROM patches, or if you do not plan to return to BASIC at any point, you may start at 4000 for maximum program length. The BASIC version must be used as written for it to be able to execute without wiping itself out as it begins to run.

Create the program presented in Listing 1 or 2. Use T-Bug or equivalent to prepare the object code, or enter the BASIC version as shown. The BASIC version is a duplicate of the machine code version for those without access to T-Bug.

Load your program from tape. RUN the BASIC program or enter /17152 in response to the second SYSTEM prompt. The statement 'Enter for Load' will appear on the screen. Place the program to be backed up in your recorder (set to play), and press Enter. The data will load into memory and simultaneously run across the screen.

At the conclusion of the load, the screen will clear and again prompt you with 'Enter for Dump.' Place a fresh tape in the recorder (set to record), and hit Enter. A copy will be dumped onto the tape, following which the computer will be returned to BASIC command level.

What You See

As you copy a BASIC program, you will see some familiar and some quite unfamiliar characters on the screen. Why are there graphics characters present in the middle of a BASIC program? The answer to this question will teach you something about the operation of BASIC and allow you to control elements of your BASIC programs.

The commands used in BASIC are not stored as whole words. For reasons of speed and economy of memory, they are turned into byte-sized tokens; PRINT is decimal 178, INPUT is decimal 137, and so on. Level II BASIC translates them back into readable words for purposes of the LIST command. Graphics characters are represented by the very same numerical figures as the tokens!

In program memory, the TRS-80 understands them to be commands, but on the screen it uses them to create graphics. Babydub copies the tape directly onto the screen, so the commands appear as various graphics blocks.

After a while you will be able to identify these, and they will help you scan the program being duplicated for obvious errors.

There is no room in this article for a complete rundown of the inner workings of machine language tapes, but take a look at Fig. 1. All tapes begin with 128 spaces of byte 00 (displayed on the screen as '@'), giving the tape player and amplifier time to

MACHINE VERSION

```
**00    21 B3 ** 11 00 3C 01 0E 00 ED B0 21 0E 3C 13 01
**10    F0 03 36 80 ED B0 3A 40 38 FE 01 20 F9 AF CD 12
**20    02 CD 96 02 11 FF 7F 18 3C C5 E5 06 08 C5 F5 21
**30    FF FF 2B 7C B5 28 36 DB FF 17 30 F6 06 41 10 FE
**40    CD 1E 02 06 50 10 FE 06 14 DB FF 10 FC 47 F1 CB
**50    10 17 F5 CD 1E 02 F1 C1 10 D3 E1 C1 77 12 23 1B
**60    0B 78 B1 20 C4 21 00 3C 01 FF 03 18 BC F1 C1 C1
**70    E1 D5 CD F8 01 21 C1 ** 11 00 3C 01 0E 00 ED B0
**80    21 0E 3C 13 01 F0 03 36 80 ED B0 3A 40 38 FE 01
**90    20 F9 AF CD 12 02 CD 87 02 D1 21 FF 7F 7C 92 47
**A0    7D 93 4F 7E CD 64 02 2B 0B 78 B1 20 F6 CD F8 01
**B0    C3 19 1A 45 4E 54 45 52 20 46 4F 52 20 4C 4F 41
**C0    44 45 4E 54 45 52 20 46 4F 52 20 44 55 4D 50 00
```

** is the MSB of the origin address

Use 43 for lowest RAM position that will not interfere with called subroutines from ROM

You may enter this program with BABYBUG in the following manner:

Use origin address 5000 (hex)). Enter the following bytes:

21 0E 50 11 00 43 01 D0 00 ED B0 C3 00 43

Follow these immediately with the complete machine program above, replacing the ** positions with byte 43 (hex). This will effect a transfer of the dubbing program to address 4300 (hex) upon execution.

First prompt:

ENTER FOR LOAD.

Load the tape to be copied into the cassette player, set to play, and press ENTER. The data will load into memory, and also wipe across the screen. With BASIC programs, this is often a valuable aid in assuring a proper load.

Second prompt:

ENTER FOR DUMP.

Load a new tape into the cassette player, set to record. Press ENTER. The data will dump onto the tape.

Last prompt:

READY>_

You have returned to BASIC. To use this program again, enter SYSTEM, followed by /17152. This will return the BABYDUB program to activity.

Listing 1.

BASIC VERSION

```
10   FOR X = 20480 TO 20701 : READ A : POKE X,A : NEXTX
20   POKE 16526,0 : POKE 16527,80
30   PRINTUSR(0)
90   DATA 33,14,80,17,0,67,1,208,0,237,176,195,0,67
100  DATA 33,179,67,17,0,60,1,14,0,237,176,33,14,60,19,1
110  DATA 240,3,54,128,237,176,58,64,56,254,1,32,249,175,205,18
120  DATA 2,205,150,2,17,255,127,24,60,197,229,6,8,197,245,33
130  DATA 255,255,43,124,181,40,54,219,255,23,48,246,6,65,16,254
140  DATA 205,30,2,6,80,16,254,6,20,219,255,16,252,71,241,203
150  DATA 16,23,245,205,30,2,241,193,16,211,225,193,119,18,35,27
160  DATA 11,120,177,32,196,33,0,60,1,255,3,24,188,241,193,193
170  DATA 225,213,205,248,1,33,193,67,17,0,60,1,14,0,237,176
180  DATA 33,14,60,19,1,240,3,54,128,237,176,58,64,56,254,1
190  DATA 32,249,175,205,18,2,205,135,2,209,33,255,127,124,146,71
200  DATA 125,147,79,126,205,100,2,43,11,120,177,32,246,205,248,1
210  DATA 195,25,26,69,78,84,69,82,32,70,79,82,32,76,79,65
220  DATA 68,69,78,84,69,82,32,70,79,82,32,68,85,77,80,0
```

Line 90 transfers the machine program from its initial address of 5000 (hex) to 4300 (hex). RUN the program.

It will prompt you with 'ENTER FOR LOAD'. Load the tape into the cassette player, and press ENTER. It will now load into memory, wiping the input information across the screen as it loads. This often helps identify the correctness of the load.

At the completion of the memory load, you will be prompted with 'ENTER FOR DUMP'. Load a cassette onto which the program is to recorded, and press ENTER. At the completion of the dump, you will be prompted with a READY>_. DO NOT attempt to RUN the BASIC program again; it has been overlaid with the BABYDUB program.

Instead, if you wish to load and dump additional programs, enter SYSTEM, followed by /17152. You are now ready for another load and dump, prompted as before. You may continue to enter SYSTEM and /17152 for each each subsequently desired dump.

Listing 2.

build up speed. The computer is searching for a symmetrical pattern of ones and zeroes to latch on to. The symmetrical pattern is hex A5, or binary 1010 0101. It is now in sync and prepared to load the next byte. If it finds hex 55 (displayed as the letter U), it knows a machine program follows, and branches to that subroutine in ROM.

This subroutine searches for the six characters that make up the program's name. You can sort out lost or mixed up tapes at this point by watching the

↙ *Reader Service—see page 290*

screen display. If the name is shorter than six characters, the last spaces are filled with blanks (hex 20). During a normal SYSTEM program load, the computer then goes on to seek the code for section start or end of program.

Section start is indicated by hex 3C, followed by a byte giving the number of program bytes to be loaded (not greater than 256, or hex [1]00, per section of program).

It will next find two bytes telling it where to place the program in memory, least significant byte first. When it digests that information, it is prepared to load the program section itself.

After the program bytes are loaded into memory, the computer finds the checksum byte, which is a single-byte verification that the load was probably correct. All program bytes (plus the starting address bytes) that are brougt into memory are totalled inside the microprocessor, disregarding the carry. The result is the checksum; that unnerving C that is displayed during a system load means the loaded bytes didn't match the recorded checksum. Listing 3 presents a sample checksum of six bytes read in from tape.

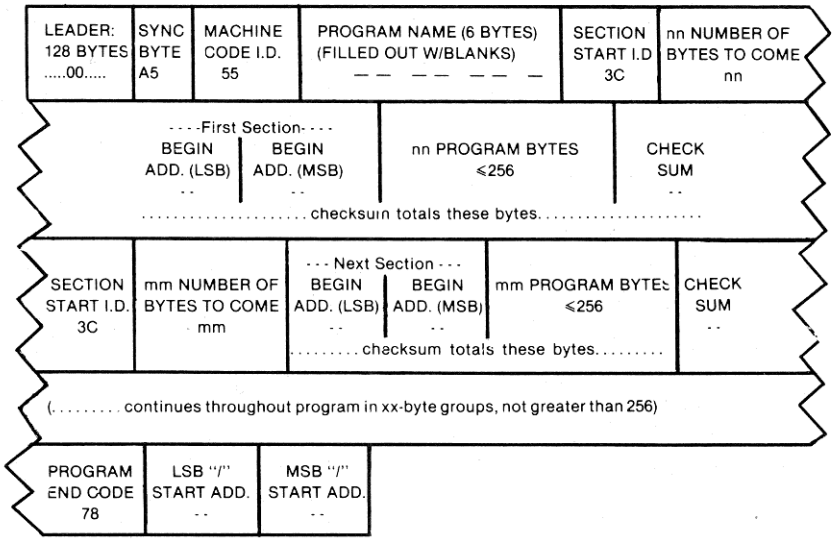


Figure 1.

Finally, the computer will loop back to search for another section start code, or the program end code (78 hex). The remaining two bytes of the load are the program's entry address, used when you enter a slash ('/') in response to the *? prompt.

By contrast, the format of a BASIC program is quite simple: after leader and sync byte, three hex B3 bytes signal the start of a BASIC program. There is the

BYTE READ FROM TAPE:	56 hex	binary:	01010110
NEXT BYTE READ FROM TAPE:	22 hex	binary:	00100010
FIRST SUM		binary	01111000
NEXT BYTE READ FROM TAPE:	AA hex	binary:	10101010
NEXT SUM:		binary:	100100010
	note ignored carry 1. . .		↑
NEXT BYTE READ FROM TAPE:	CI hex	binary:	11000001
NEXT SUM		binary:	11100011
NEXT BYTE READ FROM TAPE:	80 hex	binary:	10000000
NEXT SUM:		binary:	101100011
	note ignored carry 1. . .		↑
NEXT BYTE READ FROM TAPE:	33 hex	binary:	00110011
NEXT SUM:		binary:	10010110

Thus, the checksum of 56 + 22 + AA + CI + 80 + 33 would be 96, or binary 10010110. It's very simple to do, yet gives some continuing assurance that the program is probably going into memory correctly. The reason I say *probably* is because if the third byte above happened to be *2A* instead of AA, and the fourth byte above happened to be *41* instead of CI, the checksum would still tally—and the loaded program would be wrong. Try it. By the way, a correct checksum is not an essential part of the format—as long as *some* value is in the checksum byte position, the tape will continue to load.

Listing 3.

one-character program name; two bytes giving the line's end address in hex; and two bytes giving the line number in hex. The commands, data, etc., follow, and the line is ended by a single zero byte. The process continues, and the program load ends with three zero bytes in a row. There is no checksum at any point, which is why you can get garbage loads with no warning.

If you would like to make more than one copy, or have finally loaded a tape only to accidentally begin to dump it without a tape in the recorder, there is an inconvenient yet serviceable way to accomplish a recovery. When you are back at BASIC command level, type:

```
SYSTEM      [ENTER]
/17269      [ENTER]
```

The computer will return the Enter for Dump prompt. Make sure the tape recorder is in record mode and that you can monitor the sound, and hit Enter. Listen to the program dump *carefully.* When you hear an *alternating set of octave pitches* of leader length, the whole program has been dumped. You must press reset to return to command level.

The Endless Load

You will occasionally want to keep a tape loading in order to make a single dump of several programs, or to copy tapes made up of large sections divided by long blanks. To do this, you'll have to modify Babydub according to Listing 4. Load this version using the same starting address. Programs can be loaded one after another but after the first one, may not appear to be in sync. Data will look like garbage. This is because the leader and sync information of subsequent programs is being copied along with program data, rather than being used as true sync information. Ignore this apparent problem; Babydub will dump this data as is, allowing future loads to synchronize properly.

To dump the collection of taped programs, hit reset, enter SYSTEM and /17269, and follow the instructions above for making subsequent dumps.

For Larger Systems

Changes for 32K and 48K systems are shown in Listing 5. Other than these, it is important to note that unless you have modified your expansion interface, reset does not function, so both the endless load and multiple-copy features are not useable.

Babydub is not intended to assist in the plunder of commercial programs. Duplication of programs other than as back-up copies is theft of an author's hard work. ■

BABYDUB NOTES: THE "ENDLESS LOAD"

Rewrite these lines of the machine version of BABYDUB this way

```
**20 02 CD 96 02 11 FF 7F 18 3C C5 E5 06 08 C5 F5 00
**30 00 00 00 00 00 00 00 DB FF 17 30 F6 06 41 10 FE
```

Rewrite these lines of the BASIC version of BABYDUB this way

```
120 DATA 2,205,150,2,17,255,127,24,60,197,229,6,8,197,245,0
130 DATA 0,0,0,0,0,0,0,219,255,23,48,246,6,65,16,254
```

(THE ONLY PROGRAM CHANGES ARE THE UNDERLINED BYTES)

Listing 4.

Machine version: 32K changes bytes 4326 and 439C to BF; 48K changes these bytes to FF.

BASIC version: change the fourth line of DATA (line 120), 7th value, to 191 or 255 (32K or 48K)
change the eleventh line of DATA (line 190), 13th value to 191 or 255 (ditto).

Listing 5.

✓ Reader Service—see page 290

Monitor the status of your CPU without being exposed to all that video.

See No Evil

Robert Macri
Barbara Gregory
452 W. San Rafael
Colorado Springs, CO 80905

We all have some type of video monitor. It's standard equipment today.

I prefer not to use my monitor, however, when lengthy control applications are involved. Consequently, I often wonder if the CPU is hung up, and if not, at what stage is my program?

Adding an LED display allows me to turn off the monitor but still keep an eye on things. I have built such a display into my console, controlling it with software.

Circuit Operation

The key to this circuit is using the DM 8880 decoder-driver IC (Fig. 1).

Originally designed for high voltage gas-filled tubes, the driver can perform its function with state-of-the-art, seven-segment LED displays.

The resistor used with the driver determines the sinking current of the seven individual constant-current sources. This arrangement prevents shifts in segment brightness because of the number of lighted segments. A one kilohm value insures each segment operates at 2.5 mA. Using a lower value resistor is not recommended.

The 8212 acts as an eight-bit latch with partial decoding. The data byte is latched when inputs $\overline{DS1}$ and DS2 are at low and high levels, respectively. This occurs when address line A0 is high during a negative transition of the OUT control line. Data is then displayed via the BASIC output instruction.

Example 1: OUT 1, 00
Sends Value 00 to the hexadecimal display.

Example 2: OUT 153, 12
Sends decimal value 12 to the output, displaying a hexadecimal OC.

Since the only address line utilized is A0, the specified part must be an odd value, thus placing A0 at a high level.

The mode line (pin 2) allows data stored in the latch to be gated to the decoder-drivers. Because pin 2 is always at a high level, gating occurs spontaneously once data is latched.

Construction

The best location for this circuit is within the console. Since it requires less than 200 mA, the TRS-80 power supply can sufficiently accommodate it. The original panel LED, no longer a necessary power indicator, is disconnected.

Be sure that the most significant digit (MSD) is located to the left of the least significant digit (LSD). Reversal of the two digits is a simple error to make during assembly.

Almost any common anode seven-segment LED display is suitable. To keep the cost relatively low, I used the MAN-1. The MAN-10, although expensive, operates more efficiently and delivers a brighter display. Other good choices include the MAN-64, MAN-72, 707 and 747. Check regularly with surplus houses for low cost components.

Depending on your selection, you may wish to add a high contrast filter. This improves the clarity of the display.

By adding a second circuit (constructed in basically the same fashion as the first) you can show continuously the time and temperature.

Whether you are efficiency-minded or just cost-conscious, an LED display with a monitor is definitely a practical tool. ■

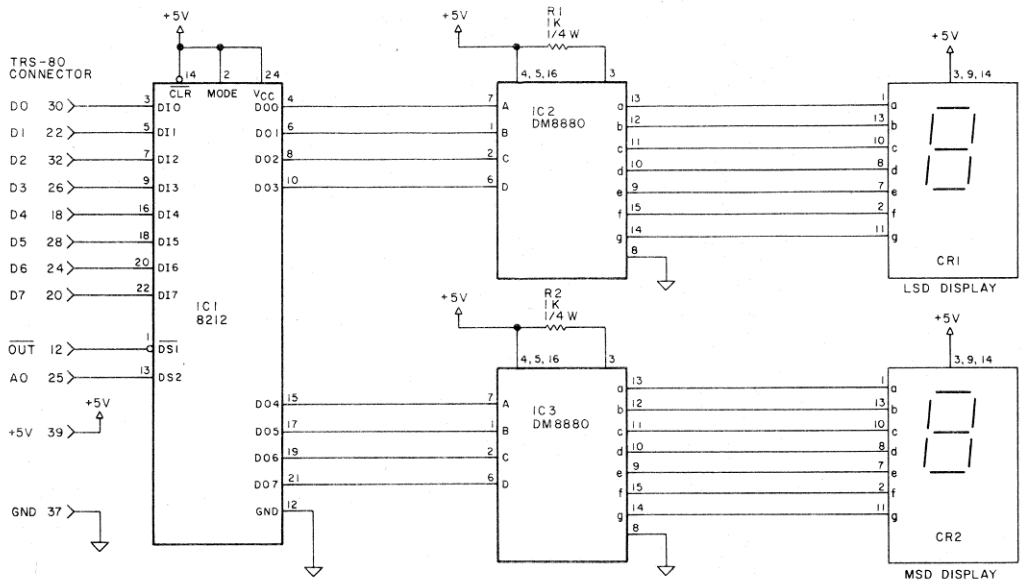
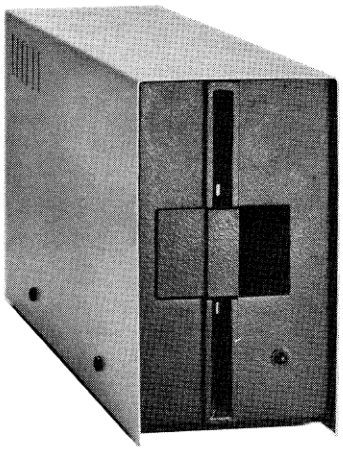


Fig. 1. Circuit Schematic

Mini-Disk Storage Systems for TRS-80* Computers



Access... Offers More. Charges less.

from $295

Check this line-by-line comparison of Access 5-inch mini-disk systems for Model I and Model III computers. *Nobody* offers more benefits, better service or lower prices.

TOLL-FREE ORDER NO.
1-800-527-4196†
(orders and literature only)

Company/Drive Model	Price 1/	Flippy	*Formatted* Diskette Storage Capacity: Dbl-Density	*Formatted* Diskette Storage Capacity: Sgl-Density	100% Tested	48 hr. Burn-In	Warranty	Trial Period
40-TRACK DRIVES								
Access Unlimited								
AFD-100†	$295.00	no	180 Kbytes	102 Kbytes	yes	yes	yes	yes
AFD-100F†	329.00	yes	360 Kbytes	204 Kbytes	yes	yes	yes	yes
MTI								
TF-5	359.00	no	?	?	?	?	?	no
Midwest Comp. & Per.								
MPI/B-51	321.00	no	?	102 Kbytes	?	?	yes	no
Aerocomp								
Mdl 40-1	349.95	yes	?	?	yes	?	yes	yes
CPU Shop								
CCI-100	314.00	no	?	102 Kbytes	?	?	yes	no
AMI								
40-track	325.00	no	?	?	?	?	?	no
80-TRACK DRIVES								
Access Unlimited								
AFD-200†	429.95	no	368 Kbytes	205 Kbytes	yes	yes	yes	yes
AFD-200F†	449.95	yes	736 Kbytes	410 Kbytes	yes	yes	yes	yes
MTI								
TF-8	639.00	no	?	200 Kbytes	?	?	?	no
Aerocomp								
80-tk mdl	459.95	yes	?	?	yes	?	yes	yes
CPU Shop								
CCI-280	429.00	no	?	204 Kbytes	?	?	yes	no
AMI								
80-track	560.00	no	?	?	?	?	?	no

1 As advertised in 80 Microcomputing, Jan. 1981.

Free trial offer

Use your brand new AFD drive system for up to 15 days. If you're not *completely* satisfied, pack it in the original shipping container and send it back to Access. We'll refund the price of the system (less shipping charges), no questions asked. (No refund for misuse or improper handling.)

Operating burn-in test — too

Every drive that leaves Access is not only 100% electrically tested and double-checked for mechanical alignment, but it is also given a full 48-hour operating burn-in test. You'll find a test list — checked and signed by one of our competent technicians — in the drive carton. If a drive has latent defects, the burn-in life test will weed them out. The drives we ship just keep on running. And running. And running.

About our warranty

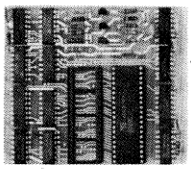
Venus de Milo has about the right number of fingers to count our warranty returns. Nevertheless, your new drive system is covered by our comprehensive 90-day limited warranty. The details are spelled out in the illustrated users manual included with each AFD drive system.

Software galore!

Games. Business programs. System software. We sell many of the best TRS-80* Model I and Model III programs, and at competitive prices. Make Access Unlimited your one-stop shopping center for all of your TRS-80* software, hardware and accessories. Save big! Call our toll-free order number, 1-800-527-4196† for free descriptive literature.

USE YOUR CREDIT CARD AND SAVE! VISA AND MASTER CARD CHARGES ARE NOT DEPOSITED UNTIL THE DAY YOUR ORDER IS SHIPPED. CALL NOW TOLL-FREE, 1-800-527-4196†.

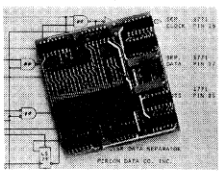
the DOUBLER™



Percom's new plug-in adapter for your Expansion Interface stores almost twice the data on a diskette track as a single-density system. You can store up to four times more data — depending on the type of drive — on one side of a diskette than you can store using a standard Model I mini-disk drive. Other features: Reads, writes and formats *either* single or double density minidiskettes. • Runs TRSDOS*, NEWDOS+‡ Percom OS-80™ or other single-density software *without* changing either software or hardware. Switch to double-density when convenient. • Includes DBLDOS™, a TRSDOS* compatible double-density operating system. • Includes on card, high-performance data separator circuit. • Installs *without* rewiring or trace cutting. • Introductory price, including DBLDOS and format conversion utility, only $219.95.

Permits Model III software to be read on Model I computers.

DATA SEPARATOR™



This PC board plug-in adapter for the TRS-80* virtually eliminates data read errors (CRC error — Track locked out!) which occur on high-density inner disk tracks, a problem that has plagued TRS-80* systems. The Percom Data Separator™ is installed in the Expansion Interface without modifying the host system. Caution: Opening the TRS-80* Expansion Interface may void the limited 90-day warranty: $29.95.

Disk System Interconnecting Cables

Improvement over RS cable design places drive 0, which includes the cable termination, at the end of the cable to eliminate the reflected noise of an unterminated cable. Better data integrity. Prices:

Two-Drive Cable $24.95
Four-Drive Cable 34.95

Power Line Filter

115/250 V, 50-400 Hz. Instructions included for easy installation in standard mini-box chassis: $19.95

Minidiskettes (Double-Density rated)

10 Disks in a convenient plastic organizer box $34.90
Single Disk 3.49

Disk Drive ID Tabs

1" x 1¼" self-adhering plastic drive identification tabs. Compatible silver with engraved black drive number. Two tabs (Nos. 0, 1): $2.50; three tabs (Nos. 0, 1, 2): $3.25; four tabs (0, 1, 2, 3): $4.50.

How to Order Order by calling Access Unlimited toll-free on 1-800-527-4196†. Mail orders also accepted. Orders may be charged to a VISA or MasterCard account or paid by a cashier's check, certified check or money order. We accept COD orders with 25% deposit. Sorry, we cannot accept personal checks. We pay shipping and insurance charges on orders over $1,000.00. Add approximate insurance and shipping charges for under $1,000.00. If in doubt about these charges, ask when you call in your order. Texas residents include 5% sales tax. Minimum order $20.00. Allow 2 to 4 weeks for delivery. †Texas residents call (214) 494-0206.

ACCESS UNLIMITED ✓229

315 N. Shiloh · Ste. D1 · Garland, TX 75042
(214) 494-0206

T trademark of Access Unlimited Company. *RADIO SHACK and TRS-80 are trademarks of Tandy Corporation. ™trademark of Percom Data Company, Inc. ‡trademark of Apparat Corporation.
ALL PRICES AND SPECIFICATIONS SUBJECT TO CHANGE AND ALL OFFERS SUBJECT TO WITHDRAWAL WITHOUT NOTICE.

Level II Manual errata and other stuff.

Between the Lines

Robert McTernan
42 Aspinwall Rd.
Red Hook, NY 12571

During my last few months of programming, I occasionally lost time due to errors present in the Level II BASIC Reference Manual. I called the Radio Shack Hotline to verify whether my problem was due to an error in the manual or my own misinterpretation of it.

This article deals with some of the problems I experienced, and, hopefully, it will trigger similar discussions from you about your particular struggles.

Writing Tape Data

The manual states that data written on tape must not exceed 255 characters, or all characters over 255 will be truncated. Key in this routine and RUN.

```
 0 CLEAR1000:CLS
 5 X$=STRING$(255,"1")
10 PRINT#-1,X$
15 INPUT"REWIND, PLAY & ENTER";X
20 INPUT#-1,Z$
25 PRINTX$:PRINT
30 PRINTZ$
```

The top string of 1s displayed on the screen is the one you tried to write on tape. The bottom string is what you read from the tape just written. Notice the six characters that were lost at the end of the record read from tape. If more variable names had been used in the line 20 statement, more characters would have been lost, due to memory overhead. Keep in mind that the maximum number of characters you can deal with in tape read/write operations is *not* 255, as stated, but 249, and *that* only when a single variable name is used in the tape statement item list.

The Lost Sign

To cut down the number of tape records written, variable-length words of array data can be concatenated into a single string and then written out on tape. For numeric data, a conversion from numeric expressions to strings must precede the concatenation. When read back from tape, the string can be disassembled and reconverted to numeric expressions by using the MID$, VAL, etc., statements. The signs of each number in the tape string serve as data separators in a disassembler routine. But watch out!

Key in this routine and RUN.

```
 0 CLEAR1000:CLS
 5 X=1234:Y=6789
10 X$=STR$(X):Y$=STR$(Y)
15 Z$=X$+Y$
20 PRINT#-1,Z$
25 INPUT"REWIND, PLAY & ENTER";X
30 INPUT#-1,A$
35 PRINTZ$:PRINTA$
40 PRINTMID$(Z$,2,2)
45 PRINTMID$(A$,2,2)
```

The top row of digits displayed is the assembled string Z$ concatenated in line 15. This string is then written on tape and read back as string A$. The second row of digits displayed is A$. Note that the sign of the left value has been lost. Also the results of an identical MID$ operation on both strings are displayed. It appears that leading blanks in a tape string are lost on the TRS-80. To prevent this, make the following changes to the routine and RUN again.

```
Add line 12          Z$=" "
Change line 15 to    Z$=Z$+X$+Y$
```

PRINT@ Where?

One fact not in the manual concerns the PRINT@ statement. Key in the routine and RUN.

```
 0 CLS
 5 X=12345
10 INPUT"ENTER POSITION";Y
15 PRINT@Y,X
20 INPUT"ENTER TO CONTINUE";Z
25 GOTO0
```

At the ENTER POSITION prompt, enter a 57. Print OK? Now enter to continue and enter a 58. Print at 58? Nope! Enter any number from 58 through 63. It appears that a numeric expression will print at a specified position only if the expression, including its trailing blank position, will fit on the line of the position specified. This may cause a problem for some applications, but can be overcome by the following: Change line 15 to PRINT@Y, STR$(X).

PRINTTAB Where?

The manual states that if the specified position of the TAB expression exceeds 63, the data will print on succeeding lines. Using the last routine again, change line 15 to: PRINTTAB(Y)X. At the ENTER POSITION prompt, enter 45. This should print OK. Now enter to continue and enter 80. It printed at position 16 of the print line, right? Repeat and enter 144. Same place! That's unfortunate because the PRINTTAB statement is useful for formatting lines for a printer. Since most printers have more than 64-character line capability, other methods must be used to format past the 64th character.

It appears that the TAB expression, when exceeding 63, is decremented by 64 until the expression is less than 64. Then the expression is printed at that position.

DIMs Anywhere?

The manual states that DIM statements may be placed anywhere in a program. Key in the routine and RUN.

```
 0 CLEAR100
 5 DEFSTRX
10 DIMXY(20,2)
15 XY(15,1)="ABCD"
20 PRINTXY(15,1)
```

If your screen displays ABCD, you're fine. Now, let's swap the statements in lines 5 and 10, then RUN. Got a BS error? When a variable is defined, following a DIM of an array variable name starting with the same character, the dimension of the array is reset to the normal array depth of 10. Therefore, always ensure that DIMs follow DEFs.

The following problem showed up so intermittently that I concluded I had a hardware problem, but actually, string concatenation was the culprit.

As I understand it, the concatenation operation sets up pointers, but doesn't clear them out until string space is exhausted. At that time, 'garbage collection' takes place to clear these pointers. This operation takes a long time. This is done outside of BASIC, which explains why the BREAK key does not work during this apparent hang.

When I called Radio Shack on their hotline, their technician said that the problem was being addressed and would result in a 'freebie' fix. He also advised that reserving more string space would alleviate the problem. A subsequent call discounted the previous statements concerning a fix; it was then stated that this problem is not unique to the TRS-80. I haven't found a solution yet. Anyone else?

After installing the cassette drive circuitry fixes by Radio Shack, installing a TBUFF, purchase and use of quality Maxell tapes, etc., I thought my tape problems were gone. Not so!

Upon using a newly-acquired batch of Maxell C46 tapes, errors again appeared. A check of the tape revealed creases across its width. A check of previously-used C90s revealed none. A detailed response to my inquiry to Maxell stated possible causes of the problem, all cassette drive related. Back to the hotline. Radio Shack's reply follows:

Yes. Certain model CTR-80 drives have this problem.

The symptom is known as 'dimpling'.

Yes. There is a 'freebie' fix.

Just send the drive to the closest Repair Center.

Well, I did and I got the drive back fast with these replies.

Never heard of 'dimpling'.

Nothing is wrong with the drive.

All previous fixes are already installed.

If you are still having problems with tapes, you may want to check for possible creases. However, if you have creases, I can't tell you what to do about it. Incidentally, creases occur immediately upon completion of a tape operation and not from leaving tape loaded in the drive for extended periods of time.

Radio Shack after-sale support leaves a lot to be desired. Articles like this one will have to make up for it. ■

Reader Service—see page 290

Wages Is Here

FOR THE TRS-80

- **Wages** MEETS PAYROLLS
- **Wages** IS GUARANTEED
- **Wages** COST ***ONLY*** $35.00 FOR MODEL I ($45.00 FOR MODEL II)

FROM
The Maine Software Library
P.O. Box 194 ✓268
Standish, Maine 04084

*A Trade name of the Tandy Corp.

LEARNING TOOLS

- Spelling Primer
- Vocabulary Builder
- Crossword Generator

EACH PROGRAM FEATURES:

- HUNDREDS OF WORDS
- GRAPHIC SCORING MONITOR
- PROGRAMMABLE SPEED CONTROL

Each Program Available in 4 Levels:
Grades 1-2; 3-4; 5-6; 7-8.
(Specify Grade)

16K - LEVEL II

$4.95 each — any two for **$7.95**

DYNATEK INFO. SYSTEMS

586 CONCORD AVE.
WILLISTON PARK, N.Y. 11596

✓253 *"Quality Software at Affordable Prices"*

DISASSEMBLED HANDBOOK FOR TRS-80

VOLUME 3—$18. POSTPAID

Chapter 1: Writing Disassembler Programs
Chapter 2: High Speed Disassemblers
Chapter 3: Spooling Theory & Practice
Chapter 4: Port Encoders & Decoders
Chapter 5: Writing Interrupt Programs
Chapter 6: D/A Converters & Construction
Chapter 7: A/D Converters & Construction
Chapter 8: High Speed Morse Code Program
Chapter 9: Comm. Bulletin Board Systems
Chapter 10: Radio Teletype From A to Z
Chapter 11: Self-Programmed Learning Q/A
Appendix A: Volumes 1-2-3 combined index
Appendix B: Vols. 1-2-3 Pgms. on Disk $30

VOLUME 1—$10. POSTPAID
6th printing

VOLUME 2—$15. POSTPAID
4th printing

—GERMAN & FRENCH LANGUAGE EDITIONS—

RICHCRAFT ENGINEERING LTD.
Drawer 1065, Wahmeda Industrial Park
Chautauqua, New York 14722
phone (716) 753-2654 for COD orders
(US funds: add $4.50 overseas airmail) ✓276

16K RAM *PLUS* $36.95

Kit includes 8 tested dynamic RAMs for memory expansion, instructions that make it very easy for anyone that's not experienced at handling, installing and testing RAMs to do the job, a RAM test program on cassette tape, etc. Completely guaranteed for 90 days. Add $2.95 for DIP shunts needed to upgrade 4K mem only.

32K RAM *PLUS* $71.95

You get the 16K kit described above plus 8 additional RAMs.

C-10 TAPES ..... $.59

You get computer quality, low noise C-10 blank cassettes with labels. Add .13 ea for plastic storage cassette box.

SOFTWARE:

RAM MEMORY TEST 1.1 The best test to find "flakey" or defective RAM $2.95

NEWBOOT 1.2 Press a single key to load and run your directory choice, BASIC or CMD programs. Assorted display lists all programs, Gran size, free disk space, etc. Has printout command, etc. for 1 or more drives $19.95

MEDICAL FORMS 1.2 The text editor with prepared office forms as patient history, surgery, etc. Saves the M.D. much time. A much used program by the surgeon $49.95

MasterCharge and Visa Cardholders
Telephone (817) 246-3778
or send check or money order to:
Digital Data Supply
P.O. Box 5061, Fort Worth, Texas 76108 ✓313
(Order now, prices may change next month)

STEPWISE

MULTIPLE LINEAR REGRESSION

Fast Compiled Machine Language
Comprehensive Data Base Manager
  * Transformations
  * Lags
Designate Any Variable As Dependent At Run Time
Descriptive Statistics
Correlation Matrix
ANOVA Table
Partial Correlation Coefficients Each Step
Future Releases Are Upward Compatible And Include: Factor Analysis, Time Series, Linear Programming
For 2 Disk, 48K TRS-80®, With Line Printer
Write For Information On Other System Configurations
Price $89.00 — Documentation Only $10.00
Visa And Master Charge Accepted

Barstrann Corporation — Dept. A ✓201
P.O. Box 265 Mid City Station
Dayton, Ohio 45402

CalData Systems Presents ✓294

WordMagic II

Complete WORD PROCESSING designed specifically for
The Radio Shack TRS-80 Model II Computer

WordMagic II™ is a Word Processor designed specifically for the Radio Shack TRS-80 Model II Computer.*

FEATURES INCLUDE:

- Mailing List/Labels Generation
- Automatic Merging of Mailing Data with Text Files to created "PERSONALIZED" Form Letters
- Automatic wrap-around in text entry
- Margination, Paging, Complete Cursor Movement
- Complete Editing Commands—Insertion, Global Substitution, Overwrite, etc.
- Centering, Smooth Right, Left Justify
- Table of Contents Generation
- Automatic Page Numbering
- Variable Form Lengths
- Underlining
- Line Numbering

TRS-80 is a trademark of the Radio Shack Division of Tandy Corporation

Requires 64K Model II, TRSDOS & BASIC (not provided with WordMagic)
Introductory Pricing: $195.00 Manual $20. (Cal. Res. add tax)

CalData Systems P.O. Box 178446
San Diego, CA 92117 (714) 272-2661

Know The Reasons Why The FORCE-80 is your "Best Buy."

FORCE-80 The computer protector

- Power dropout protection
- Suppresses transient voltages
- Suppresses RFI
- Disc protection
- Factory warranty
- Satisfaction guaranteed
- Direct from factory

FORCE-80

Only $121.50 plus $5.00 shipping & handling

Check, Mastercharge, Visa accepted
(5% discount for cash)
No C.O.D.'s please

P&S Electronics, Inc.
P.O. Box 23014 Dept. EM
Nashville, TN 37202

Allow 6-8 weeks for delivery. ✓406
Add 14 days for personal checks.

SOFTWARE * CRITIC

PROGRAM REVIEWS FOR THE "80'S" *

STOP wasting your software dollars. START getting timely, complete, understandable, objective, and critical software reviews. Subscribe NOW to the SOFTWARE CRITIC. Published bi-monthly, charter subscription only $9 per year. (Regular $15.00) Canada and Mexico $18.00, others $24.00. Unused portion refunded if you're not satisfied. Send check or money order for $9 to:

NEW! SOFTWARE CRITIC
P. O. BOX 3CH
UNIVERSITY PARK,
NEW MEXICO 88003

$9/YR

*Currently MODS I & III ✓220

Money, that's what. Save some with this electrical utility monitor.

Watt's It All About?

Don Hubert
613 Hartless Court
Hampton, VA 23669

If you can't beat 'em and they won't let you join 'em, then you have to outwit 'em.

Everyone is complaining about the spiraling cost of energy these days, but there is little we can do about it, except pay the bill when it comes each month. We are a people who are accustomed to use now and pay later.

Unfortunately, when you use electricity or gas in the home, you can't see how much, or where it is going until the day of reckoning when your bill for payment arrives. Your first inclination is to believe the power company made a mistake—no way could you have used that much electricity in one month. But, alas, it is true and you did indeed use that much. So it's off to your friendly bank to float a loan so you can do the same thing next month.

No Magic Cure

This program is not a magic cure for your energy problems, nor will it give you a better rate for the electricity you consume. It will, however, give you an insight on how much energy you are using as the month progresses. It may lessen the shock when payment comes due.

Maybe you can outwit them.

The program allows you to monitor the amount of electricity used in a given period on every day, every other day, or whatever period you wish. I've found it useful to check what it really costs to run the air conditioner and clothes dryer on typical days as opposed to opening the windows or hanging the clothes out on a nice day. I almost wish both appliances were never invented.

You'll need to know how to read your electric meter. For some strange reason a lot of people think this is a difficult task. Some don't even know where their meter is. Next time the dog barks, go outside and you'll find the meterman.

Once you find the meter the hard part is over. There it is, five smiling faces looking at you (Fig. 1). To read it, keep two important things in mind: (1) When the pointer is between two numbers, record the smaller number, (2) the dial on the far right is always read as zero.

You can always tell which way the dials are turning by looking at the numbers on the dial. To give you an idea of how much a kilowatt hour is, if you have a 100-watt light on for ten hours, you have used 1 KWH (Kilowatt Hour). I told you reading a meter was easy.

The bill you receive each month should show the previous reading and date, the present reading and date, plus the number of days between readings. Mine varies between 29

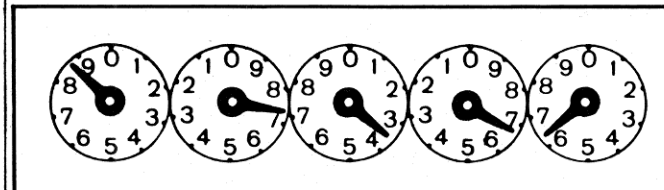


Fig. 1. Meter Reads 87360

and 32 days as a billing period.

Armed with this information, you can even check the accuracy of their computer (so far I haven't found an error in their figures).

My idea of a computer is a device into which I put a little data and it does all the work and gives me some meaningful data in return. To this end, I designed the program, so each time I want to use it, I don't need to gather a lot of input data. To use the program, all you need is two readings of your meter and the number of days between readings. The computer will give you the KWH used to date, total cost to date and average cost per day.

figure the cost, if winter rates are in effect. Lines 260–300 the actual figuring of the amount of electricity used, plus the average cost per day. Line 290 figures the total cost of actual KWHs used to date. The 5 in Line 290 is the five dollar minimum charge levied by the company.

Lines 340–390 print out the KWHs used, actual cost and average cost per day. The GOSUB statement in Line 405 is required to figure out the estimated cost, if consumption is continued at the actual rate.

On return from the subroutine, the estimated KWHs and cost are printed out.

"I've tried to detail the program sufficiently so that no matter what your rates are, . . . you can modify the program to fit your needs."

In addition, it will figure out your total KWH and cost for the billing period, if you maintain this rate of consumption.

The Program

The program is written in TRS-80 Level II using standard functions and statements which should be easily converted to other types of BASICs. The program makes use of the CHR$ function and print-using statement, which adds a little class to the printout.

Line 100 is the image line for the print-using statement that adds the $ sign and shows the cost to two decimal places. Line 110 is needed if the electric rate differs from summer to winter as it does in Virginia, where I live.

Lines 130–170 set up the required input data. Lines 190–230 separate the amount of kilowatts used, according to the rates established by the local power company. Lines 240 and 250 are the cost per KWH as used by the company, VEPCO (Virginia Electric and Power Company), and are current as of 1 Feb 80. Line 270 branches to

Line 391 sets up my billing period as every 32 days, and adjusts the printout if the period is greater. You can change Z to reflect your normal billing period as necessary. REM lines 480–530 show how the cost per KWH is used in the program for summer and winter rates.

Lines 240 and 250 show the cost of power for each category. As the rates change, these two lines must be changed to maintain the program in a current status.

I've tried to detail the program sufficiently so that no matter what your rates are, or how they are applied, you can modify the program to fit your needs.

Conclusion

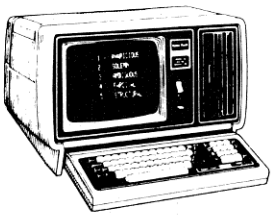
Some experts will tell you it costs less to keep your air conditioner running all day, even though no one is in the house, than it does to turn it off and restart it after you get home after work. I don't know who is right, but I intend to find out.

We may not be able to beat 'em, but we sure can try to outwit 'em.■

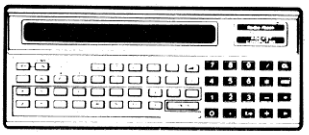
Program Listing

```
10 CLS:PRINT CHR$(23)
20 PRINT"* * *  ENERGY MONITOR  * * *"
30 PRINT
40 PRINT"TO USE THIS PROGRAM, YOU MUST"
50 PRINT"TAKE TWO READINGS OF YOUR "
60 PRINT"METER AND KNOW THE NUMBER OF"
70 PRINT"DAYS BETWEEN READINGS":PRINT:PRINT
80 INPUT"PRESS ENTER TO CONTINUE";Z$
90 CLS:PRINTCHR$(23):PRINT"    OR YOU MAY CHECK YOUR"
91 PRINT"         FINAL BILL":PRINT
92 INPUT"PRESS ENTER TO CONTINUE";Z$;:CLS
93 INPUT"TO CHECK FINAL BILL ENTER 1, ELSE PRESS ENTER"
    ;Y

100 A$="$$#####.## DOLLARS"
101 CLS:IF Y=1 GOTO600
110 INPUT"FOR MONTHS JUNE THRU SEP,TYPE 1,ELSE PRESS EN
    TER";M1
120 PRINT
130 INPUT"ENTER FIRST READING ";R1
140 PRINT
150 INPUT"ENTER SECOND READING ";R2
160 PRINT
170 INPUT"ENTER NUMBER OF DAYS BETWEEN READINGS ";D
175 CLS
180 KW=R2-R1
185 IF FK>0 THEN KW=FK
190 IF KW>600 THEN A=600 ELSE A=KW
200 B=KW-A:IFB>0 THEN B=B ELSE B=0
210 IF B>900 THEN B=900 ELSE B=KW-A
220 C=KW-1500
230 IF C>0 THEN C=KW-1500 ELSE C=0
240 O=.06638:P=.05334:Q=.04289
250 M=.06024:N=.06924
260 TK=A+B+C
270 IF M1=0 THEN 285
280 C1=M*A:C2=N*B:C3=O*C:GOTO 290
285 C1=M*A:C2=P*B:C3=Q*C
290 CD=C1+C2+C3+5
295 IF FK>0 GOTO655

300 CP=CD/D
330 PRINT
340 PRINT"YOU HAVE USED ";TK;" KWH'S IN";D;" DAYS AT A
    COST OF"
350 PRINTUSING A$;CD
360 PRINT
370 PRINT"THATS AN AVERAGE OF";
380 PRINTUSING A$;CP;
390 PRINT"    PER DAY"
391 Z=32:IF D>Z THEN Z=D
400 PRINT:EK=(TK/D)*Z
405 GOSUB1010
410 PRINT"BASED ON THE ABOVE,IT IS ESTIMATED YOU WILL"
420 PRINT"USE ";EK;"KWH'S  THIS MONTH AND YOUR NEXT"
430 PRINT"VEPCO BILL WILL COST YOU "
440 PRINT:PRINTUSING A$;EC;:PRINT"  BASED ON ";Z;" DAY
    READING"
441 PRINT
442 INPUT"DO YOU WISH ANOTHER RUN ,YES OR NO ";R$
443 IF R$="YES" GOTO93
444 CLS:PRINTCHR$(23):PRINT"GOOD-BY FOR NOW"
445 PRINT
450 END
480 REM * * RATES FOR KWH * * *
490 REM M=COST PER EACH KWH FOR FIRST 600 KWH -ALL YEAR
500 REM P=COST PER EACH KWH FOR NEXT 900 KWH -OCT THRU
    MAY
510 REM Q=COST PER EACH KWH OVER 1500 KWH -OCT THRU MAY
520 REM O=COST PER EACH KWH OVER 1500 KWH -JUN THRU SEP
530 REM N=COST PER EACH KWH OVER 900 KWH -JUN THRU SEP
540 END
600 INPUT"ENTER TOTAL KWH'S SHOWN ON BILL";FK
610 PRINT
620 INPUT"ENTER NUMBER OF BILLING DAYS";D
630 PRINT
640 INPUT"FOR MONTHS JUNE THRU SEP, TYPE 1, ELSE PRESS
    ENTER";M1
645 GOTO185
650 GOSUB1010
655 CLS:PRINTCHR$(23)
660 PRINT"YOUR BILL SHOULD SHOW A TOTAL "
661 PRINT:PRINT"COST OF"
670 PRINTUSINGA$;CD:PRINT:PRINT   "FOR ";D;" DAYS"
671 PRINT:PRINT

1000 END:REM FIGURE ESTIMATED COST
1010 IF EK>600 THEN A=600 ELSE A=EK
1020 B=EK-A:IF B>0 THEN B=B ELSE B=0
1030 IF B>900 THEN B=900 ELSE B=EK-A
1050 IF C>0 THEN C=EK-1500 ELSE C=0
1060 IF M1=0 THEN 1080
1070 C1=M*A:C2=N*B:C3=O*C:GOTO1090
1080 C1=M*A:C2=P*B:C3=Q*C
1090 EC=C1+C2+C3+5
1095 RETURN
```

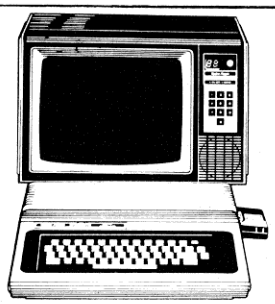
meet a REAL PLUS



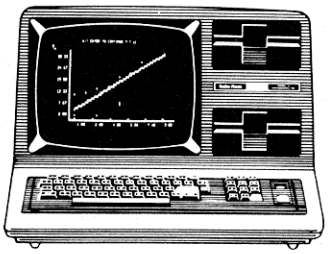
Model II 64K $3375



Pocket Computer $199



Color Computer 4K $310



Model III 16K $859

☐ **check out these unusual package values for price and warranty**

Here are just a few of our fine offers . . . call toll-free for full information.

COMPUTERS	
Model II 64K	$3375
Model III 4K LEV I	599
Model III 16K	859
Model III 32K	989
+ Model III 32K	929
Model III 48K	1099
+ Model III 48K	999
Model III 32K 2 Disc & RS232 c	2149
Color Computer 4K	310
Color Computer 16K	439.95
+ Color Computer 16K	399.95
Color Computer 16K w/extended basic	489
Pocket Computer	199
VIDEOTEX	329
APPLE 48K only	1119
ATARI 800 16K	789

PERIPHERALS	
Expansion Interface 0K	$249
Expansion Interface 16K	359.95
+ Expansion Interface 16K	339
Expansion Interface 32K	469.95
+ Expansion Interface 32K	399
16K RAM N.E.C. 200 N.S. chips	49
MODEMS	
Lynx Direct Connect	219
COMM 80 Interface	159.95
Chatterbox Interface	259.95
Telephone Interface II	169
PRINTERS	
Line Printer IV	849
Daisy Wheel II	1695
Line Printer VI	999
NEC Spinwriter 5530	2650
OKI DATA Microline 80	595
EPSON MX80	545
DISK DRIVES	
Model III 1-Drive	712
PERCOM TFD 100	389
TEAC 40 Track	319

+ Computer Plus New Equipment. 180 Day Extended Warranty

MODEL I 32K E.I., 1st. Drive, Line Printer IV, 10 Diskettes Pure R.S. $1875
combined warranty 1699

MODEL II 64K w/2 Disks, Daisy Wheel II, Scripsit Pure R.S. 6350

MODEL III 48K, 1st. Drive, Line printer IV, Scripsit Word Processor, 10 Diskettes Pure R.S. 2800
combined warranty 2699

COLOR COMPUTER 16K CC w/extended basic cassette recorder, joy sticks, dust cover Pure R.S. 569

POCKET COMPUTER w/interface Pure R.S. 239

ATARI
48K Atari 800, Atari 825 Printer, Atari 810 Disk Drive 2125

APPLE
48K Apple II & Apple II Disk w/controller, D.O.S. 3.3, Printer Card and Cable, Epson MX80 Printer 2350

PLUS real back-up warrantees —

Pure Radio Shack equipment warranteed at any Radio Shack store or dealer. Factory warrantees on Apple and Atari equipment. Other equipment carries manufacturer's warranty or Computer Plus 180 day extended warranty. Combined warrantees carry Computer Plus 180 day warranty or original manufacturer's warranty.

call TOLL FREE
1-800-343-8124

computer plus

Write for your free catalog ...

✓130

Dept. D
245A Great Road
Littleton, MA 01460

DEALER INQUIRIES ARE INVITED (617) 486-3193
Prices subject to change without notice.
Not responsible for typographical errors.
TRS-80 is a registered trademark of Tandy Corp.

✓ Reader Service—see page 290

Sail from SCRIPSIT to DOS without losing text—No jibe!

Scrip Patch

Daniel B. Allred
13 Larkspur Circle
Pease AFB, NH 03801

Out there in a sea of software float programs with various berths.

There are those that are true classics. They are cherished and passed from System 360 to 370, Model I to Model II, and 8080 to 8085.

This particular program, however, doesn't sail tall among such giant hulks—it is only intended for one time use. It does serve a useful, if limited, function, however, to a small wavelet of the computering public.

The purpose of this program is to patch Scripsit so it may be exited to DOS to perform a number of tasks (even run a small BASIC program!) without loss of text. There is provision in the patch to re-initialize it if the text area has been overwritten, or if it is the first time Scripsit has been used since powering up the system.

The modification is done in the following steps:

1) From DOS, with Scripsit in drive 0, type: RENAME SCRIPSIT/LC TO SCR/CMD. This must be done because the length of the DOS command is important to the patch.

2) Get another disk, with Disk BASIC 2.2, and bring up BASIC (FILES and MEM SIZE default by pressing ENTER).

3) Carefully type the BASIC program in Program Listing 1. *Do not run yet*!

4) Place disk with newly-RENAMEd SCR/CMD program in drive 0. (It is easier for you to do this than to type in INPUT prompts to the BASIC program telling you to do it anyway.)

5) Type RUN.

6) If an error is made in typing the data, you'll get a checksum error message. Recheck your data. If not, the modification is complete and you can try out your new Scripsit.

Scripsit (now summoned by typing SCR) has been modified to recognize the DOS command SCR * (one space between R and *). This is a request for program reentry. When DOS activity is desired while in Scripsit, exit the program normally to DOS by typing END.

When all the work is completed, type SCR * from DOS and everything should be left exactly the same—tabs and all!

For those with special printer driver routines: as long as there are no conflicts with memory area (this routine lies between hex 7AA4 and 7AC3, and uses location 7C00), your special driver routine can be included by doing the following:

1) Creating a modified program exactly as above.

2) RENAMEing SCR/CMD to whatever you are going to use in your driver routine.

3) Changing Scripsit entry point defined in the driver to 7AA4 (previously 5200).

4) Giving driver a three-letter name (like DRV/CMD)

Now, if reentry is desired, type from DOS your file name plus one space then * (e.g.: DRV*).

Assembly Language Patch

For those familiar with as-

```
10  OPEN"R",1,"SCR/CMD"
20  GET1,42
30  T=0 : FORI=73TO114 : READJ : T=T+J : POKE&H66DF+I,J
40  NEXT
50  IF T=3486 THEN 70
60  CLOSE : PRINT"CHECK YOUR DATA WITH LISTING" :PRINT : LIST100 –
70  PUT 1,42 : CLOSE
100 DATA 1 , 36 , 164 , 122 , 58 , 28 , 67 , 254 , 42 , 32
110 DATA 7 , 58 , 0 , 124 , 254 , 4 , 40 , 8 , 62 , 4 , 50
120 DATA 0 , 124 , 195 , 0 , 82 , 33 , 192 , 82 , 229 , 62
130 DATA 139 , 50 , 28 , 67 , 195 , 221 , 82 , 2 , 2 , 164 , 122
```

Program Listing 1. Scripsit Modification

```
        ORG     7AA4H
START   LD      A,(SW1)     ; Was there an asterisk in
        CP      2AH         ; the DOS command?
        JR      NZ,BP1      ; If not, initialize
        LD      A,(SW2)     ; Is there still a 04 in 7C00?
        CP      4
        JR      Z,BP2       ; If so, prepare for re-entry
BP1     LD      A,4
        LD      (SW2),A     ; Write 4 into 7C00
        JP      5200H       ; Initialize SCRIPSIT
BP2     LD      HL,52C0H
        PUSH    HL          ; Copy SCRIPSIT internal condition
        LD      A,8BH       ; Shift up-arrow code
        LD      (SW1),A     ; Eliminate asterisk in command buffer
        JP      52DDH       ; Jump into program
BUF     EQU     4317H       ; DOS 2.2 Command Buffer start
SW1     EQU     BUF+5       ; Fifth position in DOS command buffer
                            ; – this location will be checked for
                            ; presence of asterisk. (Hex 2A)
                            ; Change this number for different
                            ; position in buffer.
SW2     EQU     7C00H       ; If this area overwritten while
                            ; outside SCRIPSIT, chances are that
                            ; data is now useless!!!
        END     START
```

Program Listing 2. Assembly Language Equivalent of Patch

sembly language: This patch tacks an extra 34 bytes on to Scripsit and changes the entry point to hex 7AA4. It first checks the DOS command buffer for an asterisk (hex 2A). If an asterisk is there, it checks if memory location 7C00 has hex 04 in it. If both conditions are met, the accumulator will get the shift up-arrow code and jump into the program right after the keyboard scan routine.

If the conditions are not met, location 7C00 will get hex 04 written into it, and Scripsit will initialize. The assembler code equivalent to the patch is in Program Listing 2.

The assembly language program is sandwiched between two sets of four-byte disk-loading pointers. The pointers are used by DOS for proper loading of programs. They can be found in the DATA statements in the BASIC program. The pointers are in the following format:

Byte 1: Type pointer (1 = more data 2 = EOF)

Byte 2: Quantity of bytes following this byte until next pointer block. If there is more data to be read, the loader will decrement this number for each byte, but will not start testing for zero until one byte of data has been read. This means that a 2 in this position will read 256 + 2, or 258, bytes.

Byte 3: LSB of this data block loading start (if EOF, program entry point)

Byte 4: MSB of this data block loading start (if EOF, program entry point)

The BASIC program alters Scripsit by GETting, POKEing, then PUTting back the last record of the program. It effectively cancels Scripsit's EOF pointer and tells the DOS to read an additional 34 bytes starting at location 7AA4. It then writes a new EOF pointer with a new start address.

The necessity of Disk BASIC 2.2 is that the first byte of file one always lies at address hex 66DF, and the BASIC program uses this value. ■

✓ Reader Service—see page 290

Combine accurate flight characteristics with the best in animation graphics and you'll have SubLOGIC's

T80-FS1 Flight Simulator
for the TRS-80

SubLOGIC's T80-FS1 is the smooth, realistic simulator that gives you a real-time, 3-D, out-of-the-cockpit view of flight.

Thanks to fast animation and accurate representation of flight, the non-pilot can now learn basic flight control, including take-offs and landings! And experienced pilots will recognize how thoroughly they can explore the aircraft's characteristics.

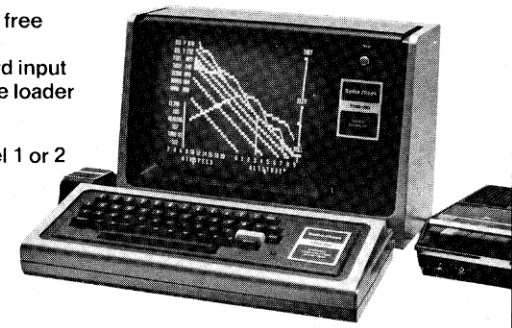
Once you've acquired flight proficiency, you can engage in the exciting British Ace 3-D Aerial Battle Game included in the package. Destroy the enemy's fuel depot while evading enemy fighters.

Computer and aviation experts call the T80-FS1 a marvel of modern technology. *You'll* simply call it *fantastic!*

Special Features:
- 3 frame-per-second flicker free animation
- Maximum transfer keyboard input
- Constant feedback cassette loader

Hardware Requirements:
- Radio Shack TRS-80, Level 1 or 2
- 16K memory
- *Nothing else!*

Only $25



See your dealer or order direct. For direct order, include $1.25 and specify UPS or first class mail. Illinois residents add 5% sales tax. Visa and Mastercard accepted.

subLOGIC
Communications Corp.
Box V, Savoy, IL 61874
(217) 359-8482
Telex: 206995 ↙150

Fantastic New Tool
for
Voice Synthesis
NO EQUAL ON THE MARKET

This peripheral adds high quality communication to the following:

TRS-80

S-100 Bus systems

Other 8 bit processors

The TLP-52 can give:

High quality speech (up to 800 words)

Phonetic speech (unlimited vocabulary)

Software generated speech

Voice recognition

From $200

For more information contact:

The Learning Place
5620-176th S.W. ↙198
Lynnwood, Washington-98036

SCHOOLMASTER

Record Keeping Made Easy!

- Analyzes grade assignments
- Keeps a cumulative grade list
- Maintains current averages
- Prints class lists; progress reports

SCHOOLMASTER is designed to replace or supplement most of the functions of the gradebook. In addition, it serves the function of analysis of scores and grades, and automates the process of producing individual progress reports. Requires 32K disk system; is compatible with NEWDOS, NEWDOS/80 and TRSDOS. $24.95 on disk. Manual alone, $1.00.

☐ Please send Schoolmaster to:

Name: _____
Address: _____
City: _____
State, Zip: _____

The Alternate Source
1806 Ada Street
Lansing, MI 48910
Ph. (517) 487-3358 or
(517) 485-0344

↙138

Add $2.00 for crush-proof box and postage; $1.50 for COD, 4% for Master Charge or Visa. PO's welcome.

SICK

OF PROGRAMS THAT TREAT YOU LIKE AN IDIOT, WASTING TIME & MONEY??

RANDOM ACCESS PAYROLL
- NO Complicated Initialization
- EDIT & LIST
- QUARTERLY Reports
- PROGRAM Loads In Less than 30 sec.

PAY ANY EMPLOYEE ANYTIME
- SALARIED OR HOURLY
- COMPLETE Including EIC
- PRINTS on NEBS 9020 CHECKS
- NO SPECIAL PAYROLL CHECKS
- SPECIAL HOURS—SPECIAL PAY
- TWO SAVINGS—INCLUDING RIA
- STATE TAX—WORKMENS COMP.
- CLASSED BY Occupation or Dept.
- PAYSTUB Shows Year-to-date
- No Filenames—All Automatic

SEND YOUR STATE TAX SCHEDULE
- We'll Customize for your State Tax

WRITTEN IN BASIC FOR COMPLETE CONTROL

Documentation & Disk $55.00
Documentation only $10.00-
credit to purchase

MASS ADD 5% or Exempt number Requires:
Min. 32K-1 Drive-Printer = 50 EMP
48K-2 Drives-Printer = 100 +
TEL. 7 AM-9:30 PM EASTERN
(617)-359-2364/6370

MEDFIELD ↙421
COMPUTER SOFTWARE
39 GREEN ST., MEDFIELD, MA 02052

Why not use decimal, hex and octal constants in Level II BASIC?

Constant Alternatives

Evan C. Hand, Sr.
RR #4, Box 185
Loogoote, IN 47553

Performing logical operations with the TRS-80 in Level II BASIC requires the use of decimal integers. With these, however, a programmer may find it difficult to see the relationship between the resulting operands. Hexadecimal and octal numbers are better suited to these operations.

Here is a machine language program (CVDNUM) that provides a convenient means for entering alternate number types. CVDNUM allows you to use decimal, hex and octal numeric constants in these and other TRS-80 functions.

In Level II BASIC, using PEEK and POKE to access and change memory locations seems simple enough when using decimal integers. Consider what occurs when you exceed 32767 in the address value of these commands (See the Level II BASIC Reference Manual, first edition, pages 8-5 and 8-6.).

PEEK and POKE require signed integer address values. Executing PEEK(32768) or any value greater than 32767 in the address value will result in an OV ERROR. Level II BASIC will perceive values above 32767 as single precision numbers.

To access memory locations above the maximum positive integer value allowed, the Level II BASIC reference manual suggests you use the following formula: $-1*(65536-\text{desired address})$ = the correct PEEK and POKE address.

The result of this operation is a signed 2's complement value (internal to the computer). Values greater than 32767 represented as an integer number must be in 2's complement form for the correct address value to be entered.

This is a very cumbersome approach to handling integer values above the allowed maximum. It would be nice to be able to directly enter positive integer values greater than those allowed. Read on and see how you can do this.

The Program: CVDNUM

CVDNUM is a machine language program that links itself to the Level II BASIC interpreter, providing the programmer with three formats to input constants: decimal, hex and octal.

CVDNUM is accessed by prefixes: &D for decimal, &H for hex and &O for octal. The ASCII source constant to be converted immediately follows the prefix (D, H or O). It must be compatible with the operation to be performed. The rules that apply for the same Disk BASIC functions also apply to CVDNUM (See the TRSDOS and Disk BASIC reference manual, pages 7-6 and 7-7.). There are only slight differences.

My program explanation assumes the reader has an elementary understanding of the mechanics of number base conversions. All the routines utilized in CVDNUM will convert numeric constants into an equivalent binary value. Binary conversion is a convenient denominator for the three number types. Also, the interpreter expects to see CVDNUM return a value in binary.

Let's take a look at the decimal conversion routine. The Level II ROM contains a subroutine to convert an ASCII decimal constant into its binary equivalent, and is used by CVDNUM to eliminate software duplication. There's one drawback: The routine will not accept a decimal value greater than 65529. This means that 65530 through 65535 cannot be converted by the decimal routine, only a slight hindrance, since the hex and octal functions will convert their equivalent values properly.

Since each digit in a decimal constant string is a factor of 10 (10000, 1000, 100, 10, 1), the individual digits must be separated and weighted by this factor. The operation starts with the most significant digit first, and works through to the least significant digit. A partial result is maintained and used to work out the binary equivalent.

When a digit is obtained from the ASCII decimal constant, it is temporarily stored and the partial result is multiplied by a value of 10 (weighted). After the partial result is put in proper perspective the new digit is added. If there are further digits to convert, the process is repeated.

```
                00010 ;       NUMERIC CONSTANTS ROUTINE
                00020
                00030 ;       WRITTEN BY EVAN C. HAND SR.
                00040
7FAB            00050         ORG     7FABH           ;MEMORY SIZE = 32682
                00060
7FAB D7         00070 CVDNUM  RST     10H             ;FIND NEXT BYTE
7FAC 110000     00080         LD      DE,0000H        ;CLEAR "DE"
7FAF FE48       00090         CP      'H'             ;HEXADECIMAL ?
7FB1 2815       00100         JR      Z,HEX           ;YES, GO TO IT
7FB3 FE4F       00110         CP      'O'             ;OCTAL ?
7FB5 282F       00120         JR      Z,OCT           ;YES, GO TO IT
7FB7 FE44       00130         CP      'D'             ;DECIMAL ?
7FB9 C24A1E     00140         JP      NZ,1E4AH        ;NO, ILLEGAL FUNCTION
7FBC CD5E1E     00150 DEC     CALL    1E5EH           ;CONVERT ASCII TO BINARY
7FBF EB         00160         EX      DE,HL           ;PUT BINARY INTO HL
7FC0 CD9A0A     00170 LDINT   CALL    0A9AH           ;PERFORM INTEGER LOAD
7FC3 EB         00180         EX      DE,HL           ;RESTORE PROGRAM POINTER
7FC4 C9         00190         RET                     ;RETURN TO "BASIC"
7FC5 C3B207     00200 OVERR   JP      07B2H           ;OVERFLOW ERROR
7FC8 D7         00210 HEX     RST     10H             ;FIND NEXT BYTE
7FC9 EB         00220         EX      DE,HL           ;SAVE PROGRAM POINTER IN DE
7FCA 300C       00230         JR      NC,ALPHA        ;GO TEST FOR "A-F"
7FCC E60F       00240         AND     0FH             ;MASK OUT MSB
7FCE 0604       00250 SHIFT   LD      B,04H           ;LOAD SHIFT COUNT
7FD0 CDFA7F     00260         CALL    SHFLFT          ;SHIFT RESULT IN "HL"
7FD3 B5         00270         OR      L               ;MERGE NEW BYTE
7FD4 6F         00280         LD      L,A             ;SAVE RESULT
7FD5 EB         00290         EX      DE,HL           ;RESTORE PROGRAM POINTER
7FD6 18F0       00300         JR      HEX             ;LOOP
7FD8 28E6       00310 ALPHA   JR      Z,LDINT         ;FINISHED, LOAD RESULT
7FDA FE41       00320         CP      'A'             ;CONTROL ?
7FDC 38E2       00330         JR      C,LDINT         ;YES, EXIT
7FDE FE47       00340         CP      'G'             ;ALPHABETIC ?
7FE0 30DE       00350         JR      NC,LDINT        ;YES, EXIT
7FE2 D637       00360         SUB     37H             ;CONVERT TO BINARY
7FE4 18E8       00370         JR      SHIFT           ;LOOP
7FE6 D7         00380 OCT     RST     10H             ;FIND NEXT BYTE
7FE7 EB         00390         EX      DE,HL           ;SAVE PROGRAM POINTER IN DE
7FE8 30D6       00400         JR      NC,LDINT        ;FINISHED, LOAD RESULT
7FEA FE38       00410         CP      '8'             ;>7 ?
7FEC 30D7       00420         JR      NC,OVERR        ;YES, ERROR
7FEE 0603       00430         LD      B,03H           ;LOAD SHIFT COUNT
7FF0 CDFA7F     00440         CALL    SHFLFT          ;SHIFT RESULT
7FF3 E60F       00450         AND     0FH             ;MASK ASCII BITS
7FF5 B5         00460         OR      L               ;MERGE NEW WITH OLD
7FF6 6F         00470         LD      L,A             ;PUT IN "L"
7FF7 EB         00480         EX      DE,HL           ;RESTORE PRGM POINTER
7FF8 18EC       00490         JR      OCT             ;LOOP
7FFA 29         00500 SHFLFT  ADD     HL,HL           ;SHIFT LEFT
7FFB 38C8       00510         JR      C,OVERR         ;ERROR, OVERFLOW
7FFD 10FB       00520         DJNZ    SHFLFT          ;LOOP TILL DONE
7FFF C9         00530         RET                     ;RETURN
4194            00540         ORG     4194H           ;SET LINKER FOR "&"
4194 C3AB7F     00550         JP      CVDNUM
1E4A            00560         END     1E4AH     ;FC ERROR, REQUIRES NO EXECUTION D
00000 TOTAL ERRORS
```

Program Listing. CDVNUM.

If not, the process is stopped and the partial result is returned as an unsigned binary value.

To understand how the operation is performed, I suggest using a disassembler to investigate the subroutine stored in the Level II ROM. The call to 1E5EH is the point of entry for the CVDNUM routine. Upon return, the binary equivalent is in the DE register pair.

The contents of the DE and HL register pairs are swapped, so the resultant binary value is in the HL registers. LDINT performs a call to an integer load routine in ROM that loads the contents of the HL registers into the arithmetic scratch pad area. This area is where the calling routine expects to find the value returned by CVDNUM.

Hex and Octal Constants

The hex and octal conversion routines are identical except that octal works with three bits representing the source digits, and hex works with four bits. Fortunately, hex and octal numbers are direct representations of their binary equivalents, thus, only a relatively simple shifting routine is required.

The HL register pair is used as the source string pointer to the current memory address where the ASCII byte being converted is located. The RST 10H instruction is a single byte call to a subroutine that increments the HL registers. It returns in the accumulator the byte pointed to by these registers. ASCII spaces are ignored and will be skipped over until a non-space character is found. If the value in the accumulator is numeric, the carry flag will be set. If the end of the string is encountered, the zero flag will be set.

In hex, after the source value is retrieved, a test is performed to see if the value is numeric or alphabetic. If the value is numeric, the four most significant bits in register A (the accumulator) are masked off, leaving only the least significant bits. The four least significant bits will represent the binary equivalent of zero to nine. If the value retrieved by the RST 10H instruction is alphabetic, a different approach is taken.

ASCII values 30H to 39H (H means hex) represent the decimal values zero to nine. Mask off the threes and the binary equivalent remains. ASCII values 41H to 46H represent the hexadecimal values A to F. Simply masking off the fours will not yield the correct equivalent binary value. The letters A to F actually represent the decimal values 10 to 15. To recover the correct value, 37H must be subtracted from the ASCII value. For example, 41H minus 37H will yield 1010 in binary, or decimal 10. See the Program Listing, lines 210 to 370, to see how this is accomplished.

After the correct binary value is recovered in register A, the result must be shifted four places to the left (weighted by a factor of 16) to make room for the new digit. When the result is shifted correctly, the new digit is merged with the old to comprise a new, resultant value. The process is repeated until a non-hexadecimal character is encountered or an overflow occurs.

In octal, the process uses only characters zero to seven. After the new ASCII value is retrieved in register A (and if the value is numeric), a test is performed to see if the value is greater than seven. Alphabetic characters cause the process to stop and control passes back to the Level II BASIC interpreter.

In both hex and octal, control is returned to the calling routine in the routine labeled LDINT (load integer and restore the BASIC program pointer).

Using the Routine

There are few restrictions on the use of expanded constants except that they cannot be entered via input statements nor appear in the data train following data statements. In some instances, it may be necessary to enclose the operation in parentheses to prevent an error.

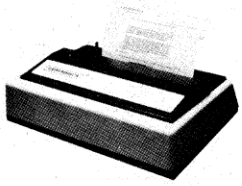
If possible, borrow a TRSDOS Disk BASIC operator's manual. Except for the &D function, the operation is outlined in the manual.

Executing PRINT &D255; &HFF;&0377, the computer responds with 255 in all three

PRINTERS & CRT'S From Orange Micro ✓296

CENTRONICS 737 (RADIO SHACK LINE PRINTER IV)

Word Processing Print Quality

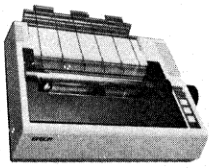


• 18 x 9 dot matrix; suitable for word processing • Underlining • proportional spacing • right margin justification • serif typeface • 50/80 CPS • 9½" Pin Feed/Friction feed • Reverse Platen • 80/132 columns

CENTRONICS 737-1 ..... (List $995) **$780**
CENTRONICS 737-3 ..... (List $1045) **$830**

EPSON MX80

Low-Priced
Professional Print Quality



• 9 x 9 dot matrix • Lower case descenders • 80 CPS • Bidirectional, Logic seeking • 40, 66, 80, 132 columns per line • 64 special graphic characters: TRS-80 Compatible • Forms handling • Multi-pass printing • Adjustable tractors

EPSON MX80 ..... (List $645) **$Call**

QUANTITY PRICING AVAILABLE

TELEVIDEO CRT'S
PRICES SLASHED!

TVI 912C
TVI 920C
TVI 950

Please Call Toll Free
Prices are too low to advertise

PRINTERS

ANACOM 150	150 CPS, wide carriage, 9 x 9 dot	(List $1350) $ Call
ANADEX 9500	wide carriage, graphics	(List $1650) $1450
VISTA V300 (C. ITOH)	Typewriter quality, daisy wheel	(List $1895) $1795
BASE 2 850	graphics printer	(List $799) 749
OKIDATA MICROLINE 80		(List) $800) 599
NEC 5530-5	letter quality, RO, parallel, tractors	(List $2970) $2599
MALIBU	Dot graphics, 132 Col, Letter quality	$ Call
PAPER TIGER IDS 445 & 460	with graphics & 2K buffer	$ Call
QUME 5/45	Typewriter quality	(List $2905) 2559

INTERFACE EQUIPMENT

APPLE II — BASE 2	parallel graphics interface board	160
SSM AIO BOARD	Serial/Parallel interface board	(List $225) 175
TRS-80 CABLES	expansion interface or direct	$ Call

TOLL FREE (800) 854-8275
CA, AL, HI (714) 630-3322

Call for FREE CATALOG

Phone orders WELCOME. Same day shipment for VISA, MASTER CHARGE, and AMERICAN EXPRESS. Personal checks require 2 weeks to clear. Add 3% for shipping and handling. California residents add 6%. Manufacturer's warranty included. Prices subject to revision.

Orange Micro, Inc.
3148 E. La Palma, Suite E
Anaheim, CA 92806

DISCOVER THE 6809 IN YOUR COLOR COMPUTER

Now you can explore the Radio Shack Color Computer's impressive potentials—as an inexpensive development system, a color peripheral, a process controller—ad infinitum. The Micro Works introduces these powerful software tools for utilizing the color computer at the assembly language level.

MONITOR TAPE: A cassette tape which allows you to:

- Examine or change memory using a formatted hex display
- Save areas of memory to cassette in binary (a "CSAVEM")
- Download/upload data or programs to a host system
- Move the video display page throughout RAM
- Send or receive RS-232 at up to 9600 baud
- Investigate and activate features of your computer, such as hi-res graphics or machine-language music
- Use your computer as an intelligent peripheral of another computer, for a color display or a 6809 program development tool

The monitor has 17 commands in all, and is relocatable and re-entrant.

80C Monitor Tape Price: $29.95

MONITOR ROM: The same program as the monitor tape, supplied on ROM. This allows BASIC to use the entire RAM space. And you don't need to re-load the monitor each time you use it.

80C Monitor ROM
Price: $39.95

C BUG IS HERE!!

INSIDE THE COLOR COMPUTER: This package is a disassembler which runs on the color computer and enables you to generate your own source listing of the BASIC interpreter ROM. Also included is a documentation package which gives useful ROM entry points, complete memory map, I/O hardware details and more. Disassembler features include cross-referencing of variables and labels; output code which can be re-assembled; output to an 80-column printer, small printer or screen; and a data table area specification which defaults to the table boundaries in the interpreter ROM. A 16K system is required for the use of this cassette.

80C Disassembler Price: $49.95

THE MICRO WORKS ✓109

Mastercharge and BankAmericard

P.O. BOX 1110 DEL MAR, CA 92014 714-942-2400

cases. Experiment with the &D facility, especially with numbers greater than 32767, to get a feel for its operation. The expanded constants may be assigned to variables but the interpreter will convert them to signed integers. Executing X = &HFFFF will assign −1 decimal to the variable X.

The maximum value allowed in each function is as follows: 65529 for &D, FFFF for &H, and 177777 for &O. Values in excess of these result in the computer's issuing an error message.

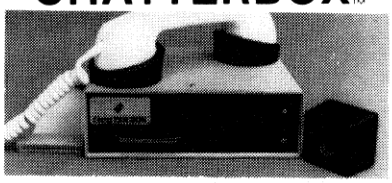
Using CVDNUM is simple. Unlike Radio Shack's utility, CVDNUM expects to find an operation declarative (D, H or O) after the & symbol. The &D function allows the computer programmer to directly enter unsigned integer values in the range 0 to 65529. The decimal value will be correctly evaluated to a signed integer value.

One final note: CVDNUM requires the TRS-80 Editor/Assembler program for assembly. After you have the program on cassette tape, turn your computer off and back on again. Answer the memory size question with 32682. Type SYSTEM, and when the computer responds with the asterisk, load the cassette tape and enter the file name the program was assembled under and ENTER. When the asterisk returns, press BREAK. Using the execution directive (/ or slash) will result in the computer issuing an FC ERROR, which is intentional (see program line 560). It does not affect loading and operation of CVDNUM. CVDNUM is now linked into the Level II BASIC interpreter and will be accessed when the computer encounters the & character.

You may enter these routines via a short BASIC program using POKE statements. If you attempt to convert these routines into a BASIC program, don't forget to set the linking address (program lines 540 and 550). Failure will result in the TRS-80 responding with the all too familiar L3 ERROR. ■

3 ALTERNATIVE INTERFACES FOR THE TRS-80

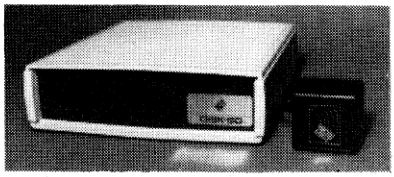
CHATTERBOX™



Featured in Aug. '80 BYTE

- 300 baud originate modem
- Centronics printer port 8-bit
- RS-232-C port (50-19.2K baud)
- connects to keyboard or E. I.
- received data automatically routed to printer ports
- includes terminal software
- only $279.95 complete

DISK-80™



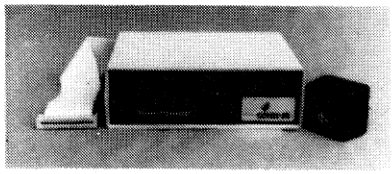
Featured in March '81 BYTE

- disk controller (4 drives)
- hardware data separator
- includes 16K of RAM provision for additional 16K
- buffered TRS-bus connector
- real-time clock
- printer port (optional)

ASSEMBLED & TESTED
with 16K of RAM $329.95
Centronics Printer
Port add . . . . . $ 50.00
with 32K RAM add. . . $ 50.00
DISK-80 pc board. . . . . $ 48.00
Printer/Power Supply
pc board . . . . . $ 16.00
Complete Kit with 16K
RAM and Printer Port . $275.00

Dealer inquiries invited.

COMM-80™



Featured in May/June '80 BYTE

- RS-232-C port (50-19.2K baud) software/hardware selectable
- Centronics printer port 8-bit
- connects to keyboard or E. I.
- chain up to 16 units
- use with I.E. for 2nd printer
- includes terminal software
- only $179.95 complete

To order call (516) 374-6793
or write: The MicroMint Inc. ✓310
917 Midway
Woodmere, NY 11598

ALL INTERFACES ARE RADIO SHACK HARDWARE AND SOFTWARE COMPATIBLE AND CARRY A 60 DAY WARRANTEE INCLUDING PARTS AND LABOR. ALL UNITS INCLUDE USER'S MANUAL, POWER SUPPLY & AUXILIARY TRS-BUS CONNECTOR FOR FUTURE EXPANSION.

MICRO MINT

TRS-80 is trademark of Tandy Corp.

TRS-80 & OTHER NEEDS FILLED FOR LESS

+ + + COMPATIBLE DISK DRIVES WITH POWER SUPPLY AND CASE—120 DAY WARRANTY + + +
- 40 TRACK (204,800 BYTE/DISK) USE BOTH SIDES, ANTI-CRIMP/POWER PROTECT **$319**
- 8 IN. DRIVE & P.S./CASE **$729** WITH P.S./CASE FOR 3 DRIVES **$929**
- 80 TRACK (204.8K BYTE) 90 DAY WARRANTY **$419**
- 4-DRIVE CABLE **$28** • • 10 DISKS-5 IN. @ **$24**-8 IN. @ **$36** HARD CASE **$3** & **5**

BASE 2 PRINTER **$599** EPSON MX-80 PRINTER **$499** MICROLINE 80 PRINTER **$499**

- CENTRONICS 737 **$699** + + + + + CABLE @ **$25**
- HARRIS SELECTRIC (WORD PROCESSING-TYPEWRITER & PRINTER) **$790**
- LOWER CASE FOR CENTRONICS 779/RADIO SHACK LINE PRINTER 1 -EASY INSTALL **$99.95**
- UPS (UNINTERRUPTIBLE POWER SUPPLY) PREVENT POWER DROP SURGE OR OUT? FROM **$195**
- CAT MODEM (ORIG/ANS) **$139** + + + + + + + + 16K MEMORY SET (200 NANO) **$34**
- 16K MODEL III RADIO SHACK SYSTEM **$849**
- APPLE, ATARI, RADIO SHACK MODEL 1/2 HARDWARE/SOFTWARE DISCOUNTED. A/R, A/P, G/L, P/R FOR **$200 or $59 ea.** (MODEL 1) & **$329 or $80 ea.** (MODEL 2). APPLICATIONS INTERACT & ARE COMPLETE & PROFESSIONAL. WILL RUN ON OTHER COMPUTERS. THIS IS A SPECIAL INTRODUCTORY PRICE.
- ASK FOR FREE FLYER WITH OUR LOW PRICES—DEALER INQUIRIES INVITED MASS. RESIDENTS ADD 5% TAX—F.O.B. TEWKSBURY—FREIGHT EXTRA.

M/C, VISA OR CHECK ACCEPTED. TRS-80 IS A REG. TRADEMARK OF TANDY CORP.

✓105

OMNITEK SYSTEMS — 24 MARCIA JEAN DR., DEPT-M, TEWKSBURY, MA 01876 CALL 617-851-3156

MICRO-80™ CASSETTES— 100% ERROR-FREE

LENGTH	12 PACK	24 PACK
C-10	69¢	59¢
C-20	89¢	79¢

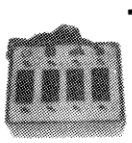
- Fully Guaranteed!
- World's Finest Media
- Premium 5-Screw Construction
- Used by Software Firms Nationwide
- Dealer and Club Discounts Available
- Cassette Shipping, Add $1.50
- Write for Wholesale Price Listing

μ80™ MICRO-80™ INC.
✓476 E-2665 NO. BUSBY ROAD
OAK HARBOR, WA 98277

THE SPIKE-SPIKERS™

with Transient Absorber and R.F. Filters

Protects Computers & Sensitive Equipment
from Most Transients & "Hash" Interference



Deluxe Power Console
with Transient Absorber & RF Filters
8 individually switched 120 Vac
Grounded outlets. Divided into two filtered
circuits. Main on/off switch.
Fuse and Indicator Light
$79.95

Mini-I
Wall mounted with
Transient Absorber
Only
$34.95

Mini-II
Two Sockets with
Transient Absorber
plus RF "Hash"
Filters
$44.95

Prices Subject to Change without Notice - PA Residents add 6% sales tax

KALGLO ELECTRONICS CO., INC.
DEPT. mc
COLONY DRIVE INDUSTRIAL PARK
6584 RUCH ROAD - E. ALLEN TWP.
BETHLEHEM, PA 18017
✓353

master charge VISA
dealers invited

ORDER FACTORY DIRECT
215-865-0006
OUT OF STATE ORDER TOLL FREE
800-523-9685

✓ Reader Service—see page 290

It's done with sound, not mirrors.

80 Appliance Control

Cass Lewart
12 Georjean Dr.
Holmdel, NJ 07733

The electronic, computer-controlled home of the future may be closer than you think. That's because the simple computer/appliance interface described here allows control of multiple ac operated home appliances—via computer commands.

Controlling Lamps, Etc.

It's done by a computer program generating specific tones which are then detected and decoded by frequency selective control circuits. The control circuits receive commands via the TRS-80 AUX plug, which is normally plugged into the recording jack of the tape recorder. This activates the solid state relays which turn on and off lamps, appliances, etc.

This novel approach is simpler than the standard decoding technique which is described in the *TRS-80 Microcomputer Technical Reference Handbook*. In the standard approach a port or a memory address has to be decoded with TTL ICs connected to the address and data lines on the TRS-80 40-pin bus. The status of a given port or memory location is then used to control the appropriate relays. However, the circuitry associated with address decoding is more complicated than the tone decoders used in my circuit. Also, this approach requires access to the TRS-80 bus with a 40-pin connector. For a hobbyist, this is difficult to obtain and wire.

The tone decoding approach *is* simpler, and here is a description of how it works:

A TRS-80 program, such as the one shown in Program Listings 1 and 2, generates different tones. These are detected with inexpensive Phase Locked Loop (PLL) NE567 integrated circuits, each tuned to a different frequency, as shown in Fig. 1. The frequency to which each IC responds is determined by C1, R1 and R2, where R1 is an adjustable potentiomenter. Depending on the setting of R1, the circuit will respond to any frequency between approximately 1 kHz and 10 kHz. When the right tone is received and detected, the output of Pin 8 on the IC goes LOW, activating the solid state relay SS1.

The result is that ac suddenly appears on the proper appliance outlet and turns on your coffee pot, or scares burglars by activating bells and sirens. It might also turn the furnace or air conditioning on before you come home, etc.

Similarly, when the tone is discontinued, the process is reversed and the power will be turned off. By connecting several tone detectors in parallel, each associated with its own relay, the computer can control several appliances one after the other by sending different tones to each one.

You will notice that the basic circuit, as shown in Fig. 1 is nonlatching and also that individual relays can only be operated in succession, one at a time. The circuit can easily be modified with additional solid state or conventional relays. Using two relays with multiple contacts, one can build latching circuits, where for example, a command (A,B) would mean: Turn appliance number A on if B = 1 and off if B = 0. An example is shown

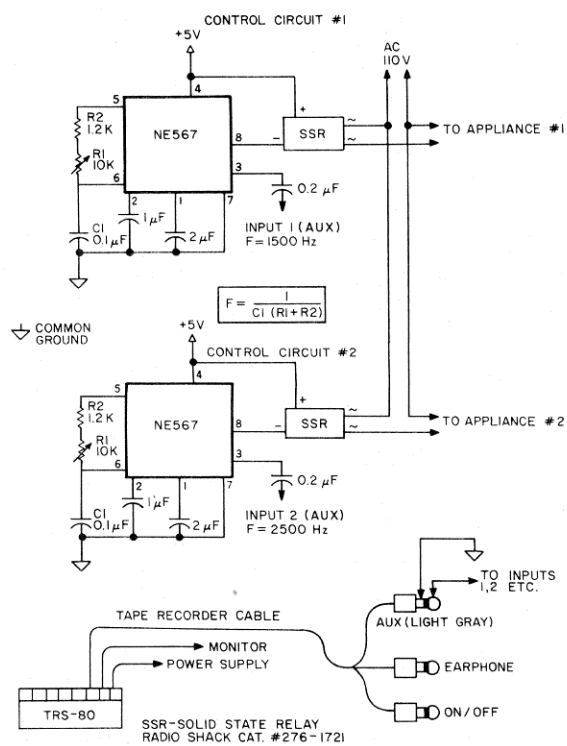


Fig. 1. Basic Multiple Appliance Control Circuit (Non-Latching)

```
30 'LET PROGRAM SET MEMORY SIZE
40 POKE 16562,111:POKE 16561,254:CLEAR 50
50 CLS:DEFINT A-Z
60 'POKE FREQUENCY GENERATING PROGRAM (FIG.2)
70 FOR A=28672 TO 28692:READ B:POKE A,B:NEXT
80 'SET USR(L) VECTOR
90 POKE 16527,112:POKE 16526,0
100 'PROMPT FOR DEVICE NUMBER
110 CLS:PRINT@465,"DEVICE NUMBER (1 OR 2)?"CHR$(14);
120 C$=INKEY$
130 'PROMPT TILL YOU GET "1" OR "2"
140 IF C$<>"1" AND C$<>"2" THEN 120
150 PRINT C$;CHR$(15);
160 'APPLIANCE #1 REQUIRES 1500 HZ
170 'APPLIANCE #2 REQUIRES 2500 HZ
180 'DETERMINE ARGUMENT FROM THE FOLLOWING APPROXIMATIO
    N:
190 'L = INT(69000/FREQUENCY - 4.1)
200 'L(1500) = 41    L(2500) = 23
210 IF C$="1" THEN L=41
220 IF C$="2" THEN L=23
230 'CALL FREQUENCY GENERATING ROUTINE, TO EXIT PRESS "
    X"
240 D=USR(L)
250 GOTO 110
260 DATA 205,127,10,14,1,62,3,169,79,211,255
270 DATA 69,16,254,58,8,56,31,48,241,201
```

Program Listing 1. Appliance Control Demonstration Program

program which is shown with comments in Program Listing 2. This program can generate square waves within the frequency response of the control circuit (1-10 kHz). Because the bandwidth of the control circuit with the component values shown is only 10 percent of the control frequency, the program could easily command 10-15 individual control circuits by allocating the control frequencies between one and ten kHz.

A control function can be executed directly by you, or can of course be driven by outside events—burglars, fire or smoke. The function can also be driven by timed events, turning on the coffee pot at 6:15 a.m., etc. Real life applications usually combine all three types of control all of which can be processed by your TRS-80. The first way is obvious, you just press the button; the second type can be originated by sensors; the third can be executed, based on the status of the Real Time Clock in the Expansion Interface. To make maximum use of the time function, you could try BASIC Level III which has the time function built-in.

Working with 110 V currents requires proper caution as you may produce "zapping" of components with fireworks displays. Always provide sufficient clearance between wires, and mount the circuit in a plastic cabinet or on a well insulated board. Make sure that the voltage and current ratings of the relays are equal to your appliances.

```
0A7F            00100 GETARG  EQU     0A7FH   ;GET ARG FROM BASIC
0001            00110 HI      EQU     1
00FF            00120 PORT    EQU     0FFH    ;CASSETTE PORT
3808            00130 ROW4    EQU     3808H   ;"X" KEY
7000            00140         ORG     7000H   ;=28672 DECIMAL
7000 CD7F0A     00150 START   CALL    GETARG  ;GET VALUE FROM USR(X)
7003 0E01       00160         LD      C,HI    ;TEMP. STORAGE
7005 3E03       00170 SPKR    LD      A,3
7007 A9         00180         XOR     C       ;FLIP HI TO LO, LO TO HI
7008 4F         00190         LD      C,A
7009 D3FF       00200         OUT     (PORT),A;OUTPUT
700B 45         00210         LD      B,L     ;LOAD DELAY LOOP
700C 10FE       00220 DEL     DJNZ    DEL     ;WASTE TIME
700E 3A0838     00230         LD      A,(ROW4);CHECK FOR "X"
7011 1F         00240         RRA
7012 30F1       00250         JR      NC,SPKR ;CONTINUE
7014 C9         00260         RET             ;RETURN TO BASIC
7000            00270         END     START
00000 TOTAL ERRORS
```

Program Listing 2. Assembled Z-80 source code of a square wave generating program included in the BASIC program shown in Listing 1.

in Fig. 2. Using two relays, one SPDT and one DPDT, and two NE567 integrated circuits with each frequency decoding circuit, independent ON/OFF commands can be issued to each appliance.

Modifications

The approach used for controlling ac operated devices can be used for low power dc applications. These include small incandescent and LED lamps; signals, model trains, etc. A change would be the replacement of solid state relays by low power dc relays. As you may know, a solid state relay will not interrupt a dc main current even when the control current is discontinued. The solid state relay just latches permanently until the main current is interrupted. Applying convenient, long lasting and inexpensive solid state relays is therefore limited to ac circuits and to dc circuits which are cyclically interrupted (for example the ON/OFF motor drive switch of a battery operated tape recorder).

The control circuit in Fig. 1, when exposed to square waves, responds reliably and has a repetition rate of at least 1 kHz. The demonstration program shown in Program Listing 1, depending on the input, generates a square wave with a repetition frequency of either 1.5 kHz or 2.5 kHz on the output of the TRS-80 (port 255). A BASIC program is too slow. Therefore to obtain frequencies above 1 kHz, the BASIC program POKEs into memory a short machine language

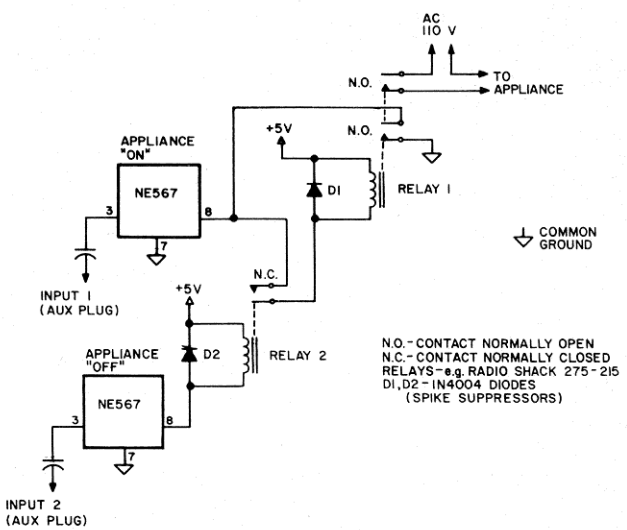


Fig. 2. A more elaborate control circuit with latching relays and independent ON/OFF commands; for details and component values see Fig. 1.

Initial Adjustments

The adjustments in the circuit shown in Fig. 2 are the settings of frequency controlling potentiometers. To perform the adjustments, connect control circuits 1 and 2 to your computer as shown in Fig. 1. Then CLOAD and RUN the demonstration program in Program Listing 1. Attach two lamps, L1 and L2, to the outlets controlled by the respective circuits. Upon prompt, press 1, and rotate R1 until lamp L1 lights up. Press X to extinguish it. Then repeat by answering 2 to the prompt and by adjusting the potentiometer in the second circuit. Leave the potentiometers in the middle of the range where the lights work.■

Use Baby-text to beat the high cost of composing.

The Pauper's Processor

Gary L. Osburn
130 Country Club Drive
Maybrook, NY 12543

Word processing is becoming a new field in computer science. What started out as a small processing chip attached to a typewriter that could read and write to a card coated with magnetic film, has now evolved into a complex microprocessor using diskettes as storage media.

Word processing equipment can even communicate with large mainframe computers now, to give those computer-generated dunning notices that personalized touch!

Every microprocessing publication available today has at least one article on the subject; the ads are filled with pitches for Electric Pencil, Scripsit and the like. However, each and every one of them has one common flaw: *cost*! Those of you who fall into the computer nut category may realize that there are nicer things available for $100-$200 —and a lot more fun than the word processing package. I am not suggesting that you refuse a reliable package if you do a lot of typing, but if you only type a few letters a month, or an occasional article for *80 Microcomputing*, I suggest you save your money.

I attacked this cost problem via my normal approach: Keep it simple, stupid! I decided that I didn't need fancy block move capability, variable margins or global search logic. I found I could control the line length myself, and as long as the left margin lines up, the right one is allowed to look a little rough.

Baby-Text

Baby-Text is a simple substitute for the more expensive word processing models. The text is entered as data statements within the program. Editing, inserting and deleting are accomplished using the EDIT mode of BASIC. This allows for a variety of data moves within the text. Tabbing, centering, paging and double spacing are controlled by inserting six simple commands before the text. Uppercase characters are entered by using the shift key. Lowercase uses the keyboard just as in normal typing. ■

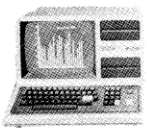
&DBL&	Double spacing
&SGL&	Single spacing (default)
&T05&-&T60&	Tab 5 through 60 spaces
&WIC&	Center wide characters
&NOC&	Center normal characters
&PAG&	Force page break

Table 1.
Six Baby-Text Commands

```
10 'BABYTEXT WORDPROCESSOR VERSION 1.0:3/80 BY GARY L.
    OSBURN
20 CLS:CLEAR 1000
25 INPUT"IS THE PRINTER ONLINE";R$:IFR$="N"POKE16422,88
    :POKE16423,4
30 READ TXT$:IF LEN(TXT$)>80 PRINT"LONG LINE ";TXT$
35 IF LEFT$(TXT$,1)<>"&"GOTO120
40 IF LEFT$(TXT$,5)="&END&"END
50 IF LEFT$(TXT$,5)="&DBL&"DBL$="D":GOTO30
60 IF LEFT$(TXT$,5)="&SGL&"DBD$="S":GOTO30
70 IF LEFT$(TXT$,2)="&T":TAB%=VAL(MID$(TXT$,3,2)):MID$(
    TXT$,1,5)="     ":TAB%=TAB% - 5:HLD$=STRING$(TAB%,
    " "):TXT$=HLD$+TXT$:GOTO120
80 IF LEFT$(TXT$,5)="&WIC&"TXT$=MID$(TXT$,6,LEN(TXT$)):
    SP%=INT(20-(LEN(TXT$)/2)):LPRINT" ":LPRINTCHR$(31)
    :LPRINTTAB(SP%) TXT$:PRINT" ":LPRINTCHR$(30):LNCT%
    =LNCT%+3:GOTO30
90 IF LEFT$(TXT$,5)="&NOC&"TXT$=MID$(TXT$,6,LEN(TXT$)):
    SP%=IN40-(LEN(TXT$)/2)):LPRINT" ":LPRINT TAB(SP%)
    TXT$:LPRINT" ":LNCT%=LNCT%+3:GOTO30
100 IF LEFT$(TXT$,5)="&PAG&"FORX%=LNCT%TO60:LPRINT" ":N
    EXTX%:LPRINT"-":FORX%=1TO4:LPRINT" ":NEXTX%:LNCT%=
    0:GOTO30
110 PRINT"UNRECOGNIZED KEYWORD ";TXT$:GOTO30
120 FORX%=1TOLEN(TXT$)
125 IF MID$(TXT$,X%,1)<CHR$(65) 150
130 IF MID$(TXT$,X%,1)<CHR$(91) MID$(TXT$,X%,1)=CHR$(AS
    C(MID$(TXT$,X%,1))+32):GOTO150
140 IF MID$(TXT$,X%,1)>CHR$(95)ANDMID$(TST$,X%,1)<CHR$(
    123)MID$(TXT$,X%,1)=CHR$(ASC(MID$(TXT$,X%,1))-32):
150 NEXTX%
160 LPRINT TAB(5) TXT$:LNCT%=LNCT%+1:
170 IFDBL$="D":LPRINT" ":LNCT%=LNCT%+1
180 IFLNCT%>55:FORX%=LNCT%TO60:LPRINT" ":NEXTX%:LPRINT"
    -":FORX%=1TO4:LPRINT" ":NEXTX%:LNCT%=0
190 GOTO30
1000 DATA "&T40&130 cOUNTRY cLUB dRIVE"
1010 DATA "&T40&mAYBROOK, nEW yORK"
1020 DATA "&T40&mAY 21, 1980"
1030 DATA" "
1035 DATA"mR. j. pERRY
1040 DATA"80 mICROCOMPUTING"
1050 DATA"pETERBOROUGH, nEW hAMPSHIRE 03458"
1060 DATA" "
1070 DATA"dEAR jIM,"
1080 DATA" "
1090 DATA"     eNCLOSED PLEASE FIND THE COMPUTER TAPE T
    HAT YOU REQUESTED OF ME."
1100 DATA"yOU INDICATED THAT i SHOULD STATE THAT MY ART
    ICLE, 'baby-text' IS"
1110 DATA"IN YOUR HOLDING FILE.  fOR YOUR CONVIENCE i A
    M ALSO INCLUDING"
1120 DATA"ANOTHER LISTING OF THE PROGRAM.  tHE TAPE WAS
     RECORDED ON A trs-80 "
1121 DATA"1EVEL-ii (dos).
1125 DATA" ","     pLEASE FEEL FREE TO CALL ON ME IF i
    CAN BE OF FURHTER HELP."
1130 DATA" ":DATA" ","&T40&sINCERELY,"," "," "," "," "
1140 DATA"&T40&gARY 1. oSBURN"
1150 DATA"&PAG&"
65000 DATA"&END&"
```

Program Listing 1

ADD DISK DRIVES TO YOUR MODEL III AND SUPERBRAIN



SAVE UP TO 34%
COMPARED TO RADIO SHACK
Now, add mini-disk drives to your TRS-80 Model III and get FREE INSTALLATION.

Fully compatible with Radio Shack's operating system TRSDOS.

- One, two, three, four drive configurations.
- 134K to over 1 Meg of storage
- 40 track and 80 track available
- Gray cases to match your system
- 90 day warranty, 100% parts and labor
- Extended warranty available

Complete upgrade includes one mini-disk drive, power supply, controller, and mounting hardware. IMMEDIATE DELIVERY.

	OUR PRICE	RADIO SHACK	SAVINGS
Complete one drive kit	$599.	$849.	29%
Additional internal drive	265.	399.	34%
External drive	340.	499.	32%
80 track drives (inc. patch)	add 150. ea.	N/A	
Additional one year warranty	45. ea.	N/A	
16K memory (required to use drives)	49.	119.	59%

To order call toll free (1) 800 345-8102 (outside PA), or (215) 461-5300 in PA.

AVAILABLE ONLY FROM VR DATA!
Increase the storage capacity of your SuperBrain to 700K with two extra 40 track drives, or over 1 Meg with QD drives.

- Fully compatible with SuperBrain operating system
- Fully compatible with all SuperBrain drives
- Easy plug-in installation
- Includes patch program
- 90 day warranty, 100% parts and labor
- Extended warranty available

If you need more storage now, don't wait for costly hard disk units and buy more storage than you need.

SAVE AND GET 1 Meg of STORAGE

Complete two drive kit (incls. cable and patch)	789.
QD drive	add 150.
Additional one year warranty	45.

VR Data, a manufacturer of innovative computer products, is known worldwide for quality, dependability and prompt, personal service since 1972.

VR Data

VISA master charge

VR Data corporation 777 Henderson Boulevard Folcroft, Pa. 19032

Positional Mathematics

HISTORY records that for thousands of years Man has endeavored to understand himself through the study of Numbers. From the Ancient Hebrew Kabbalah to contemporary **Numerology** the search for knowledge has surfaced and subsided with the permissiveness of the politics and religions of each era. The **UNIVERSE** has within it all that Man is and ever will be. **POSITIONAL MATHEMATICS** is the number and words that represent the design of our universe and the experiences of Mankind. The Programs offered below are the first Positional Math programs to be published. A series of Lessons will be offered to introduce Positional Mathematics. **LESSON ONE** is ready. Lessons Two - Seven will be offered as completed.

1. **LESSON ONE** — *An Introduction to Positional Mathematics:* PART I, *Words & Numbers, the Universe;* PART II, *The Name of Man;* PART III, *Numbers & Health;* PART IV, *Wealth & Love Balance. Lessons are interactive with student's personal data input. Each* PART *cloads in 16K, Level II.*
2. **LOVE & SEX FOR A COUPLE** — *gives Love & Sex balance & conflicts, comments & positional dates.*
3. **INDIVIDUAL LIFE WEALTH** — *gives balance for five positions & the Life Wealth Balance.*
4. **INTRODUCTORY OFFER** — LESSON ONE *and the Program* INDIVIDUAL GROWTH . . . *for only* **$19.95.** *A savings of $29.95 . . . Lesson One price after May 30, 1981 will be $29.95 . . . most single programs will be $29.95 after July, 1981.*

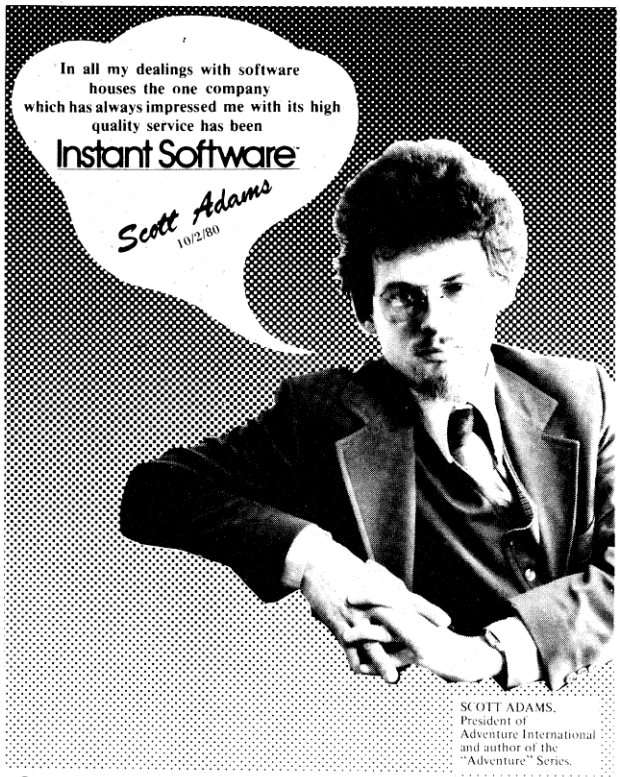
Programs are on cassette for TRS-80* Level II, 16K . . . introductory single program price . . . **$19.95.**

Order by Program Name . . . send check or money order . . . Texas residents add 5% sales tax . . . allow two weeks for domestic personal checks to clear . . . See future **TURTLE** ads for new programs.

*TRS-80 is a Trademark of Tandy Corp

Turtle Computer Software Corporation

10400 N. Central Expwy., Suite 307, Dallas, Texas 75231 ✓402



SCOTT ADAMS, President of Adventure International and author of the "Adventure" Series.

Instant Software

PETERBOROUGH, NEW HAMPSHIRE 03458 603-9247296

Are You Sitting on a Gold Mine?

Thousands of computerists have written programs and could be receiving royalty checks every month but they aren't....and won't, until they do something about publishing their programs.

Have you written any microcomputer programs that may be useful to others? Few applications or utilities are unique so the chances are excellent that many of the programs you have written are marketable, RIGHT NOW!

What kind of programs? Just about anything that others can use such as business, scientific, utility, educational, and games programs. Which microcomputer? Instant Software publishes programs for ALL popular microcomputer systems. (Instant Software also converts programs for use on other microcomputer systems and translates them into other languages, such as German, French, Italian, etc., for our international market).

Instant Software can transform your programs into royalty checks. We've set the standard for royalties to programmers. We have the publishing know-how, a world-wide distribution network and can do the most for you in selling the product of your genius.

Submit your programs on tape or disk, with complete documentation, to: Instant Software, Inc., Submissions Dept., Peterborough, NH 03458. If we can't use your program, you should hear from us in a few days. The good news takes a bit longer, but comes in the form of royalty checks.

Instant Software™

Peterborough, N.H. 03458

SAVE YOUR TIME ...AND YOUR MONEY.

Instant Software has the two best mail list programs available for your TRS-80 Model I and Model II.

Mail/List for Model I and Model II

This mailing list program maintains separate alphabetical and zip code files in *constant sort*. When you add a name to your list, it will *automatically* be inserted into its correct position in the files; therefore it's always ready to print labels!

It will record your information in these fields: NAME, ADDRESS, CITY, STATE, ZIP, PHONE NUMBER, PHONE EXTENSION and a five character CODE field. You have the choice of a 3 line, 4 line, or user defined label format. It can even include (optionally) a message line on your label!

The programs most outstanding feature is its sorting capabilities. Mail/List allows you to choose which names you want to be printed from the whole list. For example, all people in one zip code, or all people named Jones, who are living in a particular city or state. For any name in your list you can assign a code within the CODE field. You can then specify the code when printing labels, and only names with that code will be printed out. You can specify up to 9 different codes!

Every business and organization will save time and money with Mail/list to keep track of customers or members.

TRS-80 Model I version, Order No. 5000RD $99.00.
Requires 16K RAM, Expansion Interface with at least 16K RAM, one disk-drive and a printer.

TRS-80 Model II version, Order No.5001RD $199.00.
Requires 64K RAM and printer.

WRITE FOR OUR NEW CATALOG

TO ORDER: See Your Local Instant Software Dealer or Call Toll-Free 1-800-258-5473.

One-D Mailing List

Here is a mailing list system that can be run on only ONE disk-drive! You can have up to 17 fields of selection for name/address retrieval. •Disk versatility allows you to add, delete, or change the numerous details stored in the system. •Features of the One-D Mailing List includes: •Automatic name sort (alphabetically or by ZIP code). •Rapid access to any name on file. •Easy error correction and recovery. •Prints selective name listings. •Revise or update listings at any time. •Up to 2500 names on-line (with 4 drives). •Prints a list of all names on file. •Prints mailing labels. This package requires the following minimum system: 1. A TRS-80 Model I Level II microcomputer with 16K RAM. 2. An Expansion interface with 0 to 32K of RAM. 3. A single disk drive (extra drives optional). 4. A printer. 5. Any TRSDOS compatible Disk Operating System.

Order No. 0123RD $24.95.

Instant Software™

PETERBOROUGH, N.H. 03458

2.

TRS-80* Business Software with 2 Purposes

1. SAVES YOU TIME

Sales Analysis

This package is divided into several modules:

Sales Analysis: Will provide guidelines to determine and analyze an individual's sales performance and will show you where it can be improved.

Data Storage: Allows you to store data in an automated processing ledger. It will keep names, addresses, phone numbers, dates, ready for easy reference. The ledger will also show the progress of each sales prospect, in completing the sale.

Management Analysis: Will take all the sales records for your group and show you who your best salespersons are, who needs more training (and in what areas), and give you a sales forecast based on the projected improvement of your group's sales techniques.

Market Analysis: Shows you where determined sales efforts can produce the most success, when you supply data on marketing history. If your specialty is sales, we have a useful package for you. For the Level II, 16K.

Order No. 0131R $24.95

TO ORDER

SEE YOUR LOCAL INSTANT SOFTWARE DEALER

OR

call now

Toll-Free

1-800-258-5473

2. SAVES YOU MONEY

Oracle-80

Oracle-80 provides you with business anaylsis and forecasting capabilities previously available only on large computer systems. It is a flexible, professional time-series analysis and forecasting package that can be used in sales forecasting, product planning, business planning, etc. Investors can analyze stocks, company trends and growth rates. Financial managers and economists can analyze the general economic climate and investigate business cycles. Even families will find Oracle-80 useful in analyzing spending or energy consumption trends. Oracle-80 can be used by anyone who needs to analyze and forecast monthly, quarterly or annual data.

Even though it uses advanced statistical analysis, you don't have to understand statistics to use it. Oracle-80 was designed to be used and understood by the typical business person. While it is designed for ease of use, its powerful analytical capabilities will satisfy even the professional forecaster. All input and output are written in plain English and the package documentation carefully explains all the functions of the program.

You can use moving average, rate of change, seasonal indices or cycle indicies methods to analyze your data. The unique graphing capability of Oracle-80 lets you visualize your historic data or any of the modified data series you calculate. Additionally, you can direct any chart or graph to your printer.

It will forecast future data values using trend, moving average or seasonal methods. You may choose either a constant unit trend or a constant percentage growth trend forecast for even more flexibility.

Requires the following minimum system:

1. A TRS-80 Mod I,Level II 16K.
2. An Expansion Interface with 16K RAM.
3. One or more disk drives.
4. Any TRSDOS compatible DOS.
5. A printer (optional).

Order No. 0152RD (disk-based version) $99.95

For the Level II, 16K.

Order No. 0140R (cassette-based version) $75.00

Executive Expense Report Generator

You have just returned from a long, successful business trip. You are now faced with that ugly beast, THE EXPENSE REPORT!

Before you left, you resolved to record every expense in a little notebook. It worked fine for about two days. Now, you stare glassily at enigmatic scraps of paper which you've dug from various jacket pockets.

This program will take away that pain! No, it *won't* tell you how to cheat on that expense report. It will simply tell you how to play the game using established rules.

When you supply your tale of woe to this program, your answer will be a clear, plausible expense layout. The program has hardcopy capability, if you own a printer. For the Level II, 16K.

Order No. 0135R $9.95

Instant Software Guarantee

OUR PROGRAMS ARE GUARANTEED TO BE QUALITY PRODUCTS. IF NOT COMPLETELY SATISFIED YOU MAY RETURN THE PROGRAM WITHIN 60 DAYS. A CREDIT OR REPLACEMENT WILL BE WILLINGLY GIVEN FOR ANY REASON.

✓2

Instant Software™

*A trademark of Tandy Corporation

PETERBOROUGH, N.H. 03458
603-924-7296

Use this utility and it won't be a long time passing before you find out.

Where Have All The GOTOs Gone?

Hubert C. Borrmann
2840 S. Circle Drive
Colorado Springs, CO 80906

Where Do All My GOTOs go to? Or my GOSUBs, THENs, ELSEs, RESUMEs, and RUNs for that matter? This is a must to know, if you are making program changes and you want to be sure that changes in one area will not conflict with other areas. How many lines does it take for you to reach the instructions you are about to change? I'll describe two programs here.

Version 1

The first simple program will list all line numbers of your target program, which contain the instructions, and their addresses. The information will be in line number order and will be displayed, 15 lines at a time.

The program is kept simple and small, so that it may be appended to larger programs. Its line numbers are 62000-62170, and you append it to the tail-end of your program with simple commands. I will describe the procedure later.

After the program has been appended to your target program, and the two programs have become one, type "RUN 62000", and your TRS-80 starts to work. Why does it take so long to get started?

If you look at BASIC programs, you'll notice that the first line numbers are usually executed one after another, without branches. But, my program has to start searching from the beginning, and that takes some time.

You will then see the screen being filled with the desired information. The lines will be PEEKed until the search program recognizes that it has processed the last line of the combined program.

How do we recognize the end of a program? You will find a nice description of how a BASIC program looks in memory in the article "APPEND IT" by C.F. Gerald, February 1980, *80 Microcomputing*. This article says that the end of a program is indicated by zeros in the field, which normally contain the address of the next line. The line number field is two bytes long, however, not one, as indicated (i.e., byte three and four).

Version 2

This version is more elaborate, and the final output will be in order of the addresses of our branching instructions. It will show you not only where your GOTOs go to, but also where they come from.

To accommodate as large a program as possible, we run this version in two phases. The first phase is like version one, and its line numbers 62000-62200 are appended the same way. Here the output goes on tape, and every record is added to an output block until 21 records occupy it. Then the block is written on tape, so that phase two can read in those data and sort them.

The format of each record is: a five-position address, followed by a five-position line number and a one-position code for the instruction (GOTO, GOSUB, etc). After Phase One has been appended to your program, remove the program cassette (don't rewind) and put it aside until you need to load Phase 2.

Then, insert a cassette into the recorder, prepare it for RECORD and type: RUN 62000. When the target program has been completely checked, the last block will be written out. Now re-insert the program cassette and load Phase Two. Type RUN and re-insert the data cassette.

Phase Two is dimensioned to accomodate 300 records, but may be changed in line 100. The data are then sorted, and will be printed, 15 lines at a time. This sequence is better than the output of version one, and I find it

```
62000 CLS:CLEAR100:A=17129:E=0:PRINT@15,"PLEASE BE PATI
    ENT"
62010 B=PEEK(A+1)*256+PEEK(A):IF B=0 THEN END
62020 A=A+2:C=PEEK(A+1)*256+PEEK(A):A=A+2
62030 IF PEEK(A)=141THEN K1$="GOTO":GOTO 62080
62040 IF PEEK(A)=145THEN K1$="GOSUB":GOTO 62080
62050 IF PEEK(A)=202THEN K1$="THEN":GOTO 62080
62060 IF PEEK(A)=149THEN K1$="ELSE":GOTO 62080
62065 IF PEEK(A)=159THEN K1$="RESUME":GOTO 62080
62068 IF PEEK(A)=142THEN K1$="RUN":GOTO 62080
62070 A=A+1:IF A=B THEN 62010 ELSE 62030
62080 K2$=""
62090 A=A+1:IF PEEK(A)=32 THEN 62090 ELSE D=A
62100 IF PEEK(D)<48 OR PEEK(D)>57 THEN 62120 ELSE K2$=K
    2$+CHR$(PEEK(D))
62110 D=D+1:GOTO 62100
62120 IF D=A THEN 62070
62130 E=E+1:PRINTTAB(10)"LINE # ";C;" CONTAINS.....";K1
    $;" ";K2$
62140 IF E<15 THEN 62170 ELSE E=0
62150 PRINT@990,"PRESS 'SPACE' WHEN READY";:K3$=INKEY$:
    IF K3$="" THEN 62150
62160 CLS:PRINT@15,"CONTINUING"
62170 A=D:IF PEEK(D)=44THEN62080 ELSE 62070
```

Program Listing 1. Version 1

worth the effort to run it in two phases.

Appending

This is a description of how to append one program to the end of another:

- Load your target program into memory. Its line numbers must be lower than the numbers of the appended program.
- Command: PRINT PEEK (16633).
- If the number printed is two or greater:

```
Command: POKE 16548,PEEK(16633)-2
         POKE 16549,PEEK(16634).
Go to Step 5.
```

- If number of Step 2 is zero or 1:

```
Command: POKE 16548,PEEK(16633)+254
         POKE 16549,PEEK(16634)-1.
```

- CLOAD the program to be appended. Do not RUN. (Line numbers must be larger than the numbers of the program in memory.)
- Command: POKE 16548,233

Now you are ready and have one combined program. RUN 62000.

Version 1 in Detail

Look at the flowchart, Fig.1, and Program Listing 1. Variable A will be our scan-pointer, and we will walk this pointer from 17129 until the end of the combined program. We start in line 62000 where we set the screen-line counter E to zero, and print the message to be patient, to keep our fingers away from the BREAK key. At 62010 we take the first two bytes and calculate the address of the next line and keep it in B. If this is two zeros, then we have processed the last line of the program, and we are done, END (line 62010).

The next two bytes are our current line number, and we keep them in C. Variable A now points to the fifth position of the program line, and we search for our desired instructions. All instructions are carried in abbreviated form, and we check lines 62030-62068 for the decimal notations.

Let's look at the flowchart and assume that A does not point to a desired instruction. We go to 62070 where we increment A and now we compare it against B. B contains the address of the next line, and if the two variables are equal, we go to 62010. Otherwise, we go and make our checks again (62030-62082).

Let us now assume that we did find a match in one of these tests. We keep the instruction in variable K1$. Now we have to find the address (line number) to which our instruction GOSUBs, etc. At 62090, we bypass any spaces and keep incrementing A until we fine a non-space. We check for numerics.

Let us assume that we encountered a numeral: We build our GOTO-address in K2$ a non-numeral byte. We now are ready to print our information on the screen. We have our line number in C, our instruction in K1$, and our address in K2$. Or do we? We check this in line 62120 where we check our work-variable D. We find out whether we encountered an instruction like IF A = 9 THEN A = 1.

This is not what we wanted. We go to 62070 and continue our search.

But if we did find an address as in IF A = 9 THEN 1230 we print our information and add one to the screen line counter and check this counter E.

Let us assume that the screen is full: We print the message to PRESS 'SPACE' WHEN READY, line 62150, and wait for keyboard action. We then clear the screen and continue. We also reset the screen-line counter E. If the screen still has room, we bypass all this.

What now? If we are at the end of a line, check the next one. If not, check the next instruction on this line. But first, we want to be sure that we completed this instruction. We check for a comma. If we find one, we work with an instrucion like ON A GOTO 1020,1030,1040. We go to 62080 and look for the address after the comma and build another line. Then check again.

Look at line 62100. Addresses are carried in memory, one numeral in one byte, except for the first two of each line (the address of the next line and the current line number).

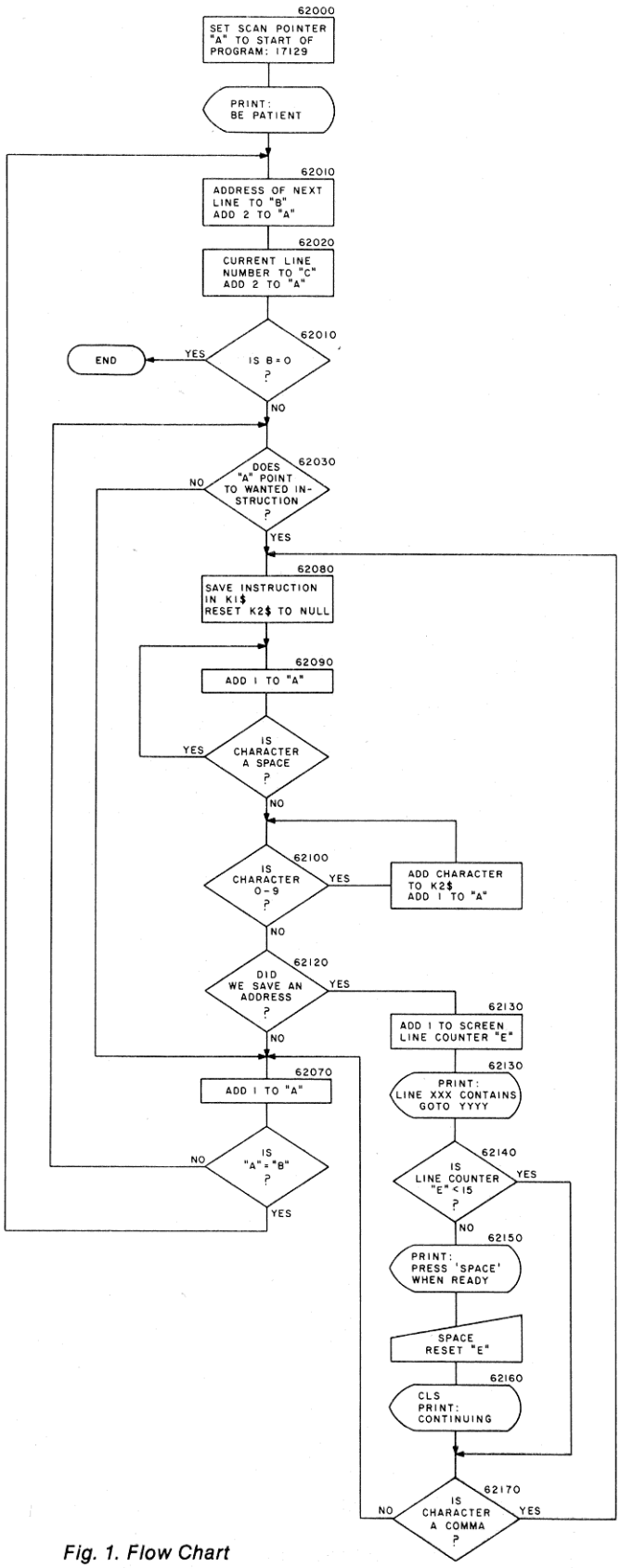


Fig. 1. Flow Chart

Comments on Version 2

Phase One is like version one. Here are some points, though. Since we build a STRING,K4$, we have to convert numerics to string-variables, as we do with the line number variable C in line 62130. We gain an extra space (from the sign and eliminate this

```
62000 CLS:CLEAR 4000:A=17129:E=0:PRINT@64,"BE PATIENT,
      I AM WORKING. READY RECORDER TO 'RECORD'":K4$="":F
      =1
62010 B=PEEK(A+1)*256+PEEK(A):IF B=0 THEN 62190
62020 A=A+2:C=PEEK(A+1)*256+PEEK(A):A=A+2
62030 IF C>62000 AND C<62201 THEN 62080
62040 IF PEEK(A)=141THEN K1$="A":GOTO 62090
62050 IF PEEK(A)=145THEN K1$="B":GOTO 62090
62060 IF PEEK(A)=202THEN K1$="C":GOTO 62090
62070 IF PEEK(A)=149THEN K1$="D":GOTO 62090
62072 IF PEEK(A)=159THEN K1$="E":GOTO62090
62074 IF PEEK(A)=142THEN K1$="F":GOTO 62090
62080 A=A+1:IF A=B THEN 62010 ELSE 62030
62090 K2$=""
62100 A=A+1:IF PEEK(A)=32 THEN 62100 ELSE D=A
62110 IF PEEK(D)<48 OR PEEK(D)>57 THEN62130ELSE K2$=K2$
      +CHR$(PEEK(D))
62120 D=D+1:GOTO 62110
62130 IF D=A THEN 62080 ELSE E=E+1:C$=STR$(C):C$=MID$(C
      $,2,LEN(C$)-1):PRINT@280,C$
62150 IF LEN(C$)<5 THEN C$="."+C$:GOTO 62150
62160 IF LEN(K2$)<5 THEN K2$="."+K2$:GOTO 62160
62170 K4$=K4$+K2$+C$+K1$:F=F+1:IF F<22 THEN 62200 ELSE
      F=1
62180 PRINT#-1,K4$:K4$="":GOTO 62200
62190 K4$=K4$+"#END#":PRINT#-1,K4$:PRINT@448,"REWIND AN
      D REMOVE DATA TAPE, CLOAD PHASE 2 AND 'RUN'":END
62200 A=D:IF PEEK(D)=44THEN 62090 ELSE 62080
```

Program Listing 2. Version 2, Phase 1

in C$ = Mid$(C$,2,LEN(C$)-1).

We make line number (C$) and address (K2$) each five positions long by inserting periods. This makes sorting easier in Phase Two.

Phase Two is straightforward (see Program Listing 3), stands alone and does not have to be appended. It gets its data from the cassette, sorts by major sequence—line number address to which the instruction branches, and sorts by minor sequence—line number which contains this instruction.

Each data block of 21 records is read into A$ and unblocked into E$(C).

Those are the two versions of our cross reference program, and either one is valuable for anybody modifying BASIC programs. ■

```
100 CLS:CLEAR 7000:C=1:DIM E$(300):PRINT "READY RECORDE
     R TO READ IN DATA FROM PHASE 1. PRESS 'SPACE'."
120 A$=INKEY$:IF A$="" THEN 120ELSE CLS
130 INPUT#-1,A$:B=1:D=1
140 IF MID$(A$,B,5)="#END#" THEN 160ELSE E$(C)=MID$(A$,
     B,11):C=C+1
150 B=B+11:D=D+1:IF D>21 THEN 130ELSE 140
160 C=C-1:PRINT:PRINTTAB(8) C;"DATA LOADED, NOW SORTING
     THEM."
170 F=C:G=1
180 G=2*G:IF G<F THEN 180
190 G=INT((G-1)/2):IF G=0 THEN 240
200 L=F-G:FOR I=1 TO L:J=I
210 K=J+G
220 IF MID$(E$(K),1,10)<MID$(E$(J),1,10) THEN TE$=E$(J)
     :E$(J)=E$(K):E$(K)=TE$:J=J-G:IF J>0 THEN 210
230 NEXT I:GOTO 190
240 CLS:M=1:N=1
250 IF N-1=C THEN PRINT"PRESS 'SPACE' TO REPEAT PRINTOU
     T, 'ENTER' TO END.":GOTO 390
260 IF MID$(E$(N),1,5) = MID$(E$(N-1),1,5) THEN 280
270 PRINT CHR$(149);STRING$(13,"-");MID$(E$(N),1,5);STR
     ING$(44,"-");CHR$(170);:M=M+1
280 PRINT CHR$(149);N;
290 IF MID$(E$(N),11,1)="A" THEN P$="GOTO":GOTO 350
300 IF MID$(E$(N),11,1)="B" THEN P$="GOSUB":GOTO 350
310 IF MID$(E$(N),11,1)="C" THEN P$="THEN":GOTO 350
320 IF MID$(E$(N),11,1)="D" THEN P$="ELSE":GOTO 350
330 IF MID$(E$(N),11,1)="E" THEN P$="RESUME":GOTO 350
340 P$="RUN"
350 PRINTTAB(14)".....";P$;TAB(27) MID$(E$(N),1,5);" -
     - - FROM LINE # ";MID$(E$(N),6,5);TAB(63) CHR$(17
     0);
360 M=M+1:N=N+1:IF M<15 THEN 250ELSE PRINT STRING$(63,"
     ");:M=1
370 PRINT@980,"PRESS 'SPACE' WHEN READY";:TE$=INKEY$:IF
     TE$="" THEN 370
380 CLS: GOTO 250
390 TE$=INKEY$:IF TE$=CHR$(32) THEN 240ELSE IF TE$=CHR$
     (13) THEN END ELSE 390
```

Program Listing 3. Version 2, Phase 2

MULTI-USER OASIS HAS THE FEATURES PROS DEMAND. READ WHY.

(THEN COMPARE.)

Computer experts (the pros) usually have big computer experience. That's why when they shop system software for Z80 micros, they look for the big system features they're used to. And that's why they like Multi-User OASIS. You will too.

DATA INTEGRITY: FILE & AUTOMATIC RECORD LOCKING

The biggest challenge for any multi-user system is co-ordinating requests from several users to change the same record at the same time.

Without proper co-ordination, the confusion and problems of inaccurate or even destroyed data can be staggering.

Our File and Automatic Record Locking features solve these problems.

For example: normally all users can view a particular record at the same time. But, if that record is being updated by one user, automatic record locking will deny all other users access to the record until the up-date is completed. So records are always accurate, up-to-date and integrity is assured.

Pros demand file & automatic record locking. OASIS has it.

SYSTEM SECURITY: LOGON, PASSWORD & USER ACCOUNTING

Controlling who gets on your system and what they do once they're on it is the essence of system security.

Without this control, unauthorized users could access your programs and data and do what they like. A frightening prospect isn't it?

And multi-users can multiply the problem.

But with the Logon, Password and Privilege Level features of Multi-User OASIS, a system manager can specify for each user which programs and files may be accessed—and for what purpose.

Security is further enhanced by User Accounting—a feature that lets you keep a history of which user has been logged on, when and for how long.

Pros insist on these security features. OASIS has them.

EFFICIENCY: RE-ENTRANT BASIC

A multi-user system is often not even practical on computers limited to 64K memory.

OASIS Re-entrant BASIC makes it practical.

How?

Because all users use a single run-time BASIC module, to execute their compiled programs, less memory is needed. Even if you have more than 64K, your pay-off is cost saving and more efficient use of all the memory you have available—because it services more users.

Sound like a pro feature? It is. And OASIS has it.

AND LOTS MORE...

Multi-User OASIS supports as many as 16 terminals and can run in as little as 56K memory. Or, with bank switching, as much as 784K.

Multi-Tasking lets each user run more than one job at the same time.

And there's our BASIC—a compiler, interpreter and debugger all in one. An OASIS exclusive.

Still more: Editor; Hard & Floppy Disk Support; Keyed (ISAM), Direct & Sequential Files; Mail-Box; Scheduler; Spooler; all from OASIS.

Our documentation is recognized as some of the best, most extensive, in the industry. And, of course, there's plenty of application software.

Put it all together and it's easy to see why the real pros like OASIS. Join them. Send your order today.

OASIS IS AVAILABLE FOR

SYSTEMS: Altos; Compucorp; Cromemco; Delta Products; Digital Group; Digital Microsystems; Dynabyte; Godbout; IBC; Index; Intersystems; North Star; Onyx; SD Systems; TRS 80 Mod II; Vector Graphic; Vorimex.

CONTROLLERS: Bell Controls; Cameo; Corvus; Konan; Micromation; Micropolis; Tarbell; Teletek; Thinkertoys; X Comp.

Write for complete free Application Software Directory.

PLEASE SEND ME:

Product	Price with Manual	Manual Only
OPERATING SYSTEM (Includes: EXEC Language; File Management; User Accounting; Device Drivers; Print Spooler; General Text Editor; etc.)		
SINGLE-USER	$150	$17.50
MULTI-USER	350	17.50
BASIC COMPILER/ INTERPRETER/DEBUGGER	100	15.00
RE-ENTRANT BASIC COMPILER/INTERPRETER/ DEBUGGER	150	15.00
DEVELOPMENT PACKAGE (Macro Assembler; Linkage Editor; Debugger)	150	25.00
TEXT EDITOR & SCRIPT PROCESSOR	150	15.00
DIAGNOSTIC & CONVERSION UTILITIES (Memory Test; Assembly Language; Converters; File Recovery; Disk Test; File Copy from other OS; etc.)	100	15.00
COMMUNICATIONS PACKAGE (Terminal Emulator; File Send & Receive)	100	15.00
PACKAGE PRICE (All of Above)		
SINGLE-USER	500	60.00
MULTI-USER	850	60.00
FILE SORT	100	15.00
COBOL-ANSI '74	750	35.00

Order OASIS from:

Phase One Systems, Inc.
7700 Edgewater Drive, Suite 830
Oakland, CA 94621
Telephone (415) 562-8085
TWX 910-366-7139

NAME _____
STREET (NO BOX #) _____
CITY _____
STATE _____ ZIP _____

AMOUNT $ _____
(Attach system description; add $3 for shipping; California residents add sales tax)
☐ Check enclosed ☐ VISA
☐ UPS C.O.D. ☐ Mastercharge
Card Number _____
Expiration Date _____
Signature _____

OASIS

MAKES MICROS RUN LIKE MINIS

A useful loan calculator program.

The Loan Wrangler

Garth Jensen
59 Oak Drive, Ancient Oak
Macungie, PA 18062

This program will take the effort out of calculating monthly payments on a loan, annual interest rates, future values of savings plans, etc. It will help you plan financial transactions in advance, and in the process prepare you to deal with lending institutions.

To be consistent with lending institutions, the Loan Calculator program assumes monthly compounding of interest. For a fixed monthly payment, you can see that the payments of principal and interest change each month. See Sample 3.

Running the Program

When you type RUN, a menu is presented allowing calculation of the amount borrowed, the future value of an annuity, the annual interest rate, the amount of one monthly payment on the life of the loan. You are prompted to enter specific details about the loan. The routine is then executed and output options presented.

"Each month's payment is split into principal and interest and a running principal balance is maintained."

Samples 1 through 4 illustrate some of the uses of the program.

Borrowing

Suppose you want to buy a new car and can afford payments of $150 per month. Assume a 12% interest rate on auto loans and a three year life. Run the Loan Calculator, specify menu item one, and enter the appropriate data when prompted:

```
Enter annual interest rate in percent ? 12
Enter amount of one monthly payment ? 150
Enter total life in months ? 36
```

If you request the Terms Only output, the results will appear as shown in Sample 1. You can afford to borrow $4516.

A second use of this routine is to determine the total savings required in order to withdraw a fixed amount each month. For example, what savings are required in a bank account paying 5.25% interest in order to provide supplement retirement income of $100 per month for 10 years? Entering the interest rate of 5.25%, the monthly payment of $100 and the total life of 120 months yields a required savings of $9321. After 10 years of withdrawing $100 per month, the account would be emptied.

Value of an Annuity

This routine provides a way to evaluate a savings program. If you save $50 per month in a 7% credit union account, how much will you have in the account after 15 years? After entering the interest rate, monthly payment, and 180 month life, you are informed that your efforts result in a savings balance of $15,848. See Sample 2.

```
LOAN TERMS

AMOUNT BORROWED:          $ 4516.08
ANNUAL INTEREST RATE:       12 %
MONTHLY PAYMENT:          $ 150
LIFE:                       36 MONTHS
```

Sample Output 1. Calculate Amount Borrowed

```
LOAN TERMS

ANNUAL INTEREST RATE:       7 %
MONTHLY PAYMENT:          $ 50
LIFE:                       180 MONTHS
FUTURE VALUE              $ 15847.9
```

Sample Output 2. Calculate Future Value of an Annuity

Program Listing

```
45 ' MAJOR VARIABLES USED IN CALCULATIONS:
50 ' A=AMOUNT BORROWED; B=REQUESTED CALCULATION; F=FUTU
   RE VALUE OF ANNUITY
55 ' I=MONTHLY INTEREST RATE; IA=ANNUAL INTEREST RATE;
   N=LOAN LIFE IN MONTHS
60 ' P=MONTHLY PAYMENT; PF=FACTOR FOR CALCULATING PRESE
   NT VALUE OF AN ANNUITY
65 ' MAJOR VARIABLES USED IN OUTPUT:
70 ' M=MONTH; Y=YEAR; TP=TOTAL PRINCIPAL PAID IN ONE YE
   AR
75 ' TI=TOTAL INTEREST PAID IN ONE YEAR; PP=PRINCIPAL P
   AID IN PERIOD
80 ' BP=BEGINNING PRINCIPAL BALANCE; EP=ENDING PRINCIPA
   L BALANCE
85 ' IP=INTEREST PAID IN PERIOD
90 CLEAR100:CLS:S$="######.##  ":S$=S$+" "+S$+S$+" "+S$
95 'MENU ROUTINE
100 PRINT:PRINTTAB(20)"LOAN CALCULATOR":PRINT:PRINT
110 PRINT"WHAT DO YOU WANT TO FIND?":PRINT
120 PRINTTAB(10)"1-AMOUNT BORROWED"
125 PRINTTAB(10)"2-FUTURE VALUE OF ANNUITY"
130 PRINTTAB(10)"3-ANNUAL INTEREST RATE"
135 PRINTTAB(10)"4-AMOUNT OF ONE MONTHLY PAYMENT"
140 PRINTTAB(10)"5-LIFE OF LOAN":PRINT
150 INPUT"ENTER ITEM NUMBER";B:IFB<1ORB>5THEN150ELSECLS
155 'ENTER RELEVANT DATA
160 IFB=1ORB=2THEN170ELSEINPUT"ENTER AMOUNT BORROWED";A
    :IFA<0THEN160
170 IFB=3THEN180ELSEINPUT"ENTER ANNUAL INTEREST RATE IN
     PERCENT";I
175 IFI<=0THEN170ELSEI=I/1200
180 IFB=4THEN190ELSEINPUT"ENTER AMOUNT OF MONTHLY PAYME
    NT";P:IFP<=0THEN180
190 IFB=5THEN200ELSEINPUT"ENTER TOTAL LIFE IN MONTHS";N
    :IFN<=0THEN190
200 ONBGOTO300,400,500,600,700
210 STOP
220 INPUT"HIT 'ENTER' TO CONTINUE";A$:RETURN 'PAUSE SUB
    ROUTINE
300 'CALCULATE AMOUNT BORROWED
310 PF=(1-(1+I)[(-N))/I:A=(INT(P*PF*100+.5))/100:GOSUB3
    50:GOTO800
350 'SUBROUTINE TO TEST FOR UNDEFINED LIFE.  IF THE PAY
    MENT DOES NOT COVER
355 'THE INTEREST REQUIREMENT, PRINCIPAL GROWS AND THE
    LIFE IS UNDEFINED
360 IFP-A*I>.1THENRETURNELSEPRINT
370 PRINT"ONE MONTHLY PAYMENT = $";P;", AND THE"
380 PRINT"INTEREST REQUIRED = $";A*I;"."
390 PRINT"THEREFORE, THE LIFE OF THE LOAN IS UNDEFINED.
    ":END
400 'CALCULATE FUTURE VALUE OF THE ANNUITY
410 F=P*((1+I)[N-1)/I:F=(INT(F*100+.5))/100:GOTO800
500 'CALCULATE ANNUAL INTEREST RATE
505 'RANGE TESTED IS 0% TO 360% USING BINARY SEARCH
510 IFN*P>ATHENT=.3:B=0:GOTO530
520 PRINT"INTEREST RATE IS NEGATIVE OR ZERO":PRINT:END
530 I=(T+B)/2:PF=(1-(1+I)[(-N))/I:PV=P*PF
535 'IF TRIAL PRESENT VALUE = AMT BORROWED, THEN
536 'INTEREST RATE IS FOUND
540 IFABS(PV-A)<=.01THEN800
545 IFABS(I-.3)<=.00001 PRINT"INTEREST RATE ABOVE 360%"
    :END
550 IFPV>ATHENB=I:GOTO530:ELSET=I:GOTO530
600 'CALCULATE MONTHLY PAYMENT
610 PF=(1-(1+I)[(-N))/I:P=(INT(A/PF*100+.5))/100:GOTO80
    0
700 'CALCULATE LIFE OF LOAN
710 GOSUB350
715 'PRINCIPAL GROWS AND LIFE IS INFINITE IF PAYMENT
716 'DOES NOT COVER INTEREST
720 N=LOG(P/(P-A*I))/LOG(1+I)
800 'DISPLAY RESULTS
805 IFB=2THENC=1:GOTO814
810 CLS:PRINTTAB(20)"SPECIFY OUTPUT FORMAT":PRINT
811 PRINT"WANT 1-TERMS ONLY, 2-ANNUAL, OR 3-MONTHLY SUM
    MARY?"
812 INPUT"ENTER 1, 2, OR 3";C:IFC<1ORC>3THEN810
814 PRINT:PRINT"WANT OUTPUT TO 1-VIDEO, OR 2-PRINTER?"
816 INPUT"ENTER 1 OR 2";C2:PRINT:IFC2<1ORC2>2THEN814
818 IFC2=1THENPRINT:GOTO825
820 IFPEEK(14312)=63THEN825
822 PRINT:PRINT"GET PRINTER READY":GOSUB220:GOTO814
825 IFC=1THEN830ELSEINPUT"ENTER MONTH OF YEAR FIRST PAY
    MENT IS MADE (1-12)";M
826 MM=M:IFM<1 ORM>12THEN825
827 INPUT"ENTER OUTPUT TABLE LENGTH IN YEARS";L:IFL<1TH
    EN827
830 CLS:IFC2=1THENPRINTTAB(14)"LOAN TERMS":PRINT:GOTO84
    0
835 LPRINTTAB(14)"LOAN TERMS":LPRINT" "
840 IFC2=1THENIFB<>2PRINT"AMOUNT BORROWED:","$";A:GOTO8
    50
845 IFB<>2LPRINT"AMOUNT BORROWED:","$";A
850 IFC2=1THENPRINT"ANNUAL INTEREST RATE:",1200*I;"%":G
    OTO860
```

Program continues

↙ *Reader Service—see page 290*

Annual Interest Rate

This routine calculates the interest rate on a loan. Assume that a book salesman offers you a set of encyclopedias for $1000 cash, or $50 per month for 24 months. What is the interest rate in this installment purchase plan? Enter $1000 as the amount borrowed, $50 as the payment, and 24 months for the life of the loan. The results are given in Sample 3.

Each month's payment is split into principal and interest and a running principal balance is maintained. Also, each year is summarized for convenient record keeping.

Monthly Payments

Situations arise when you need to determine monthly payments. Assume that you are considering the purchase of a new house costing $85,000. You have funds available to pay closing costs and make a 20% down payment on the house. The 80% to be borrowed is $68,000. At a 12% interest rate over a 30 year life, what is the monthly payment? Sample 4 answers our question, illustrating the Annual Summary output format.

Another use of this routine calculates withdrawals. Suppose you have $10,000 in a bank account earning 5.25% interest, and want to withdraw equal monthly amounts over five years. What amount can you withdraw monthly? The Loan Calculator provides the answer of $190.

Life of the Loan

Let's extend the example of purchasing encyclopedias. As shown in Sample 3, the principal remaining to be paid at the end of the first year is $666. If you make a payment of $300 at the end of year one and continue with the $50 per month payments, how many more payments will be required? Inputting $366 as the amount borrowed, 18.16% as the interest rate, and $50 as the payment, you learn that 7.8 payments remain.

It is also a simple matter to calculate the interest you avoid by making the $300 advance payment. Add the interest paid through year one ($116.49 from Sample 3) to the interest in the remaining payments ($24.90 in 7.8 payments). The total of $141.39 compares to the $199.97 total of the three years' interest from Sample 3. You save $58.58 in interest by making the $300 payment.

Summary

Monthly compounding of interest guarantees accurate results in almost all common borrowing situations. Monthly compounding also yields close approximations of daily interest savings plans.

I hope this program proves to be useful in helping you make optimal financing decisions. For a program cassette, send $6.00 to the author. ■

```
                LOAN TERMS

AMOUNT BORROWED:                  $ 68000
ANNUAL INTEREST RATE:               12 %
MONTHLY PAYMENT:                  $ 699.46
LIFE:                               360 MONTHS

                BEGINNING   PRINCIPAL                      ENDING
YEAR  MONTH     PRINCIPAL   REPAYMENT     INTEREST       PRINCIPAL
  1   TOTAL     68000.00      246.79       8146.73       67753.20
  2   TOTAL     67753.20      278.10       8115.42       67475.10
  3   TOTAL     67475.10      313.38       8080.14       67161.70
  4   TOTAL     67161.70      353.11       8040.41       66808.60
  5   TOTAL     66808.60      397.88       7995.64       66410.80
  6   TOTAL     66410.80      448.35       7945.17       65962.40
  7   TOTAL     65962.40      505.22       7888.30       65457.20
  8   TOTAL     65457.20      569.30       7824.22       64887.90
  9   TOTAL     64887.90      641.48       7752.04       64246.40
 10   TOTAL     64246.40      722.86       7670.66       63523.60
 11   TOTAL     63523.60      814.53       7578.99       62709.00
 12   TOTAL     62709.00      917.82       7475.70       61791.20
 13   TOTAL     61791.20     1034.23       7359.29       60757.00
 14   TOTAL     60757.00     1165.39       7228.13       59591.60

                    Etc.
```

Sample Output 3. Calculate Annual Interest Rate

```
855 LPRINT"ANNUAL INTEREST RATE:",1200*I;"%"
860 IFC2=1THENPRINT"MONTHLY PAYMENT:","$";P:GOTO870
865 LPRINT"MONTHLY PAYMENT:","$";P
870 IFC2=1THENPRINT"LIFE:",,N;"MONTHS":GOTO880
875 LPRINT"LIFE:",,N;"MONTHS"
880 IFC2=1THENIFB=2PRINT"FUTURE VALUE",,"$";F:GOTO890
885 IFB=2LPRINT"FUTURE VALUE",,"$";F
890 IFC=1ORB=2THENPRINT:END
900 'PRINT SUMMARIES
910 Y=0:EP=A:PRINT:PRINT:GOSUB220:CLS:GOSUB1100
920 Y=Y+1:TP=0:TI=0:IFY>1THENMM=1
930 FORM=MMTO12:BP=EP
940 IP=INT(100*BP*I+.5)/100:TI=TI+IP:PP=P-IP:IFPP>BPTHE
    NPP=BP:L=0
950 TP=TP+PP:EP=BP-PP
955 'IF REQUESTED, PRINT MONTHLY DETAILS
960 IFC2=1ANDC=3PRINTUSING"####     ##     ";Y,M;:PRINTUS
    INGS$;BP,PP,IP,EP:GOTO980
970 IFC=3 LPRINTUSING"####     ##     ";Y,M;:LPRINTUSINGS
    $;BP,PP,IP,EP
980 IFEP=0THENL=0:GOTO990:ELSEIFY<=LTHEN NEXTM
985 'PRINT ANNUAL SUMMARIES
990 IFC2=1THENPRINTUSING"####    TOTAL  ";Y;:PRINTUSINGS
    $;EP+TP,TP,TI,EP:GOTO1010
1000 LPRINTUSING"####    TOTAL  ";Y;:LPRINTUSINGS$;EP+TP
    ,TP,TI,EP
1010 IFL=0ORY=LTHENEND
1020 IFC=3THENGOSUB220:GOSUB1100
1030 GOTO920
1100 'PRINT SUMMARY HEADINGS
1105 IFC2=2THEN1130
1110 PRINT:PRINTTAB(15)"BEGINNING    PRINCIPAL
         ENDING"
1120 PRINT" YEAR  MONTH    PRINCIPAL    REPAYMENT    INTER
    EST    PRINCIPAL":RETURN
1130 LPRINT" ":LPRINTTAB(15)"BEGINNING    PRINCIPAL
                ENDING"
1140 LPRINT" YEAR  MONTH    PRINCIPAL    REPAYMENT    INTE
    REST    PRINCIPAL":RETURN
```

"Monthly compounding of interest guarantees accurate results in almost all common borrowing situations."

```
                LOAN TERMS

AMOUNT BORROWED:                    $ 1000
ANNUAL INTEREST RATE:                18.1563 %
MONTHLY PAYMENT:                    $ 50
LIFE:                                24 MONTHS

                BEGINNING   PRINCIPAL                 ENDING
YEAR   MONTH    PRINCIPAL   REPAYMENT   INTEREST    PRINCIPAL
  1      4       1000.00      34.87       15.13       965.13
  1      5        965.13      35.40       14.60       929.73
  1      6        929.73      35.93       14.07       893.80
  1      7        893.80      36.48       13.52       857.32
  1      8        857.32      37.03       12.97       820.29
  1      9        820.29      37.59       12.41       782.70
  1     10        782.70      38.16       11.84       744.54
  1     11        744.54      38.73       11.27       705.81
  1     12        705.81      39.32       10.68       666.49
  1    TOTAL     1000.00     333.51      116.49       666.49

                BEGINNING   PRINCIPAL                 ENDING
YEAR   MONTH    PRINCIPAL   REPAYMENT   INTEREST    PRINCIPAL
  2      1        666.49      39.92       10.08       626.57
  2      2        626.57      40.52        9.48       586.05
  2      3        586.05      41.13        8.87       544.92
  2      4        544.92      41.76        8.24       503.16
  2      5        503.16      42.39        7.61       460.77
  2      6        460.77      43.03        6.97       417.74
  2      7        417.74      43.68        6.32       374.06
  2      8        374.06      44.34        5.66       329.72
  2      9        329.72      45.01        4.99       284.71
  2     10        284.71      45.69        4.31       239.02
  2     11        239.02      46.38        3.62       192.64
  2     12        192.64      47.09        2.91       145.55
  2    TOTAL      666.49      520.94       79.06       145.55

                BEGINNING   PRINCIPAL                 ENDING
YEAR   MONTH    PRINCIPAL   REPAYMENT   INTEREST    PRINCIPAL
  3      1        145.55      47.80        2.20        97.75
  3      2         97.75      48.52        1.48        49.23
  3      3         49.23      49.23        0.74         0.00
  3    TOTAL      145.55     145.55        4.42         0.00
```

Sample Output 4. Calculate Monthly Payment

✓ Reader Service—see page 290

Find out what lurks in your TRS-80.

The Shadow Knows

John L. Romkey
27 Blue Hill West
Bangor, ME 04401

Have you ever wondered what dread secrets the TRS-80 holds? How about the reserved areas in RAM that eat part of your memory? With the help of a good disassembler and a lot of good luck, I've discovered interesting things about these valleys of shadows.

Memory Map

Radio Shack's memory map shows reserved areas extending from 402D to 41E6 (I'll use hex notation). Keyboard RAM starts at 4000; when you PRINT MEM and are told you have only 15572 bytes of free RAM, BASIC is protecting RAM areas.

You can't put them in ROM because BASIC has to keep track of many things, and uses the reserved areas for holding status information, passing values to functions, and implementing tables so its capabilities can be complemented. Table 1 lists some locations in the reserved area, most requiring two bytes per word to store status information. Those needing only one byte are preceded by a *b* in the first column of the Table.

Suppose we want to find out where the program starts. Table 1 refers you to the word in 40A4. Enter: PRINT PEEK(16548)+PEEK(16549)*256. Hex 40A4 is 16548; multiply the second byte by 256 so that the address will be calculated correctly. The Z-80 stores a number like 3A7C in memory as 7C 3A, since 7C and 3A are both bytes.

If you've never worked with machine language, don't panic. Everything in this part of the article can be used directly from BASIC.

403D This address has a single byte which determines whether the video display is in the 32 or 64 character mode. If the byte is an eight, the screen will display 32 characters. If it is a 0, the screen will display 64 characters. Clear the screen and try this: POKE 16445,8. Your display should be 32 characters. CHR$(28) will get you out of the 32-character mode without clearing the screen, but the cursor will be placed in the upper left corner of the screen.

Put the screen in the 32-character mode and try this: POKE 16445,0. The cursor is where it should be. There are spaces between letters because only every other video location is used since the characters are twice the usual width.

409A This code indicates the last error. When you use ERR, you get the contents of this location. Here's how the code goes: If it is a 0, the error is a NEXT without FOR; if it is a 2, the error is syntax; 4—the error is RETURN without GOSUB, etc. The codes are all even because this simplifies the way BASIC issues its error messages. Take a look at ROM. At address 18C9 (6435 decimal) there is a group of letters. They are NFSNRGODFCOVMUL, which are Level II's error messages! BASIC figures out which message to print by adding the error code to 18C9 and showing the two letters at the resultant address. Address 40F2 is the error flag. A flag is a value that lets the computer know something has happened. If the byte at 40F2 is not 0, there is an

Length	Address Hex	Address Decimal	Function
b	403D	16445	Current screen status
b	409A	16538	Current internal error code
	40A0	16544	Pointer to bottom of string storage
	40A2	16546	Current line number
	40A4	16548	Pointer to start of program
b	40AF	16559	Type of argument in function
	40B1	16561	Top of BASIC RAM
	40D6	16598	Pointer to bottom of used string storage
	40DF	16607	Pointer to most recent change in variable table
b	40E1	16609	Auto flag
	40E2	16610	Current auto line number
	40E4	16612	Auto increment
	40E6	16614	Pointer to current character
	40E8	16616	Highest possible stack address with current string storage
	40EA	16618	Current error's line number
	40EC	16620	Current line number
	40EE	16622	Pointer to I/O buffer
	40F0	16624	Pointer to line of BASIC error handling
b	40F2	16626	Error flag
	40F9	16633	Pointer to end of program
	40FB	16635	Pointer to top of variable table
	40FD	16637	Pointer to top of array table
	40FF	16639	Pointer to current data item
	4101-411A	16641-16666	Implicit variable type table
b	411B	16667	Trace flag
	411D-	16673-	Argument for function
	4152-41A5	16722-16805	Disk BASIC command table
	41A6-41E5	16806-16869	Misc. entry points

Table 1. Reserved Area Addresses

Code(dec.)	Command	Address
128	END	1DAE
129	FOR	1CA1
130	RESET	0138
131	SET	0135
132	CLS	01C9
133	CMD	4173/56CB
134	RANDOM	01D3
135	NEXT	22B6
136	DATA	1F05
137	INPUT	219A
138	DIM	2608
139	READ	21EF
140	LET	1F21
141	GOTO	1EC2
142	RUN	1EA3
143	IF	2039
144	RESTORE	1D19
145	GOSUB	1EB1
146	RETURN	1EDE
147	REM	1F07
148	STOP	1DA9
149	ELSE	1F07
150	TRON	1DF7
151	TROFF	1DF8
152	DEFSTR	1E00
153	DEFINT	1E03
154	DEFSNG	1E06
155	DEFDBL	1E09
156	LINE	41A3/5786
157	EDIT	2E60
158	ERROR	1FF4
159	RESUME	1FAF
160	OUT	2AFB
161	ON	1F6C
162	OPEN	4179/6434
163	FIELD	417C/61AF
164	GET	417F/6355
165	PUT	4182/6354
166	CLOSE	4185/6173
167	LOAD	4188/606E
168	MERGE	418B/6109
169	NAME	418E/6544
170	KILL	4191/6521
171	LSET	4197/620B
172	RSET	419A/620A
173	SAVE	41A0/6148
174	SYSTEM	02B2
175	LPRINT	2067
176	DEF	415B/5655
177	POKE	2CB1
178	PRINT	206F
179	CONT	1DE4
180	LIST	2B2E
181	LLIST	2B29
182	DELETE	2BC6
183	AUTO	2008
184	CLEAR	1E7A
185	CLOAD	2C1F
186	CSAVE	2BF5
187	NEW	1B49
188	TAB(	2173
189	TO	?
190	FN	4155/558E
191	USING	2CBD
192	VARPTR	24EB
193	USR	27FE
194	ERL	24DC
195	ERR	24CE
196	STRING$	2A2F
197	INSTR	419D/588B
198	POINT	0132
199	TIME$	4176/5745
200	MEM	27C9
201	INKEY$	019D
202	THEN	1D78
203	NOT	25BF
204	STEP	2BO1
205	+	see Table 4
206	–	see Table 4
207	*	see Table 4
208	/	see Table 4
209	↑	?
210	AND	25FD
211	OR	25F7
212	<	?
213	=	?
214	>	?
215	SGN	098A
216	INT	0B37
217	ABS	0977
218	FRE	27D4
219	INP	2AEF
220	POS	27E5
221	SQR	13E7
222	RND	14C9
223	LOG	0809
224	EXP	1439
225	COS	1541
226	SIN	1547
227	TAN	15A8
228	ATN	15BD
229	PEEK	2CAA
230	CVI	4152/5F1B
231	CVS	4158/5655
232	CVD	415E/5F1E
233	EOF	4161/62B8
234	LOC	4164/62FE
235	LOF	4167/630F
236	MKI$	416A/5EFF
237	MKS$	416D/5F02
238	MKD$	4170/5F05
239	CINT	0A7F
240	CSNG	0AB1
241	CDBL	0ADB
242	FIX	0B26
243	LEN	2A03
244	STR$	2836
245	VAL	2AC5
246	ASC	2A0F
247	CHR$	2A1F
248	LEFT$	2A61
249	RIGHT$	2A91
250	MID$	2A9A

Table 2

error in the statement most recently evaluated.

40A0 This is the bottom of the reserved string storage. When you CLEAR 100, this number will be 100 less than the word at 40B1, which holds the highest address that BASIC will use.

40A2 BASIC has to know which line number it is on, so it stores that number here. If you're not running a program, the word here will be FFFF.

40A4 This holds the address where BASIC puts the first line of the program. If you want programs elsewhere in memory, put the address of that location in 40A4. Then POKE a 0 into the address stored in 40A4, minus one. Type NEW to reset the other pointers. If you don't POKE the 0, BASIC will probably give you a syntax error when you type NEW.

40AF When a function is used, it must know the type of its argument. A number is stored here to indicate entity. The codes are: 2 = integer, 3 = string, 4 = single precision, 8 = double precision. If you enter PRINT PEEK(16559), the computer should reply 2, because PEEK always returns an integer.

40E1-40E4 These locations are used with the AUTO command. When you AUTO, BASIC has to remember what line you're on, as well as the increment between lines. The word in 40E4 is the increment between lines, and 40E2 contains the last line number. 40E1 is a flag that tells BASIC it is in AUTO. When the byte is zero, everything is normal. When it's not, BASIC enters the AUTO mode. If a non-zero byte is poked into 40E1, as soon as BASIC resurfaces into the command mode it will start AUTOing. Turn off the BREAK key and set the increment to zero. Do this only if there's nothing in memory.

```
10 POKE 16396,23:POKE
16397,201 ' TURN OFF BREAK
20 POKE 16612,0:POKE 16613,0 '
SET INCR. TO 0
30 POKE 16609,1 ' TURN AUTO
ON (HEE-HEE-HEE)
```

Once you run this program, you'll have to turn off the system to do anything unless you have a monitor activated by a keypress.

40EA This contains the number of the last line with an error, and is used by ERL. If the error is committed in the command mode, it will be FFFF.

40F0 When you use an ON ERROR GOTO, BASIC gets the address of the specified line and stores it here.

40FF This points to the data for the next READ statement. When you RESTORE, it is reset to the first data statement. This can give you selection capabilities from two or more banks of data. You can position this pointer anywhere by POKEing, and treat a string variable as a data statement. When you change your program, the data address will change. This may be disastrous, so be careful.

Operation	Precision		
	Integer	Single	Double
+	BD7	716	C77
–	BC7	713	C70
*	BF2	847	DA1
/	2490	8A2	DE5

Table 3. Arithmetic Routines

4101-411A This table represents 26 bytes; DEFINT, DEFSTR, DEFSNG, and DEFDBL determine its contents. The byte at 4101 determines the implicit type of a variable starting with A; 4102, the type of a B variable. The numbers here are the same as those under address 40AF. If you want to make all E variables implicitly strings, add 5 to 4101 to get the address for the byte for E variables, and then POKE 16646, 3, because 3 specifies a string. Don't POKE anything but an 8, 4, 3, or 2 in this table. The result can be a crash.

411B This byte is the trace flag. If it is 0 (TROFF) then tracing is off. If it is not (TRON), then tracing is on.

Now you know something about the reserved areas. But BASIC has many other mysteries...

BASIC Commands

In ROM, starting at 1650,

there is a table of semi-recognizable words. These are BASIC's reserve words. The first letter of each word has 80 added to its ASCII code, letting BASIC tell one word from another without wasting memory. Table 2, at 1822, lists the routines for these commands. Every two-byte entry in the table is an address, corresponding with a reserved word, ending with the address for NEW. The next address would be 7979 which is wrong—that's in semi-high RAM, which a 4K system wouldn't have. My table ran out. I found addresses for TAB(by step-through disassembly. The only one I failed to find was the address for TO. I found the table for functions, SGN through MID$, at 1608. That finished up the list for everything but the infix operators (+, −, *, /, =, . . .).

The addresses for the arithmetic routines are listed in Table 3. They vary with the precision of the operands. The table is at 18AB. For these routines, integer operands are DE and HL (the register pairs); single precision numbers are stored in 4121, and register pairs BC and DE, where DE has the least significant bytes; double precision numbers are stored at 411D-4121 and 4127-412E.

The results of any of the operations or of any BASIC functions called are stored in an area of memory called the accumulator. For strings, integers, and single precision numbers, the accumulator is at 4121. For double precision numbers it is at 411D through 4124. When a function is called, it expects its argument to be the accumulator, and the type flag to be set accordingly.

Address Hex	Address Decimal	Function
41A6	16806	Called when error occurs
41A9	16809	USR function
41AC	16812	Called on BASIC reentry from 1A1C
41AF	16815	Called on the beginning of a line input
41B2	16818	Called after a line of BASIC text is encoded
41C4	16836	Called when keyboard is strobed for byte
41C7	16839	Called on INPUT# or a RUN command with something after it
41CA	16842	Called on a PRINT#
41DF	16863	Called when LIST is executed
41E2	16866	Called when SYSTEM is executed

Table 4. Miscellaneous ROM Exits

Disk Basic Functions

In the list of BASIC keywords in Table 2, the first number is the code for that keyword. When you type a line of program, or anything in the command mode, each keyword is replaced with a single byte. This process saves time and memory, and makes it easier for BASIC to interpret the program.

When the address is written as number/number, it is a Disk BASIC command. The first address is the point in RAM to which BASIC jumps when it finds the command. The second address is the end address of the command under Radio Shack's Disk BASICR for TRS-DOS version 2.3.

These commands are the ones that give you an L3 error in Level II. They were implemented to allow upward expandibility of BASIC. You can create your own routines and have them executed directly from BASIC by patching them into the table at the proper address. End the routines with a machine language RET instruction. HL will hold the pointer to the current character when your routine is called.

Suppose you want to call the routine NAME. The table entry for NAME is 418E. Start by loading the program and protecting memory, then poke JP (hex C3) into 418E. Put the address of your routine in 418F. You can now execute the routine directly by entering the NAME command. This allows you versatility in defining your set of commands for BASIC.

The Disk Basic Command Table occupies addresses 4152 through 41A5; the I/O buffer for Level II doesn't start until 41E6. Another table lies between them, consisting of miscellaneous places BASIC jumps to under certain conditions. Some interesting locations are listed in Table 4.

Whenever BASIC encounters an error, it calls 41A6 before it prints the error message. By using this table entry, Disk BASIC and Level III issue long error messages. They patch themselves into 41A6, look at the error code when called, and print the appropriate message.

Level III and Disk BASIC also use 41A9 to give you 10 USR calls. Disk BASIC uses 41C7 to allow sequential disk input. Using address 41B2, you can mask out a BASIC command.

Other ROM Routines

Each of the following routines evaluates an expression which has been compressed. It may contain functions, variables, constants and arithmetic operators, like any standard BASIC expression. On entry to the subroutine, HL points to the expression in RAM. On return, HL points to 1 + the last byte it evaluated. It stops evaluating at the first unrecognized byte, so always end your expression with a 0 or carriage return.

2337 Gets a general (string, integer, single, or double precision) parameter in the accumulator and sets the type flag (40AF, remember?) accordingly.

2B02 Gets a two byte integer in DE.

2B1C Gets a one byte integer in A.

There is another subroutine at 2857. It gives you the address of a line of BASIC program. When you call it, put the line number in DE. On return, DE will be unchanged. If the line does not exist, the C flag will be reset. BC will point to the line if it exists, and HL will point to the next line.■

This program takes the guesswork out of setting printer character width and line length.

EZ Wider

Milan D. Chepko, M.D.
119 Belleville Court
Thief River Falls, MN 56701

One of the problems with the Centronics 779 line printer from Radio Shack is that character width—and the number of characters per printed line—is determined by an adjustable potentiometer, instead of a step switch. As a result, changing character size can be a hit-or-miss affair.

The greatest problem is that if the print head reaches the end of the paper *before* it has printed all the characters in its buffer memory, they are lost. The next line is fed into the buffer.

This program generates a line of print containing a constant number of characters. While this standard line is sent repeatedly to the printer, you can adjust the control on the back of the printer until the line just fills the width of the paper. By selecting the number of characters contained in the line, you can set your printer for any size of character or line width.

This program should be compatible with any line printer, even those using step switches. You might want to use it to test your system to see that it is actually printing the size of characters you want. ■

```
10 '*** PROGRAM TO ADJUST PRINTER LINE-WIDTH
20 'BY MILAN D. CHEPKO, M.D.
30 'THIEF RIVER FALLS, MN
40 CLS:CLEAR 200
50 IF PEEK(14312)<>63 PRINT"ACTIVATE LINE PRINTER":GOTO
    40
60 PRINT"SELECT A LINE WIDTH FROM THE FOLLOWING TABLE:"
    :PRINT
70 PRINT"   1)   72 CHARACTERS/LINE
80 PRINT"   2)   80 CHARACTERS/LINE
90 PRINT"   3)   96 CHARACTERS/LINE
100 PRINT"   4)  120 CHARACTERS/LINE
110 PRINT"   5)  132 CHARACTERS/LINE
120 PRINT:INPUT X:IF X>5 GOTO40
130 CLS:PRINT"HOLD DOWN 'BREAK' KEY TO TERMINATE TEST..

140 ON X GOTO150,160,170,180,190
150 LPRINT"<  72 CHARACTERS/LINE ";STRING$(49,":");">":
    GOTO150
160 LPRINT"<  80 CHARACTERS/LINE ";STRING$(57,":");">":
    GOTO160
170 LPRINT"<  96 CHARACTERS/LINE ";STRING$(73,":");">":
    GOTO170
180 LPRINT"< 120 CHARACTERS/LINE ";STRING$(97,":");">":
    GOTO180
190 LPRINT"< 132 CHARACTERS/LINE ";STRING$(109,":");">"
    :GOTO190
```

Program Listing

✓ Reader Service—see page 290

Dear Radio Shack. . .

OPINION = PEEK (Mail)

Jim Glosser
152 Clover Ave.
Marion, OH 43302

Dear Radio Shack,

Let me introduce myself. I'm the owner of a Model I Level II 48K TRS-80, with two disks and a printer. I'm also one of those people who make harsh comments about Radio Shack's attitude toward the owners of their computers.

Jon Shirley and Ed Juge, both of Radio Shack, have mentioned the Shack's "bad press" and have expressed a lack of understanding for the hard feelings expressed in print. I will attempt to explain those feelings.

As Wayne Green has pointed out in his columns, owners of Radio Shack computer systems find themselves in a love-hate relationship. Radio Shack was the first to make computing power really available to the ordinary guy on the street.

Other microcomputers were marketed before the TRS-80, but Radio Shack put together the right combination of consumer marketing and after sale support to make the micro really practical for me and hundreds of thousands of others.

Believe me, I really studied before I bought the TRS-80. Your ads got my attention. Then I saw a demonstration in my local store. The computer catalog was my next step. I read and re-read the computer catalog many times before making my next move. I finally felt that you had put together an equipment combination that could offer me practical value.

After deciding your equipment was financially practical, I bought the Level I manual and studied it from front to back, just as if I had already bought a computer. I then bought the Level II manual and studied it.

By this time, I was convinced that your equipment could be put to use in personal and work-related applications. I finally bought the Level II computer and started through the two manuals again, but this time it was for real. I thought that if I successfully mastered program techniques, I would be able to complete the tasks I dreamed of before buying the computer. I thought it would be just me and the computer.

Imagine my surprise when I met another fellow at work who also had a Shack computer. He introduced me to *Kilobaud Microcomputing* magazine and the newsletter put out by H&E Computronics. I found that I wasn't alone and that we could communicate.

Then came *80-US Journal*, *80 Microcomputing* magazine, and the Alternate Source newsletter. It became apparent that others were trying to make practical applications out of their dreams of using computer power.

I found that my problems of programming technique and operating efficiency were being encountered by many others.

Did I try any Radio Shack software? Yes; Editor Assembler, T-Bug, and In Memory Information. Were they good? Well, the Editor Assembler certainly was. In Memory Information was only so-so for a fast startup of a very limited requirement. T-Bug only convinced me to immediately buy RSM from Small System Software and Monitor-3 by Dr. Hubert Howe.

My main application is to classify information about materials used in an industrial maintenance operation, however, it has required programming which is thus far unique among other applications I've seen.

Soon after buying the Level II, I bought the expansion interface and went to a full 48K to get the necessary memory for information storage during processing. I bought the disk manual soon after buying the computer, and I went ahead and bought my first disk unit approximately 15 months later. By this time, I had spent at least 1500 hours with the computer or studying various reference materials.

The Radio Shack computer system's capabilities far outstrip my wildest imaginings and most likely go way beyond anything that Radio Shack officials had in mind when computer marketing began. I fully expect that in 1981 we will see a full network-type data base management system released by the Shack to revolutionize information management for small operators like me.

There are two sources of strength for the Shack computer. The first is from Radio Shack. You've made available a versatile computing system at a reasonable cost. I thank you many times for doing that. Without the hardware system, microcomputing for the masses would be a pipe dream.

However, a good system with large numbers in service doesn't make you better than any other system on the market.

There is, however, a second source of strength for the Shack. This is the 200,000 plus users trying to accomplish something worthwhile. Those 200,000 are both a market for professional programming organizations like Apparat, and a source of new applications technology which will transform the Shack computer from a sophisticated electronic gadget into a useful tool.

No person or organization can possible imagine all the incredible uses for the computer being successfully accomplished by the sum of interested users.

It's not just a matter of marketing software. Shack users are anxious to promote applications of this computer due to the snowballing of so many published sources for information, and so many professionals working exclusively for the Shack market.

Why have I gotten angry with Radio Shack? Because Racet provided me with the means to

make sorting an efficient operation. Because Roger Fuller provided me with the most insight into the workings of Level II BASIC. Because Allan Moluf provided me with the most information about storage of pointers and markers used in the execution of BASIC programs. These sources, among others, have helped me improve the efficiency of my applications programs many times.

Why couldn't I get that help from Radio Shack?

When I started using disks, H.C. Pennington published his book about the working of disks. Before it came out, I treated the disks like holy objects, out of fear which came from lack of understanding. Thanks to Pennington and Apparat's Superzap, I now understand how they work. Disk drives are now just another tool which work for me and improve the efficiency of my efforts beyond the tape drives. Why didn't Radio Shack give me the information necessary to really use the disk drives?

TRSDOS is great. I really admire Randy Cook for his ability to write such a comprehensive piece of machine language programming. But TRSDOS has not grown. It's essentially the same operating system it was when first released. NEWDOS21 and NEWDOS80 represent what should be natural extensions of growth to the operating system. Who made them available? Apparat did, not Radio Shack. I bought TRSDOS so that Radio Shack would have its deserved royalties. But I never used it!

I was willing to pay the price for the enhanced NEWDOS2.1, so I bought it. And when NEWDOS80 came out, I paid the price to upgrade to it and its significant enhancements. But why doesn't Radio Shack acknowledge and market such systems as a supplement to TRSDOS for those who feel they need it and are willing to pay?

Sure, I know you released two upgrades free of charge. For the most part, they only correct problems in previous issues. Apparat's not perfect, but it sends me documented corrections to its operating system soon after learning about errors. It doesn't make me wait several months for them to accumulate for a new release.

So what's ahead for the Shack? Well, if Radio Shack keeps making the Model I available, the applications of that computer will continue to become available at accelerating rates.

What's the most important thing Radio Shack can do to help me obtain maximum potential from its computer? Keep it compatible. I realize that you have the right to make changes. But a small change, such as the reported changes in ROM when the two-chip set was introduced, concerns me. It may put me back as far as knowledge about the system is concerned.

Do I want to upgrade to Model II? No! Model I is where the vast majority of the support is. I want as many sources of support as possible.

Will I upgrade to Model III? (Having a single-unit computer without all the connection cables would add much to the reliability of the system.) As it is now, I'm making do quite well with homebrew modifications. All my cables are connected to the PC boards with gold plated connectors.

But will the Model III put me back to square one? Statements like 'Most Model I BASIC programs will be compatible' and '80 percent compatible' lead me to believe that the Model III will have to be considered a new computer. This is because many applications already require interfacing with the BASIC interpreter from machine language subroutines in order to have efficiency and effectiveness. If changes can't be documented, I expect to stay with the Model I or even the copycat of the Model I being marketed with Level II ROM.

When making decisions about the future of the Shack computer, please remember the ramifications of having several hundred thousand identical units in service, and what consistency means to the user and even you, the marketer of such versatile systems.■

Always,
Jim Glosser

✓ Reader Service—see page 290

Backing up disk programs.

Disk to Tape

Robert J. Hocking
1126 Karney Ave.
Palm Bay, FL 32905

Like most owners of the TRS-80, I put my cassette recorder into a closet and forgot it when I purchased my first disk drive. I put all of my programs on the new disk and soon had about six diskettes full.

All went well until one of my diskettes wouldn't boot. Try as I might, I couldn't recover the lost programs.

I went back and read through my DOS manual hoping for enlightenment and found that I should have made backup copies of all of my diskettes. So, I bought another box of diskettes and copied everything. All was fine—no more losses. But as my software library grew, so did my backup library, 'til I had about 30 diskettes with programs and 30 with backup programs.

Income tax time came, and while figuring out what paltry sum Uncle Sam was going to let me keep, I realized that I had approximately $120.00 in working diskettes and approximately $120.00 in backup. This didn't seem right to me. After all, backups are like insurance, and I certainly wouldn't spend $5000 to insure a $5000 car. I forgot about my income tax.

I started to think about cheap program insurance, when I spied the power cord of my stashed cassette recorded trailing out of the closet. I still had 50 tapes that I no longer used after my disk arrived, and cassette tapes are relatively cheap.

I saved all my BASIC programs to cassette, and my pile of backup diskettes dwindled until the only programs left to backup were machine code programs. This time I hit a brick wall. There I sat with RSM2D, ELECTRIC PENCIL, NEWDOS utilities and FORTRAN programs all on disks, and no way to get them onto cassettes.

Radio Shack provides Tapedisk to get machine code programs from tape to disk, but not the reverse.

I took a look at T-BUG. It would create machine code tapes, but it resides in low RAM around 4000 hex as do most machine code programs. I took a look at RSM2D, from Apparat, Inc. It has three versions, one at the end of 16K, one at the end of 32K, and one at the end of 48K. I could therefore pick the version I needed.

In order to create a save tape with RSM2D I needed the following RAM addresses—the first location used, the last location used and the starting address. NEWDOS+ contains a program called LMOFFSET, which will load a machine code program from tape or disk, and display the required addresses on the screen.

Armed with this knowledge, I tried saving a program. I found that it loaded from 5500 hex to 6A00 hex with 5500 hex as an entry point. It appeared to save correctly, but when I tried to load it back, my TRS-80 went off into the woods and wouldn't return without a reset. I tried again and discovered that LMOFFSET was trying to tell me what the problem was: My program loaded into an area used by DOS and was being overwritten before I could save it.

It took a few minutes before I realized I could correct the problem right then. LMOFFSET will relocate a program. When it asked for the new start address, I answered 8000 hex. It then asked if the appendage should be suppressed. This is a short sequence of code that will move the program back to its original address when executed. Since I was going to move it with RSM2D I answered, yes, suppress it. LMOFFSET told me that the new program was moved to reside from 8000 hex to 9500 hex. I wrote that down and had it store the relocated program under a dummy file name. I used its original name with the extension /REL, for relocated. I then went back to DOS. I loaded the relocated program with the load command and then called RSM2D. Using the addresses from LMOFFSET I typed:

```
P 5500 9500 5500 <ENTER>
```

This moved the program back to its original location. Then I loaded a tape and typed:

```
P 5500 6A00 5500 <ENTER>
```

This caused RSM2D to create the machine code tape, and after checking the program, I found that all was well.

I have used this procedure on many programs and all those have worked well.

My cassette recorder is now serving a useful function, and my backup library is down to six diskettes which are used to backup various DOS systems.

Hey, can any of you readers suggest software to put on 24 blank diskettes? ■

Stop plugging away at CTR control. . . just flick your switch.

Switched-on CLOAD

John E. Bickel WA3VRI
III Strasburg Pike
Lancaster, PA 17602

Here is a simple modification for Radio Shack's CTR-80 cassette recorder allowing the user to play a tape without unplugging the cables of the TRS-80's keyboard. It all happens at the flick of a switch installed in the CTR-80 unit.

No other parts are required.

To understand this, consult Fig. 1 or the schematic with the CTR-80. Both J3 and J4 are closed circuit jacks. In other words, when a plug is inserted, the normal circuit is opened and a new circuit connected to the plug is substituted.

In the case of the ear jack (J3), inserting the plug disables the cassette unit's built-in speaker and substitutes the TRS-80 audio input circuitry which "listens" to the recorded data. Plugging into the mic rem jack (J4), takes cassette motor control away from the play push-button and transfers it to the relay inside the TRS-80. By placing a switch across the jacks, the user restores the original audio and motor control circuits. This bypasses the fact that the plugs are inserted.

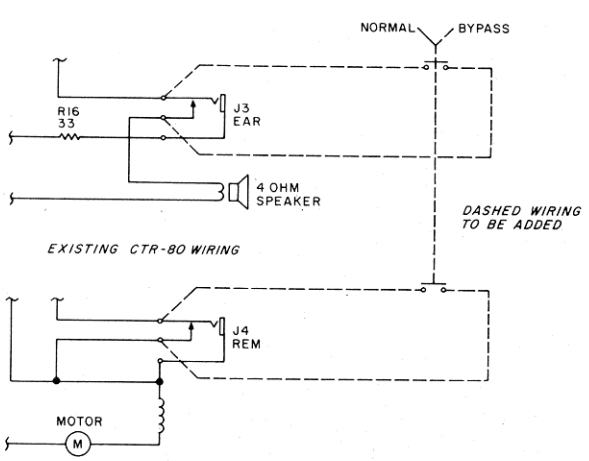


Fig. 1. CTR Modification

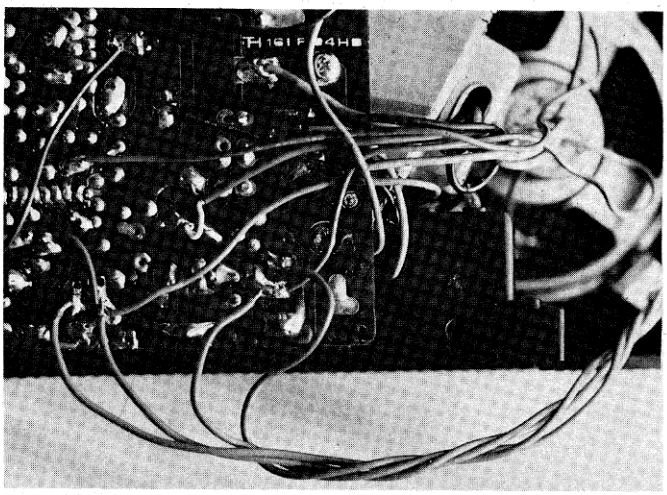


Photo 1. PC board attach points for switch wiring

Construction

The modification calls for installation of a double pole switch. I chose the location in the photo because it was easy to drill indentations in the CTR-80 case leaving plenty of space directly underneath for the switch base and wiring. I used Radio Shack's 275-663 DPDT toggle switch.

Those who object to a toggle sticking up out of the cassette unit may wish to take the extra time to fit a slide switch. With tools and material in hand, it will take about an hour to finish the job.

Begin by removing the screws holding the CTR-80 case halves together. Two are on the bottom near the handle end; one is at the bottom of the battery compartment. Separate the case halves, being careful not to disconnect the wiring connecting the battery holder. The PC board which you have exposed is held together by two screws and the locking rings on the four side-mounted jacks. Removing the screws is easy. To remove the locking rings you will need a spanner wrench, or better yet, a paper clip will do the job. Bend the paper clip into a "U" shape which will engage the slots in the locking rings. Use long-nose pliers to turn the whole works. After all the rings are off, the PC board is free. Wiggle it back and up, away from the unit until you can get at both sides.

Preparing the Switch

Prepare your switch by soldering four wires to it—long enough to reach from the switch

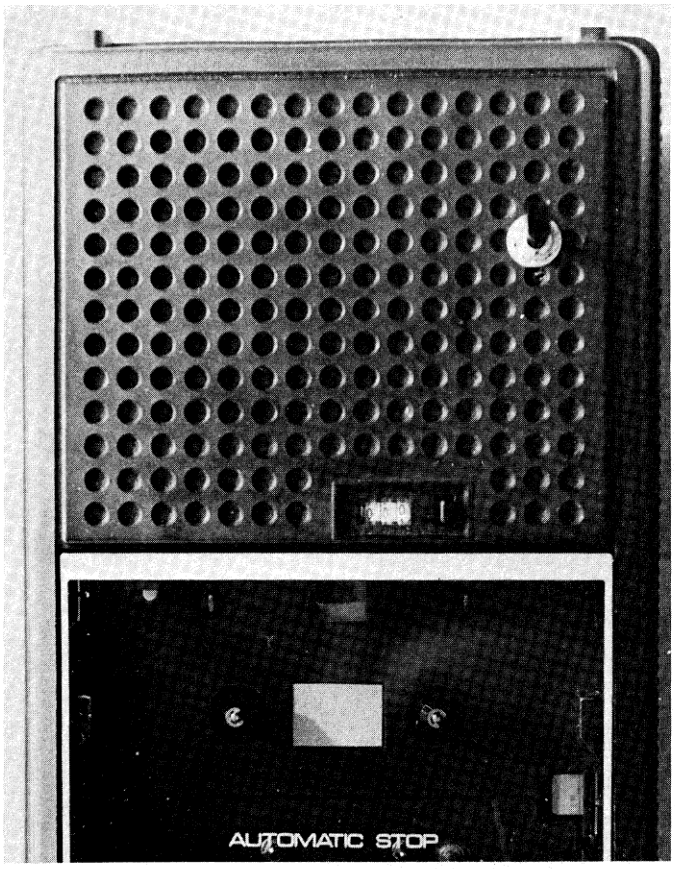


Photo 2. Location of switch in CTR-80 case

location to the foil side of the PC board, opposite the EAR and REM jacks. Tin the bare wire ends. Drill a hole in the CTR-80 case to accept the switch. Make sure that nothing interferes with the switch on re-assembly.

Find the points on the foil pattern which correspond to the jacks on the opposite side of the PC board. Carefully solder the switch leads to the foil side, in accordance with Figs. 1 and 2. Check to see that none of the CTR-80 wiring popped loose during soldering, and that switch leads don't touch anything but the intended points on the board. Establish which direction you want the switch to throw and mount it accordingly. Reassemble the entire unit and watch out for pinched wiring.

Use a dummy plug to verify that your CTR-80 is operable before connecting it to the keyboard. With the plug in the mic rem jack, and then the new switch to start the recorder. Transfer the dummy plug to the ear jack. Next, turn the CTR-80 speaker on, with the switch in

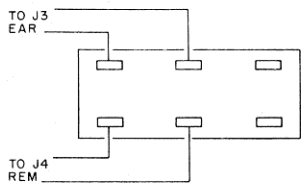


Fig. 2. DPDT Switch

the position which started the recorder in the first test. If all is well, reconnect to the TRS-80.

Some experiments now will determine how the switch can best help your operation. Since I installed mine, I always use it to cue up a tape before CLOADing. This avoids waiting an unknown length of time for the bit stream to start and asterisks to appear on the CRT. Within a second or two, I can see if I'm getting a good CLOAD.

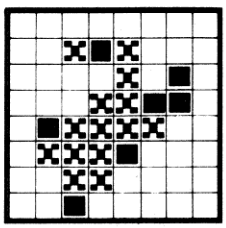
Being able to hear the tape start allows me to set the proper volume setting. I've had no trouble CLOADing since I began to use the switch to cue tapes.

The switch also makes finding the end of previously recorded data simple. ■

Proven Favorites... These 10 Popular Games Offer Exciting New Challenges

Here are 10 games that have never lost popularity. They've been played in many forms for many years. And now they're among the favorite software packages in the computerized home-entertainment field.

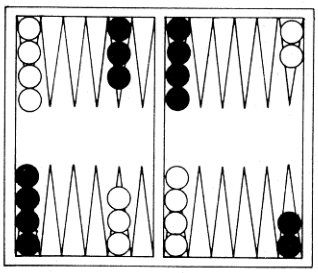
Don't miss out on the excitement, challenge and downright fun of these programs. Take the word of thousands of satisfied players. They're still tops. Order yours today and play your kind of game.



TRS-80*

OTHELLO

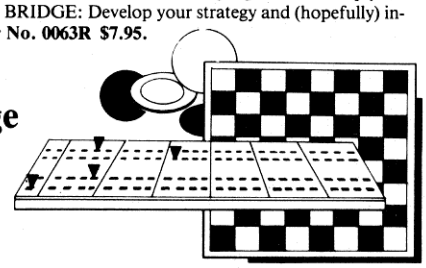
OTHELLO—Pit your strategic powers against a merciless, computerized opponent. You play on a board of 64 squares. When you capture your opponent's game disks (by bracketing them with your own disks), they immediately change sides to become members of your set. You can capture several game pieces at one time as long as they are in-line horizontally, vertically or diagonally. Major reversals of score are commonplace as whole blocks of men change side in an instant. **(T1) Order No. 0046R $9.95**



Beginner's Backgammon and Keno

BEGINNER'S BACKGAMMON/KENO—Why sit alone when you can play these fascinating games: ● BACKGAMMON: Play against the computer in a game that's sure to sharpen your skills; ● KENO: Enjoy this popular Las Vegas gambling game—guess the right numbers and win big! **(T1) Order No. 0004R $7.95.**

CARDS—A one-player package to let you play, with your computer, these famous games: ● DRAW AND STUD POKER: These programs will keep your game sharp; ● NO-TRUMP BRIDGE: Develop your strategy and (hopefully) increase your skill. **(T1) Order No. 0063R $7.95.**



Your Cribbage and Checkers Partner

YOUR CRIBBAGE AND CHECKERS PARTNER— ● CRIBBAGE is a two-person game that you are sure to enjoy. This is NOT a tutorial—it is a game worthy adversary. ● CHECKERS: An old favorite which follows international rules, including multiple jumps. **(T1) Order No. 0068R $9.95.**

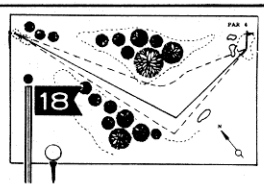
(T1)=TRS-80 Model 1, Level II, 16K RAM

CHESSMATE-80—This versatile chess opponent gives you a choice of ten levels of play, from the "blitz" level (the computer has 3 seconds to move) to the infinity level (where the computer will consider every possible move—which could take years). This machine-language program is a conservative player and follows all the rules of international play. CHESSMATE-80 can teach you how to move and allow you to set up the board and play end games or special problems. CHESSMATE-80 battled Sargon II to a draw at two minutes a move and beat Microchess 1.5 in six moves. **(T1) Order No. 0057R $19.95.**

BOWLING—Let your TRS-80 set up the pins and keep score. One player can pick up spares and get strikes. **(T1) Order No. 0033R $7.95.**

Apple**

GOLF



GOLF—Without leaving the comfort of your living room you can enjoy a challenging 18 holes of golf with a complete choice of clubs and shooting angles. Now, you need never cancel your game because of rain. One or two players can enjoy this game on your Apple II with Applesoft and 32K. **Order No. 0018A $7.95**

PET***

CASINO I—Gamblers, gather round for: ● BLACKJACK—a fun-to-play and tutorial version that allows you to play every combination that they play at the MGM Grand Hotel in Las Vegas; ● ROULETTE—also a game *and* a tutorial which could enable you to develop the skill to make a fortune at the tables. **Order No. 0014P $7.95**

CASINO II—This is not simply a craps game, it is a tutorial program that will teach you the odds on every type of bet, so you can steer clear of the sucker bets that impoverish the unwary. Using the exact rules used at the MGM Grand Hotel in Las Vegas, you'll play under actual gambling house conditions. Get the edge you need to "get lucky." **Order No. 0015P $7.95**

CHECKERS & BACCARAT—Using International Rules, you'll play ● CHECKERS with your computer and the computer will keep score, time the moves and even tell you when to jump (it'll be so busy, you may have an edge!); ● BACCARAT—gives you the choice of playing Las Vegas or Blackjack style and the computer will deal the cards, figure the payoff and keep track of your bankroll. **Order No. 0022P $7.95**

ORDER YOUR INSTANT SOFTWARE TODAY!

Just Call Toll-Free 1-800-258-5473

Instant Software™

PETERBOROUGH, N.H. 03458 603-924-7296

* A trademark of Tandy Corporation
** A trademark of Apple Computer Inc.
*** A trademark of Commodore Business Machines Inc.

Ask for Instant Software at a computer store near you.

Alabama

Anderson Computers
3156 University Dr., Huntsville

Computerland of Huntsville
3020 University Dr., Huntsville

Olensky Bros.
3763 Airport Blvd., Mobile

Arizona

Professional Data Systems
4506-A N. 16th St., Phoenix

Millets TV & Radio
621 East Broadway, Mesa

Software Station
1419 E. Commodore Pl., Tempe

California

AMCO Elect. Supply
635 E. Arrow Hwy., Azusa

Byte Industries
3501 Arden Rd., Hayward

Byte Shop
8038 Clairmont Mesa Blvd., San Diego

Byte Shop
123 E. Yorba Linda, Placentia

Byte Shop of Mt. View
1415 West El Camino Real, Mt. View

Byte Shop of Sacramento
6041 Greenback Ln., Citrus Heights

Capital Computer Systems
3396 El Camino Ave., Sacramento

Coast Electronics
6905 Alcameno Real, Altascadero

Coast Electronics
2360 N. Main St., Morro Bay

Computers Made Easy
819 East Ave. Q-9, Palmdale

Comp:.ter Store of San Leandro
701 MacArthur Blvd., San Leandro

Computer World
6791 Westminster Ave., Westminster

Computerland
16720 S. Hawthorne, Lawndale

Computerland of W. LA
6840 La Cienega Blvd., Inglewood

Coast Electronics
3118 No. Main St., Morro Bay

Computerland
24001 via Fabricante No 904, Mission Viejo

Computer Mart of California
315 Diamond Bar Blvd., Diamond Bar

Electronic Systems
4883 Tonino, San Jose

Hobbi-tronics
1378 So. Bascom Ave., San Jose

Hobby World
19511 Business Ctr. Dr., Unit 6, Northridge

Huntington Computing
2020 Charles St., Corcoran

Jade Computer Products
4901 W. Rosecrans, Hawthorne

Malibu Microcomputing
23910A Deville Way, Malibu

Marfam Co.
6351 Almaden Rd., San Jose

Opamp/Technical Books
1033 N. Sycamore Ave., Los Angeles

PC Computers
10166 San Pablo Ave., El Cerrito

Q.I. Computers, Inc.
15818 Hawthorne Blvd., Lawndale

Radio Shack Dealer
8250 Mira Mesa Blvd., San Diego

Radio Shack Dealer
50 N. Cabrillo Hwy., Half Moon Bay

Santa Rosa Computer Center
604 7th St., Santa Rosa

Silver Spur Elect. Comm.
3873 Unit F, Schaefer Ave., Chino

The Computer Store
820 Broadway, Santa Monica

Colorado

Apparat Inc.
4401 South Tamarac Pky., Denver

Colorado Computer Systems
311 W. 74th Ave., Westminster

Computerland of North Denver
8749 Wadsworth Blvd., Arvada

Computer Shack
1635 South Prairie, Pueblo

Software Gourmet
1111 S. Pearl St., Denver

The Computer Store
2300 Welton St., Denver

Connecticut

American Business Computers
454 Thames St., Groton

Computerlab
130 Jefferson, New London

Computerland
1700 Post Rd., Fairfield

Computerland
60 Skiff St., Hamden

Computer Works
1439 Post Rd. E., Liberty Plaza, Westport

Diversified Electronics
2 Amity Rd., New Haven

Instructional Systems Computers
807 Hartford Rd., Manchester

Technology Systems
208 Greenwood Ave., Bethel

D.C.

The Program Store
4200 Wisconsin Ave., N.W., Washington, D.C.

Florida

Al Personal Computer
178 Oxford Rd., Fern Park

AMF Microcomputer Center
11158 N. 30th St., Tampa

Computer Junction
5450 So. State Rd. 7, Ft. Lauderdale

Computerland
7374 S. Tamiami Trail, Sarasota

Computerland of Ft. Lauderdale
3963 N. Federal Hwy., Ft. Lauderdale

Computerland of Jacksonville
2777-6 University Blvd. W. Jacksonville

Computerland of Tampa
1520 E. Fowler Ave., Tampa

Computerland of West Palm Beach
4275 Okeechobee Blvd., West Palm Beach

Computer Shack
3336 Beach Blvd., Jacksonville

Computer System Resources Inc.
3222 S.W. 35th Blvd., Gainesville

Heath Kit Electronic
4705 W. 16th Ave. Center, Hialeah

HIS Computermation
1295 Cypress Ave., Melbourne

MicroComputer Systems, Inc.
144 S. Dale Mabry Hwy., Tampa

Williams Radio & TV Inc.
2062 Liberty St., Jacksonville

Your Basic Computer Store
2729 So. US 1, Suite 11, Fort Pierce

Georgia

Atlanta Computer Mart
5091 Buford Hwy., Atlanta

Computerland of Atlanta
2423 Cobb Parkway, Smyrna

Micro Computer Systems
3104 E. Shadowlawn N.E., Atlanta

Hawaii

Computerland of Hawaii
567 N. Federal Hwy., Honolulu

Radio Shack Assoc. Store
1712 S. King St., Honolulu

Idaho

Electronic Specialists
8411 Fairview Ave., Boise

Illinois

Computerland
4507 North Sterling, Peoria

Computerland
9511 N. Milwaukee Ave., Niles

Computer Station
3659 Nameoki Rd., Granite City

Garcia & Associates
203 No. Wabash Ave., Suite 1510, Chicago

Midwest Micro Computers, Inc.
708 S. Main St., Lombard

Indiana

Computer Center of South Bend
51591 US 31 North, South Bend

Data Domain
221 W. Dodds, Bloomington

Fall Creek Electronics Store
732 Center St., Pendleton

Iowa

Memory Bank
1721 Grant St., Bettenborf

Kansas

Central Kansas Computers
6 S. Broadway, Herington

Maine

Maine Computronics
Intown Plaza, Bangor

Mid Maine Computer Co.
158 Turner St., Auburn

Radio Shack
315 Main Mall Rd., So. Portland

Maryland

Computer Age
9433 Georgia Ave., Silver Springs

Jack Fives Electronics
4608 Debilen Circle, Pikesville

The Comm Center
9624 Ft. Meade Rd., Laurel

Massachusetts

ComputerCity
175 Main St., Charlestown

ComputerCity
50 Worcester Rd., Framingham

Computer Packages Unlimited
342 Boston Turnpike, Shrewsbury

Land of Electronics
1127 Western Ave., Lynn

Lighthouse Computer Software
14 Fall River Ave., Rehobath

Mark Gordon Computers
15 Kenwood St., Cambridge

Small Business System Group
Main St., Dunstable

The Computer Store
120 Cambridge St., Burlington

Tufts Radio & Electronics
206 Mystic Ave., Medford

Michigan

Computer Center
28251 Ford Rd., Garden City

Computer Connections
38437 Grand River, Farmington Hills

Computerland of Grand Rapids
2927 28th St. S.E., Kentwood

Computerland of Southfield
29673 Northwestern Hwy., Southfield

Computer Mart
560 W. 14 Mile Rd., Clawson

Computer Room
455 E. Michigan Ave., Kalamazoo

Computronix Corp.
423 S. Saginaw Rd., Midland

Hobby House
1035 W. Territorial Rd., Battle Creek

Main Systems Inc.
1161 No. Ballenger Hwy., Flint

The Alternate Source
1806 Ada, Lansing

The Eight Bit Corner
722 Evanston Ave., Muskegon

TRI Country Electronics & Sound Center
1537 North Leroy, Fenton

Ye Olde Teacher Shoppe
1823 Witmyre St., Ypsilanti

Minnesota

Computerland of Hopkins
11319 Hwy F., Hopkins

Digital Den
Burnsville Center

Minnesota Software Inc.
5422 Fisher St., White Bear Lake

The Code Room
18216 Ginavale Lane, Eden Prairie

Zim Computers
5717 Xerxes Ave., N. Brooklin Center

Mississippi

Dyer's, Inc.
200 E. Main St., West Point

Softwarehouse
816 Foley St., Jackson

Missouri

Century Next Computers
1001 E. Walnut, Columbia

Computer Center
212 W. 4th St., Joplin

Comp-U-Trs Software Center
51 Florissant Oaks Shopping Center, Florissant

Software Shack
16501 Greenwald Court, Belton

Montana

Intermountain Computer
529 So. 9th St., Livingston

Personal Computer
121 Red Oak Dr., Carl Junction

The Computer Store
1216 16th St. W. #35, Billings

Nebraska

Computerland of Omaha
11031 Elm St., Omaha

Midwest Computer Co. Inc.
8625 I St., Omaha

Midwest Computer Co. Inc.
4442 S. 84th St., Omaha

Midwest Computer Co. Inc.
4403 S. 87th St., Omaha

Scottsbluff Typewriters Inc.
1824 Broadway, Scottsbluff

Nevada

Century 23
4566 Spring Mountain Rd., Las Vegas

Hurley Electronics
1112 S. Casino Center, Las Vegas

New Hampshire

Bitsnbytes Computer Center
568 Pleasant St., Concord

ComputerCity
1525 S. Willow, Manchester

Paul's TV
Main St., Fremont

Portsmouth Computer Center
31 Raynes Ave., Portsmouth

Radio Shack Assoc. Store
Fairbanks Plaza, Keene

New Jersey

Abe's TV Sales & Service
College Town Shopping Center, Glassboro

Computer Corner of NJ
439 Rte. #23, Pompton Plains

Computer Encounter
2 Nassau St., Princeton

Computerland
35 Plaza Rte. #4, W. Paramus

Computer Mart of NJ
501 Rte. 27, Iselin

Crowley's
Rd. #3, Whitehouse Station

Dave's Electronics
Pennsville Shopping Ctr., Pennsville

GHB Enterprises Inc.
Rte. 38, Rudderaw Ave., Mapleshade

Lashen Electronics Inc.
21 Broadway, Denville

Personal Computing Inc.
51 Central Sq., Linwood

Radio Shack/J&J Electronic
Mansfield Shopping Ctr.
Rt. 57 Allen Rd., Hackettstown

Radio Shack Assoc. Store
Moorestown Mall, Moorestown

The Bargain Brothers
Glen Roc Shopping Center
216 Scotch Road, Trenton

The Computer Emporium
Bldg. 103, Avenues of Commerce
2428 Rte. 38, Cherry Hill

New Mexico

Autel Electronics Co.
146 Wisconsin NE, Albuquerque

J&W Enterprises
121 Purdue, Clovis

South West Computer Center
121 Wyatt Drive, Suite 7, Las Cruces

Thomas E. Carr Jeweler
1300A Tenth St., Alamogordo

New York

Aristo Craft
314 Fifth Ave., NYC

Berliner Computer Center
102 Jericho Turnpk, New Hyde Park

Computer Corner
200 Hamilton Ave., White Plains

Computer Era Corp.
1570 3rd Ave., New York

Computer Factory
485 Lexington Ave., NYC

Computerland of Nassau
79 Westbury Ave., Carle Place

Computerland of New York City
58 W. 44th St., New York

Computer Resources
5560 Main St., Williamsville

Comtek Electronics, Inc.
2666 Coney Island Ave., Brooklyn

Comtek Electronics, Inc.
Staten Island Mall
Store 220A, Staten Island

Digibyte Systems Corp.
31E. 31st St., New York

80-Microcomputer Services
118 Masten Ave., Cohoes

Future Visions Computer Store
70 Broad Hallow Rd., Melville

Home Computer Center
671 Monroe Ave., Rochester

Mr. Computer
Imp. Plaza, Rte. 9, Wappingers Falls

Softron Systems
308 Columbia Turnpike, Rensselaer

The Computer Tree Inc.
409 Hooper Rd., Endwell

Upstate Computer Shop
629 French Rd., Campus Plaza, New Hartford

A World of Computers
519 Boston Post Rd., Port Chester

North Carolina

Sound Mill
Slocum Shopping Ctr., Havelock

TD's Record Shop
27 E. Main St., Sylva

Ohio

Altair Business Systems, Inc.
5252 North Dixie Dr., Dayton

Astro Video Electronics
504 E. Main St., Lancaster

Cincinnati Computer Store
4816 Interstate Dr., Cincinnati

Computerland
4579 Great Northern Blvd.,
N. Olmstead

Computerland
6429 Busch Blvd., Columbus

Computerland
1288 Som Rd., Mayfield Heights

Computerland
2000 North Rd. SE, Warren

Computer Store of Toledo
18 Hillwyck Dr., Toledo

H. Gabriel & Co.
1469 Rosena Ave., Madison

Microcomputer Center
7900 Paragon Rd., Dayton

Micro-Mini Computer World
74 Robinwood, Columbus

21st Century Shop
16 Convention Way, Cincinnati

Universal Amateur Radio, Inc.
1280 Aida Dr., Columbus

Oklahoma

Sounds, Etc.
Hyw. 33, Watonga

Vern Street Products
114 W. Taft St., Sapulpa

Oregon

Computerland of Portland
12020 S.W. Main St., Tigard

Computer Pathways Unlimited, Inc.
2151 Davcor St. S.E., Salem

TRS-80 Products Ltd.
3520 S.E. Vineyard Rd., Portland

Pennsylvania

Artco Elect.
302 Wyoming Ave., Kingston

Artco Elect.
Back Mountain Shopping Center, Shavertown

Audio Mart
518 Fifth Ave., New Brighton

Computer Workshoppe
3848 William Penn Hwy, Monroeville

Computerland of Harrisburg
4644 Carlisle Pike, Mechanicsburg

Computerland of Pittsburgh
5499 William Flynn Hwy., Gibsonia

Erie Computer Co.
2127 West 8th St., Erie

J + E Communications
617 3rd Ave., Duncansville

Mighty Byte Computer Center
537 Easton Rd., Horsham

Personal Computer Corp.
24-26 West Lancaster Ave., Paoli

Personal Computer Corp.
Frazer Mall, Lancaster Ave., Frazer

Pittsburgh Computer Store
2945 Banksville Rd., Pittsburgh

Rhode Island

Computer City
165 Angell St., Providence

South Carolina

Omni Electronics
Northwoods Mall, Charleston

Tennessee

ACS
1100 8th Ave. So., Nashville

Ace Mini Systems
1598 Tradewinds North, Clarksville

Computerlab
671 S. Menden Hall Rd., Memphis

Computer World
625 Main St., Nashville

H & H Electronics Inc.
509 N. Jackson St., Tullahoma

Texas

RL Coles Communications Center
3907 Fredricksburg, Rd., San Antonio

Computer Hobby Center
813½ South Lamar Blvd., Austin

Computer Port
2142 N. Collins, Arlington

Gateway Electronics
8932 Clarkcrest, Houston

Houston Computer Tech
5313 Bissonet, Bellaire

K.A. Elect.
9090 Stemmons Frwy., Dallas

Pan American Elect. Inc.
1117 Conway, Mission

Radio Shack Dealer
21969 Katy Freeway, Houston

The Compu Shop
6353 Camp Bowie Blvd., Ft. Worth

Waghalter Books Inc.
3 Greenway Plaza E., Houston

Utah

DC Computer Co.
1911 West 70 South, Provo

Quality Technology
470 E. 2nd So., Salt Lake City

Virginia

Computer Works
Rte. 6, Box 65A, Harrisonburg

Home Computer Center
2927 Virginia Beach Blvd.
Virginia Beach

Southside Radio Comm.
135 Pickwick Ave., Colonial Heights

Washington

American Mercantile Co. Inc.
2418 1st Ave. S., Seattle

Byte Shop of Bellevue
14701 N.E. 20th St., Bellevue

Computer Connection Inc.
3100 NW Bucklin Hill Rd., Silverdale

Computer Products Division
10316 Hwy 99 S., Everett

Computerland of South King Co.
1500 S. 336 St., Suite 12, Federal Way

Personal Computers
S 104 Freva, Spokane

Ye Old Computer Shop
1301 G. Washington, Richland

West Virginia

The Computer Corner Inc.
22 Beechurst Ave., Morgantown

Wisconsin

Byte Shop Of Milwaukee
6019 West Layton Ave., Greenfield

Computerland
690 S. Whitney Way, Madison

Computerland-Fox River Valley
1526 S. Koehlder St., Oshkosh

Computerworld
3015 W. Wisconsin Ave., Appleton

Magic Lantern Computed
3313 University Ave., Madison

Petted Microsystems
4265 W. Loomis Rd., Milwaukee

Wyoming

Computer Concepts
1104 Logan Ave., Cheyenne

Puerto Rico

The Microcomputer Store
1568 Ave. Jesus T. Pinero
Caparra Terrace

Canada

CANADIAN DISTRIBUTOR:
Micron Distributing
409 Queen St., W. Toronto, Ont.
M5V 2A5

Computerland of Winnipeg
715 Portage Ave., Winnipeg, Man.

Compumart
411 Roosevelt Ave., Ottawa, Ontario

Micromatic Systems Inc.
1303 Powell St., Vancouver

Micro Shack of W. Canada
333 Park Street, Regina. Sask.

Orthon Holdings Ltd.
12411 Stony Plain Road
Edmonton, Alberta

Total Computer Systems
Ajax, Ontario

✓2

Instant Software™

Peterborough, N.H.
03458

A renumbering program that does it all—in assembly language no less!

Speedy Renumberer

Robert J. Dowd
13141 Yockey Street #303
Garden Grove, CA 92644

Those of us who do a lot of BASIC programming learn the hard way how important a good line renumbering program can be. It's very frustrating to find no room between lines just when you want it. This happened to me once too often, and I did something about it.

Each renumbering program I tried added to my frustration. In addition to being written in BASIC, making them very slow, the programs wouldn't update the references to line numbers. This left the hardest part of the job to me.

I thought about running out to buy one of the many machine language renumbering programs on the market. Fortunately, my better judgement held out. I'd been looking for a good assembly language programming exercise: I could think of nothing better than writing my own assembly language renumber program and save $10-$20 by not buying a commercial program!

Renumbering

My renumber program (RENUMB) allows you to specify a line number range, new starting line number, and the increment to be used between the new line numbers.

RENUMB displays descriptive error messages on the screen whenever there are problems renumbering. Most problems which you may encounter are not terminal, and RENUMB will run to completion, leaving you to correct only a few minor errors.

Upon execution, RENUMB clears the screen and displays the message: ENTER START LINE.

Enter the line number where you want renumbering to start. Pressing the ENTER key without entering a line number is a default to the first line of the program. RENUMB next displays the message: ENTER END LINE. Enter the line number where renumbering is to stop. This line number will also be renumbered. The default for this value is the last line of the program.

RENUMB will then display: ENTER NEW START LINE. Enter the new number, or press the ENTER key which defaults to line number 100. Finally, RENUMB displays: ENTER INCREMENT. The default is an increment between renumbered lines of 10. If a non-numeric value is entered for any parameter, RENUMB will display PARAMETER NOT NUMERIC and ask for that parameter again.

When all four parameters are accepted, RENUMB begins renumbering. If no errors are detected, within a second or two, RENUMB displays: FINISHED, followed by the READY prompt from BASIC.

The renumbered program can then be run, listed, etc., since all references to the renumbered lines are updated. If the program contains the commands RUN or DELETE followed by a line number, these line numbers won't be updated.

Persistent Errors

If errors in your program interfere with renumbering, an appropriate error message is displayed. LINE # TOO LARGE is displayed if the calculation of a new line number exceeds 65529. This error ends renumbering. RANGE VALUE ERROR is displayed if the specified range of line numbers cannot be found in the program. It is also shown if the specified starting line number is larger than the specified ending line number. This error also terminates renumbering.

GOTO/GOSUB MISSING LINE # IN xxxxx is displayed when a GOTO or GOSUB is followed by anything but a valid line number. (Xxxxx represents the line number with the error after renumbering.) This error does *not* terminate renumbering.

UNDEFINED LINE # IN xxxxx is displayed when a reference is made to a line number falling within the renumbering range not contained in the program. This error does *not* terminate renumbering.

RENUMB will operate only in a 16K or larger Level II machine. This is due to the extensive use of ROM routines to provide sophistication with reasonably small size.

The program occupies 1125 bytes of RAM and requires an additional four for each line number renumbered. Very large programs in a 16K machine may be renumbered a small section at a time. ■

Program Listing

```
41E6            00010 INPBUF  EQU     41E6H           ;Keyboard input buffer
7B9A            00020         ORG     7B9AH
0001            00030 ARRAY   DEFS    1               ;Line # cross reference table
7B9B AF         00040 START   XOR     A               ;Clear A reg
7B9C 0616       00050         LD      B,22
7B9E 21687F     00060         LD      HL,INDEX        ;Initialize
7BA1 77         00070 LOOP1   LD      (HL),A          ;Program
7BA2 23         00080         INC     HL              ;Parameters
7BA3 10FC       00090         DJNZ    LOOP1
7BA5 2F         00100         CPL
7BA6 32707F     00110         LD      (ELINE),A       ;Set end line to
7BA9 32717F     00120         LD      (ELINE+1),A     ;Maximum value
7BAC 21767F     00130         LD      HL,NXTLPT       ;Set next line
7BAF 36E9       00140         LD      (HL),0E9H       ;Pointer
7BB1 23         00150         INC     HL              ;to start of
7BB2 3642       00160         LD      (HL),42H        ;program
7BB4 219A7B     00170         LD      HL,ARRAY        ;Initialize
7BB7 22687F     00180         LD      (INDEX),HL      ;array index
7BBA 3E64       00190         LD      A,100           ;Default value for
7BBC 32727F     00200         LD      (SLINE),A       ;new start line #
7BBF 3E0A       00210         LD      A,10            ;Default value for
7BC1 32747F     00220         LD      (INCR),A        ;increment
7BC4 2AF940     00230         LD      HL,(16633)      ;Get address of prog. end
7BC7 2B         00240         DEC     HL              ;Adjust
7BC8 2B         00250         DEC     HL              ;it
7BC9 227C7F     00260         LD      (PEND),HL       ;Save it
7BCC CDD47E     00270         CALL    OPTION          ;Get any operator values
7BCF 2A767F     00280 BEGIN   LD      HL,(NXTLPT)     ;Get current line address
7BD2 E5         00290         PUSH    HL              ;Save it
7BD3 ED5B7C7F   00300         LD      DE,(PEND)       ;Check for
7BD7 CD1D7D     00310         CALL    COMPAR          ;end of
7BDA D2717C     00320         JP      NC,PASS2        ;first pass
7BDD E1         00330         POP     HL              ;Get back start of line
7BDE E5         00340         PUSH    HL              ;Save it
7BDF 23         00350         INC     HL              ;Get
7BE0 23         00360         INC     HL              ;current
7BE1 227A7F     00370         LD      (CURLPT),HL     ;line # address
7BE4 E1         00380         POP     HL             ;Get back start of line
7BE5 0604       00390         LD      B,4
7BE7 DD21767F   00400         LD      IX,NXTLPT
7BEB 7E         00410 LOOP2   LD      A,(HL)          ;Get address of next line
7BEC DD7700     00420         LD      (IX+0),A        ;and current line #
7BEF 23         00430         INC     HL
7BF0 DD23       00440         INC     IX
7BF2 10F7       00450         DJNZ    LOOP2
7BF4 3A6D7F     00460         LD      A,(GOFLAG)      ;Have we already
7BF7 B7         00470         OR      A               ;found the start point
7BF8 200E       00480         JR      NZ,CKEND        ;if yes, continue
7BFA 2A787F     00490         LD      HL,(CURLIN)     ;look for
7BFD ED5B6E7F   00500         LD      DE,(BLINE)      ;start of
7C01 CD1D7D     00510         CALL    COMPAR          ;renumber range
7C04 2810       00520         JR      Z,SAVEM        ;yes. go renumber
7C06 38C7       00530         JR      C,BEGIN         ;No. Go get next line
7C08 2A787F     00540 CKEND   LD      HL,(CURLIN)     ;Check for
7C0B ED5B707F   00550         LD      DE,(ELINE)      ;end of
7C0F CD1D7D     00560         CALL    COMPAR          ;renumber range
7C12 2802       00570         JR      Z,SAVEM         ;Last line so save it
7C14 305B       00580         JR      NC,PASS2        ;Go resole references
7C16 AF         00590 SAVEM   XOR     A               ;set start point
7C17 2F         00600         CPL                     ;found
7C18 326D7F     00610         LD      (GOFLAG),A      ;flag
7C1B 2A727F     00620         LD      HL,(SLINE)      ;test for new line #
7C1E 11FAFF     00630         LD      DE,65530        ;greater than max.
7C21 CD1D7D     00640         CALL    COMPAR
7C24 3806       00650         JR      C,LINEOK        ;jump if line # less than max.
7C26 CDF17C     00660         CALL    ERROR1          ;go print error message
7C29 C37200     00670         JP      0072H           ;return to BASIC
7C2C 2A687F     00680 LINEOK  LD      HL,(INDEX)      ;INDEX points to
7C2F E5         00690         PUSH    HL              ;next ARRAY entry
7C30 DDE1       00700         POP     IX
```

Program continues

```
7C32 2A787F    00710          LD      HL,(CURLIN)     ;save
7C35 DD7400    00720          LD      (IX+0),H        ;old
7C38 DD2B      00730          DEC     IX              ;line
7C3A DD7500    00740          LD      (IX+00),L       ;number
7C3D DD2B      00750          DEC     IX
7C3F 2A727F    00760          LD      HL,(SLINE)      ;save
7C42 DD7400    00770          LD      (IX+0),H        ;new
7C45 DD2B      00780          DEC     IX              ;line
7C47 DD7500    00790          LD      (IX+0),L        ;number
7C4A DD2B      00800          DEC     IX
7C4C DD22687F  00810          LD      (INDEX),IX      ;update array pointer
7C50 2A6A7F    00820          LD      HL,(LINECT)     ;increment
7C53 23        00830          INC     HL              ;line
7C54 226A7F    00840          LD      (LINECT),HL     ;counter
7C57 2A7A7F    00850          LD      HL,(CURLPT)     ;now
7C5A E5        00860          PUSH    HL              ;update
7C5B DDE1      00870          POP     IX              ;the line #
7C5D 2A727F    00880          LD      HL,(SLINE)      ;in the line
7C60 DD7500    00890          LD      (IX+0),L
7C63 DD7401    00900          LD      (IX+1),H
7C66 ED5B747F  00910          LD      DE,(INCR)       ;calculate next
7C6A 19        00920          ADD     HL,DE           ;new
7C6B 22727F    00930          LD      (SLINE),HL      ;line number
7C6E C3CF7B    00940          JP      BEGIN           ;continue
7C71 2A6A7F    00950 PASS2    LD      HL,(LINECT)     ;see if
7C74 7C        00960          LD      A,H             ;any lines
7C75 B5        00970          OR      L               ;were renumbered
7C76 CAF87C    00980          JP      Z,ERROR2        ;if no, print error message
7C79 21E942    00990          LD      HL,42E9H        ;set HL to program start
7C7C 22767F    01000 PASS22   LD      (NXTLPT),HL     ;save it
7C7F AF        01010          XOR     A               ;clear the
7C80 326C7F    01020          LD      (ONFLAG),A      ;"ON GOTO/GOSUB" flag
7C83 110300    01030          LD      DE,3            ;now point HL to
7C86 19        01040          ADD     HL,DE           ;first byte of this
7C87 23        01050 GETBYT   INC     HL              ;BASIC statement
7C88 7E        01060          LD      A,(HL)          ;get the next byte
7C89 B7        01070          OR      A               ;is it a zero
7C8A 200C      01080          JR      NZ,COLON        ;if not, continue
7C8C 23        01090          INC     HL              ;ah,ha! end of statement so
7C8D ED5B7C7F  01100          LD      DE,(PEND)       ;check for
7C91 CD1D7D    01110          CALL    COMPAR          ;end of program
7C94 20E6      01120          JR      NZ,PASS22       ;if not, go get next statement
7C96 1848      01130          JR      FINISH          ;else go to end routine
7C98 FE3A      01140 COLON    CP      58              ;is it a colon
7C9A 2006      01150          JR      NZ,COMMA        ;if no, go check for comma
7C9C AF        01160          XOR     A               ;if yes
7C9D 326C7F    01170          LD      (ONFLAG),A      ;clear "ON GOTO/GOSUB" flag
7CA0 18E5      01180          JR      GETBYT          ;go get next byte
7CA2 FE2C      01190 COMMA    CP      44              ;is it a comma
7CA4 200A      01200          JR      NZ,CHEKON       ;if no, look for "ON"
7CA6 3A6C7F    01210          LD      A,(ONFLAG)      ;if an "ON" is active
7CA9 B7        01220          OR      A               ;make LOOKUP think it's a "GOTO"
7CAA 28DB      01230          JR      Z,GETBYT        ;if "ON" not active, get next byte
7CAC 3E8D      01240          LD      A,141           ;141=GOTO
7CAE 1820      01250          JR      GOFORT
7CB0 FEA1      01260 CHEKON   CP      161             ;is it "ON"
7CB2 2006      01270          JR      NZ,KEYWRD       ;if no, go look for other keywords
7CB4 2F        01280          CPL                     ;if yes
7CB5 326C7F    01290          LD      (ONFLAG),A      ;set "ON" flag
7CB8 18CD      01300          JR      GETBYT
7CBA FE8D      01310 KEYWRD   CP      141             ;is it "GOTO"
7CBC 2812      01320          JR      Z,GOFORT        ;yes
7CBE FE91      01330          CP      145             ;is it "GOSUB"
7CC0 280E      01340          JR      Z,GOFORT        ;yes
7CC2 FECA      01350          CP      202             ;is it "THEN"
7CC4 280A      01360          JR      Z,GOFORT        ;yes
7CC6 FE95      01370          CP      149             ;is it "ELSE"
7CC8 20BD      01380          JR      NZ,GETBYT       ;if not, get next byte
7CCA 08        01390          EX      AF,AF'          ;if "ELSE"
7CCB AF        01400          XOR     A               ;clear the
7CCC 326C7F    01410          LD      (ONFLAG),A      ;"ON" flag
7CCF 08        01420          EX      AF,AF'
```

Program continues

```
7CD0 CD8F7D   01430 GOFORT  CALL    LOOKUP          ;get line # and search table
7CD3 B7       01440         OR      A               ;A reg holds result of search
7CD4 CC017D   01450         CALL    Z,ERROR3        ;0=not found in table
7CD7 CB4F     01460         BIT     1,A             ;2=THEN/ELSE without line #
7CD9 20AD     01470         JR      NZ,GETBYT+1     ;which is OK
7CDB CD387E   01480         CALL    INSERT          ;insert the new line #
7CDE 18A8     01490         JR      GETBYT+1        ;continue
7CE0 2A7C7F   01500 FINISH  LD      HL,(PEND)       ;update
7CE3 23       01510         INC     HL              ;new
7CE4 23       01520         INC     HL              ;end location
7CE5 22F940   01530         LD      (16633),HL      ;for BASIC
7CE8 217E7F   01540         LD      HL,FINI         ;get address of 'FINISHED'
7CEB CDA728   01550         CALL    28A7H           ;display it
7CEE C37200   01560         JP      0072H           ;return to BASIC
7CF1 21887F   01570 ERROR1  LD      HL,EMSG1        ;'LINE # TOO LARGE'
7CF4 CDA728   01580         CALL    28A7H           ;display it
7CF7 C9       01590         RET
7CF8 219A7F   01600 ERROR2  LD      HL,EMSG2        ;'RANGE VALUE ERROR'
7CFB CDA728   01610         CALL    28A7H           ;display it
7CFE C37200   01620         JP      0072H           ;return to BASIC
7D01 E5       01630 ERROR3  PUSH    HL              ;save address of next byte
7D02 21AD7F   01640         LD      HL,EMSG3        ;'GOTO/GOSUB MISSING LINE #'
7D05 CDA728   01650         CALL    28A7H           ;display it
7D08 CD107D   01660         CALL    ERRET           ;display rest of error message
7D0B 21887C   01670         LD      HL,GETBYT+1     ;get abnormal return address
7D0E E3       01680         EX      (SP),HL         ;get next byte address off stack
7D0F C9       01690         RET                     ;and return to GETBYT+1
7D10 2A767F   01700 ERRET   LD      HL,(NXTLPT)     ;get number of this line
7D13 23       01710         INC     HL
7D14 23       01720         INC     HL
7D15 5E       01730         LD      E,(HL)
7D16 23       01740         INC     HL
7D17 56       01750         LD      D,(HL)
7D18 EB       01760         EX      DE,HL
7D19 CDA70F   01770         CALL    0FA7H           ;display 'IN LINE XXX'
7D1C C9       01780         RET
7D1D C5       01790 COMPAR  PUSH    BC              ;logically compare  HL:DE
7D1E D5       01800         PUSH    DE              ;save all registers
7D1F E5       01810         PUSH    HL              ;all will be restored except F
7D20 F5       01820         PUSH    AF
7D21 45       01830         LD      B,L             ;save LSBs of
7D22 4B       01840         LD      C,E             ;both operands
7D23 6C       01850         LD      L,H             ;move MSBs of
7D24 5A       01860         LD      E,D             ;both to LSB positions
7D25 AF       01870         XOR     A               ;clear A reg and carry flag
7D26 67       01880         LD      H,A             ;zero MSB positions
7D27 57       01890         LD      D,A             ;of HL and DE
7D28 ED52     01900         SBC     HL,DE           ;subtract DE from HL
7D2A 2805     01910         JR      Z,MSBEQU        ;jump if MSBS were equal
7D2C F2447D   01920         JP      P,HLGRTR        ;jump if HL greater DE
7D2F 180C     01930         JR      HLLESS          ;jump if HL less DE
7D31 68       01940 MSBEQU  LD      L,B             ;now test the
7D32 59       01950         LD      E,C             ;LSBs
7D33 AF       01960         XOR     A               ;clear A reg and carry flag
7D34 67       01970         LD      H,A             ;zero the MSB
7D35 57       01980         LD      D,A             ;positions of HL and DE
7D36 ED52     01990         SBC     HL,DE           ;subtract
7D38 2810     02000         JR      Z,EQUAL         ;jump if HL = DE
7D3A F2447D   02010         JP      P,HLGRTR        ;jump if HL greater DE
7D3D F1       02020 HLLESS  POP     AF              ;get original AF regs back
7D3E 47       02030         LD      B,A             ;save A
7D3F F6FF     02040         OR      0FFH            ;clear zero and carry flags
7D41 37       02050         SCF                     ;set carry to show HL less DE
7D42 1809     02060         JR      EXITPT          ;go bye-bye
7D44 F1       02070 HLGRTR  POP     AF              ;get original AF regs back
7D45 47       02080         LD      B,A             ;save A
7D46 F6FF     02090         OR      0FFH            ;clear zero & carry (HL > DE)
7D48 1803     02100         JR      EXITPT          ;go bye-bye
7D4A F1       02110 EQUAL   POP     AF              ;get back original AF
7D4B 47       02120         LD      B,A             ;save A
7D4C AF       02130         XOR     A               ;HL=DE. SET ZERO FLAG
```

Program continues

```
7D4D 78        02140 EXITPT  LD      A,B             ;restore A reg
7D4E E1        02150         POP     HL
7D4F D1        02160         POP     DE
7D50 C1        02170         POP     BC
7D51 C9        02180         RET
7D52 CD1D7D    02190 HLDESB  CALL    COMPAR    ;16-bit subtract (HL-DE), no sign
7D55 D8        02200         RET     C         ;HL must be greater DE, else return
7D56 C5        02210         PUSH    BC              ;save BC regs
7D57 4F        02220         LD      C,A             ;save A reg
7D58 2826      02230         JR      Z,ZEROIT        ;if HL=DE go set HL to zero
7D5A 0600      02240         LD      B,0             ;clear 'adjust' flag
7D5C CB7C      02250         BIT     7,H             ;is MSB of HL set
7D5E 280F      02260         JR      Z,CLEAR         ;if not, don't adjust
7D60 D5        02270         PUSH    DE              ;save DE
7D61 B7        02280         OR      A               ;clear carry flag
7D62 CB1C      02290         RR      H               ;divide HL by 2
7D64 CB1D      02300         RR      L         ;with any remainder in carry flag
7D66 F5        02310         PUSH    AF              ;save remainder (carry flag)
7D67 B7        02320         OR      A               ;clear carry flag
7D68 CB1A      02330         RR      D               ;divide DE
7D6A CB1B      02340         RR      E               ;by 2
7D6C 3816      02350         JR      C,CHKCRY ;go see if carry adjust is needed
7D6E 04        02360         INC     B               ;set 'adjust' flag
7D6F B7        02370 CLEAR   OR      A               ;clear carry flag
7D70 ED52      02380         SBC     HL,DE           ;do the subtract
7D72 CB40      02390         BIT     0,B             ;test the 'adjust' flag
7D74 2806      02400         JR      Z,SKIPIT        ;if not set, skip readjustment
7D76 F1        02410         POP     AF              ;get back remainder, if any
7D77 CB15      02420         RL      L               ;multiply result by
7D79 CB14      02430         RL      H               ;2 to get true result
7D7B D1        02440         POP     DE              ;get back original DE
7D7C B7        02450 SKIPIT  OR      A               ;clear carry flag
7D7D 79        02460         LD      A,C             ;restore original A reg
7D7E C1        02470         POP     BC              ;restore original BC
7D7F C9        02480         RET                     ;go bye-bye
7D80 ED62      02490 ZEROIT  SBC     HL,HL           ;clear HL reg
7D82 18F8      02500         JR      SKIPIT          ;go to exit
7D84 F1        02510 CHKCRY  POP     AF              ;get carry from HL/2
7D85 3003      02520         JR      NC,CHKCR1       ;if none, make adjustment
7D87 B7        02530         OR      A               ;if there was, get rid of it
7D88 1802      02540         JR      CHKCR1+2        ;go clean up
7D8A 13        02550 CHKCR1  INC     DE              ;adjust DE
7D8B 37        02560         SCF                     ;set the carry flag
7D8C F5        02570         PUSH    AF              ;save it
7D8D 18DF      02580         JR      CLEAR-1         ;go do subtract
7D8F 23        02590 LOOKUP  INC     HL              ;point to 1st digit of line #
7D90 E5        02600         PUSH    HL              ;save that address
7D91 F5        02610         PUSH    AF              ;save keyword code
7D92 CD5A1E    02620         CALL    1E5AH           ;convert line # to binary
7D95 7A        02630         LD      A,D             ;test for non-digit
7D96 B3        02640         OR      E
7D97 CA237E    02650         JP      Z,POSERR        ;if no line#, POSSIBLE error
7D9A F1        02660         POP     AF              ;clear stack
7D9B 2A6E7F    02670         LD      HL,(BLINE)      ;test line # to see
7D9E CD1D7D    02680         CALL    COMPAR          ;if it's in the renumber range
7DA1 2810      02690         JR      Z,FINDIT        ;it's ok
7DA3 300A      02700         JR      NC,RETURN       ;jump if not in range
7DA5 2A707F    02710         LD      HL,(ELINE)      ;check other end of range
7DA8 CD1D7D    02720         CALL    COMPAR
7DAB 2806      02730         JR      Z,FINDIT        ;if ok, go look for it
7DAD 3004      02740         JR      NC,FINDIT       ;same as above
7DAF E1        02750 RETURN  POP     HL              ;it wasn't in range
7DB0 C32F7E    02760         JP      NOERR           ;so get out of here
7DB3 ED4B6A7F  02770 FINDIT  LD      BC,(LINECT)     ;get array search limit
7DB7 DD219A7B  02780         LD      IX,ARRAY        ;set up start of table
7DBB DD6600    02790 LOOKLP  LD      H,(IX+0)        ;look up the number
7DBE DD2B      02800         DEC     IX              ;HL=old line # from table
7DC0 DD6E00    02810         LD      L,(IX+0)
7DC3 DD2B      02820         DEC     IX
7DC5 CD1D7D    02830         CALL    COMPAR          ;see if we found it
7DC8 280C      02840         JR      Z,GOTMAT        ;hooray! we found it
7DCA DD2B      02850         DEC     IX              ;point to the next
```

Program continues

N E W PEN737

FINALLY - THE ULTIMATE IN WORD PROCESSING FROM COMPUTEX

Superscripts / Subscripts / Underlines / **Elongated** printing / BOLD face type and many other features can now be accomplished with your TRS80 computer, Electric Pencil, Centronics 737 Printer and our new PEN737 program.

PEN737 program (available on disk or cassette) **$29.95**
When purchased with Centronics 737 printer **$19.95**

CENTRONICS 737 PRINTER
Special $750.00

FEATURES: Fan fold, roll and single sheet paper handling • True underlining capabilities • Superscripts and Subscripts • True UPPER & lower case • 80 or 132 column lines • Fast 80 cps printing • Paper forward and reverse switch • Supports 10 and 16.7 c.p.i

TRS80 MODEL I or MODEL III PRINTER CABLE $29.95

16K MEMORY EXPANSION KITS FOR MODEL I AND MODEL III SYSTEMS
$35.95/kit 2-5 kits $32.95/kit 6 up $29.95/kit

LNW EXPANSION I/F and COLOR COMPUTER KITS

COMPLETE LNW EXPANSION INTERFACE
..... $249.00

Includes all components, sockets, and hardware to build the complete kit. Does not include the relay for dual cassette, cabinet or Ram.

RAM MEMORY $29.95/16K WHEN PURCHASED WITH LNW

Complete CPT Systems with disk drive, 32K Ram, RS232 and Custom cabinet as low as $799.00
(CALL FOR QUOTE)

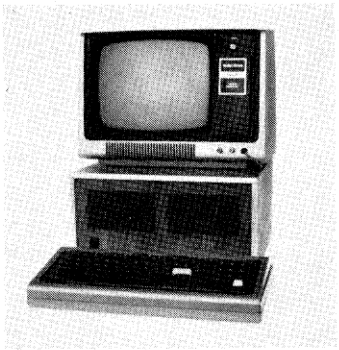
LNW COLOR Computer Kit $599.00

Includes all components to assemble the LNW Color Computer including: High Resolution Video Ram and 16K user ram, Keyboard, and Level II compatible Roms.

CUSTOM CABINET FOR LNW COLOR COMPUTER AND EXPANSION I/F **$149.95**

CUSTOM CABINET FOR LNW EXPANSION I/F, AND DISK DRIVES ..... **$ 99.95**

We also stock a complete line of components for both the LNW Color Computer and the LNW System Expansion at NEVER UNDERSOLD PRICES



TRS80 is a registered trade mark of Radio Shack
Certified Checks, Money Orders
Or call to arrange direct wire transfer
Visa and Mastercharge accepted, add 4% to order.
Personal or company checks require 2 weeks to clear.

COMPUTEX ✓392
15502 Highway 3, Suite 715
Webster, Texas 77598
(713) 488-8022

DISK DRIVES - Compare !!!!

RELIABILITY / FEATURES / WARRANTY
PRICE / DELIVERY
Computex offers the best of all with TANDON Disk Drives

Warranty: 120 day repair / replace warranty
Reliability: 8000 hrs. mean time between failure
Features: 5 m.s. Track to Track Access Time
0 sec. Head load time (no solenoid either!)
Single or Double Density Capabilities
40 / 80 Track - Single & Double Sided
Maintenance Manual Provided with drive

TM100-1 Single headed 40 track disk drive, unformatted double density storage = 250 KB
TRS80[tm] Storage single density = 102 KB Double = 184KB
Bare drive $225.00 **With case/supply $299.95**

TM100-2 Double headed 40 track drive, unformatted double density storage = 500 KB
TRS80[tm] Storage single density = 204KB, Double = 367KB
Bare drive $345.00 **With case/supply $419.95**

TM100-3 Single headed 80 track drive, unformatted double density storage 500KB
TRS80[tm] Storage single density = 204KB, Double = 367KB
Bare drive $375.00 **With case/supply $449.00**

TM100-4 Double headed 80 track drive, unformatted double density storage = 1.0mB
TRS80[tm] Storage single density = 408KB, Double = 735KB

Bare drive $475.00 **With case/supply $549.00**

2 Drive Disk Cable **$24.95**
4 Drive Disk Cable **$34.95**
Drive Extender Cable **$13.95**

MODEL III DISK DRIVE KITS

Complete kit installs in minutes, Interfaces any Tandon disk drive to your Model III, Detailed instructions included **$429.95**

TIME ...

METHUSELAH™ has time for you and your TRS·80®

The name Methuselah has always been associated with long life. The Methuselah computer clock board with its 24 hour clock and its perpetual calendar is indeed worthy of the title. Methuselah puts state of the art technology into action with the on board four year lithium battery back-up to keep it running even when your computer is off. That means no more fooling around with the software or hardware "clocks" that become Rip Van Winkles when the computer is reset or turned off.

Methuselah has many other timely features. Software patches keep the DOS and BASIC time and clock commands ticking. Only two screws mount Methuselah inside the expansion interface without soldering, clipping, cutting, or jumpering. There is even a spare 1/0 port and four different interrupts available for people who love to tinker.

SPECS: MM/DD/YY, HH:MM:SS and day of the week. Four year lithium battery back-up. Crystal controlled timing (adjustable 32.768 oscillator). Twenty four hour clock and perpetual calendar.

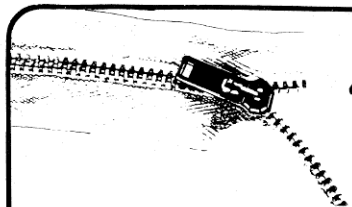
Send check or money order to:

M-S1 ..... $97.00
(add $4.50 P and H)
S-S1 ..... $17.50
(add $2.50 P and H)
(If you wish to use Methuselah and the RS-232 board (26-1145) together, order this option which includes a new cover and connector.)
WI residents add 4% sales tax.

NDM DESIGNS ✓135
6330 W. North Ave.
Wauwatosa, WI 53213
(414) 259-0120

METHUSELAH is a trade mark of NDM Designs

® Reg. T.M. Tandy Corp.



ZIP UP YOUR MODEL I

SPEED-UP UNIT - an easy to install electronic device that enables programs to run 2 to 3 times faster. It's the fastest and finest quality speed-up unit on the market.

- **guaranteed to double processing speed - no additional purchases required.**
- **programs will run up to 3 times faster (5.3 mhz) by adding a Z80B microprocessor and delay line - not supplied.**
- **returns automatically to normal speed during disk and cassette operation - no software patches required.**
- **keyboard power-on light changes color to indicate operating speed.**

$45.00

VIDEO I. Reverses the video display to provide black characters and graphics on an all white screen for a much easier to read presentation. Software controlled. Cures pulling and distortion problems commonly experienced with TRS-80 monitors. For use with TRS-80 monitors only. Assembled.

$24.00

"SATISFACTION GUARANTEED"

Add $2.00 postage & handling - California residents add 6% sales tax -Foreign orders add 15%.

ARCHBOLD ELECTRONICS
10708 Segovia Way **Rancho Cordova, CA 95670**
(916) 635-5408
Dealer inquiries invited

```
7DCC DD2B      02860          DEC     IX              ;line # in the table
7DCE 0B        02870          DEC     BC              ;see if we've
7DCF 78        02880          LD      A,B             ;looked at the
7DD0 B1        02890          OR      C               ;entire table
7DD1 20E8      02900          JR      NZ,LOOKLP       ;if not, try again
7DD3 C3327E    02910          JP      ERROR5          ;go print the error message
7DD6 DD6600    02920 GOTMAT   LD      H,(IX+0)        ;get new line # into HL
7DD9 DD2B      02930          DEC     IX
7DDB DD6E00    02940          LD      L,(IX+0)
7DDE AF        02950          XOR     A               ;clear the 'go' flag
7DDF 326D7F    02960          LD      (GOFLAG),A
7DE2 FD21FB7F  02970          LD      IY,ASCIIN       ;convert the line #
7DE6 DD21F17F  02980          LD      IX,POWERS       ;to ASCII
7DEA 0605      02990          LD      B,5             ;max. of 5 digits
7DEC DD5E00    03000 LOOKL2   LD      E,(IX+0)        ;load DE with
7DEF DD5601    03010          LD      D,(IX+1)        ;power of ten
7DF2 3E00      03020          LD      A,0             ;A=INT(line#/power of ten)
7DF4 0E30      03030          LD      C,'0'           ;C+A will equal the ASCII digit
7DF6 CD527D    03040 LOOKL3   CALL    HLDESB          ;subtract the power
7DF9 3021      03050          JR      NC,INCRA        ;it went so inc A
7DFB B7        03060          OR      A               ;the power didn't go
7DFC 2811      03070          JR      Z,BLNKIT        ;go see if leading zero
7DFE 81        03080          ADD     A,C             ;calculate the digit
7DFF 326D7F    03090          LD      (GOFLAG),A      ;set the 'go' flag to non-zero
7E02 FD7700    03100 LOOKL4   LD      (IY+0),A        ;save the digit in ASCIIN
7E05 FD23      03110          INC     IY              ;point to next posn in ASCIIN
7E07 DD23      03120          INC     IX
7E09 DD23      03130          INC     IX
7E0B 10DF      03140          DJNZ    LOOKL2          ;continue the conversion
7E0D 1810      03150          JR      ALLDON
7E0F 3A6D7F    03160 BLNKIT   LD      A,(GOFLAG)      ;is this a leading zero
7E12 B7        03170          OR      A
7E13 2004      03180          JR      NZ,BLNK1        ;jump if not a leading blank
7E15 3E20      03190          LD      A,' '           ;it's a leading zero, so
7E17 18E9      03200          JR      LOOKL4          ;put a blank in
7E19 79        03210 BLNK1    LD      A,C             ;put a zero in
7E1A 18E6      03220          JR      LOOKL4
7E1C 3C        03230 INCRA    INC     A
7E1D 18D7      03240          JR      LOOKL3          ;try same power of ten
7E1F 3E01      03250 ALLDON   LD      A,1             ;tell 'em we got it
7E21 E1        03260          POP     HL              ;get back pointerto lst digit
7E22 C9        03270          RET
7E23 F1        03280 POSERR   POP     AF              ;get keyword code
7E24 E1        03290          POP     HL              ;get back pointer
7E25 FECA      03300          CP      202             ;if it's 'THEN'
7E27 2806      03310          JR      Z,NOERR         ;there's no error
7E29 FE95      03320          CP      149             ;or if it's 'ELSE'
7E2B 2802      03330          JR      Z,NOERR         ;there's no error
7E2D AF        03340          XOR     A               ;uh, oh! it's an error
7E2E C9        03350          RET
7E2F 3E02      03360 NOERR    LD      A,2             ;tell 'em it's ok
7E31 C9        03370          RET
7E32 21DF7F    03380 ERROR5   LD      HL,EMSG5        ;get the message location
7E35 C3057D    03390          JP      ERROR3+4        ;display it
7E38 7E        03400 INSERT   LD      A,(HL)          ;get a digit from the line
7E39 FE20      03410          CP      ' '             ;is it a blank
7E3B 2003      03420          JR      NZ,INSRT1       ;if not, insert the new number
7E3D 23        03430          INC     HL              ;it's a blank
7E3E 18F8      03440          JR      INSERT          ;so skip it
7E40 0605      03450 INSRT1   LD      B,5
7E42 DD21FB7F  03460          LD      IX,ASCIIN       ;ASCIIN has the new # in ASCII
7E46 DD7E00    03470 INSRT2   LD      A,(IX+0)        ;get a byte from the new line #
7E49 FE20      03480          CP      ' '             ;see if it's a leading blank
7E4B 2004      03490          JR      NZ,INSRT3       ;if not, go to insert the digit
7E4D DD23      03500          INC     IX              ;skip the leading blank
7E4F 10F5      03510          DJNZ    INSRT2
7E51 F5        03520 INSRT3   PUSH    AF              ;save the new digit
7E52 7E        03530          LD      A,(HL)          ;see if there's room for it
7E53 FE30      03540          CP      '0'             ;if nxt byte not numeric
7E55 FA737E    03550          JP      M,INSRT5        ;make room for the new digit
7E58 FE3A      03560          CP      ':'
```

Program continues

STOCK MARKET SOFTWARE

By H & H Trading Company for Mod. I, II, & III TRS-80®s

STOCK TRACKER™ times trades on individual stocks and options. Uses a technical volume analysis. Completely objective. Gives buy, sell signals.

MARKET TRACKER™ times tops & bottoms on the Dow Industrials; a composite of six technical indicators. Some judgement required. Gives buy, sell signals.

These two programs are accurate and profitable. Hundreds of satisfied users.

ASK FOR THE EVIDENCE!

FOR MORE INFORMATION OR TO ORDER, CONTACT:

H & H TRADING COMPANY
POST OFFICE BOX 23546
PLEASANT HILL, CALIFORNIA 94523
Telephone 415/937-1030

✓223

VISA & MASTERCARD

®Regd. T.M. of Radio Shack ©

COMPUTER EQUIPMENT & SOFTWARE BARGAINS

EVERY MONTH

BUY, SELL OR TRADE ALL TYPES OF COMPUTER EQUIPMENT AND SOFTWARE (pre-owned and new) among 20,000 readers nationwide.

FEATURES:

- Low classified ad rates - 10¢ a word
- Hundreds of ads from individuals
- Categorized ads so you can find them instantly
- Large (11 by 14") easy to read pages

Subscribe now for $10 and receive 13 issues/year (one FREE plus 12 regular issues). After receiving your first issue if you're not completely satisfied you may have a 100% refund and you still keep the first issue free. Bank cards accepted.

BONUS: *If you have something to advertise (pre-owned or software) send in a classified ad with your subscription and we'll run it FREE.*

The Nationwide Marketplace for Computer Equipment
COMPUTER SHOPPER ✓212
P.O. BOX F27 • TITUSVILLE, FL 32780 • 305-269-3211

MasterCard & VISA subscriptions only, call **TOLL FREE 1-800-528-6050 Ex. 184**.

This Month Only

Lowest Price
and we guarantee it.

on CENTRONICS 737

$789
Same as Line Printer 4

If you find a lower price advertised in this magazine this month, just . . .

Deduct $20 from their price.

AND send check or M.O. or call in Visa or M.C.

This special price includes shipping anywhere in the continental U.S.

* This price good on pre-paid orders only. Sorry, no C.O.D. on this item.

Supplies limited, so hurry!

master charge VISA

SSM SOFT SECTOR MARKETING, INCORPORATED
6250 Middlebelt • Garden City, MI 48135 ✓434
1 (313) 425-4020 **Dealer inquiries invited.**

Most orders shipped next day. Michigan residents, please add 4% tax. Personal checks take 3 weeks to clear. Send $1.00 for catalog - get $2.00 credit on next order.

✓Reader Service—see page 290

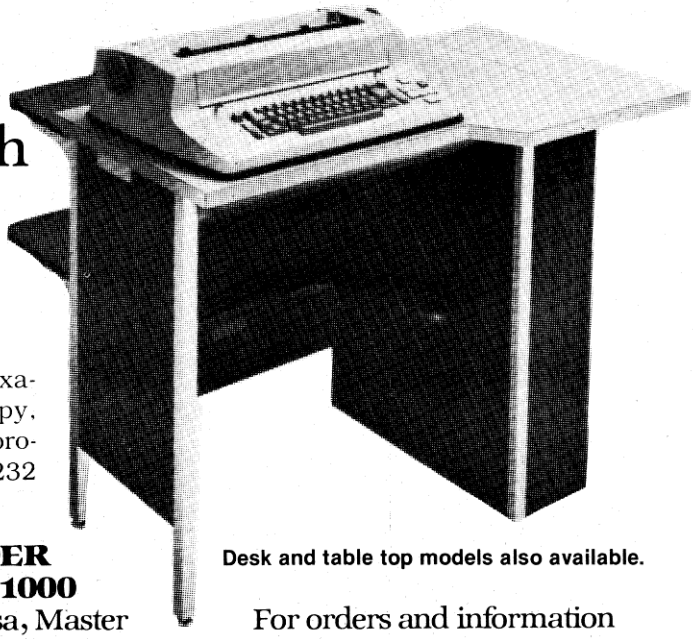
```
7E5A F2737E   03570          JP      P,INSRT5        ;gomake room for the new digit
7E5D F1       03580 INSRT6   POP     AF              ;restore new digit
7E5E 77       03590          LD      (HL),A          ;insert it into the line
7E5F DD23     03600          INC     IX
7E61 23       03610          INC     HL
7E62 DD7E00   03620          LD      A,(IX+0)        ;get next digit
7E65 10EA     03630          DJNZ    INSRT3          ;GO til all digits inserted
7E67 7E       03640 INSRT4   LD      A,(HL)          ;all digits inserted so
7E68 FE30     03650          CP      '0'             ;check nxt character in line
7E6A F8       03660          RET     M               ;not a digit so exit
7E6B FE3A     03670          CP      ':'
7E6D F0       03680          RET     P               ;not a digit so exit
7E6E CD967E   03690          CALL    COMPRS          ;compress the program
7E71 18F4     03700          JR      INSRT4          ;go check the next character
7E73 CD787E   03710 INSRT5   CALL    EXPAND          ;make room for a digit
7E76 18E5     03720          JR      INSRT6          ;now go insert it
7E78 E5       03730 EXPAND   PUSH    HL
7E79 08       03740          EX      AF,AF'          ;use alternate
7E7A D9       03750          EXX                     ;registers
7E7B D1       03760          POP     DE
7E7C 2A7C7F   03770          LD      HL,(PEND)       ;add 1 to
7E7F 23       03780          INC     HL              ;program
7E80 E5       03790          PUSH    HL              ;end
7E81 227C7F   03800          LD      (PEND),HL       ;address
7E84 23       03810          INC     HL
7E85 E5       03820          PUSH    HL
7E86 CD527D   03830          CALL    HLDESB          ;calculate # of bytes to move
7E89 E5       03840          PUSH    HL
7E8A C1       03850          POP     BC              ;BC=# BYTES TO MOVE
7E8B D1       03860          POP     DE              ;DE=(PEND)
7E8C E1       03870          POP     HL              ;HL=(PEND)-1
7E8D EDB8     03880          LDDR                    ;move the program to make room
7E8F AF       03890          XOR     A               ;A=0 means text was expanded
7E90 CDB57E   03900          CALL    FIXPTS          ;fix up line pointers
7E93 08       03910          EX      AF,AF'
7E94 D9       03920          EXX
7E95 C9       03930          RET
7E96 E5       03940 COMPRS   PUSH    HL
7E97 08       03950          EX      AF,AF'          ;use alternate registers
7E98 D9       03960          EXX
7E99 D1       03970          POP     DE
7E9A 2A7C7F   03980          LD      HL,(PEND)       ;subtract 1
7E9D 2B       03990          DEC     HL              ;from program
7E9E 227C7F   04000          LD      (PEND),HL       ;end address
7EA1 23       04010          INC     HL
7EA2 23       04020          INC     HL
7EA3 CD527D   04030          CALL    HLDESB          ;calculate # of bytes to move
7EA6 E5       04040          PUSH    HL
7EA7 C1       04050          POP     BC              ;BC=# of bytes to move
7EA8 D5       04060          PUSH    DE              ;DE=address of byte to remove
7EA9 E1       04070          POP     HL
7EAA 23       04080          INC     HL              ;HL=DE+1
7EAB EDB0     04090          LDIR
7EAD AF       04100          XOR     A
7EAE 2F       04110          CPL                     ;A=NZ IF text was compressed
7EAF CDB57E   04120          CALL    FIXPTS          ;fix up line pointers
7EB2 08       04130          EX      AF,AF'
7EB3 D9       04140          EXX
7EB4 C9       04150          RET
7EB5 2A767F   04160 FIXPTS   LD      HL,(NXTLPT)     ;get address of next line
7EB8 5E       04170 FIXPT1   LD      E,(HL)          ;into the
7EB9 23       04180          INC     HL              ;DE
7EBA 56       04190          LD      D,(HL)
7EBB B7       04200          OR      A               ;see if we compressed or expanded
7EBC 2813     04210          JR      Z,ADDON         ;jump if expanded
7EBE 1B       04220          DEC     DE              ;reduce pointer since we compressed
7EBF 72       04230 FIXPT2   LD      (HL),D          ;put in the new pointer
7EC0 2B       04240          DEC     HL
7EC1 73       04250          LD      (HL),E
7EC2 D5       04260          PUSH    DE              ;see if
7EC3 C1       04270          POP     BC              ;we've done
7EC4 EB       04280          EX      DE,HL           ;all
```

Program continues

The DATA-TRANS 1000

A completely refurbished **IBM** Selectric Terminal with built-in **ASCII** Interface.

***FOR YOUR TRS-80 WITH OR WITHOUT EXPANSION INTERFACE. AVAILABLE WITH CENTRONICS TYPE PARALLEL PORT.**



Desk and table top models also available.

Features:

- 300 Baud Serial
- 14.9 characters per second printout
- Reliable heavy duty Selectric mechanism
- RS-232C Interface
- Documentation included
- 60 day warranty—parts and labor
- High quality Selectric printing Off-line use as typewriter
- Optional tractor feed available
- 15 inch carriage width

Also works with Exatron's Stringy floppy, for fast loading of programs. (Has RS232 built in stringy)

HOW TO ORDER DATA-TRANS 1000

1. We accept Visa, Master Charge. Make cashiers checks or personal check payable to:

DATA-TRANS

2. All orders are shipped F.O.B. San Jose, CA
3. Deliveries are immediate

For orders and information

DATA-TRANS

45277 Fremont Blvd., #7 ✓274
Fremont CA 94538

Phone: (408) 263-9246

Bayesian Investment Services

Send for free catalog discussing TRS-80 programs available for evaluating:

Options & Futures
Stocks & Bonds
Convertible
& Other Hedges
Portfolio Position
Tax Positions
Market Timing

Write or call:

Bayesian ✓186
Investment Services
Dept. H3
757 Santa Rosita
Solana Beach, CA 92075
(714) 755-6225

New Software for the New TRS-80 Color Computer

TRS-80* **COLOR** TRS-80*

SPACE INVADERS — The full-feature popular arcade game with super fast HIGH RESOLUTION (256 x 192) graphics and sound. Can you beat the invaders?
16K, cassette $19.96

GAME OF LIFE — Fastest game of LIFE ever in exciting, HIGH RESOLUTION graphics. Set up any starting colony and see if it will survive or die.
16K, cassette $19.95

ETCH-A-SKETCH — Your joysticks and computer make an etch-a-sketch pad.
4K, cassette $9.95

ARTILLERY DUEL — Fight computer or friend to destroy his artillery.
4K, cassette $9.95

16K RAM — Expand your 4K system to 16K. No hardware modifications or special tools required. Full instructions included, easy to install.
$75.00

EDITOR/ASSEMBLER — Write your own 6809 assembly language programs. Generates system tapes and allows storage of source files on cassette.
16K, cassette $29.95

BUGOUT — Full Feature MONITOR with EDIT, FILL, Register EXAMINE/MODIFY, HEX/ASCII dumps, breakpoints and TRACE, a powerful debugging tool.
16K, cassette $19.95

DISASSEMBLER — The best disassembler for the 6809 available — ANYWHERE. Unlock the secrets of COLOR BASIC. A MUST for the serious programmer.
16K, cassette $14.95

BUGOUT PLUS — BUGOUT & DISASSEMBLER mixed together.
16K, cassette $29.95

COMING SOON — MAGIC BOX — load MOD I & II tapes into the color computer. 16K expansion — put 32K RAM inside the computer, or put 16K RAM on the ROM PAK port. GAMES — fast action HIGH RESOLUTION games like ASTEROID and SPACE WAR. ROM UTILITIES — put utilities like EDITOR/ ASSEMBLER and BUGOUT in ROM on the ROM PAK port.

WRITE FOR COMPLETE CATALOG OF GAMES & UTILITIES

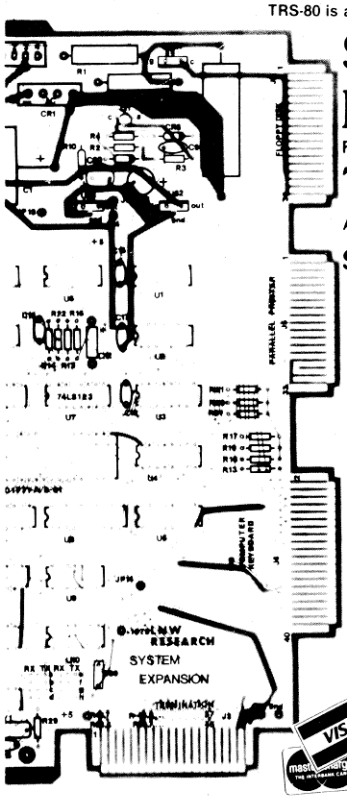
*TRS-80 IS A TM OF TANDY CORP.

SPECTRAL ASSOCIATES ✓132
141 Harvard Ave.
Tacoma, Washington 98466
(206) 475-8483

```
7EC5 ED5B7C7F 04290          LD      DE,(PEND)       ;the pointers
7EC9 CD1D7D  04300          CALL    COMPAR
7ECC D0      04310          RET     NC              ;return if we have
7ECD C5      04320          PUSH    BC              ;keep going
7ECE E1      04330          POP     HL
7ECF 18E7    04340          JR      FIXPT1
7ED1 13      04350 ADDON    INC     DE              ;increment the pointer
7ED2 18EB    04360          JR      FIXPT2
7ED4 CDC901  04370 OPTION   CALL    1C9H            ;clear screen
7ED7 DD21177F 04380         LD      IX,AMSG1
7EDB FD216E7F 04390         LD      IY,BLINE
7EDF 0604    04400          LD      B,4             ;request and accept
7EE1 C5      04410 LOOPOP   PUSH    BC              ;operator values if any
7EE2 DD6E00  04420          LD      L,(IX+0)
7EE5 DD6601  04430          LD      H,(IX+1)
7EE8 CDA728  04440          CALL    28A7H           ;output a request message
7EEB 21E641  04450          LD      HL,INPBUF
7EEE CDD905  04460          CALL    5D9H            ;get operator input
7EF1 AF      04470          XOR     A
7EF2 B0      04480          OR      B               ;any input?
7EF3 280D    04490          JR      Z,NEXTOP        ;if none, try next
7EF5 CD5A1E  04500          CALL    1E5AH           ;convert value to binary
7EF8 7A      04510          LD      A,D
7EF9 B3      04520          OR      E               ;if not numeric
7EFA 2812    04530          JR      Z,ERROR4        ;print error message
7EFC FD7300  04540          LD      (IY+0),E        ;else, save it
7EFF FD7201  04550          LD      (IY+1),D
7F02 DD23    04560 NEXTOP   INC     IX
7F04 DD23    04570          INC     IX
7F06 FD23    04580          INC     IY
7F08 FD23    04590          INC     IY
7F0A C1      04600          POP     BC              ;getloop ctr back
7F0B 10D4    04610          DJNZ    LOOPOP          ;try next message
7F0D C9      04620          RET
7F0E 21C87F  04630 ERROR4   LD      HL,EMSG4
7F11 CDA728  04640          CALL    28A7H           ;display it
7F14 C1      04650          POP     BC              ;retrieve loop counter
7F15 18CA    04660          JR      LOOPOP          ;try same one again
7F17 1F7F    04670 AMSG1    DEFW    MESG1           ;table of message addresses
7F19 317F    04680          DEFW    MESG2
7F1B 417F    04690          DEFW    MESG3
7F1D 577F    04700          DEFW    MESG4
7F1F 45      04710 MESG1    DEFM    'ENTE'
7F23 52      04720          DEFM    'R ST'
7F27 41      04730          DEFM    'ART '
7F2B 4C      04740          DEFM    'LINE'
7F2F 20      04750          DEFM    ' '
7F30 00      04760          DEFB    0
7F31 45      04770 MESG2    DEFM    'ENTE'
7F35 52      04780          DEFM    'R EN'
7F39 44      04790          DEFM    'D LI'
7F3D 4E      04800          DEFM    'NE '
7F40 00      04810          DEFB    0
7F41 45      04820 MESG3    DEFM    'ENTE'
7F45 52      04830          DEFM    'R NE'
7F49 57      04840          DEFM    'W ST'
7F4D 41      04850          DEFM    'ART '
7F51 4C      04860          DEFM    'LINE'
7F55 20      04870          DEFM    ' '
7F56 00      04880          DEFB    0
7F57 45      04890 MESG4    DEFM    'ENTE'
7F5B 52      04900          DEFM    'R IN'
7F5F 43      04910          DEFM    'CREM'
7F63 45      04920          DEFM    'ENT '
7F67 00      04930          DEFB    0
7F68 0000    04940 INDEX    DEFW    0
7F6A 0000    04950 LINECT   DEFW    0
7F6C 00      04960 ONFLAG   DEFB    0
7F6D 00      04970 GOFLAG   DEFB    0
7F6E 0000    04980 BLINE    DEFW    0
7F70 0000    04990 ELINE    DEFW    0
7F72 0000    05000 SLINE    DEFW    0
```

Program continues

TRS-80 is a registered trademark of TANDY CORP.



SYSTEM EXPANSION

FOR THE

TRS-80™

AT

$69.95 [PC BOARD & USER MANUAL]

- SERIAL RS232C/20mA I/O
- FLOPPY CONTROLLER
- 32K BYTES MEMORY
- PARALLEL PRINTER PORT
- DUAL CASSETTE PORT
- REAL-TIME CLOCK
- SCREEN PRINTER BUS
- ONBOARD POWER SUPPLY
- SOFTWARE COMPATIBLE
- SOLDER MASK,SILK SCREEN

LNW RESEARCH ↙53

714-641-8850

TO ORDER

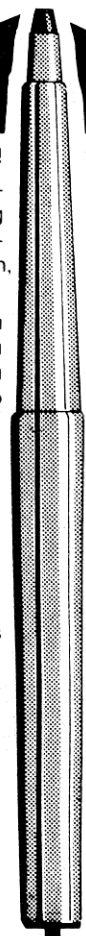
P.O. Box 16216 Irvine CA 92713
Add $3 for postage and handling.
CA residents add 6% sales tax

Card No. _____ Expiration Date_____ Signature _____

The Original "Photo point"™ Light Pen

ONLY **$19.95**

Why pay $30.00, $35.00, $62.00 for the same results??



SILVER & BLACK

Don't be misled by more expensive imitations!

This is the **original** *Photo point* light pen preferred and supported by some of the leading software sources like, "Quality Software"—"Instant Software"—"Level IV"products and so on.

Just imagine . . .

In playing backgammon, (included) when you want to move a man, you just point at where you want to move from, then point at where you want to move to, and your man moves!!! No more fumbling with keyboards—YEA!

Your Photo Point package comes complete;

- 1 Photo point light pen (of course)
- Info sheets on how to connect the pen and how to write your own programs ALL IN BASIC
- Two apertures
- AND two sensitivity settings
- A cassette tape with 4 informative programs and games
- Ready to connect to your TRS-80 System. **(DOS too!)**
- Does not void any Radio Shack warranties

Requirements:

- Level II basic
- And a little *imagination!!*

For fast real time programming it is your lowest cost peripheral at $19.95

Announcing

NEW PEN BASIC by Steve Bjork

Steve is one of the Best Assembly Lang. programmers around, and he has come up with PEN BASIC. This low memory routine will add 10 more commands to Level II such as PENGET which searches the entire screen for the pen and returns a number between 0-1024 in about 1 sec. Plus 9 other commands. Perfect for you lightware authors and **NEW** light pen owners too! . . . . . .only **$14.95**

(COUPON)

Micro Matrix ↙68
P.O. Box 938 • Pacifica, CA 94044
Send for yours **NOW:** **(415) 355-4635**

Name _____ Photopoint ☐ $19.95

Address _____

City _____ St._____ Pen Basic ☐ $14.95

Zip _____

Card # _____ Ex. Date _____ CK. ☐ Money Order ☐ Visa ☐ MC ☐

!!NOW AVAILABLE!!

INDEX SEQUENTIAL ACCESS METHOD

★ Get and Put Records to Disk File by "KEY"
★ Read File in Key Sequence Without Sorting
★ Delete Records Without Recopying File
★ Add Records to Disk Files in Any Sequence
★ Variable Key Length From 1 to 50 Characters
★ Machine Language or Basic Subroutines.

BUSINESS APPLICATION ADVANTAGES

- Improved Disk Utilization
- Easier Program Development
- Improved Operating Characteristics
- Reduce or Eliminate Sorting
- Improved Performance

ISAM SUBROUTINES **Documentation**
ISAM UTILITIES **On Diskette $90.00**

– PLUS – Free Mailing List Sample Application
Add 6% Sales Tax for California Orders

TRS-80 **MODEL I, II, & III** SOFTWARE FROM:

Johnson Associates -or- Telephone Order Line
P.O. Box 1402M ↙85 For Bank Card Sales
Redding, CA 96001 (916) 221-0740

WRITE FOR FREE CATALOG

```
7F74 0000      05010 INCR    DEFW    0
7F76 0000      05020 NXTLPT  DEFW    0
7F78 0000      05030 CURLIN  DEFW    0
7F7A 0000      05040 CURLPT  DEFW    0
7F7C 0000      05050 PEND    DEFW    0
7F7E 0D        05060 FINI    DEFB    13
7F7F 46        05070         DEFM    'FINI'
7F83 53        05080         DEFM    'SHED'
7F87 00        05090         DEFB    0
7F88 0D        05100 EMSG1   DEFB    13
7F89 4C        05110         DEFM    'LINE'
7F8D 20        05120         DEFM    ' # T'
7F91 4F        05130         DEFM    'OO L'
7F95 41        05140         DEFM    'ARGE'
7F99 00        05150         DEFB    0
7F9A 0D        05160 EMSG2   DEFB    13
7F9B 52        05170         DEFM    'RANG'
7F9F 45        05180         DEFM    'E VA'
7FA3 4C        05190         DEFM    'LUE '
7FA7 45        05200         DEFM    'ERRO'
7FAB 52        05210         DEFM    'R'
7FAC 00        05220         DEFB    0
7FAD 0D        05230 EMSG3   DEFB    13
7FAE 47        05240         DEFM    'GOTO'
7FB2 2F        05250         DEFM    '/GOS'
7FB6 55        05260         DEFM    'UB M'
7FBA 49        05270         DEFM    'ISSI'
7FBE 4E        05280         DEFM    'NG L'
7FC2 49        05290         DEFM    'INE '
7FC6 23        05300         DEFM    '#'
7FC7 00        05310         DEFB    0
7FC8 50        05320 EMSG4   DEFM    'PARA'
7FCC 4D        05330         DEFM    'METE'
7FD0 52        05340         DEFM    'R NO'
7FD4 54        05350         DEFM    'T NU'
7FD8 4D        05360         DEFM    'MERI'
7FDC 43        05370         DEFM    'C'
7FDD 0D        05380         DEFB    13
7FDE 00        05390         DEFB    0
7FDF 0D        05400 EMSG5   DEFB    13
7FE0 55        05410         DEFM    'UNDE'
7FE4 46        05420         DEFM    'FINE'
7FE8 44        05430         DEFM    'D LI'
7FEC 4E        05440         DEFM    'NE #'
7FF0 00        05450         DEFB    0
7FF1 1027      05460 POWERS  DEFW    10000
7FF3 E803      05470         DEFW    1000
7FF5 6400      05480         DEFW    100
7FF7 0A00      05490         DEFW    10
7FF9 0100      05500         DEFW    1
0005           05510 ASCIIN  DEFS    5
7B9B           05520         END     START
00000 TOTAL ERRORS
```


Subscription Problem?

80 Microcomputing does not keep subscription records on the premises, therefore calling us only adds time and doesn't solve the problem.

Please send a description of the problem and your most recent address label to:

80 Microcomputing
Subscription Dept.
PO Box 981
Farmingdale, NY 11737

Thank you and enjoy your subscription.

✓ Reader Service—see page 290

The disco version.

KBEEPFIX Revisited

Darell R. Whitehead
11 Patterson Rd.
Bedford, MA 01730

Those of us with disk-based TRS-80 systems often find it hard to modify and run machine language utility routines written for BASIC Level II machines, without disks. I tried to run Dennis Kitsz' KBEEPFIX (*80 Microcomputing*, February 1980, pp. 14-15) and had that problem.

KBEEPFIX jumps straight into the Level II BASIC initialization sequence. If one has a disk system, this jumps into BASIC2 instead of disk BASIC, and, unfortunately, BASIC2 doesn't have disk capabilities.

I found out how to modify KBEEPFIX to run with disk by adding five bytes of code to the beginning of the routine and

LOCATION	MACHINE LANGUAGE			Z-80 INSTRUCTION	
FF8A	3E	89		LD	A,89H
FF8C	32	49	40	LD	(4049H),A
FF8F	21	98	FF	LD	HL,0FF98H
FF92	22	16	40	LD	(4016H),HL
FF95	C3	2D	40	JP	402DH
FF98	21	36	40	LD	HL,4036H
FF9B	01	01	38	LD	BC,3801H
FF9E	16	00		LD	D,00H
FFA0	0A			LD	A,(BC)
FFA1	5F			LD	E,A
FFA2	A3			AND	E
FFA3	20	1A		JR	NZ,$+28
FFA5	77			LD	(HL),A
FFA6	14			INC	D
FFA7	2C			INC	L
FFA8	CB	01		RLC	C
FFAA	79			LD	A,C
FFAB	D6	80		SUB	80H
FFAD	20	F1		JR	NZ,$-13
FFAF	7E			LD	A,(HL)
FFB0	06	07		LD	B,07H
FFB2	2D			DEC	L
FFB3	86			ADD	A,(HL)
FFB4	10	FC		DJNZ	$-2
FFB6	FE	00		CP	00H
FFB8	3E	00		LD	A,00H
FFBA	C0			RET	NZ
FFBB	32	1A	40	LD	(401AH),A
FFBE	C9			RET	
FFBF	A6			AND	(HL)
FFC0	28	10		JR	Z,$+18
FFC2	3A	1A	40	LD	A,(401AH)
FFC5	3C			INC	A
FFC6	32	1A	40	LD	(401AH),A
FFC9	FE	FF		CP	0FFH
FFCB	20	D9		JR	NZ,$-37
FFCD	3D			DEC	A

LOCATION	MACHINE LANGUAGE			Z-80 INSTRUCTION	
FFCE	32	1A	40	LD	(401AH),A
FFD1	7B			LD	A,E
FFD2	73			LD	(HL),E
FFD3	C5			PUSH	BC
FFD4	01	00	02	LD	BC,0200H
FFD7	CD	60	00	CALL	0060H
FFDA	C1			POP	BC
FFDB	0A			LD	A,(BC)
FFDC	A3			AND	E
FFDD	C8			RET	Z
FFDE	C5			PUSH	BC
FFDF	E5			PUSH	HL
FFE0	F5			PUSH	AF
FFE1	06	40		LD	B,40H
FFE3	3A	3D	40	LD	A,(403DH)
FFE6	E6	FD		AND	0FDH
FFE8	67			LD	H,A
FFE9	F6	02		OR	02H
FFEB	6F			LD	L,A
FFEC	7D			LD	A,L
FFED	D3	FF		OUT	(0FFH),A
FFEF	7C			LD	A,H
FFF0	D3	FF		OUT	(0FFH),A
FFF2	C5			PUSH	BC
FFF3	06	40		LD	B,40H
FFF5	10	FE		DJNZ	$+0
FFF7	C1			POP	BC
FFF8	10	F2		DJNZ	$-12
FFFA	F1			POP	AF
FFFB	E1			POP	HL
FFFC	C1			POP	BC
FFFD	C3	FB	03	JP	03FBH

Note: For a 16K system, the program will occupy locations 7F8AH-7FFFH, and location FF91H is changed from FFH to 7FH. For a 32K, locations BF8AH-BFFFH are occupied, and location FF91H is changed to BFH.

Program Listing 1. KBEEPFIX for a 48K system. Machine language hex codes can be entered using DEBUG and Z-80 instructions by using Editor/Assembler.

changing two program bytes.

The first problem is getting KBEEPFIX to jump into a disk initialization sequence instead of BASIC2. The instruction currently in locations 7F95H-7F97H is a jump to address 1A19H, the address which begins BASIC2 initialization. If we change this jump to address 402DH, KBEEPFIX will transfer to the DOS READY. Thus, the machine language hex codes for program locations 7F95H-7F97H should be changed to read: C3 2D 40. KBEEPFIX can then be executed from DOS READY and will return to DOS READY.

This creates a second problem. Whenever disk BASIC is loaded, the highest 64 bytes of RAM are used by the TRSDOS BASIC loading sequence, but this is where we have stored KBEEPFIX. We have to fool the machine about where the top of memory (TOPMEM) is located, to protect our program. We do this by modifying address 4049H, the storage location for the LSB of TOPMEM. Location 404AH contains the MSB of TOPMEM, but we do not have to change that because TRSDOS loads that location with the correct value.

In a 16K machine, TOPMEM is 7FFFH (BFFFH in 32K, FFFFH in 48K) and the LSB is FFH (same in 32K and 48K). We need to change the LSB to point to the first address preceding KBEEPFIX.

The first available address preceding KBEEPFIX is 7F8EH (BF8EH in 32K, FF8EH in 48K) so the LSB of TOPMEM is 8EH. We could use DEBUG to go into RAM and modify location 4049H (which currently contains FFH for a 16K, 32K and 48K system) to read 8EH. Why not let KBEEPFIX do it for you? This is done by adding two instructions to the beginning of KBEEPFIX, which only take up five bytes of RAM. Note that this will also change our TOPMEM address from 7F8EH to 7F89H, so our LSB of TOPMEM becomes 89H.

The two instructions added to the beginning of KBEEPFIX will load 89H into the A register, then store A to location 4049H:

"I found out how to modify KBEEPFIX to run with disk by adding five bytes of code to the beginning of the routine and changing two program bytes."

Location	Hex Code	Instruction
7F8A	3E 89	LD 1,89H
7F8A	32 89	LD (A,89H)

The five bytes of hex code are all that need to be added at the indicated program locations (BF8AH and BF8CH for a 32K system, FF8AH and FF8CH for a 48K system). The highest 64 bytes of RAM will still be used by the TRSDOS BASIC loading sequence but they will be the 64 bytes before KBEEPFIX.

The new KBEEPFIX is only 118 bytes long and can be created using the DEBUG modify memory command.

Save it on disk with: DUMP KBEEPFIX/CMD (START = X'7F8A',END = X'7FFF',TRA = X'7F8A'). Now everytime DOS is rebooted (at powerup or reset), just type in KBEEPFIX and you have the debounce, repeat keys, and the audible beep in TRSDOS or disk BASIC.

An advantage to this is that you can still use the MEMORY SIZE? protection for other machine language programs. Make sure the machine language program you want to protect from BASIC doesn't overlay KBEEPFIX; i.e., make sure it doesn't use addresses beyond 7F89H (BF89H in 32K, FF89H in 48K).

You can now get real fancy and use the TRSDOS AUTO command to modify the power up sequence. This executes KBEEPFIX immediately at power up or reset. But, I am not that far in the DOS manual yet. ■

✓ Reader Service—see page 290

One-liners for your TRS-80.

Short and Sweet

Chris Gundlach
618 Hal Greer Boulevard
Huntington, WV 25701

Everyone has heard of the proverbial "one-liner," and those delivered by comedians are funny, usually. There are plenty of one-liners for the TRS-80 too, but they're useful. Try this one:

```
POKE 16396, 23
```

It turns off the BREAK key. Add a line number with this in your program, and you will no longer step out of a program by accidentally hitting BREAK. (TRSDOS 2.1 or NEWDOS+: POKE 23461,0; TRSDOS 2.3: POKE 23886.)

Of course, BREAK will remain off until you do the opposite one-liner:

```
POKE 16396, 201
```

If you forget, and LIST with the BREAK turned off, it isn't a funny one-liner at all. If you get stuck in a loop with the BREAK key off—only the rear reset will rescue you. (TRSDOS 2.1: POKE 23461,1; TRSDOS 2.3: POKE 23886,1.)

Here's another:

```
FOR F=0 TO 20:CSAVE "A":
FOR FF=0 TO 50:NEXT FF,F
```

This records your program 21 times on the tape, with a short space between each copy. Great for filling up those long audio cassettes with your favorite utility.

Our next one-liner does a bit of nonsense that seems to infatuate TRS-80 users.

```
CLEAR 300:FOR F=0 TO 3:?
STRING$(255,191):NEXT.
```

This one liner whites out the screen in a flash (almost).

Folks with the expansion box will like this one:

```
SYSTEM After the *?, answer /177.
```

It brings up MEMORY SIZE? and lets you reset the size. In case you forgot, like when you bbegin gettting double charracters because yyou forgot to load KBFIX.

This lets you reset memory size without disturbing the number of files available, under disk. Though the computer will print LEVEL II BASIC, you'll be in Disk BASIC, but will lose your program.

How to let your TRS-80 brag that you have Level II:

```
SYSTEM After the *?, answer /250.
```

Your computer says RADIO SHACK LEVEL II BASIC.

This one-liner may be the world's smallest text-editor and typing timer. Change the number in the FOR loop to set goals for yourself. How many words can you type in before READY comes back?

```
CLS:? CHR$(14);:FOR F=0 TO
20000: Z$=INKEY$:?Z$: NEXT
```

To build up your typing speed, reduce the FOR-loop number a little each time. Try the shift-arrow keys to watch the cursor. The CLEAR key erases from the cursor to the end of the screen in this micro text machine.

Running an LPRINT program without a line-printer? Or want to check your LPRINT output without wasting paper? We've got a one-liner for that too:

```
POKE 16422, PEEK(16414) :
POKE 16423, PEEK(16415).
```

It's crazy, but it works. Your line-printer output is directed to the video screen. It will print over text that may be there, but you can see what you're outputting.

This one-liner: POKE 16425,0 will start your substitute line-printer at the top of the screen. And this one: LPRINT CHR$(14) makes the phony line printer's cursor appear, giving you two cursors on the screen.

To restore normal line printer operation, another one-liner:

```
POKE 16422, 141 : POKE 16423, 5
```

If you're using a line printer machine language driver, you should substitute the appropriate address values when restoring locations 16422 and 16423. This is necessary if you use a non-standard TRS-80 printer without the built-in printer routine in the Level II ROM.

A FOR-NEXT loop written as a one-liner lets you test various values for timers or whatever:

```
FOR F=0 TO 5000:NEXT
```

Try adding the type declarer to see what difference integers make. Hit you stopwatch when you ENTER and stop it when READY reappears. These two one-liners run at different speeds:

```
FOR F%=0 TO 5000:NEXT

FOR F!=0 TO 5000:NEXT
```

Finally, there's nothing wrong with using GOSUB in a one-liner to test your subroutines (good programs, they say, are made up of purposeful subroutines). If your subroutine at line 15 is a timer, just tell your computer a one-liner:

```
GOSUB 15
```

The READY reappears when the routine is done.

More one-liners, anyone?■

STEREO HI-FI OWNERS
MUSIC COLLECTORS
RADIO PROGRAM MANAGERS

CUSTOM STORAGE AND RETRIEVAL SYSTEM

Now you can have instant access to any record or tape in your music collection with your TRS-80 MODEL I or III.

RECORD AND TAPE COLLECTION INVENTORY will store your collection on tape or disk. You will then be able to search for any record or group of records and list to the display or a printer. The collection can be sorted (alphabetically) for display, printing and storage. File editing allows quick and easy addition or deletion of records, and adding to or changing any individual record description. Complete error-trapping makes program crashes almost impossible.

The 16K TAPE version will conveniently handle up to 350 records and tapes per data file. An unlimited number of data files can be stored on tape. PRICE: $15.00

The enhanced 32K and 48K DISK versions are upgraded for heavy commercial use and can handle up to 500 and 1000 records per data file, respectively. Up to 1500 records can be stored on a one-drive system. Any number of drives can be used, allowing instant access to thousands of records. An unlimited number of data files can be stored on diskettes, allowing even a one-drive system convenient access to thousands of records. Many other improvements over the tape version. PRICE: $25.00

Programs provided on cassette; easily transferred to disk. Complete user's manual with examples.

Mailed First Class, include $1.00 S&H.
(Arkansas orders add 3% sales tax.)

Order from:

OZARK MOUNTAIN ✓308
SOFTWARE

FRANK L. WHITSELL, P.O. Box 2945, Fayetteville, Arkansas 72701 Ph. (501) 521-2801

TRS 80*
CANADIAN INCOME TAX 1980

CAN TAX1 Disk

A complete T1 form including schedules 1 to 10 and Provincial forms. Formulated printouts for all forms. Requires 32K and 1 Disk Drive.
$150.00

CAN TAX2 Tape

A complete T1 form only. No schedules. Requires 16K.
$80.00

TAX COURSE

A complete income tax course for home study, plus a complete listing of the tax program.
$200.00

NEW OLIVITTI ET-201

Daisey wheel, letter quality typewriter printer completely interfaced with any TRS 80*. No hardware required. 16K minimum.
$2799.00

All orders shipped FOB Warehouse
Ontario Residents add 7% SALES TAX
Terms: Cheque, Money Order, Visa, NO COD

J R Software ✓155
910 Wilson Ave.
Downsview Ont M3K 1E7
(416) 636-8690

Exact Replacement
TRS-80* PRINTER RIBBONS or CARTRIDGE RELOADS
(HEAVY INKING - LONGER LIFE)

TRS-80#	CENTRONICS#	PRICE/PACK
LP I (26-1413)	#779,700-704	11.00/3 PACK
LP II (26-1413)	#730	11.00/3 PACK
LP III (26-1414)		9.00/RELOAD
III (YOU INSERT IN CARTRIDGE)		21.00/3 PACK
LP IV (26-1413)	#737	11.00/3 PACK
LP VI (26-1418)		9.00/RELOAD
VI (YOU INSERT IN CARTRIDGE)		21.00/3 PACK
Daisy W II (26-1419 CARBON FILM)		21.00/3 PACK

SEND PRINTER NUMBER and/or OLD CARTRIDGES (for reloads)

MODEL II 8" GAME DISCS

VOLUME I	VOLUME 2
Biorhythms, Trap	Ugly, Bingo
Concentration	Towers, Blackjack
Rip Cord, Yacht Sea	Farkle, Chomp
Programmer's Nightmare	Piglatin Translator
Snoopy Picture	A. E. Newmann
Slot Machine	+ Others!

$24.95 per volume

SEND CHECK or MONEY ORDER TO:
BCCOMPCO ✓237
Box 246
Summersville, Mo. 65571
417 932-4196 *TANDY CORP TM

---

its...

SOFTWARE for the 80's

Programs for the full line of TRS-80* Microcomputers

MODEL I – Level II and DISK
I/O Utility, Inventory
Machine Language Retrieval

MODEL II – Reading Aids

MODEL III – Retrieval
I/O Utility
Machine Language Aids

COLOR – Games for Everyone
Quick Tac Toe
Asteroid Pilot & Tag

POCKET – Traveling Salesperson
Weight Forecasting
Estimating

B A S I C I O
Save Time and Space with this Machine Language Utility
Large Data Arrays on Tape or Disk up to 10 Times Faster!
Model I, Level II and Disk BASIC, ONLY $14.95

Reserve your Spring Catalog NOW!
its... ✓300
Information Technology Systems
POB 2667 Sarasota FL 33578
(813) 366 – 0064
*TRS-80 is a trademark of Radio Shack, a Tandy Corporation

TRS 80

INCOME TAX ESTIMATES
BAP$

INCOME TAX PROGRAM 8K

NEVER WONDER ABOUT TAXES AGAIN!

You input income & deductions, the program does the rest. Follows Fm. 1040, computes taxable income, self employment tax, capital gains/losses & taxes due or refund. All 4 tax tables written into program. Super program for tax return and spot check on where you stand during the year.

$19.95
CASSETTE 8K

Add $1.50 for shipping. Add $4.00 for disk.

BAP$ SOFTWARE ✓152
6221 Richmond Suite 220
Houston, TX 77057
(713) 783-3433

FINALLY... **SOFTWARE for the TRS-80* POCKET COMPUTER**

CALENDAR—Remembers your important dates and events with instant recall $14.95
GAMBLING—Bet on horse races, slot machines, roulette wheel and craps $9.95
TRAVEL COMPANION—Approximates MPG, ETA, average MPH, time to next fill up, etc. $9.95
EDUCATIONAL—Let the Pocket Computer teach your children math at various skill levels $9.95
INVENTORY—Take inventory in the palm of your hand $14.95
CHECKBOOK—Computerize your checkbook with hand held automation $9.95
GOLF—Play your friends or against par on Simulated Golf's constantly changing course $9.95

Write or call for a FREE PAMPHLET containing a description of these and many other amazing hand held programs (Scientific, Real Estate, Civil Eng., Horse Race Predicting, Games, Etc.). Prices from $5.95.

E.F. DREYER—*"A PIONER IN POCKET SOFTWARE"*

For more information or to order
CALL 1-313-553-4978
WRITE E. F. DREYER ✓488
P.O. Box 2391, Farmington Hills MI 48018
*TRS-80 is a trademark of Radio Shack, A Tandy Corp.

JOY·6

THE ULTIMATE JOYSTICK INTERFACE FOR YOUR TRS-80*

Don't be left out. The best action and simulation games require smooth, clean, jitter free joystick control. That's what you get with **JOY-6**, *and more!*

- 2 joysticks with pushbuttons for fast action.
- Sound effects capability. Great for space games.
- Simple software control—write your own programs.
- Manual—schematic, theory of operation, software.
- 2 additional A to D channels for game paddles, sensors.

PLUS: Joypak-1—A 16K L2 cassette with 6 fascinating new games — **SOLO, FRENZY, WIPEOUT, DOODLE, ENDPLAY** and **BLOCKADE** — an action and strategy game with 9 levels of difficulty. Try to break through your opponent's defenses. But watch out! While the enemy is rebuilding his blockade, he's destroying yours! Games can last for seconds or hours.

SPECIAL INTRODUCTORY OFFER! *ALL THE ABOVE, PLUS POWER SUPPLY— ASSEMBLED, TESTED, READY TO PLAY* ONLY **$99.95**
COMPLETE KIT—ONLY **$84.95** QUALITY GUARANTEED

MEGA SYSTEMS INC. ✓272
262 PARK LANE
KING OF PRUSSIA, PA 19406

PHONE ORDERS WELCOME
215-337-3876
C.O.D., CHECK, M.O., M.C., VISA
Add $2.00 postage & handling
Pa. Res. add 6% tax
*TRS-80 is a Trademark of Tandy Corp

---

word processor

a low-cost, yet powerful disk-based word processor

miniSCRIPT*

only $ 17.50

for your business and personal letters, termpapers...

- easy-to-use, menu-like command table
- store/retrieve text fast using your disk
- UPPER and lower case at printhead or
- support for lower case modifications
- selectable margin with automatic centering
- optional justification, page numbering
- flexible editing: global search and replace
- insert/delete/change line or character
- plus many more...

(for Model 1, Level 2 systems, 1 disk 32K min.)

ORDER by MAIL (check,MO,VISA,MC) or CHARGE IT on our 24-hour phone
overseas: please add $4.00 for air mail
N.J. residents please add 5% tax

micrOdome ✓248
P.O. Box 392, Denville, New Jersey 07834 USA
(201) 627-8554

SUPERIOR PRODUCTS FROM SSI

MIND-WARP - An exciting new program from SSI. If you like challenging, educational action that improves your memory skills, then MIND-WARP should be in your computer library. Enjoyment for the entire family.
Requires 32k-disk $13.95
Write SSI for a brochure of our SUPERIOR PRODUCTS.

EPSON MX-80 PRINTER - A printer that prints bidirectionally, upper & lower case, disposable print head, and adjustable tractors just to name a few of its features. If you need a high quality printer at a very reasonable price: The EPSON MX-80 is the one for you.
Only $599.95 + $25.00 shipping & handling

Memorex Diskettes - 5 1/4" soft sector, box of 10 diskettes: Only $29.95 + $1.00 handling & postage.

SUPERIOR SOFTWARE INC
P.O. BOX 11676
KANSAS CITY MO 64138

VISA master charge ✓403

MISSOURI RESIDENTS PLEASE ADD 4% STATE SALES TAX

Write SSI for information about marketing your programs.

Another super marketing application.

Shoplist

Hal Smith
1334 Keith Drive
Colorado Springs, CO 80916

I hate to go to the store!" Do these words sound familiar? The shoplist program I'm about to describe may not take the pain out of grocery shopping, but it will make preparation easier. It will also keep a running tab on what you have on hand.

The original version of this program was written by James McClure and was designed to run under the CP/M operating system. The version here has been re-written and expanded for the TRS-80, and is designed for a disk system. It could be reworked for systems using tape.

Operation

While instructions for Shoplist may appear complicated, the program is really simple. Operation is based on a master list of grocery items. Up to 500 items may be entered through the ENTER command. If your computer doesn't have enough memory for that many items, this is changed easily in lines 40 and 50. All commands have been INKEYed for ease of operation.

When running the program for the first time, the computer will ask if you already have a file created. If you answer N, it will prompt you for an item to be put on your master list. It will then proceed to create the Shoplist/Dat file. If you answer Y it will proceed normally. Warning: be sure to answer Y if you already have a file because otherwise it will write over the existing file and you'll have to start over.

As the names are entered, they are stored alphabetically on the master list, reprinted on the screen. Each item is assigned a number which appears in brackets next to the name.

These numbers are used for convenience so that the item name need not be typed out each time. They will change as items are entered, deleted and modified.

When first running Shoplist, enter the grocery items that your family regularly buys. After this, don't re-enter the items because they will be stored automatically when you quit the program.

If any mistakes are made while entering items, correct them by pressing the C (change) key after the item is entered. The computer will then ask for the code number of the item. If an illegal code is entered (a number less than one, or greater than the total items on the list), it will be ignored. Once the proper code is entered, you will be asked for the correct item, and the list will be reprinted.

To delete an item from the master list, press D and then the code number of the item to be deleted. This will cause the item to disappear from the master list.

Normally, there will be more items entered than can be simultaneously displayed on the screen. This is where the PAGE command comes in. A page consists of the total item names printed simultaneously on the screen. This has been set for two columns, seven items each, or fourteen items per page. The pages are numbered in the upper right hand corner of the screen with a running total of the number of items on the master list. If an item is not visible, press P followed by the page number desired, N for the next page, or P for the previous page. This allows you to "flip" pages. Since the names are alphabetical you will have a general idea of where they are.

As you run out of groceries, use the SELECT command to mark them. Just press the S followed by the code of the item. The computer will then ask for the quantity desired. Type the number of cans (or whatever) needed. Whenever the grocery list is printed the quantity needed will appear immediately in front of each item.

Shopping Trip

When the time comes for that dreaded trip to the market, use the LIST command. The computer will ask if a full or short list is desired. Press F for the full list. You will then be asked if the printer is on. A Y reply will result in a printed list of the required grocery items, in alphabetical order, preceded by the quantity needed. An N reply will cause the list to be printed only on the screen.

When finished, the RESET command may reset the quantities of items back to zero, or to however many need to purchased the next time. Pressing R will result in the computer asking if all items are to be reset. An N response will give you the selected items one at a time to re-

set as needed. A Y response will automatically reset all items to zero.

Sometimes a quick trip to a convenience store is in order. A list may not be necessary, but a short form of the LIST command is available. Following LIST, press S for the short form. You will be asked if any items from the master list are desired. Answer Y and you will be asked for the code number and quantity of the desired items; answer N and the next prompt will ask if you wish to enter any special items not appearing on the list. Here you enter the name and quantity desired. The computer will print out the short list on completion.

When you are finished with Shoplist and wish to terminate the program, just press the Q for QUIT. If there have been any changes in the master list, they will be automatically re-written as the data is saved back to disk. This way you won't lose any data changes, unless you BREAK out of the program.

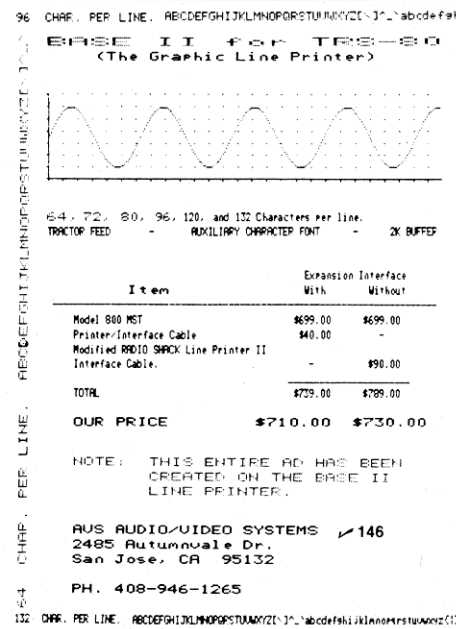
Most of the commands are self-prompting, so you won't have to refer to this list of instructions.■

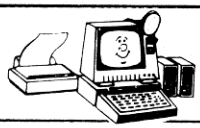
Program Listing

```
1 REM  ++++++++++++++++++++++++++++++++++
2 REM  +             SHOPLIST            +
3 REM  +                                 +
4 REM  +               BY                +
5 REM  +          JIM MCCLURE            +
6 REM  +                                 +
7 REM  +                                 +
8 REM  +                                 +
9 REM  +       TRS-80 REVISION BY        +
10 REM +           HAL SMITH             +
11 REM +               &                 +
12 REM +        JOHN BRESNAHAN           +
13 REM ++++++++++++++++++++++++++++++++++
14 REM
20 CLS:PRINT@468,CHR$(23);"SHOP  LIST"
30 FORX=1TO1500:NEXT
40 CLEAR1000:DEFINTA-Z
50 DIM ML$(500),ML(500),TL$(50),TL(50)
60 R$=STRING$(26," ")
70 F1$="(###) ## "
80 FB$="          ## "
90 F2$="(###) "
100 F3$="    ## "
110 F4$="      ## "
120 CLS:GOSUB4030
130 CLS:PRINT@468,CHR$(23);"SHOP  LIST"
140 REM  READ LIST FROM DISK
150 DATAC,D,E,L,P,S,Q,R
160 OPEN "I",1,"SHOPLIST/DAT"
170 N=1
180 J=1
190 INPUT #1,ML$(N),ML(N)
200 N=N+1
210 IF NOT EOF(1) THEN 190
220 I=N
230 CLOSE 1
240 START=1:REM  ASSIGN 1 AS PAGE TO BE PRINTED
250 COUNT=2*7:REM  NUMBER OF LINES OF ITEMS PRINTED (7
    HERE)
300 REM  MENU PRINT
310 V=0
320 CLS
330 G=INT((I-1)/14)
340 IFG<>((I-1)/14)THENG=INT((I-1)/14)+1ELSEG=INT((I-1)
    /14)
350 J=J:IFJ<1THENJ=1
360 IFJ>GTHENJ=G
370 PRINT"          MASTER LIST: ";I-1;" ITEMS FOR ";G;
    " PAGES          PAGE";J
380 PRINT"          FORMAT: (CODE #) (QTY NEEDED) (ITEM
    NAME)"
390 PRINT
400 GOSUB 2800
410 PRINT
420 PRINT@704," <C>HANGE "," <D>ELETE "," <E>NTER "," <
    L>IST ":PRINT" <P>AGE",
430 PRINT" <R>ESET "," <S>ELECT "," <Q>UIT"
440 FORY=1TO8:READB$(Y):NEXT
450 ONERRORGOTO470
460 GOTO490
470 RESUME480
480 FORT=1TO20:NEXTT
490 PRINT"YOUR CHOICE:  "
500 GOSUB4010:D=1
510 IFB$(D)=A$THEN520:ELSED=D+1:IFD<10THEN510ELSE500
520 ONDGOSUB1400,1900,600,1100,900,800,3100,1600
530 GOTO300
600 REM ENTER
610 INPUT"NAME OF ITEM TO ADD TO LIST ('0' TO ABORT)  "
    ;N$
620 IFN$="0"THEN300
630 FOR N=1 TO I-1
640 IF N$>ML$(N) THEN NEXT N
650 FOR N1=I TO N+1 STEP -1
660 ML$(N1)=ML$(N1-1)
670 ML(N1)=ML(N1-1)
680 NEXT N1
690 ML$(N)=N$: ML(N)=0
700 I=I+1
710 RETURN
800 REM  CHOOSE
810 INPUT"CODE NUMBER OF ITEM TO BE BOUGHT ('0' TO ABOR
    T)  ";C
820 IF C>=I OR C<0 THEN PRINTCHR$(27)CHR$(27):GOTO810
830 IFC=0THEN300
840 PRINTCHR$(27)CHR$(27):PRINT"WHAT QUANTITY OF ";ML$(
    C);
850 INPUT ML(C)
860 RETURN
900 REM  PAGE COMMAND
910 INPUT "WHAT PAGE  #  (ENTER N FOR NEXT, P FOR PREVI
    OUS, OR PG #) ";PA$
920 IFPA$="P"THENV=1:J=J-1:GOTO940
930 IF PA$="N" THENJ=J+1:GOTO970
940 IFV=1THENSTART=START-1*COUNT:GOTO990
950 J=VAL(PA$)
960 START=(VAL(PA$)-1)*COUNT+1:GOTO990
970 IF (START+1)+COUNT>I THEN RETURN
980 START=START+COUNT
990 IF VAL(PA$)>GTHENPRINTCHR$(27)CHR$(27):GOTO910
1000 IF I-COUNT<1 THEN RETURN
1010 IF START <1 THEN START=1
1020 IF START>I-COUNT THEN START=START
1030 RETURN
1100 REM  PRINT LIST TO PRINTER
1110 A$="":PRINTCHR$(27)CHR$(27):PRINT"DO YOU WANT THE
    FULL LIST OR A SHORT ONE (F OR S) ?  ";
1120 A$=INKEY$:IF A$="" THEN1120
1130 IF A$="S" THENPRINTA$:FORZ=1TO250:NEXT:GOTO2100
1140 IF A$="F" THENPRINTA$:FORZ=1TO250:NEXT:GOTO1160
1150 GOTO1120
1160 PRINTCHR$(27)CHR$(27):PRINT"IS THE PRINTER SWITCHE
    D ON ?  ";
1170 GOSUB 3000
1180 P=ANSWER
1200 CLS:PRINTTAB(22);"SHOPPING LIST"
1210 IF P THEN LPRINT TAB(32);"SHOPPING LIST"
1220 PRINT:IF P THEN FORK=1TO3:LPRINT" ":NEXTK
1230 FOR N=1 TO I-1
1240 IF ML(N)=0 THEN 1270
1250 PRINTUSING F1$;N,ML(N);:PRINTLEFT$(ML$(N)+R$,23);
1260 IF P THENLPRINTUSING FB$;ML(N);:LPRINTLEFT$(ML$(N)
    +R$,26);
1270 NEXT N
1280 IFNOTPTHENPRINT:PRINT:INPUT"HIT <ENTER> TO CONTINU
    E  ";AN$
1290 RETURN
1400 REM  CHANGE COMMAND
1410 INPUT"CODE NUMBER OF ITEM TO CHANGE ('0' TO ABORT)
    ";C
1420 IF C>=I OR C<0 THEN PRINTCHR$(27)CHR$(27):GOTO1410
1430 IFC=0THEN300
1440 PRINTCHR$(27)CHR$(27):PRINT"CHANGE ";ML$(C);" TO W
    HAT  ";
1450 INPUT N$
1460 Q=ML(C)
1470 GOSUB 1940: REM  DELETE PREVIOUS ENTRY
1480 GOSUB 630: REM  ENTER NEW ENTRY
1490 ML(N)=Q
1500 RETURN
1600 REM  RESET COMMAND
1610 PRINT"ARE ALL ITEMS TO BE RESET ? ";
1620 GOSUB 3000
1630 IF NOT ANSWER THEN 1680
1640 FOR N=1 TO I-1
1650 ML(N)=0
1660 NEXT N
1670 RETURN
1680 PRINT
1690 CLS:PRINT"HERE IS A LIST OF THE ITEMS YOU WERE TO
    BUY."
1700 PRINT"FOR EACH ITEM, ENTER A RETURN IF IT WAS PURC
    HASED,"
1710 PRINT"OR, IF NOT, THE QUANTITY REMAINING TO BE BOU
    GHT."
1720 PRINT
1730 FOR N=1 TO I-1
1740 IF ML(N)=0 THEN 1780
1750 PRINTCHR$(27)CHR$(27):PRINT@395, ML(N);"  ";ML$(N)
    ;
1760 ML(N)=0
1770 INPUT ML(N)
1780 NEXT N
1790 CLS:PRINT@463,"THERE ARE NO MORE ITEMS TO BE RESET
    ":FORX=1TO1500:NEXT:RETURN
1900 REM  DELETE
```

Program continues

```
1910 INPUT"CODE NUMBER OF ITEM TO DELETE ('0' TO ABORT)
     ";C
1920 IF C>=I OR C<0 THEN PRINTCHR$(27)CHR$(27):GOTO1910
1930 IFC=0THENGOTO300
1940 FOR N=C+1 TO I-1
1950 ML$(N-1)=ML$(N)
1960 ML(N-1)=ML(N)
1970 NEXT N
1980 I=I-1
1990 H=INT((I-1)/14)
2000 IFH<>((I-1)/14)THENRETURN
2010 START=START-14
2020 RETURN
2100 REM  TEMPORARY LIST
2110 TI=1
2120 COUNT=2*7:REM  NUMBER OF ITEMS TO PRINT
2130 PRINTCHR$(27)CHR$(27):PRINT"DO YOU WANT ANY ITEMS
     FROM THE MASTER LIST ?  ";
2140 GOSUB 3000
2150 CLS
2160 IF ANSWER THEN GOSUB 2400ELSEPRINT:PRINT:PRINT
2170 PRINT"DO YOU WISH TO TYPE IN ANY SPECIAL ITEMS ?
     ";
2180 GOSUB 3000
2190 IF ANSWER THEN GOSUB 2670
2200 PRINTCHR$(27)CHR$(27):PRINT"IS THE PRINTER SWITCHE
     D ON ?  ";
2210 GOSUB 3000
2220 P=ANSWER
2230 CLS:PRINTTAB(24);"QUICK LIST"
2240 IF P THEN LPRINT TAB(32);"QUICK LIST"
2250 PRINT:IF P THEN FORE=1TO3:LPRINT" ":NEXTE
2260 FOR N=1 TO TI-1
2270 PRINTUSING F3$;TL(N);:PRINTLEFT$(TL$(N)+R$,26);
2280 IF P THENLPRINTUSING F4$;TL(N);:LPRINTLEFT$(TL$(N)
     +R$,26);
2290 NEXT N
2300 IF NOT P THEN PRINT:PRINT:INPUT"HIT <ENTER> WHEN Y
     OU ARE FINISHED READING";AN$
2310 COUNT=2*7
2320 PA$="1"
2330 GOSUB 930
2340 RETURN
2400 REM  CLEAR SCREEN
2410 PRINTTAB(22);"QUICK LIST":PRINT:PRINT
2420 GOSUB 2800
2430 PRINT
2440 PRINT@704,"COMMANDS:";:PRINT@716,"<N>EXT PAGE";:PR
     INT@737,"<P>REVIOUS PAGE";:PRINT@780,"<S>TOP";:PRI
     NT@801,"<SPACE> TO ORDER"
2450 PA$=INKEY$:IFPA$=""THEN2480
2460 IFPA$="N"THEN2530
2470 IFPA$="P"THEN2540
2480 IFPA$=" "THEN2510
2490 IFPA$="S"THEN2550
2500 GOTO2150
2510 PRINTCHR$(27)CHR$(27):PRINTUSING"CODE NUMBER OF IT
     EM ##";TI;
2520 INPUT PA$:GOTO2550
2530 CLS: GOSUB 930:GOTO 2400
2540 CLS:V=1:GOSUB940:GOTO2400
2550 IFPA$="S"THENRETURN
2560 C=VAL(PA$)
2570 IF C>I-1 OR C<1 THEN 2510
2580 TL$(TI)=ML$(C)
2590 PRINTCHR$(27)CHR$(27):PRINT"WHAT QUANTITY OF ";TL$
     (TI);
2600 INPUT TL(TI)
2610 PRINTCHR$(27)CHR$(27):PRINT"SUBTRACT FROM MASTER L
     IST ?  ";
2620 GOSUB 3000
2630 IF ANSWER THEN ML(C)=ML(C)-TL(TI)
2640 IF ML(C)<0 THEN ML(C)=0
2650 TI=TI+1
2660 CLS:GOTO 2400
2670 PRINTCHR$(27)CHR$(27):PRINT"HIT <ENTER> TO STOP"
2680 PRINTUSING"NAME OF ITEM ##";TI;
2690 INPUT TL$(TI)
2700 IF LEN(TL$(TI))=0 THEN RETURN
2710 PRINTCHR$(27);"WHAT QUANTITY OF ";TL$(TI);
2720 INPUT TL(TI)
2730 TI=TI+1
2740 PRINTCHR$(27)CHR$(27):GOTO 2680
2800 REM  PRINT LIST
2810 FOR N=START TO START+COUNT-1
2820 IF ML(N)=0 THEN 2840
2830 PRINTUSING F1$;N,ML(N);:PRINTLEFT$(ML$(N)+R$,23);:
     GOTO2850
2840 PRINTUSING F2$;N;:PRINTLEFT$(ML$(N)+R$,26);
2850 IF N<I-1 THEN NEXT N
2860 RETURN
2900 REM  UPDATE DATA FILE
2910 OPEN"O",1,"SHOPLIST/DAT
2920 FOR N=1 TO I-1
2930 PRINT#1,CHR$(34);ML$(N);CHR$(34);",";ML(N)
2940 NEXT N
2950 CLOSE 1
2960 RETURN
3000 REM  YES/NO ANSWER
3010 ANSWER =1
3020 AN$=INKEY$:IFAN$=""THEN3020
3030 IF AN$="Y" THEN ANSWER=-1:PRINTAN$:FORZ=1TO250:NEX
     T:RETURN
3040 IF AN$="N" THEN ANSWER=0:PRINTAN$:FORZ=1TO250 NEXT
     :RETURN
3050 GOTO3020
3100 REM  QUIT
3110 GOSUB 2900
3120 END
4000 REM DATA FILE CREATION
4010 A$=INKEY$:IFA$=""THEN4010:RETURN
4020 RETURN
4030 A$="":PRINT"DO YOU ALREADY HAVE A DATA FILE CREATE
     D ?  ";
4040 A$=INKEY$:IFA$=""THEN4040ELSEPRINTA$:FOR Z=1 TO 25
     0:NEXT
4050 IF A$="Y" THEN RETURN
4060 IF A$="N" THEN 4080
4070 GOTO 4040
4080 CLS:PRINT"NOTE: THIS SECTION WILL CREATE YOUR SHOP
     LIST DATA FOR ONE ITEM."
4090 PRINT"AFTER YOU PLACE IN THE FIRST ITEM, THE PROGR
     AM WILL THEN START"
4100 PRINT"RUNNING."
4110 PRINT:PRINT:PRINT
4120 INPUT"NAME OF AN ITEM TO PLACE ON YOUR SHOPPING LI
     ST ";N$
4130 ML$(N)=N$:ML(N)=0
4140 I=I
4150 OPEN"O",1,"SHOPLIST/DAT"
4160 N=0
4170 PRINT#1,CHR$(34);ML$(N);CHR$(34);",";ML(N)
4180 CLOSE
4190 RETURN
```





THE MICRO CLINIC

✓214

EVERY MODEL I DISK SYSTEM NEEDS A FLOPPY DOCTOR!

Double-density recording (using the Percom Doubler**) and 77/80 track drives put greater demands on your TRS-80* and make it imperative that you assure yourself of data integrity BEFORE committing it to disk. THE FLOPPY DOCTOR 3.0 is the **only** machine language diagnostic which can test your Model I in either single density or, if you have a Doubler**, in double density. This program is a comprehensive test of every sector on every track of 35, 40, 77, or 80 track drives, single or double sided; checks drive motor speed; use your computer as an intelligent disk exerciser; MORE! Also includes memory diagnostic and 16-page instruction manual.

*TM Tandy Corp. **TM Percom Data Co.

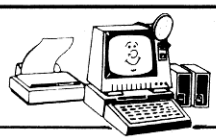
VISA®

PRICE: $25.95
Includes P/H
CA Add 6% Tax

On diskette for a min. 16K single disk system. Drive 0 must be 35 or 40 tracks. VISA/MC include card #, exp. date, sign.

master charge

THE MICRO CLINIC • 17375 Brookhurst • Suite 114 • Fountain Valley, CA 92708 • 714-962-0621



THE MICRO CLINIC ✓214

DUAL CHARACTER SETS FOR YOUR CENTRONICS 779/RS PRINTER I

Expand the capabilities of your Centronics 779/Radio Shack Printer I with our CLC-1 Lowercase Adapter. The CLC-1 gives you the option of selecting between 2 different UPPERCASE/lowercase character sets so you can choose the print style that is best for your application, all at a fraction of the cost of a new printer. Send for a free sample printout and discover what your Centronics 779/RS Printer I can do for your word processing needs. We think you'll find that its dual character sets, easy installation, and FAST delivery make the CLC-1 an outstanding value!

master charge

CLC-1 PRICE: $99 A/T. Includes P/H. CA add 6% tax.
VISA/MC include card number, signature, exp. date, phone number.
MC include interbank number.

VISA®

THE MICRO CLINIC • 17375 Brookhurst • Suite 114 • Fountain Valley, CA 92708 • (714) 962-0621

STOP

DON'T THROW AWAY YOUR BACK ISSUES OF "80 MICROCOMPUTING"*

TURN THEM INTO A VALUABLE REFERENCE USING YOUR TRS-80†AND INDEX 80

INDEX-80 Instantly searches all 1980 issues of 80 Microcomputing* for hundreds of subjects. Lists month, page no., and title of all articles on desired subject(s) Specify LV II or disk basic $16.95

OVERRUN YOUR PAYCHECK? Use your TRS-80† to control your monthly expenditures.

BUDGET 80 Will record all family expenditures into accounts and compare each with budgeted amounts to any date. Indicates projected over-expenditures before they occur $16.95

NOW USE YOUR TRS-80† TO PLOT WHAT YOU WANT WITH - - -

CURVPLOT Rapidly plots nearly any user defined function. User controls range of X and Y and program labels both axes. Excellent for instruction $16.95

CURVFIT Determines coefficients of all polynomials up to 14th degree through a large no. of data points (limited by memory size). Data points may be input in any order. Program tabulates correlation coefficients for selection of best fit $16.95

Any two programs for $29.95

System requirements: 16K, LVII, or 32K, Disk, except *Budget 80* only 32K, 48K, Disk.

VISA

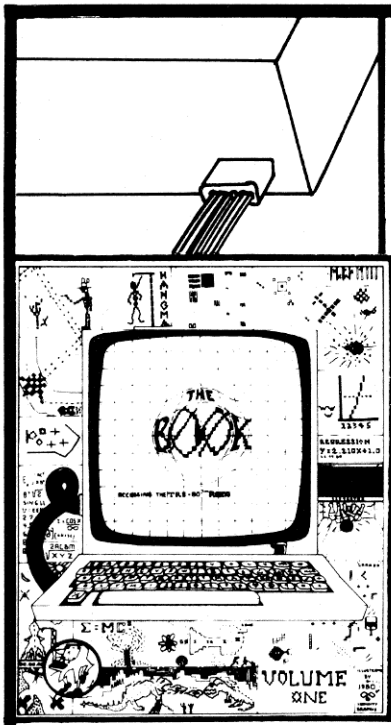
Mail or Phone (904) 897-3741
FL. residents add 4% sales tax
Foreign orders add $2.00

master charge

✓451

MS ENTERPRISES
P.O. Box 596 Niceville, FL 32578
MICROCOMPUTER TECHNOLOGY AND SOFTWARE

*TM Wayne Green Inc. † TM Tandy Corp.



THE BØØK

TRS-80™ ROM

If you ever do Assembly language programming, or you just want to know more about your TRS-80 ROM, "THE BOOK"s are for you.

*TRS-80 is a trademark of Tandy Corp.

Each volume is priced at: $14.95 + $1.50 S&H = $16.45 ($17.05 in VA) Overseas add $2.50 for air shipment

Insiders Software Consultants, Inc. ✓305
P.O. Box 2441, Dept. SUM2
Springfield, VA 22152

☐ **Please send me Volume I of THE BOOK**

☐ **Please send me Volume II of THE BOOK**

NAME _____

ADDRESS _____

CITY, STATE _____ ZIP CODE _____

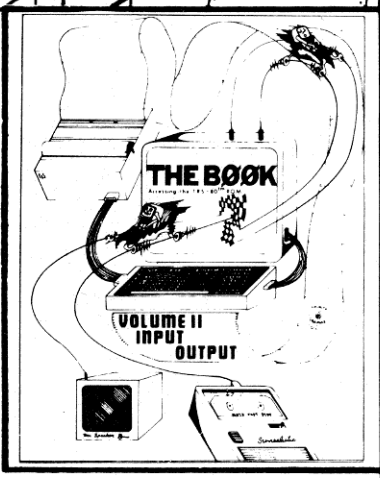
[ ] Check payable to Insiders Software Consultants, Inc.

[ ] MASTER CHARGE MC Bank Code

[ ] VISA Exp. Date _____ Card Number _____

Signature _____

Dealer Inquiries Invited



Volume I: The most comprehensive book yet on the math routines in Level II ROM, Models I & III. Includes a fully commented listing from 0708H to 1607H and an incredibly complete map of the ROM and reserved areas of RAM.

Volume II: Everything you want to know about the video, keyboard, cassette, and printer driver routines. Learn how to write your own! Remarkably detailed listings illustrate well-commented source code. Complement Volume I, now.

A game for rolling stones.

Compukala

Peter K. Moller
126 Jamesville Ave. J-4
Syracuse, NY 13210

Today the game Kala is played on an oblong or circular surface about the size of a bread board, but the game has a long history. Even before the Phoenicians were reading and writing, burly competitors were rolling boulders around huge open fields in the name of Kala.

The latest version of the game, which I call Compukala, fits nicely on the CRT of the TRS-80 and the opposition can be the not-so-burly TRS-80 itself.

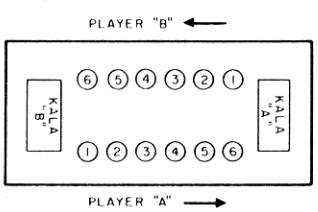


Fig. 1. Geography of the Kala Board. Moves are made in a counter-clockwise direction.

Kala: Pips in Pits

To understand how the game is played, you must first be familiar with the geography of the board. (See Fig. 1.) On each side of the board are six indentations, called pits. They are numbered one to six in opposite directions. At each end of the board are larger indentations —one for each player—which are called the kala.

When playing on a board, the pips or playing pieces can be any small, simple objects like matches, coins, or dried beans. At the beginning of a round, each player has the same number of pips in each of his six pits. The two kala are empty. The players determine between themselves how many pips will be in each pit at the start.

The complexity of the game varies depending on the number of the pips used.

Moves are made by picking up all the pips in a pit and placing one in each successive pit, including the Kala, counterclockwise, until all the pips are re-deposited. Often, you'll find yourself placing pips from your side of the board into your opponent's pits. This is, in fact, part of the strategy of Kala.

The object of the game is for one player to accumulate as many pips as possible on his side of the board *and* in his kala. A round is over when either player cannot move because he has no pips on his side of the board. Scores are determined by the total number of pips each player retains at the end of a round. The end of the match occurs when one player accumulates a mutually agreed upon total of pips.

The rules of Kala are surprisingly simple for a game which requires the strategy of backgammon and returns the endless permutations of cribbage. There are only three rules:

- A repeat turn is earned when the last pip of any player's move lands in *his* kala.
- If a player's last pip lands in an empty pit on his side of the board, any pips in the pit directly opposite are captured and deposited in the captor's kala along with the capturing pip.
- Once a pip is placed in a kala, it remains there until the end of the round.

Compukala: No Pips, No Pits

When this computer version of Kala is loaded and run, you are asked to enter the number of pips that will determine the end of a match. This can be any number depending on how many rounds of the game you wish to play. Next, enter the number of pips to start the round. In this version the limits are between two and five to allow for beginning and advanced levels of play. The two players are then asked to enter their names. If you wish to play against the computer, the word *computer* should be entered as player number two.

The player to make the first move is determined by a random number. From that point on in the round, plays are made in turn. Each player enters his move by inputting the pit number from which he wishes to move. After each turn, the computer displays the distribution of the board and updates scores. Illegal moves (such as moving from an empty pit or entering a number larger than six) are trapped and the

player is prompted to make another entry.

If at any time you wish to refresh your memory about the pit number designations, enter the letter P and the board numbers will be displayed. Entering a P again will return the display to the pip distribution mode.

Compukala is written in Disk BASIC and requires less than 16K of RAM. Error traps and features which provide a clear and symmetrical display can be deleted if memory overhead is critical. The residue function defined in line 80 gives the game its circular nature. Once the 14th position of the vector A(14) has been reached, the first position fo the vector is returned.

If you decide to play against the computer, you will find a low level of intelligence. The computer knows only the rules of the game and its only strategy will be to attempt to empty pits four, five, and six in the first two moves. Additional strategies could be programmed once you discover the subtleties of the game.

"Even before the Phoenicians were reading and writing, burly competitors were rolling boulders around huge open fields in the name of Kala."

The word kala is taken from Jainist philosophy, a reformist sect of Hinduism founded in the sixth century B.C. It signifies time, the eternal aspect of existence. There couldn't be a better name for this game: you'll find its pleasures are endless.■

Program Listing. Compukala

```
                    **INITIALIZE
50 '
60 CLS:CLEAR 100
70 DIM A(14)
80 DEF FNRES(A,B)=A-(INT(A/B)*B)
90 GOSUB 740
100 PRINT@130,"ENTER MAXIMUM NUMBER OF PIECES FOR END O
    F MATCH----->";:INPUT PS
110 PRINT@130,"ENTER NUMBER OF PIECES (BETWEEN 2 AND 5)
    TO BEGIN----->";:INPUT P
120 IF P>5 OR P<2 PRINT@70,CHR$(30):GOTO 110
130 GOSUB 740  :GOSUB 1970
140 IF G=>1 GOTO 240
150 '
160 '                    **INPUT PLAYER NAMES
170 '
180 PRINT@128,CHR$(30)
190 PRINT@141,"ENTER NAME OF PLAYER #1==>";:LINEINPUT N
    $(0)
200 PRINT@141,CHR$(30)
210 PRINT@141,"ENTER NAME OF PLAYER #2==>";:LINEINPUT N
    $(1)
220 PRINT@141,CHR$(30)
230 '
240 '                    **DETERMINE FIRST MOVE
250 '
260 I=RND(2)-1
270 GOTO 350
280 '
290 '                    **EXCHANGE PLAYERS
300 '
310 IF I=1 THEN I=0 ELSE I=1
320 '
330 '                    **GOSUB BOARD ROUTINE AND SCORES
340 '
350 GOSUB 790  :GOSUB 920
360 '
370 '                    **INPUT AND COMPUTE MOVES
380 '
```

Program continues

```
390 IF N$(1)="COMPUTER" AND I=1  GOSUB 1710
400 PRINT@75,"      ENTER NUMBER OF PIT TO EMPTY.
    ";
410 M$=INKEY$:IF M$="" GOTO 410
420 PRINT@64,CHR$(30);:PRINT@192,CHR$(30);
430 IF M$="P" GOSUB 1000 :GOTO 350
440 PRINT@78,"      ";N$(I);" MOVES FROM PIT # ";M$;
450 M=VAL(M$)
460 IF M>6 OR M<1 THEN GOSUB 1180
470 IF I=1 THEN M=M+7
480 IF A(M)=0 THEN GOSUB 1180
490 IF I=0 GOSUB 1860 ELSE GOSUB 1910
500 D=FNRES(M+A(M),14)
510 A(M)=0
520 C=M
530 C=C+1
540 IF C>13 THEN C=0
550 A(C)=A(C)+1
560 IF C<>D THEN 530
570 GOSUB 790
580 '
590 '                    **CHECK FOR ENDGAME
600 '
610 IF I=0 AND C>=7 OR I=1 AND C<=7 OR C=0 OR C=7 GOTO
    630
620 IF A(C)=1 AND A(14-C)=>1 GOSUB 1240
630 IF A(1)+A(2)+A(3)+A(4)+A(5)+A(6)=0 GOSUB 1380
640 IF A(8)+A(9)+A(10)+A(11)+A(12)+A(13)=0 GOSUB 1380
650 '
660 '                    **CHECK FOR REPEAT TURN
670 '
680 IF C=0 OR C=7 GOSUB 1650
690 '
700 '                    **NEXT TURN
710 '
720 GOTO 310
730 '
740 '                    **PAINT BORDER
750 '
760 PRINT@0,STRING$(63,191);:PRINT@256,STRING$(63,191);
    :PRINT@384,STRING$(63,179);
770 RETURN
780 '
790 '                    **DRAW THE BOARD
800 '
810 PRINT@205,"  =ENTER 'P' TO SEE PIT NUMBERS=  ";
820 PRINT@710,A(0);:PRINT@754,A(7);
830 PRINT@847,A(1);:PRINT@852,A(2);:PRINT@857,A(3);:PRI
    NT@862,A(4);:PRINT@867,A(5);:PRINT@872,A(6);
840 PRINT@616,A(8);:PRINT@611,A(9);:PRINT@606,A(10);:PR
    INT@601,A(11);:PRINT@596,A(12);:PRINT@591,A(13);
850 IF I=0 PRINT@921,N$(0);"--->"; ELSE GOTO 880
860 PRINT@535,CHR$(30);
870 GOTO 900
880 PRINT@535,"<---";N$(1);
890 PRINT@921,CHR$(30);
900 RETURN
910 '
920 '                    **KEEP SCORES
930 '
940 T1=0:T2=0
950 T1=A(1)+A(2)+A(3)+A(4)+A(5)+A(6)+A(7)
960 T2=A(8)+A(9)+A(10)+A(11)+A(12)+A(13)+A(0)
970 PRINT@330,N$(0);"'S SCORE=";T1;:PRINT@353,N$(1);"'S
    SCORE=";T2;
980 RETURN
990 '
1000 '                    **DISPLAY BOARD NUMBERS
1010 '
1020 PRINT@711,"KALA";:PRINT@753,"KALA";
1030 FOR W=847 TO 872 STEP 5
1040 Q=Q+1
1050 PRINT@W,Q;
1060 NEXT W
1070 U=7
1080 FOR W=591 TO 616 STEP 5
1090 U=U-1
1100 PRINT@W,U;
1110 NEXT W
1120 Q=0
1130 PRINT@73,"   TO RETURN TO GAME BOARD....ENTER 'P'.
    ";
1140 E$=INKEY$:IF E$="" GOTO 1140
1150 PRINT@704,CHR$(30)
1160 GOTO 790
1170 '
1180 '                    **ILLEGAL MOVE ROUTINE
1190 '
```

Program continues

MAKE $$$$$ $$$$ MONEY

Selling 80 Microcomputing, the only major journal for the users of the TRS-80*, is a sure bet for getting the computer enthusiast into your store. Once through the door you can sell him anything.

We know "80" will make you money . . . it's the only magazine for the TRS-80* users and you know how many of those there are. So call today and join the dealers who make money with "80".

For information on selling 80 Microcomputing, call 603-924-7296 and speak with Ginnie Boudrieau, our Bulk Sales Manager. Or write to her at 80 Microcomputing, Pine Street, Peterborough, NH 03458.

*TRS-80 is a trademark of the Tandy Corp.

DISK BASED WORD PROCESSOR 7.50

A complete word processing system for your TRS-80.
Provides full editing capability including paragraph move, line deletion, insertion and correction.
Store text on disk, print business/personal letters, reports with numbered pages and title pages.
Text is stored on disk as blocks are created so texts are not limited by the available memory.
Requires 16K and one or more disk drives.
Comes complete on cassette with software to produce upper/lower case at printhead, and keyboard reverse.
Full right/left justification and much more.
Send cheque, money order or order by phone, 24 hours, 7 days a week. Mastercharge and Visa cards welcome.
Please include $1.00 extra for first class post.

MAILING LIST OPTION 7.50 AVAILABLE NOW!

A complete mailing list option for the owners of the Pensa-Write Word Processor.
Capacity for 300 names per disk.
Sorts by postal code, prints business letters (created by Pensa-Write) against mailing list or portion thereof depending on selectable criteria.
Prints labels in user definable format: Any number of columns (up to 5) across the page, any tab postions, and any number of spaces between rows.
Interfaces directly with Pensa-Write Software to form the complete word processing/mailing system.
Requires 32K and one disk drive.
Please note: Pensa-Mail is not a "stand alone" program and is designed to be used with Pensa-Write.

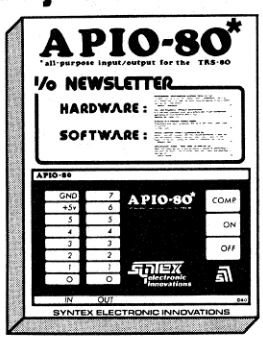
Both systems available on Diskette for $19.95

PENSADYNE ✓207

MasterCard VISA

4441 WEST FIRST AVE.
VANCOUVER, B.C., V6R 4H9
604-224-3107

EXCITING NEW uses for the TRS-80*



all purpose INPUT/OUTPUT PORT + teaching newsletter

- **the most useful peripheral**
- **don't spend more for less** **$45**

→ **8 outputs 6 inputs**
→ **assembled & tested PC board**
→ **customizing kit included**
→ **LII Basic or LI T-Bug**
→ **WE TEACH YOU HOW**

with the free newsletter subscription you will learn how to use the TRS-80 for:

* home security * robotics * test equipment simulation * RTTY * printer interface * remote control * EKG * human body interfacing * energy controllers * amateur radio * computer games * model railroad * computer music *

✓139 and a multiplicity of other uses WE don't just say it, we DO it!

INSTANT-MAIL RECORDING LINE
for more info. call 717/733-4769
SYNTEX electronic innovations
Box 4034 LANCASTER PA 17604

*TRS-80 IS A TRADEMARK OF TANDY CORP. (C) 80 BY SYNTEX ELECTRONIC INNOVATIONS

PPI-80

PARALLEL I/O FOR THE TRS-80

*See Article in Sept. Issue.

The PPI-80 is a complete parallel I/O interface designed specifically for the TRS-80, consisting of 3 complete 8 bit I/O ports including such features as:

- switch selectable address decoding
- complete on board regulated power supply
- TTL compatible I/O lines conveniently available through 16 pin sockets
- +5 volts and ground at each socket
- 3 software selectable modes of operation
- handshaking
- plugs into keyboard or expansion interface
- on board kluge area for experimenting
- provisions for interfacing Sears-BSR-RS home controller

Possible applications include:

- bidirectional communication between microcomputers
- parallel printer interface
- wireless home control via BSR home controller
- direct control of lights, appliances, and motors
- interfaces to many popular boards including A/D-D/A converter and an EPROM Programmer

PPI-80 is available now and can be purchased in several forms

Completely assembled and tested	$119.95
Complete kit with all parts	89.95
Bare board drilled and etched with assembly manual	25.95
Accessories:	
8 channel A/D - 2 channel D/A by Optimal Technology	$115.00
EPROM Programmer Model EP-2A-79 by Optimal Technology	155.00

To order, send payment plus $2.00 shipping and handling to:
QUANT SYSTEMS
P.O. BOX 628
CHARLESTON, S.C. 29402

S C residents add 4% sales tax
Overseas orders add $5 for shipping

✓269 Quant Systems **p.o. box 628 charleston sc 29402**

```
1200 PRINT@78,"ILLEGAL MOVE.  PLEASE ENTER AGAIN!!!";
1210 FOR TT=1 TO 550:NEXT
1220 GOTO 350
1230 '
1240 '                    **CAPTURE ROUTINE
1250 '
1260 FOR TT=1 TO 50
1270 PRINT@728,"**CAPTURE**";
1280 PRINT@728,"           ";
1290 NEXT TT
1300 IF I=1 GOTO 1340
1310 A(7)=A(C)+A(14-C)+A(7)
1320 A(C)=0:A(14-C)=0
1330 GOTO 1360
1340 A(0)=A(C)+A(14-C)+A(0)
1350 A(C)=0:A(14-C)=0
1360 GOTO 790
1370 '
1380 '                  **END OF ROUND ROUTINE
1390 '
1400 G=G+1
1410 FOR TT=1 TO 50
1420 PRINT@711,"   <<  GAME IS OVER!  HIGH SCORE WINS!
     >>";
1430 PRINT@711,CHR$(30);
1440 NEXT TT
1450 GOSUB 920
1460 PRINT@80,"           ROUND";G;"
     ";
1470 IF T1>T2 THEN N4$=N$(0) ELSE N4$=N$(1)
1480 PRINT@330,N$(0);" ENDS WITH:";T1;:PRINT@353,N$(1);
     " ENDS WITH:";T2;
1490 TS=T1+TS:TQ=T2+TQ
1500 PRINT@133,N$(0);" PIECES WON=";TS;:PRINT@165,N$(1)
     ;" PIECES WON=";TQ;
1510 FOR TT=1 TO 100
1520 PRINT@206,N4$;" WON THIS ROUND OF THE GAME!!";
1530 PRINT@206,CHR$(30);
1540 NEXT TT
1550 IF TS=>PS OR TQ=>PS GOTO 1610
1560 PRINT@717,"TO PLAY ANOTHER ROUND, HIT 'ENTER'.";:L
     INEINPUT E$
1570 K=0:CLS:GOTO 130
1580 '
1590 '                  **END OF MATCH ROUTINE
1600 '
1610 IF TS>TQ THEN N4$=N$(0) ELSE N4$=N$(1)
1620 PRINT@711,"    ";N4$;" HAS WON THE KALA MATCH!  CO
     NGRATS!          ";
1630 PRINT@970,"FOR ANOTHER MATCH, SIMPLY HIT ENTER.";:
     LINEINPUT Z$
1635 RUN
1640 '
1650 '                  **REPEAT TURN ROUTINE
1660 '
1670 IF N$(1)="COMPUTER" AND I=1  GOTO 1750
1680 PRINT@78,"     ";N$(I);", TAKE ANOTHER TURN!!   ";
1690 GOTO 410
1700 '
1710 '                   **THE COMPUTER'S MOVES
1720 '
1730 PRINT@205,CHR$(30);
1740 K=K+1
1750 M=RND(14)
1760 IF K<3 AND M<11 GOTO 1750
1770 IF M=>1 AND M<=7 GOTO 1750
1780 IF A(M)=0 GOTO 1750
1790 PRINT@76,"     ";N$(1);" MOVES FROM PIT #";M-7;
1800 GOSUB 1910
1810 FOR TT=1 TO 50:NEXT TT
1820 GOTO 500
1830 '
1840 '                  **MOVE INDICATORS ROUTINE
1850 '
1860 FOR TT=1 TO 50
1870 PRINT@844+(M*5),">";
1880 PRINT@844+(M*5)," ";
1890 NEXT TT
1900 RETURN
1910 FOR TT=1 TO 50
1920 PRINT@621-(M*5)+35,"<";
1930 PRINT@621-(M*5)+35," ";
1940 NEXT TT
1950 RETURN
1960 '
1970 '                  **INITIALIZE POSITIONS ROUTINE
1980 '
1990 FOR X=0 TO 13:A(X)=P:NEXT
2000 A(0)=0:A(7)=0
2010 RETURN
```

80 microcomputing T.M. bookshelf

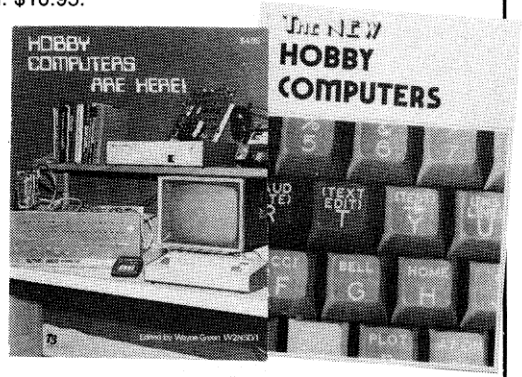
3 new books from the editors of KB & 80 Microcomputing

● **40 COMPUTER GAMES**—BK7381—Forty games in all in nine different categories. Games for large and small systems, and even a section on calculator games. Many versions of BASIC used and a wide variety of systems represented. A must for the serious computer gamesman. $7.95*

● **UNDERSTANDING AND PROGRAMMING MICROCOMPUTERS** – BK7382 – A valuable addition to your computing library. This two part text includes the best articles that have appeared in 73 and Kilobaud Microcomputing magazines on the hardware and software aspects of the new microcomputing hobby. Well known authors and well structured text helps the reader get involved in America's fastest growing hobby. $10.95*

● **SOME OF THE BEST FROM KILOBAUD/MICROCOMPUTING**—BK7311—A collection of the best articles that have recently appeared in Kilobaud/MICROCOMPUTING. Included is material on the TRS-80 and PET systems, CP/M, the 8080/8085/Z80 chips, the ASR-33 terminal. Data base management, word processing, text editors and file structures are covered too. Programming techniques and hardcore hardware construction projects for modems, high speed cassette interfaces and TVTs are also included in this large format, 200 plus page edition. $10.95.*

INTRODUCTORY



● **THE NEW HOBBY COMPUTERS**—BK7340—This book takes it from where "HOBBY COMPUTERS ARE HERE!" leaves off, with chapters on Large Scale Integration, how to choose a microprocessor chip, an introduction to programming, low cost I/O for a computer, computer arithmetic, checking memory boards . . . and much, much more! Don't miss this tremendous value! Only $4.95.*

● **HOBBY COMPUTERS ARE HERE!**—BK7322—If you (or a friend) want to come up to speed on how computers work . . . hardware and software . . . this is an excellent book. It starts with the fundamentals and explains the circuits, and the basics of programming. This book has the highest recommendations as a teaching aid for newcomers. $4.95.*

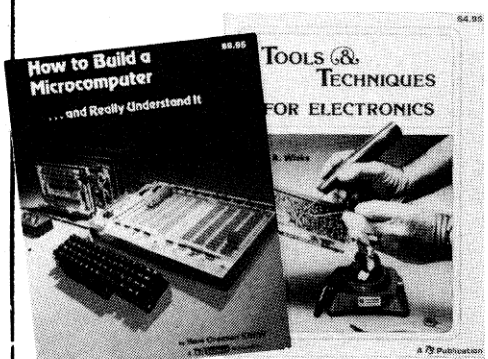
INTRODUCTION TO MICROCOMPUTERS (VOL. 0→III)

● **AN INTRODUCTION TO MICROCOMPUTERS, VOL. 0**—BK1130—The Beginner's Book—Written for readers who know nothing about computers—for those who have an interest in how to use computers—and for everyone else who must live with computers and should know a little about them. The first in a series of 4 volumes, this book will explain how computers work and what they can do. Computers have become an integral part of life and society. During any given day you are affected by computers, so start learning more about them with Volume 0. $7.95.*

● **VOL. I**—BK1030—**2nd Edition completely revised.** Dedicated to the basic concepts of microcomputers and hardware theory. The purpose of Volume I is to give you a thorough understanding of what microcomputers are. From basic concepts (which are covered in detail), Volume I builds the necessary components of a microcomputer system. This book highlights the difference between minicomputers and microcomputers. $12.99.*

● **VOL. II**—BK1040 (with binder)—Contains descriptions of individual microprocessors and support devices used only with the parent microprocessor. Volume II describes all available chips. $31.99*

● **VOL. III**—BK1133 (with binder)—Contains descriptions of all support devices that can be used with any microprocessor. $21.99*



● **HOW TO BUILD A MICROCOMPUTER – AND REALLY UNDERSTAND IT** – BK7325 – by Sam Creason. The electronics hobbyist who wants to build his own microcomputer system now has a practical "How-To" guidebook. This book is a combination technical manual and programming guide that takes the hobbyist step-by-step through the design, construction, testing and debugging of a complete microcomputer system. Must reading for anyone desiring a true understanding of small computer systems. $9.95.*

● **TOOLS & TECHNIQUES FOR ELECTRONICS** – BK7348 – by A. A. Wicks is an easy-to-understand book written for the beginning kit builder as well as the experienced hobbyist. It has numerous pictures and descriptions of the safe and correct ways to use basic and specialized tools for electronic projects as well as specialized metal working tools and the chemical aids which are used in repair shops. $4.95.*

"PRICES SUBJECT TO CHANGE WITHOUT NOTICE"

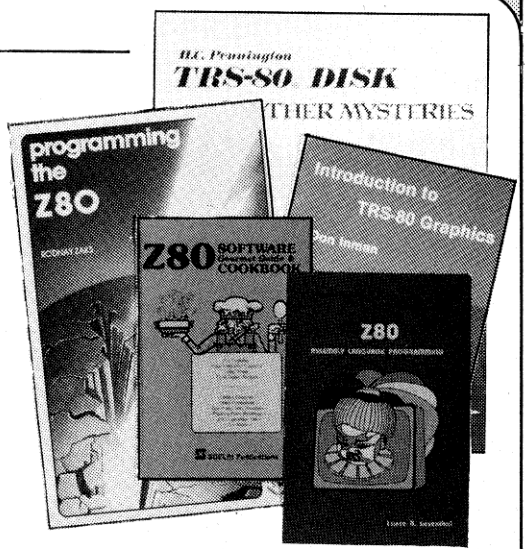
*Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to 80 Microcomputing Bookshelf ● Peterborough NH 03458. Be sure to include check or detailed credit card information. No C.O.D. orders accepted. All above add $1.00 handling. Please allow 4–6 weeks for delivery. Questions regarding your order? Please write Customer Service at the above address.

FOR TOLL FREE ORDERING CALL 1-800-258-5473

80 microcomputing T.M. bookshelf

Z80 BOOKS

NEW!



● **MICROSOFT BASIC DECODED AND OTHER MYSTERIES**—BK1186—by James Farvour. From the company that brought you *TRS-80 DISK AND OTHER MYSTERIES*! Contains more than 6500 lines of comments for the disassembled Level II ROMs, six additional chapters describing every BASIC subroutine, with assembly language routines showing how to use them. Flow charts for all major routines give the reader a real insight into how the interpreter works. $29.50 (Available after December 20th).

● **TRS-80 DISK AND OTHER MYSTERIES**—BK1181—by Harvard C. Pennington. This is the definitive work on the TRS-80 disk system. It is full of detailed "How to" information with examples, samples and in-depth explanations suitable for beginners and professionals alike. The recovery of one lost file is worth the price alone. $22.50.*

● **PROGRAMMING THE Z-80**—BK1122—by Rodnay Zaks. Here is assembly language programming for the Z-80 presented as a progressive, step-by-step course. This book is both an educational text *and* a self-contained reference book, useful to both the beginning and the experienced programmer who wish to learn about the Z-80. Exercises to test the reader are included. $14.95.*

● **Z-80 ASSEMBLY LANGUAGE PROGRAMMING**—BK1177—by Lance A. Leventhal. This book thoroughly covers the Z80 instruction set, abounding in simple programming examples which illustrate software development concepts and actual assembly language usage. Features include Z80 I/O devices and interfacing methods, assembler conventions, and comparisons with 8080A/8085 instruction sets and interrupt structure. $16.99.*

● **Z-80 SOFTWARE GOURMET GUIDE AND COOKBOOK**—BK1045—by Nat Wadsworth. Scelbi's newest cookbook! This book contains a complete description of the powerful Z-80 instruction set and a wide variety of programming information. Use the author's ingredients including routines, subroutines and short programs, choose a time-tested recipe and start cooking! $16.99.*

● **INTRODUCTION TO TRS-80 GRAPHICS**—BK1180—by Don Inman. Dissatisfied with your Level I or Level II manual's coverage of graphics capabilities? This well-structured book (suitable for classroom use) is ideal for those who want to use all the graphics capabilities built into the TRS-80. A tutorial method is used with many demonstrations. It is based on the Level I, but all material is suitable for Level II use. $8.95.*

BASIC & PASCAL

● **BASIC BASIC (2ND EDITION)**—BK1026—by James S. Coan. This is a textbook which incorporates the learning of computer programming using the BASIC language with the teaching of mathematics. Over 100 sample programs illustrate the techniques of the BASIC language and every section is followed by practical problems. This second edition covers character string handling and the use of data files. $10.50.*

● **LEARNING LEVEL II**—BK1175—by David Lien. Written especially for the TRS-80, this book concentrates on Level II BASIC, exploring every important BASIC language capability. Updates are included for those who have studied the Level I User's Manual. Sections include: how to use the Editor, dual cassette operation, printers and peripheral devices, and the conversion of Level I programs to Level II. $15.95.*

● **THE BASIC HANDBOOK**—BK1174—by David Lien. This book is unique. It is a virtual ENCYCLOPEDIA of BASIC. While not favoring one computer over another, it explains over 250 BASIC words, how to use them and alternate strategies. If a computer does not possess the capabilities of a needed or specified word, there are often ways to accomplish the same function by using another word or combination of words. That's where the HANDBOOK comes in. It helps you get the most from your computer, be it a "bottom-of-the-line" micro or an oversized monster. $14.95.*

● **INTRODUCTION TO PASCAL**—BK1189—by Rodnay Zaks. A step-by-step introduction for anyone wanting to learn the language quickly and completely. Each concept is explained simply and in a logical order. All features of the language are presented in a clear, easy-to-understand format with exercises to test the reader at the end of each chapter. It describes both standard PASCAL and UCSD PASCAL, the most widely used dialect for small computers. No computer or programming experience is necessary. $12.95.*

● **ADVANCED BASIC**—BK1000—Applications, including strings and files, coordinate geometry, area, sequences and series, simulation, graphing and games. $10.75*.

"PRICES SUBJECT TO CHANGE WITHOUT NOTICE"

*Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to 80 Microcomputing Bookshelf ● Peterborough NH 03458. Be sure to include check or detailed credit card information. No C.O.D. orders accepted. All above add $1.00 handling. Please allow 4–6 weeks for delivery. Questions regarding your order? Please write Customer Service at the above address.

FOR TOLL FREE ORDERING CALL 1-800-258-5473

80 microcomputing T.M. bookshelf

GAMES

NEW!

● **MORE BASIC COMPUTER GAMES** – BK1182 – edited by David H. Ahl. More fun in BASIC! 84 new games from the people who brought you *BASIC Computer Games.* Includes such favorites as Minotaur (battle the mythical beast) and Eliza (unload your troubles on the doctor at bargain rates). Complete with game description, listing and sample run. $7.50.*

● **WHAT TO DO AFTER YOU HIT RETURN** – BK1071 – PCC's first book of computer games . . . 48 different computer games you can play in BASIC . . . programs, descriptions, many illustrations. Lunar Landing, Hammurabi, King, Civel 2, Qubic 5, Taxman, Star Trek, Crash, Market, etc. $10.95.*

● **BASIC COMPUTER GAMES** – BK1074 – Okay, so once you get your computer and are running in BASIC, then what? Then you need some programs in BASIC, that's what. This book has 101 games for you from very simple to real buggers. You get the games, a description of the games, the listing to put in your computer and a sample run to show you how they work. Fun. Any one game will be worth more than the price of the book for the fun you and your family will have with it. $7.50.*

SPECIAL INTERESTS

● **THE CP/M HANDBOOK (with MP/M)**—BK1187—by Rodnay Zaks. A complete guide and reference handbook for CP/M—the industry standard in operating systems. Step-by-step instruction for everything from turning on the system and inserting the diskette to correct user discipline and remedial action for problem situations. This also includes a complete discussion of all versions of CP/M up to and including 2.2, MP/M and CDOS. $13.95.

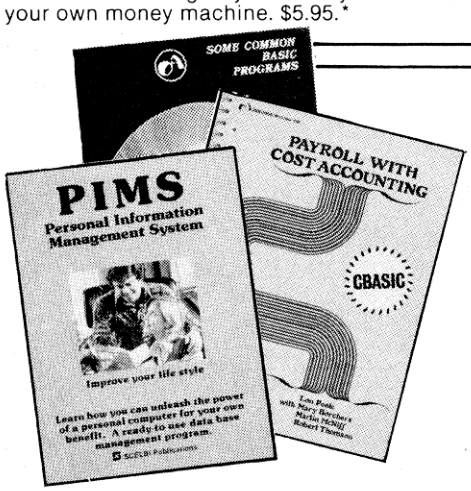
● **HOW TO MAKE MONEY WITH COMPUTERS** – BK1003 – In 10 information-packed chapters, Jerry Felsen describes more than 30 computer-related, money-making, high profit, low capital investment opportunities. $15.00.*

● **HOW TO SELL ANYTHING TO ANYBODY** – BK7306 – According to *The Guinness Book of World Records*, the author, Joe Girard, is "the world's greatest salesman." This book reveals how he made a fortune – and how you can, too. $2.25.*

NEW!

● **THE INCREDIBLE SECRET MONEY MACHINE** – BK1178 – by Don Lancaster. A different kind of "cookbook" from Don Lancaster. Want to slash taxes? Get free vacations? Win at investments? Make money from something that you *like* to do? You'll find this book essential to give you the key insider details of what is really involved in starting up your own money machine. $5.95.*

BUSINESS



● **PAYROLL WITH COST ACCOUNTING – IN BASIC** – BK1001 – by L. Poole & M. Borchers, includes program listings with remarks, descriptions, discussions of the principle behind each program, file layouts, and a complete user's manual with step-by-step instructions, flowcharts, and simple reports and CRT displays. Payroll and cost accounting features include separate payrolls for up to 10 companies, time-tested interactive data entry, easy correction of data entry errors, job costing (labor of distribution), check printing with full deduction and pay detail, and 16 different printed reports, including W-2 and 941 (in **CBASIC**). $20.00.*

● **SOME COMMON BASIC PROGRAMS**—BK1053—published by Adam Osborne & Associates, Inc. Perfect for non-technical computerists requiring ready-to-use programs. Business programs, plus miscellaneous programs. Invaluable for the user who is not an experienced programmer. All will operate in the stand-alone mode. $14.99 paperback.

● **PIMS: PERSONAL INFORMATION MANAGEMENT SYSTEM** – BK1009 – Learn how to unleash the power of a personal computer for your own benefit in this ready-to-use data-base management program. $11.95.*

"PRICES SUBJECT TO CHANGE WITHOUT NOTICE"

*Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to 80 Microcomputing Bookshelf ● Peterborough NH 03458. Be sure to include check or detailed credit card information. No C.O.D. orders accepted. All above add $1.00 handling. Please allow 4–6 weeks for delivery. Questions regarding your order? Please write Customer Service at the above address.

FOR TOLL FREE ORDERING CALL 1-800-258-5473

FLOPPY DISK STORAGE

CORVUS 11T 10 MEGABYTE HARD DISK *$4029
Fully compatible with **TRS-80, APPLE II,** and **S-100**

LOBO 400T $399
TRS-80 plug-in compatible 5.25" single headed 35 track drive. With Power Supply and case.

MPI MODEL B51 $321
TRS-80 plug-in compatible 5.25" single headed 40 track drive. Comes complete with case and power supply. **90 DAY WARRANTY.**

BARE DRIVES
(No Case or Power Supply)

5" MODELS

MPI MODEL 51	40 tracks, 5 ms.	$254
MPI MODEL 52	Double headed, 80 tracks	$299
MPI MODEL 91	80 tracks	$299
MPI MODEL 92	160 Tracks, Double headed	$399

5.25" POWER SUPPLIES WITH CASE

STANDARD MINI DISK POWER SUPPLY WITH CASE	$80
DELUXE MINI DISK POWER SUPPLY w/CASE & extender card	$131
DUAL MINI DISK POWER SUPPLY w/CASE	$121
DISK DRIVE EXTENDER CABLE	$16

DRIVE CABLES

1 DRIVE TRS-80 CABLE	$24
2 DRIVE TRS-80 CABLE	$29
3 DRIVE TRS-80 CABLE	$35
4 DRIVE TRS-80 CABLE	$40

We'll make any cable! Call us with specs.

THE DOUBLER by PERCOM DATA $203
Double density controller for **TRS-80 Model I**. With data separator. (Use your **PERCOM DISCOUNT** coupon)
DATA SEPARATOR by Percom Data $29
Stops read errors on inner tracks.

SOFTWARE

NEWDOS (40 Track)	$49
NEWDOS + (40 Track)	$99
NEWDOS 80	$139
TBS Business Mail	$100
Super Utility	$49
Make 80	$14
Electric Pencil (disk)	$130

10% off all **RADIO SHACK** software

DISKETTES

		(Prices per box of 10) VERBATIM	DYSAN	BASF
5 1/4"	Single sided, single density	$31	$41	$28
	Single sided, double density	$40	$42	
	Double sided, single density	$48	$45	
	Double sided, double density	$48	$48	
8"	Single sided, single density	$41	$41	
	Single sided, double density	$51	$61	
	Double sided, single density	$61	$51	
	Double sided, double density	$61	$61	

When ordering diskettes, please give us your make and model number of computer or disk drive, or give us the brand and part number of the diskettes you are currently using.

5 1/4"	**LIBRARY CASE** holds 10 disks	$3.95
	BINDER HOLDER holds 2 disks	10 for $7.95
8"	**LIBRARY CASE** holds 10 disks	$4.95
	BINDER HOLDER holds 2 disks	10 for $19.95
FLOPPY SAVER		5 1/4" $14.95
		8" $16.95
		Refills also available!

16k **MEMORY KITS** 200 ns $39.00
NEC UPD416C-2 prime dynamic RAM. Compatible with TRS-80, APPLE, or any computer specifying 4116 dynamic RAM.

HOW TO ORDER:
PHONE ORDERS: HOURS 10AM - 5PM CST MON - FRI
BY CHECK: Call in your order then send us a cashiers check or money order. This way we can have your order waiting for you when your check arrives. Personal checks require 10-14 days to clear.
BY CREDIT CARD: Just have your **MASTER CHARGE** or **VISA** or **AMERICAN EXPRESS** handy, we'll do the rest.
MAIL ORDERS: NEW ADDRESS 1467 South Michigan Ave., Chicago, IL 60605.
ALL MAIL ORDERS: Send your **NAME** and **STREET ADDRESS** (we ship UPS), **CITY, STATE, ZIP CODE AND TELEPHONE NUMBER** along with your order.
BY CHECK: Include a cashiers check or money order with your order. Personal checks require 10-14 days to clear. For your safety, mail your order via insured mail. **BETTER SAFE THAN SORRY.**
BY CREDIT CARD: Neatly type or print your **MASTER CHARGE** or **VISA** account number along with the expiration date and 4 digit interbank number (MC only).
IL. RES. ADD 6% SALES TAX

SHIPPING INFORMATION: We pay all UPS ground service charges (USA only, BLUE LABEL EXTRA) on all items except those denoted with an asterisk (*). (*) items must go airfreight or truck and can be sent freight collect.
FOREIGN ORDERS WELCOME, CALL OR WRITE.
ALL orders F.O.B. Chicago, Il.

WARRANTY INFORMATION: All Equipment carries a 90 day parts and labor warranty, unless otherwise noted. Warranty repair or replacement will be provided at no charge, excluding shipping costs. **10 DAY MONEY BACK GUARANTEE:** If within ten days you return merchandise in new and re-saleable condition we will refund your original purchase price. All refunds, exchanges or repairs include original cartons, packaging, and manuals and be in new and re-saleable condition and accompanied by your original invoice. A 10% re-stocking fee may be charged on returns at our discretion.
Not responsible for typographical errors
Prices subject to change without notice
Dealer inquiries welcome
TRS-80™ TANDYCORP.

SAVE TIME. . . ORDER BY PHONE
TOLL FREE: 800-621-3229
IN ILLINOIS: 312-987-1024

MCP MIDWEST COMPUTER PERIPHERALS

1467 South Michigan Av.
Chicago, IL. 60605

We pay UPS charges on all orders in continental U.S.

Reader Service—see page 290

HOBBYWORLD ELECTRONICS, INC.

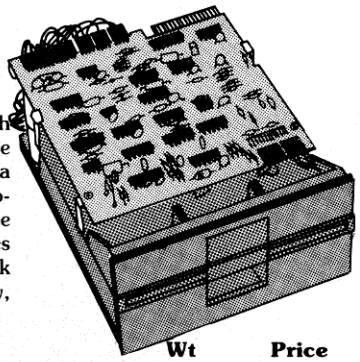
19511 Business Center Dr. Dept. V3 Northridge, CA 91324

Call Toll Free: (800) 423-5387 ↙23 **Local & Outside USA: (213) 886-9200**

March Specials for your TRS-80

TANDON MAGNETICS

5¼" Double Density Disk Drives



Tandon disk drives offer you high storage capacity, quality and performance at an affordable price. The TM-100-1 is a single sided, double density drive providing 40 tracks and 250K bytes storage capacity (unformatted). These features and others are the reason TANDON disk drives are used by companies like Tandy, Vector Graphic, North Star and others!

Cat No.	Description	Wt	Price
2933	Tandon TM-100-1 bare drive	5 lb	$275.00
2941	(1) TM-100-1 w/pwr sply & sgl encl.	8 lb	$325.00
2942	(2) TM-100-1 w/pwr sply & dual encl.	13 lb.	$645.00

SUB-LOGIC

Flight Simulator

$21.95

YOU are the pilot, with a real time, 3D view of flight and regulation FAA instrument panel and controls. Practice your skills, then begin your first mission as a wartime fighter pilot! Fly into enemy territory, destroy the enemy fuel depot, and return home safely (If you can). Along the way you'll encounter enemy fighters who are cunning, fast and sometimes work as a team. Wt. 8 oz.

Cat No. 2850 TRS-80, L2 & L3, 16K cass

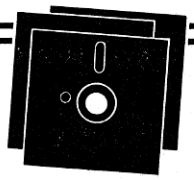
16K Memory Add-On

for the TRS-80

$32.00

Everything you need to upgrade your system! Complete with RAMs and preprogrammed jumpers—No special tools required. Four pages of illustrated instructions are also included. Wt 4 oz.

Cat No.	Description
1156	For TRS-80 keyboard unit
1156A	Exp. Interface purchased before 4/1/79
1156B	Exp. Interface purchased after 4/1/79



MEMOREX 5¼" Diskettes

- *All 40 track tested*
- *Single sided, double density*
- *Soft Sector, unformatted*
- *Wt 12 oz.*

$29.00 (box of 10)

Cat No. 2909

FREE HOBBYWORLD CATALOG

Name _____

Company _____

Address _____

City _____

State _____ *Zip* _____

FREE CATALOG

Forty-four pages of computers, terminals, printers, disk drives and many more products that can add dimension to your personal computing. We also carry a complete line of computerized toys and games, application boards, integrated circuits, software, p.c. & soldering accessories, electronics books and much more!

Call or write today and ask for your free catalog (Or circle the reader's service number in this magazine).

HOW TO ORDER

Order toll-free by phone, by mail, or at our retail stores. Pay by check, Mastercard, VISA or C.O.D. Please include expiration date and interbank number with charge card orders. US dollars only. Please include phone number, and magazine issue from which you are ordering. Add $1.25 for C.O.D.

Prices valid through month of magazine issue. Subject to prior sale or quantity limits. Not responsible for typographical errors. 120 day guaranteed satisfaction (Exception: Partially assembled kits, abuse or misuse. Software will be replaced with another copy only).

SHIPPING: USA

Ground: $2.25 for first 2 lbs. 40¢ each additional lb.
Air: $3.25 for first 2 lbs. 70¢ each additional lb.

SHIPPING: FOREIGN

Ground: $3.00 for first 2 lbs. 60¢ each additional lb.
Air: $11.25 for first 2 lbs. $5.00 each additional lb.

TRS-80 is a registered trademark of the Radio Shack division of Tandy Corp.

$uper $avings on printers & disks for the TRS-80™.

CENTRONICS 779

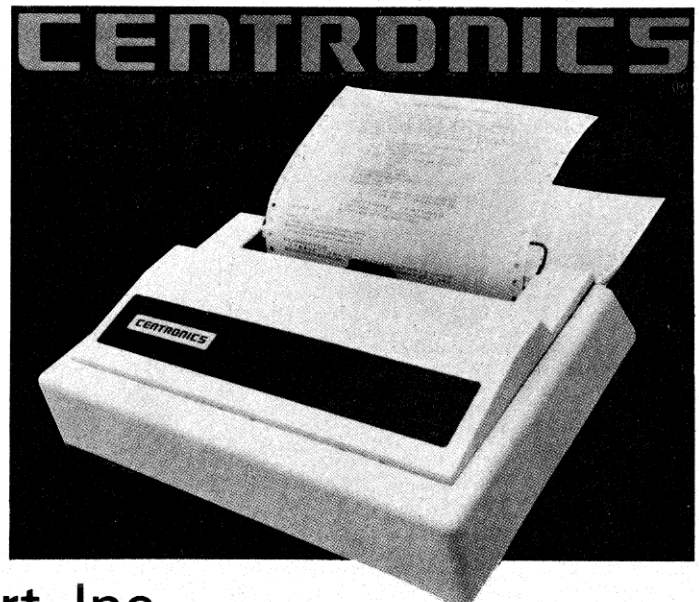
Same as Radio Shack Line Printer I
LIST PRICE $1350 ***OUR PRICE*** **$849**
(Ship freight collect)

CENTRONICS 737

Featuring Correspondence, Quality and Proportional Spacing
LIST $995 ***OUR PRICE*** **$829**

CENTRONICS 730

Same as Radio Shack Line Printer II
LIST $795
OUR PRICE **$649**
(add $7.50 for shipping)



CENTRONICS 704-11
(same as 703-9)
CENTRONICS PARALLEL INTERFACE
Ideal for TRS-80

180 CPS Logic-Seeking Tractors
Adjustable to 16''
Former List Price $2975

NOW ONLY **$1695**

DISK DRIVES FOR TRS-80
5 1/4'' Disk Drives
Ideal for TRS-80

Featuring MPI-51 Drive, 40 Track Capability, Fast Seeking

$499 VALUE

MMM PRICE ONLY **$329**
2 for $638

TM TRS-80 is a trademark of Radio Shack

24

MiniMicroMart, Inc. 1618 James St., Syracuse, NY 13203 (315) 422-4467 TWX 710 541-0431

LIST OF ADVERTISERS

RS Number		Page
282	ACR Consultants	72
452	A. M. Electronics	51
445	AT-80	162
229	Access Unlimited	215
34	Acorn Software Products	89
184	Advanced Technology Corp	58
213	Adventure International	81
97	Adventure International	37
163	Adventure International	59
387	Aerocomp, Inc	183
69	Alpha Byte Storage	53
401	Alpha Products Company	26, 27
124	Alphanetics	127
138	The Alternate Source	44, 126, 190, 224, 282
396	American Business Computers	101
397	American Business Computers	101
483	American Business Computers	101
484	American Business Computers	101
192	American Business Computers	101
317	American Business Computers	101
322	American Business Computers	101
227	American Computer Development	241
461	Ancie Labs	197
264	Apparat, Inc	61, 139
*	Archbold Electronics	261
166	Aspen Ribbons, Inc	59
146	Audio Video Systems	102, 162, 276
48	Automated Simulations	8
162	The Avalon Hill Game Co	59
*	Balcode Software	88
152	BAP$ Software	273
201	Barstarnn Corporation	218
49	Basics and Beyond, Inc	251
186	Baysian Investment Services	265
237	BCCOMPCO	273
357	Big Five Software Company	109
181	Blechman Enterprises	57
76	Breeze Computing	56
382	Business Micro Products	80
325	CC Exchange	60
316	CFR Associates	208
145	C & S Electronics Mart Ltd	271
298	The CPU Shop	143
294	Caldata Systems	218
38	Case Computer Products Co	78
62	Cecdat, Inc	64, 269
32	Cload Magazine	177
100	CompuCover	162
107	Computer Applications Unlimited	282
199	Computer Case Company	242
169	Computer Control Systems, Inc	58
372	Computer Disc. of America	112
22	Computer Information Exchange	108
130	Computer Plus	221
212	Computer Shopper	263
392	Computex	261
9	Computronics, Inc	144–151
204	Comsoft	251
327	Contact Electronic Specialists, Inc	60
10	Contract Services Associates	99
78	Coosol, Inc	202
233	Cottage Software	138
7	Custom Computer Center	130
121	Custom Electronics	162
*	Cybernetics, Inc	123
83	Cyberware	164
134	DAR Sales	269
44	Data Train, Inc	98
274	Data Trans	265
407	Datasoft	29, 66
302	Decision Master	71
123	Design Solution, Inc	43
175	Design Solution, Inc	59
361	Digital Concepts Inc	279
313	Digital Data Supply Co	218
368	Digital Systems Engineering	130
440	Discount Software Group	130
*	Disk Supply Company	175
88	Documan Software	190
314	Dorsette Educations	128
488	Eugene F. Dreyer	273
253	Dynatek Information Systems Inc	218
117	EBG & Associates	268
*	80 Microcomputing	129, 158, 217, 269, 281, 283–285
33	80 US Journal	Cov.IV
329	E-Z Software	60
58	Electronic Specialists	271
327	Electronic Specialists	60
278	Emtrol Systems, Inc	228
404	Epson America	35
3	Exatron	Cov.IV
141	FEC Ltd	280
12	FMG Corporation	107
115	Financial Computer Systems	205
39	G & L Software Enterprises	156
79	Allen Gelder Software	120, 170
75	Godbout Electronics	122
218	Good-Lyddon Data Systems	268
394	Gooth Software	242
171	Gooth Software	59
223	H & H Trading Company	263
125	Hexagon Systems	133
23	Hobby World Electronics	288
457	Howard W. Sams	201
366	Howard W. Sams	95
103	Howe Software	282
37	IJG Inc	39
300	Information Technology Systems	273
305	Insiders Software Consultants Inc	277
2	Instant Software	90, 91, 125, 195, 234, 235, 254, 255
179	Interaction Systems Inc	57
246	Interface, Inc	142
287	Interlude	79
190	JPC Products	198, 199
155	JR Software	273
193	Joe Computer	140
85	Johnson Associates	267
239	Joseph Bros	276
353	Kalglo	229
230	Kengore Corp	159
53	LNW Research	208, 267
173	Lakefront Software	59
198	The Learning Place	224
14	Level IV Products, Inc	156, 202
182	Lifeboat Associates	57
208	Little Bee Educational Programs	120
15	Lobo Drives Int'l	Cov.III
306	Mr. X Consulting Inc	268
451	MTS Enterprises	277
268	Maine Software	218
87	Management Systems Software	176
90	Manhatten Software, Inc	127
270	Mark Gordon Computers	171
128	Med Systems Software	161
421	Medfield Computer Software	224
272	Mega Systems	273
104	Mercer Systems	128
20	Meta Technologies Corp	11, 15, 17, 21, 23
219	Meta Media	9
54	Micro Architect	133
185	Micro Architect	60
205	Micro Blajak Systems, Inc	213
214	The Micro Clinic	277
379	Micro-Design	140
476	Micro-80	229
89	Micro Learningware	162
72	Micro Management Systems Inc	65
68	Micro Matrix	267
29	Micro Mega	88, 223
310	Micro Mint	229
384	Micro Systems Software Inc	6, 194
109	Micro Works	227
176	Microcomputer Information Services	57
28	Microcomputer Tech. Inc	166, 167
178	Microcosm, Inc	58
6	Microperipheral Corp	187
168	Microperipheral Corp	58
248	MicrOdome	273
378	Microproof	122
165	Microsette Co	57
8	Midwest Computer Peripherals	286, 287
112	Miller Microcomputer Services	96
24	Mini Micro Mart Inc	289
98	Minis 'n Micros Inc	162
221	MISOSYS	190
177	Modular Software Assoc	60
*	Mullen Computer Products	176
144	Mumford Micro Systems	115
135	NDM Design	261
*	NRI Schools	193
381	National Computer Conference	113
430	National Computer Shows	46, 47
142	National Tricor, Inc	268
55	Nautilus	164
194	New England Business Service, Inc	133
74	Northeast Microware	126
389	Omega Sales	157
174	Omni Systems Co	59
105	Omnitek	229
172	Opportunities for Learning	58
296	Orange Micro	227
376	Orion Instruments	228
308	Ozark Mountain Software	273
160	PGI Publishing	58
406	P & S Electronics	218
370	Pacific Exchanges	102
153	Pacific Office Systems	177
228	Palomar Software	280
64	Pan American Electronics	123
207	Pensadyne Computer Services	127, 281
1	Percom Data Company	Cov.II
408	Percom Data Company	3
183	Percom Data Company	60
51	Perry Gas & Oil	164
422	Personal Micro Computers, Inc	121
328	Personal Micro Computers, Inc	60
*	Phase I Systems	239
273	Pickles & Trout	246
17	The Program Store	116, 117
21	Programma International	85, 141
364	The Programmers Guild	185
110	Programs Unlimited	188, 189
180	Programs Unlimited	58
441	Prosoft	169
395	QC Microsystems	212
170	QC Microsystems	59
269	Quant Systems	281
304	Quarp Publishing	142
385	RKS Enterprises, Inc	211
41	Racet Computes	97
*	Realty Software Company	213
70	Remsoft, Inc	100
276	Richcraft Engineering Ltd	218
468	Rochester Data Inc	208
291	Scientific Engineering Lab	190
297	Service Technologies Inc	243
19	Simutek	247, 251
67	Sirius Systems	94
232	Snapp, Inc	31, 67
434	Soft Sector Marketing Inc	172, 173, 263
299	Software Concepts	238
220	Software Critic	218
*	Software Discount	213
42	Software Etc	103
478	Software Innovations	77
380	Software Innovations	77
373	Southern Innovative Design (SID)	33
132	Spectral Associates	265
154	Standard & Poors	73
438	Stocking Source	32
82	Sturdivant & Dunn, Inc	268
164	Sturdivant & Dunn, Inc	57
150	Sublogic	224
326	Suma Microware	60
403	Superior Software	273
167	Sybex, Inc	58
231	Synergistic Solar Inc	162
139	Syntex Electronic Innovations	281
161	Syntex Electronic Innovations	60
358	Syracuse R & D Center	124
157	System Shacks	124
5	Systemworks	223
148	Tab Sales Company	138
45	Taranto & Associates	134, 135
147	Task Computer Applications	280
25	Texas Computer Systems	170
402	Turtle Computer Software Corp	233
*	V R Data Corporation	233
137	Van Horn Office Supply	279
432	Vern Street Products/Keyline Computer Products	66
111	Vern Street Products/Keyline Computer Products	249
363	Walonick Associates	253

***This advertiser prefers to be contacted directly.**

READER SERVICE

This card valid until May 31, 1981.

Please help us to bring you a better magazine—by answering these questions.

I. What is your age?
- ☐ A. under 18
- ☐ B. 18-22
- ☐ C. 23-40
- ☐ D. 41-60
- ☐ E. over 60

II. What is your occupation?
- ☐ 1. Professional
- ☐ 2. Engineer
- ☐ 3. Data processing
- ☐ 4. Business
- ☐ 5. Education
- ☐ 6. Technician
- ☐ 7. Student
- ☐ 8. Other

III. What are your primary applications of your TRS-80 (check only two)?
- ☐ A. Business
- ☐ B. Games
- ☐ C. Home
- ☐ D. Education
- ☐ E. Scientific
- ☐ F. Control
- ☐ G. Music

IV. Your TRS-80, is it a
- ☐ 1. Level I
- ☐ 2. Level II
- ☐ 3. Model II
- ☐ 4. Don't own one yet

V. What peripherals do you have (check all that apply)?
- ☐ A. Expansion interface
- ☐ B. Disk
- ☐ C. Printer

VI. How much have you spent on hardware?
- ☐ 1. less than $500
- ☐ 2. $500-1,000
- ☐ 3. $1,000-2,000
- ☐ 4. $2,000-4,000
- ☐ 5. $4,000-6,000
- ☐ 6. more than $6,000

VII. How much have you spent on software?
- ☐ A. less than $100
- ☐ B. $100-250
- ☐ C. $250-500
- ☐ D. $500-1,000
- ☐ E. more than $1,000

VIII. What is your level of education?
- ☐ 1. Post-graduate
- ☐ 2. College
- ☐ 3. High school

IX. How many people read your copy of 80 Microcomputing?
- ☐ A. 1
- ☐ B. 2
- ☐ C. 3
- ☐ D. 4 or more

X. If you are not a subscriber please circle number 500.

80 Microcomputing • POB 2743 • Clinton IA 52735

Reader Service: To receive more information from any of the advertisers in this issue of 80 Microcomputing, circle the number on the postage-paid 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	126	131	136	141	146	251	256	261	266	271	376	381	386	391	396
2	7	12	17	22	127	132	137	142	147	252	257	262	267	272	377	382	387	392	397
3	8	13	18	23	128	133	138	143	148	253	258	263	268	273	378	383	388	393	398
4	9	14	19	24	129	134	139	144	149	254	259	264	269	274	379	384	389	394	399
5	10	15	20	25	130	135	140	145	150	255	260	265	270	275	380	385	390	395	400
26	31	36	41	46	151	156	161	166	171	276	281	286	291	296	401	406	411	416	421
27	32	37	42	47	152	157	162	167	172	277	282	287	292	297	402	407	412	417	422
28	33	38	43	48	153	158	163	168	173	278	283	288	293	298	403	408	413	418	423
29	34	39	44	49	154	159	164	169	174	279	284	289	294	299	404	409	414	419	424
30	35	40	45	50	155	160	165	170	175	280	285	290	295	300	405	410	415	420	425
51	56	61	66	71	176	181	186	191	196	301	306	311	316	321	426	431	436	441	446
52	57	62	67	72	177	182	187	192	197	302	307	312	317	322	427	432	437	442	447
53	58	63	68	73	178	183	188	193	198	303	308	313	318	323	428	433	438	443	448
54	59	64	69	74	179	184	189	194	199	304	309	314	319	324	429	434	439	444	449
55	60	65	70	75	180	185	190	195	200	305	310	315	320	325	430	435	440	445	450
76	81	86	91	96	201	206	211	216	221	326	331	336	341	346	451	456	461	466	471
77	82	87	92	97	202	207	212	217	222	327	332	337	342	347	452	457	462	467	472
78	83	88	93	98	203	208	213	218	223	328	333	338	343	348	453	458	463	468	473
79	84	89	94	99	204	209	214	219	224	329	334	339	344	349	454	459	464	469	474
80	85	90	95	100	205	210	215	220	225	330	335	340	345	350	455	460	465	470	475
101	106	111	116	121	226	231	236	241	246	351	356	361	366	371	476	481	486	491	496
102	107	112	117	122	227	232	237	242	247	352	357	362	367	372	477	482	487	492	497
103	108	113	118	123	228	233	238	243	248	353	358	363	368	373	478	483	488	493	498
104	109	114	119	124	229	234	239	244	249	354	359	364	369	374	479	484	489	494	499
105	110	115	120	125	230	235	240	245	250	355	360	365	370	375	480	485	490	495	500

Name _____

Address _____

City _____ State _____ Zip _____

BOOKS

80 Microcomputing
Peterborough NH 03458

Please send me the following 80 Microcomputing products:

Qty.	Catalog #	Title	Unit Price	Total

Add $1 shipping/handling _____

Total _____

Enclosed $_____ ☐ Check ☐ M.O.
Bill: ☐ AE ☐ MC ☐ VISA

Card # _____

Exp date_____ Interbank # _____

Signature _____

Name _____

Address _____

City_____ State_____ Zip_____

3/81

Please allow 4–6 weeks for delivery. No C.O.D. orders accepted.

SUBSCRIPTION

80 Microcomputing POB 981
Farmingdale NY 11737

80 Microcomputing subscribers SAVE 40% Subscribe now.

Please allow 4–6 weeks for delivery

☐ New Subscription ☐ Renewal
☐ 1 year—$18 ☐ 2 years—$30 ☐ 3 years—$45
☐ Enclosed $_____ ☐ Check ☐ M.O.
Bill: ☐ MC ☐ VISA ☐ AE ☐ Me

Card # _____ Exp. date_____

Signature _____ Interbank # _____

Name _____

Address _____

City _____ State _____ Zip _____

Canada: $20 1 year only, US funds
Other foreign: $28 one year only, US funds

313B9

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 217 CLINTON IA 52735

POSTAGE WILL BE PAID BY ADDRESSEE

80 microcomputing T.M.

POB 2743
Clinton IA 52735

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 780 FARMINGDALE NY 11737

POSTAGE WILL BE PAID BY ADDRESSEE

80 microcomputing

Subscription Dept.
POB 981
Farmingdale NY 11737

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY CARD

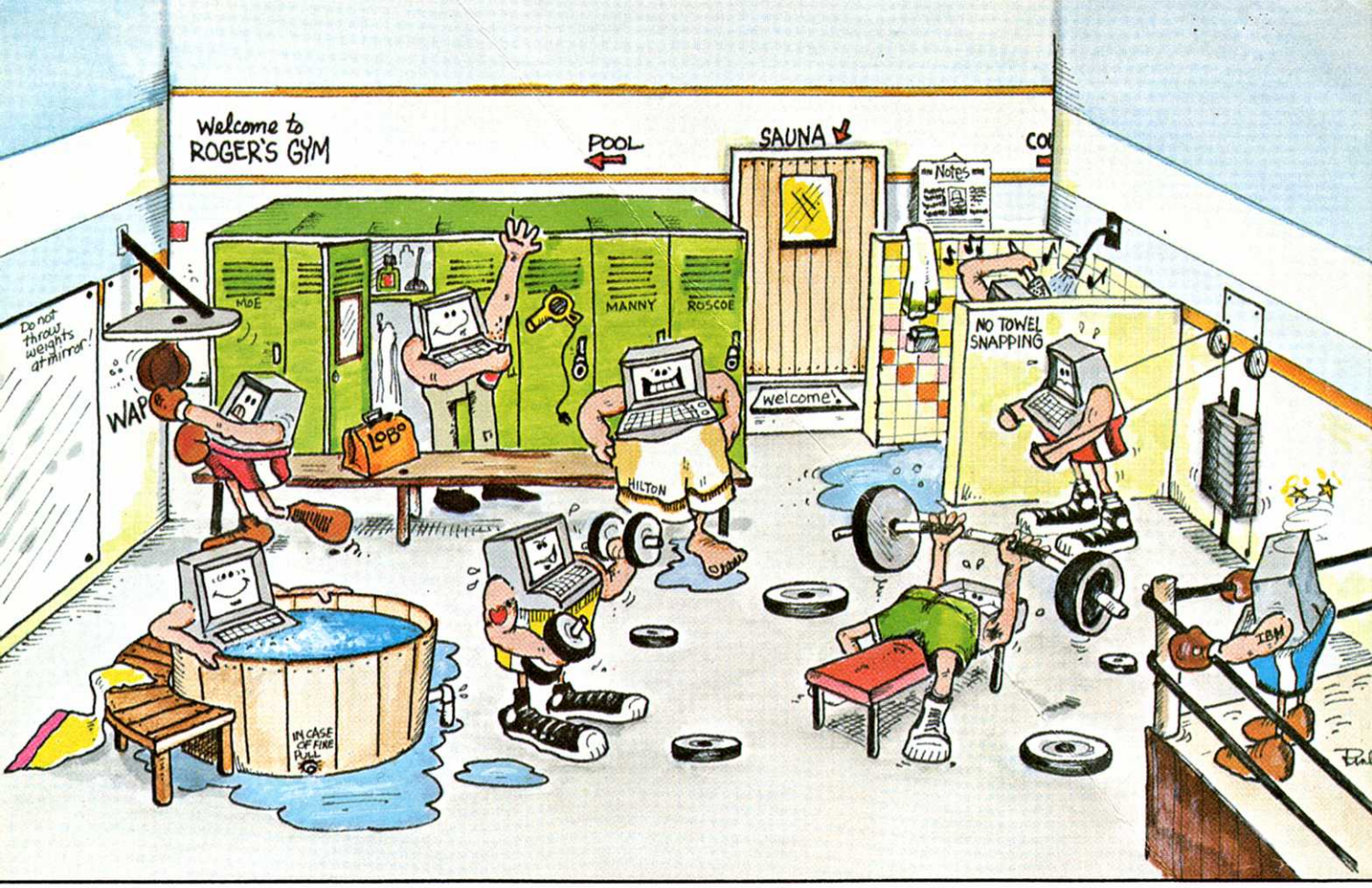
FIRST CLASS PERMIT NO. 80 PETERBOROUGH NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE

80 microcomputing

Peterborough NH 03458

Att. Mail Order



LOBO'S NEW LDOS™ Puts muscle in your TRS-80*

LOBO DRIVES' new LDOS™ Disk Operating System is loaded with outstanding features that will enable you to realize the full power and potential of your TRS-80.* With LDOS, you can support up to eight drives (5¼ and 8-inch drives, double-sided drives, double-density drives, 80-track drives), including the new 8-inch and 5¼-inch Winchester fixed disk drives, in any combination.

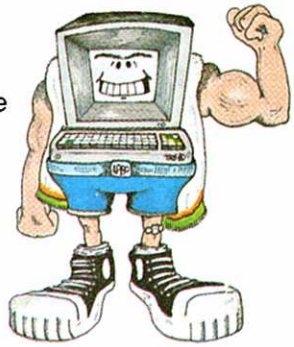
Other LDOS muscle building features include: ISAM accessing techniques; keyboard typeahead; Graphic string packer; Dated files, Marked files; File transfer by class; Built-in lower case display drivers; Non-breakable AUTO and DO commands, and many, many more.

LDOS is the perfect operating system to use with your LOBO DRIVES LX-50 or LX-80 expansion interface and disk drive subsystems. There's even an 800 number for instant service. To find out how you can put more muscle into your TRS-80, contact your nearest LOBO Drives dealer or call or write.

LDOS is available:

A. Operating System Diskette with Master Reference Manual
$139

B. Master Reference Manual Only
$25



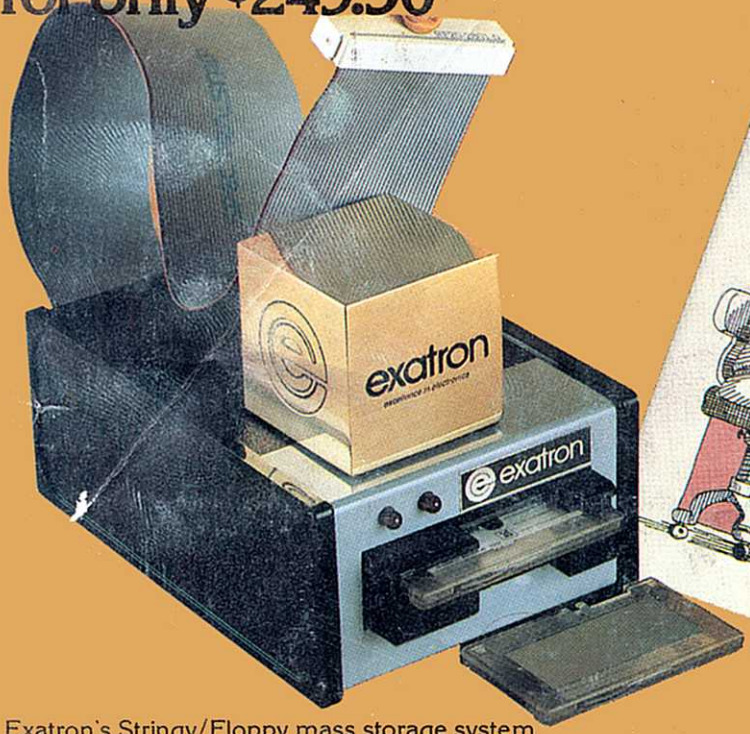
* TRS-80 is a registered trademark of Radio Shack, a Tandy Company.

LOBO drives INTERNATIONAL

LOBO DRIVES, INT'L ✓15
354 South Fairview Ave.
Goleta, CA 93117
(805) 683-1576

Two proven ways to expand your TRS-80* capability

Exatron's Stringy/Floppy... speed, capacity and reliability for only $249.50



Exatron's Stringy/Floppy mass storage system gives you the speed, capacity and reliability of a mini-disk system at far less cost. Thousands of ES/F users agree.

Here's your complete Starter System:

- ES/F Operating Manual
- Basic ES/F System ($249.50)
- 10 Blank Wafers
- ES/F Monitor Program
- Tutorial Demo Program
- 2 for 1 Bus Connector
- Data I/O Program
- FREE 1 Year Subscription to 80-U.S., the User's Journal
- Complete Info Package
- Complete Starter Kit: $299.50

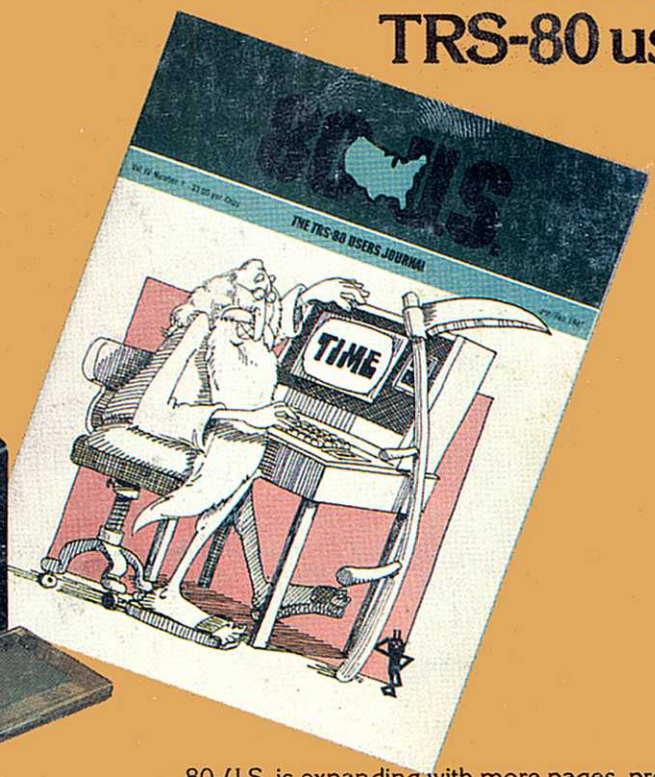
CALL OUR FREE HOTLINE
(800) 538-8559
IN CALIFORNIA: (408) 737-7111

Or contact:

exatron ↙3
181 Commercial Street
Sunnyvale, CA 94086

*TRS-80 Trademark of Tandy Co.

80-U.S. TRS-80 User's Journal New "How-To" info for the serious TRS-80 user



80-U.S. is expanding with more pages, programs and color with heavy emphasis on advanced TRS-80 applications... plus a regular Exatron Stringy/Floppy feature.

Special Introductory offer $9.95 per year

We'll send you six issues of 80-U.S. for only $9.95... almost half the regular price. If you purchase an Exatron Stringy/Floppy System, we'll send them FREE.

OFFER VALID UNTIL MARCH 31, 1981.

Send to:

80-U.S. ↙33
80-U.S. Journal
3838 South Warner Street
Tacoma, WA 98409
(206) 475-2219

Sign me up! I can't resist at $9.95!

☐ Check ☐ Money Order ☐ M/C or Visa

CARD NUMBER _____ EXP DATE _____

NAME _____

ADDRESS _____

CITY _____ STATE _____ ZIP _____

Offer good only in the United States.