February 1980 $2.00 U.S., $2.50 Canada

THE TRS-80* MAGAZINE

80 microcomputing™

THE magazine for TRS-80* users

Mailing List Programs: 4 reviewed – which is best? Pg. 24.

ROM Secrets: Use the built-in subroutines, and increase program speed. **Pg. 106.**

Disks Exposed: All about the workings of your disks and how to fix errors. **Pg. 32.**

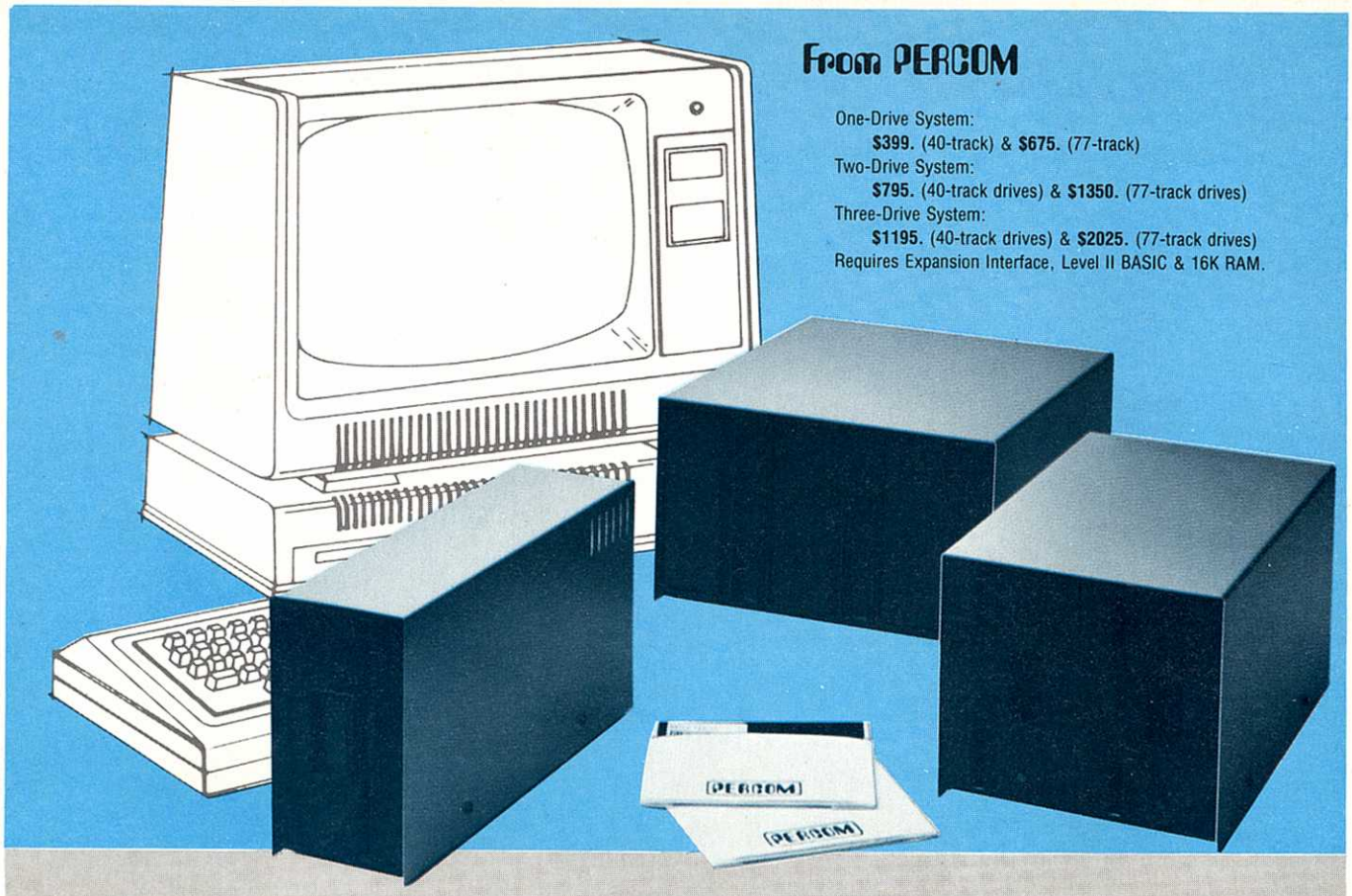
Psychotherapy & the Micro: Confessions of former lives and hidden meanings. **Pg. 9.**

Faster! Faster!: Double the speed of your system, under program control. **Pg. 50.**

Plus:
11 Programs with complete listings.
Source Code for 3 Assembly Language programs.
Software Reviews, Hardware Reviews, Book Reviews.
More than 30 articles and columns.

*a trademark of Tandy Corporation

0 74470 65947 02



Low Cost Add-On Storage for Your TRS-80*. In the Size You Want.

When you're ready for add-on disk storage, we're ready for you. Ready with six mini-disk storage systems — 102K bytes to 591K bytes of additional *on-line* storage for your TRS-80*.

- **Choose either 40-track TFD-100™ drives or 77-track TFD-200™ drives.**
- **One-, two- and three-drive systems immediately available.**
- **Systems include Percom PATCH PAK #1™, on disk, at no extra charge. PATCH PAK #1™ de-glitches and upgrades TRSDOS* for 40- and 77-track operation.**
- **TFD-100™ drives accommodate "flippy disks." Store 205K bytes per mini-disk.**
- **Low prices. A single-drive TFD-100™ costs just $399. Price includes PATCH PAK #1™ disk.**
- **Enclosures are finished in system-compatible "Tandy-silver" enamel.**

Whether you need a single, 40-track TFD-100™ add-on or a three-drive add-on with 77-track TFD-200™s, you get more data storage for less money from Percom.

Our TFD-100™ drive, for example, lets you store 102.4K bytes of data on one side of a disk — compared to 80K bytes on a TRS-80* mini-disk drive — and 102.4K bytes on the other side, too. Something you can't do with a TRS-80* drive. That's almost 205K bytes per mini-disk.

And the TFD-200™ drives provide 197K bytes of on-line storage per drive — 197K, 394K and 591K bytes for one-, two and three-drive systems.

PATCH PAK #1™, our upgrade program for your TRSDOS*, not only extends TRSDOS* to accommodate 40- and 77-track drives, it enhances TRSDOS* in other ways as well. PATCH PAK #1™ is supplied with each drive system at no additional charge.

The reason you get more for less from Percom is simple. Peripherals are not a sideline at Percom. Selling disk systems and other peripherals is our main business — the reason you get more engineering, more reliability and more back up support for less money.

In the Product Development Queue . . . a *printer interface* for using your TRS-80* with any serial printer, and . . . the *Electric Crayon*™ to map your computer memory onto your color TV screen — for games, animated shows, business displays, graphs, etc. Coming PDQ!

™ TFD-100, TFD-200, PATCH PAK and Electric Crayon are trademarks of PERCOM DATA COMPANY.
*TRS-80 and TRSDOS are trademarks of Tandy Corporation and Radio Shack which have no relationship to PERCOM DATA COMPANY.

PERCOM

PERCOM DATA COMPANY, INC.
211 N. KIRBY • GARLAND, TX. • 75042

To order add-on mini-disk storage for your TRS-80*, or request additional literature, call Percom's toll-free number: 1-800-527-1592. For detailed Technical information call (214) 272-3421.

Orders may be paid by check or money order, or charged to Visa or Master Charge credit accounts. Texas residents must add 5% sales tax.

Percom *'peripherals for personal computing'*

NOW... Exclusively Yours from ACS

TRS-80-DOS 3.0

Disk Operating System by the Original Author

ACS is not affiliated with Radio Shack or Tandy Corp.

☐ DOS 3.0
EFFECTIVE SEPTEMBER 1
ONLY $49.95

Check these DOS 3.0 features . . . that go far beyond 2.1

NO KEY BOUNCE • UPWARDS COMPATABLE FROM 2.1 • CHECK EOF
SEEK • RE-READ • RE-WRITE • LOC • VARABLE LENGTH RECORDS
SKIP • DISK LOGGING OF MESSAGES • BOOT • CHAIN • WRITE EOF
PURGE • SET • RESET • ROUTE • RUN & LOAD FOR 1 DRIVE SYS.
XFER • FORMAT W/O ERASE • DIR FROM BASIC • PATCH • LINK
USER DEFINED KEYS • KEY AUTO REPEAT • UPPER AND LOWER
CASE DRIVER • SHIFT LOCK • RS-232-C DRIVERS
• PAUSE • USER DEVICE CONTROL BLOCKS
• MULTI PROTOCOL COMMUNICATIONS

HARDWARE FOR TRS-80®

☐ **Pertec Disk Drives FD-200 ..... $375.00 ea.**
These are 40-track Drives that are completely compatible with the TRS-80 and Radio Shack Drives. 3.0 DOS $20.00 extra with disk drive. Will allow Turning Diskette over and Write on other side.

☐ **4-Drive Cable for Pertec Drives $35.00**

☐ **DECwriter III, 132 Character $2500.00**
110 to 9600 band EIA tractor feed keyboard printer. This is truly the nicest printer available. (30 day delivery)

☐ **701 Centronics TRACTOR FEED Bidirectional Printer $1500.00**
2½ times as fast as the Radio Shack 779 Printer, has full size 132 Char. Carriage Bell tone. Complete with Cable plug in and use. Shipped Freight COD.

☐ **200 ns 16K Dynamic Memory Clips for Keyboard or Expansion Interface, Lifetime Guarantee, complete $110.00**
Lifetime Guarantee. Complete with Instructions and Jumper Blocks.

☐ **10 Key Numerical Keypad Kit $79.95**

☐ **TRS-80® Level II – 16k $750.00**
- ☐ Expansion Interface $275.00
- ☐ RS-232-C Interface ..... $ 89.00

ORDER NOW AND SAVE

Just list the items you want and mail this convenient coupon.

SOFTWARE BY ACS

☐ **Monitor No. 3 ..... $29.95**
Complete Machine Language Monitor for TRS-80® features: Find, EDIT, Relocate, Symbolic Dump to Tape, etc.

☐ **Monitor No. 4 ..... $49.95**
All of the commands that reside in Monitor No. 3, plus: RS-232 I/O, Disk Program I/O, Symbolic Dump to Disk for Loading into Disk Editor/ASM., Track & Sec I/O for modification.

☐ **PCLEND $15.95**
Will Patch ASCII files of Basic Programs or text or DATA FILES so that they may be loaded into the Disk Version of the Electric Pencil for Editing purposes comes on Cassette that will automatically create a Disk file of PCLEND.

☐ **MAKE TAPE AND MAKE DISK for Cassette Dealers $69.95**
These are two programs that will allow you to take any type of Program from Disk and store it on tape for mailing purposes. When the user receives the program in the mail on cassette, it is loaded into the computer which will automatically make a Disk file of the program.

☐ **CP/M & C BASIC for the TRS-80®**
CP/M Includes: MOVCPM, STAT, PIP, Dump, DDT, ASM (8080), ED, plus 6 user manuals.
CP/M $150.00
C Basic-2 Includes: XREF2, CBAS2, and manuals.
C BASIC 2 ..... $99.95

☐ **G2 LEVEL III BASIC for TRS-80® ..... Special $39.95**

☐ **TELCOM – Telecommunications for the TRS-80® $29.95**
Telecommunications for the TRS-80® allows one TRS-80® to communicate with another through the RS-232-C over the phone line.

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

Orders received by 6:00 p.m. shipped next day on Master Charge, Visa, Certified Check or Money Order. Personal Checks require 14 days to clear. No C.O.D. Collect calls not accepted. All Hardware warranted through ACS for 1 full year. Software guaranteed for replacement only. Prices subject to change without notice.

Send Check or Money Order payable to –
SOFTWARE • P.O. Box 60097 • Nashville, TN 37206

Quan.	Description	Unit Price	Total
	HANDLING CHARGE $1.50 TENN. RES. ADD 6% SALES TAX **TOTAL**		

☐ Check
☐ Money Order
☐ MasterCharge
☐ Visa
Card No. _____
Exp. Date _____

Name _____

Address _____

City _____ State _____ Zip _____

AUTOMATED COMPUTER SOFTWARE SERVICE
(615) 244-2798
Division of Computer World INC.
625 Main Street • Nashville, TN 37206

80 microcomputing T.M.

DATA

February 1980 Issue #2

APPLICATION

70 **Passing the Plate.** Fleece the flock. *Henry Reikers.*
104 **Process Control.** Control the world. *Robert Hoffman.*

GENERAL

32 **DISK & Other Mysteries.** The definitive disk book. *Harv Pennington.*
36 **The Bottom Shelf.** The bottom line. *Dan B. Shuford.*
106 **ROM Routines.** Basic shortcuts. *Wes Thielke.*

HARDWARE

4 **Cassette Problems II.** Enhance your CLOAD. *Don Stoner & Dick Barker.*
50 **Faster! Faster!** Give your CPU adrenalin. *Dennis Kitsz.*
62 **Video Tune-up.** Focus on your monitor. *David Miller.*
94 **A Simple Interface.** Turn on your 80. *Arthur Mullin Jr.*
100 **LPRINT "Cheap."** Hard times copy. *Fred Blechman.*

HOME

114 **Household Accountant.** Rent? Spent! *David Andresen.*

REVIEWS

24 **Three Mailing Programs.** All for disk. *Reese Fowler.*
26 **Radio Shack's Mailer.** Good documentation. *Dr. Duncan Buell.*
38 **Rival Publications II.** More competition? *Rod Hallen.*
58 **Software Review II.** Utilities galore. *Rod Hallen.*
66 **Percom Drives.** Alternative to Shugart. *D. Buffington & C. Wagner.*
79 **Anadex Printer.** Purple printers. *Walter June.*
80 **Floppy PIMS.** A disk database. *Morris Herman.*
97 **Games Review.** Gambling androids. *Rod Hallen.*
111 **Poor Man's Text Editor.** $10 text editor. *Fred Blechman.*

TUTORIAL

88 **Relocate with PEEKPOKE.** A basic idea. *Irwin Rappaport.*

STYLE

73 **Root Routines.** Get to the root of the problem. *Curtis Gerald.*

UTILITY

42 **BABYBUG.** Miniature monitor. *Dennis Kitsz.*
68 **CLOAD Machine Language.** Modify your monitor. *Mark Schimelman.*
82 **Append it!** Two into one. *Curtis F. Gerald.*
118 **Compress It!** Program shrinker. *David Powers.*
120 **LPRINT Formatter.** Page your printouts. *E. M. McCormick.*
124 **Disk Index.** Keep your programs in order. *Ron Cheshire.*

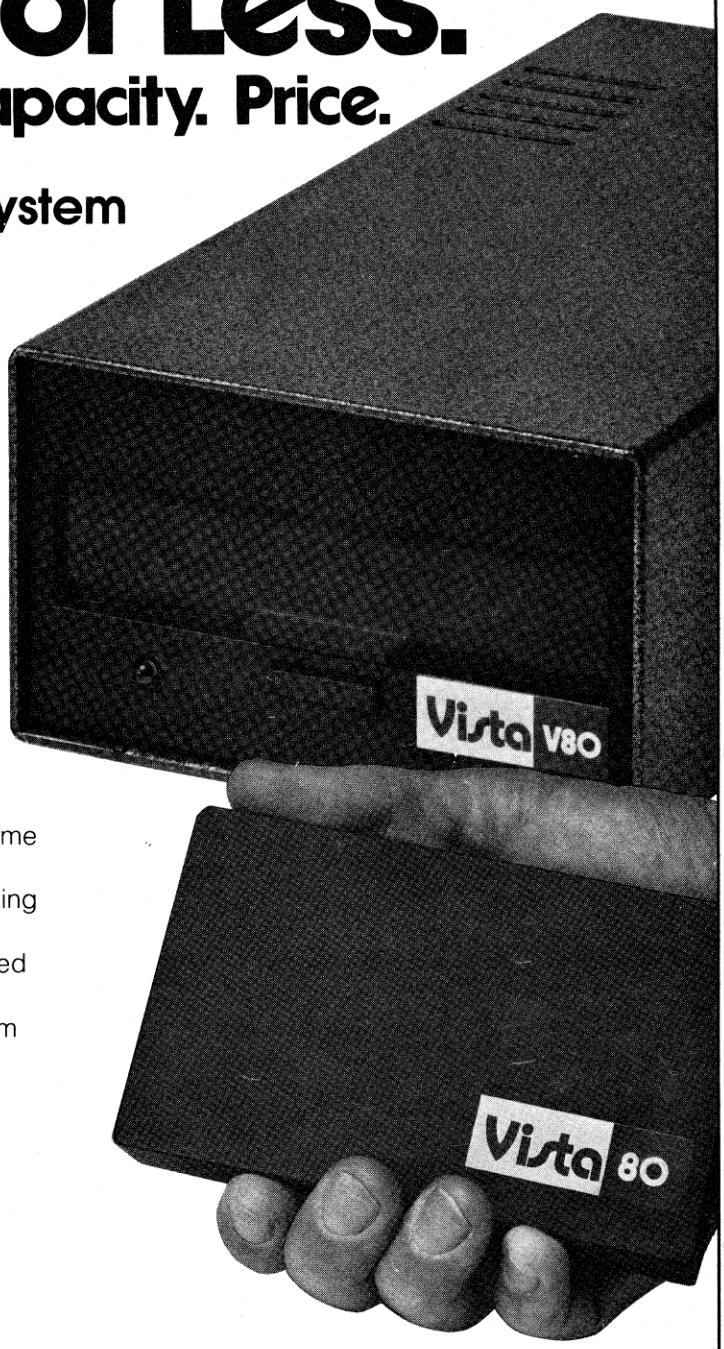
REGULARS

9 **80 Remarks.** *Wayne Green.*
11 **80 Accountant.** *Michael Tannenbaum.*
12 **Captain 80.** *Bob Liddil.*
12 **Input.** *Why not you?*
13 **Club 80.** *Ross Wirth.*
14 **Unlimited 80's.** *Sherry Smythe.*
14 **80 Applications.** *Dennis Kitsz.*
15 **80 Review.** *Chris Brown.*
16 **NEWS.** *Michael Comendul.*
130 **Preview.** *Next month in 80.*
130 **Advertisers Index.**

80 Microcomputing (ISSN # applied for) is published monthly by 1001001 Inc., Pine Street, Peterborough, NH 03458. Application to mail second class postage rate is pending at Peterborough, NH 03458 and at additional mailing offices. Phone: 603-924-3873/4. Subscription rates in the U.S. are $15 for one year and $40 for three years. In Canada $17 for one year and $46 for three years. In Europe please contact Monika Nedela, Markstr. 3, D-7778, Markdorf, W. Germany. In South Africa contact *80 Microcomputing*, P.O. Box 782815, Sandton, S. Africa 2146. Australian distribution by Katherine Thirkell, Sonotron Instruments, 17 Arawatta Street, Carnegie, Victoria 3163, Australia. All other foreign subscriptions $20 (one year only) surface mail. 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.

More for Less.

Speed. Capacity. Price.



The Vista V-80 mini disk system is 8 times faster than the TRS-80, 23% more storage capacity, and costs only $395.

Compare our performance to Radio Shack's TRS-80*. Then match our price with theirs. Then decide which one is for you.

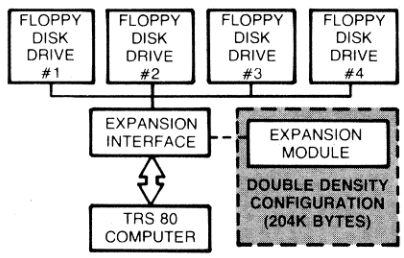
Features

- Vista offers 102K bytes to Radio Shack's 89K. That's 13K more bytes per drive for Vista.
- The V-80 operates at 12ms versus 40ms for TRS-80. Our drive can operate at 5ms, but only 50% of TRS-80 will operate at that speed; therefore, Vista has purposely set the access time at 12ms.
- Totally compatible with all available disk operating systems.
- Upgraded system. Increased storage and speed patch supplied at no charge by Vista.
- Drives are interchangeable for any location from Drive 0—thru Drive 3.
- Immediate Delivery.
- **120 Day Warranty**

Prices:
Single Drive System $395
Two Drive System $770
Four Drive System $1450

*TRS-80 is a registered trademark of Radio Shack, a Tandy Company

TYPICAL CONFIGURATION



Single Density Configuration (102K Bytes)

Vista Expansion Module

The expansion Module provides a double density modification to your current Radio Shack interface that allows you to format diskettes in either single or double density. In double density format, your Vista Drive increases your storage capacity up to 204K bytes on a single 40-track drive.

To insure the highest performance possible, without compromise, we recommend that you use Vista disk drives in conjunction with our Expansion Module.* For a demonstration on your system call **TOLL-FREE 800-854-8017**

Price $239.00 (includes all hardware and software)

**Vista cannot guarantee Radio Shack drives to operate 100% in double density.*

Vista

The Vista Computer Company 1401 Borchard Street • Santa Ana, California 92705 • 714/953-0523

Bigger Is Better.

Expandable storage. Greater Versatility.

The Vista Model II* Disk Expansion system provides one, two or three drives, and adds up to one-half million bytes of storage.

You say you want more disk storage, more programming versatility . . . at a reasonable price. Say no more. Our Vista Model II Disk Expansion System may be just the answer for you. Choose from 1, 2 or 3 drives, already mounted. Each additional disk drive will add about one-half million bytes of on-line storage to your system. Buy only what you need now. As your needs grow, you can continue to expand your capacity by adding another drive. It's that simple.

But the best part we've saved for last. The price. It's cheaper than Radio Shack, yet our Model II will do everything Radio Shack's expansion system can do. The only difference is our Model II will keep on working long after most others have stopped. That's why we are justifiably proud of our product's high reliability. And our **120-day warranty**. Actually, it's not hard to stand behind a product — that works — if you know what we mean.

*Model II is a registered trademark of Radio Shack, a Tandy Company

COMPARE OUR INTRODUCTORY LOW PRICES

1-drive Expansion System $1,000
2-drive Expansion System $1,550
3-drive Expansion System $2,100
Additional Drives Only ..... $ 525

NO WAITING • IMMEDIATE DELIVERY.

NAME _____
ADDRESS _____
CITY _____ STATE ___ ZIP _____
TELEPHONE ( ) _____
OPTIONAL
COMPANY AFFILIATION _____
TITLE _____
INTENDED APPLICATION _____

PLEASE ☑ ONE:
☐ CHECK OR MONEY ORDER ☐ CREDIT CARD ☐ C.O.D.
PLEASE CHARGE MY CREDIT CARD: _____
☐ MASTER CHARGE ☐ BANK AMERICARD VISA
CARD NO. _____
EXPIRATION DATE _____
PRINT EXACT NAME OF CARD HOLDER _____
SIGNATURE _____

Item	Quantity	Price

Subtotal _____
Calif. Res. add 6% sales tax _____
Per-Unit Shipping and Handling _____
($5.00 UPS - $7.50 UPS Blue) _____
Total _____

Vista

✓65

CALL TOLL-FREE 800-854-8017

The Vista Computer Company 1401 Borchard Street • Santa Ana, California 92705 • 714/953-0523

New TRS-80™ Model II

More power, more savings, and quicker delivery to your small business.

More Computer for Your Money! Radio Shack is now delivering TRS-80 Model II, our second computer breakthrough, designed especially for business/professional applications and priced 33% to 66% less than comparable systems from IBM, H-P, DEC and others. The basic Model II system is contained in one desktop cabinet, with a built-in 8″ dual-density floppy and a detachable, full-sized keyboard. A convenient calculator-type keypad is provided. Model II's 12″ high-resolution screen displays upper/lower case alphabet, numerals and 32 graphics characters. Format is 80x24, and expanded characters — 40 per line — are software selectable. Price for the 32K 1-Disk basic Model II system is a surprisingly low $3450.*

RAM Expansion. Model II's are available in either 32K or 64K models and, of course, you can upgrade a 32K to 64K anytime.

Disk Expansion. Model II's built-in floppy stores about ½ megabyte. For more storage, plug in our Disk Expansion Unit with one, two or three drives. Each drive provides another 486K bytes. Price for the DEU with one drive is $1,150.* Drives 2 and 3 are just $600,* each, which means a 64K, 4-Disk Model II, as shown, with *2 megabytes of memory* and heavy-duty desk is yours for just $6,599.* Visit your nearest Radio Shack Computer Center or any of our 7300 stores, or write today for the exciting details.

TRS-80 Model II Power and Features

Software Now Available

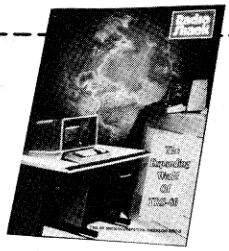
- General Ledger
- Inventory Control
- Accounts Receivable
- Mailing List
- Payroll

More is on the way!

- 32K or 64K RAM
- Built-in 8″ Floppy (500K Bytes)
- Supports 4 Floppies (Up to 2 Megabytes)
- DMA and Vectored Interrupts for Faster Throughput
- Interpreter BASIC (Other Languages Available 2nd Quarter 1980)
- Z-80A Processor at 4 MHz
- Separate Keyboard Processor
- 2 RS-232C I/O Ports, 1 Centronics Parallel Port

* Retail prices may vary at individual stores and dealers.

Free 24-Page TRS-80 Catalog



Write: Radio Shack, Dept. CMA-447, 1300 One Tandy Center, Fort Worth, Texas 76102

Radio Shack®
The biggest name in little computers™

✓5

80 REMARKS
by Wayne Green

The Mind Works Much Like a Computer

Surely I am jesting . . . no one could be serious about using a computer in place of a trained psychotherapist? No, I'm serious about this and I'll explain what I mean.

Firstly, I don't expect the application of microcomputers to psychotherapy to spring full blown into our lives. I suspect it will sort of creep in slowly; at first, possibly, as an adjunct to diagnostic programs used by doctors. You see, we can devise programs to help doctors with their preliminary diagnostics and to take a lot of the work from nurses in writing up patient histories. But most illnesses have an emotional aspect to them and this should be taken into account.

The emotional factor in an illness is always hidden, even from the patient and, if asked about it, the patient won't be of much help . . . not without some input from the subconscious mind. A doctor who tries to treat only the illness and its symptoms, but who ignores the emotional components of the illness, is not going to have a lot of success with the patient.

Ah, but now do we tap the patient and ferret out the emotional problems which may be contributing to an illness, particularly when, as I said, the patient has no awareness of them? Fortunately this is not a complicated matter, once you understand how the mind works. Let's wander off into this branch of computers—the human mind.

There is a tendency to think of the brain and the mind as one. Unfortunately, the more you study the brain, the more confusing much of the operation of the mind becomes. Are you aware that no one yet has even a good hint as to where its memory is? That's right! Oh, we can poke and probe into the brain and restimulate memories to flash back, but this is akin to poking a probe into a computer and bringing a memory out of a disk or a chip. The brain is a complicated switching arrangement, much like a computer.

I have an idea about our memory, but unless you are geared to think in terms of the occult, you'll prefer to wait for further data. Luckily we don't have to know where the memory is in order to work with the brain.

Former Lives

We have some hints on memory. We know that under hypnosis we are able to recall in complete detail all of the percepts from any time of our life. Everything is recorded in the mind in some way. Consciously, we have trouble accessing this data and we term this: forgetting. The mind is much like a computer when you've lost your index to a disk and, though the data is there, you can no longer access it.

This complete recall of a lifetime in full detail tells us something more. When you add up the number of bits of memory making up the percepts of the eyes, ears, nose, fingers and other bodily contacts, as well as temperature, pressure, kinesthesia, etc., you have a formidable storage problem. I suggest that the human brain's storage medium is both infinite and permanent.

There is more to life than the physical body and the physical brain. Experiences of mental telepathy and psychokinesis have been documented. (Perhaps you've seen the movies of a Russian woman moving things without touching them.) If such experiences have been reported for thousands of years, is it so difficult to conceive of another dimension where we store our memories? No one has anything better to offer.

In case you have neither seen hypnosis nor read about it, people can be made to regress easily and experience events from the past with all the realism of a current happening. There is a good deal of argument in scientific circles about this because some hypnotists have regressed people back to their birth . . . and then beyond that, to their conception.

Since our perceptions are being recorded automatically and no recognition or understanding is involved—much like a computer recording a disk or a phonograph record—there is no reason to be surprised that the recording starts fairly early.

This gets murkier when people regress to past lives. People are able to bring up memories of past lives in rather remarkable detail. I've done this a number of times myself and the facts that I've brought up checked out surprisingly well.

As fascinating as the concept is and as real as the memories brought back seem to be, I'm not completely convinced of the existence of past lives. This may be a fanciful construction of the mind, aimed at helping people to handle current problems more easily. There is a need for research on this. I do know that when people are made to experience previous deaths, this can have amazing results in curing present-time complications. My view is this: Whether the previous lives are real or not, using them to help people in their present lives often works.

Cut Out the Middle Man

You don't have to go to a hypnotist to experience your own past. It's right there, easily contacted at any time . . . if you know how. You have to learn how to bypass the conscious mind, that acts as a filter, keeping out subconscious material.

The conscious mind has trouble getting data if it has instructions that program it to have trouble. These instructions can take the form of parents telling a child that he doesn't remember something right. This can take the form of a pain equivalent that turns off a memory. The body, geared for survival, records a painful experience and any surrounding percepts as the equivalent of that painful experience. These percepts are to be avoided in the future. Certain words or phrases can equal pain on a subconscious level without a person's being aware of it. No amount of later programming can erase the permanent recordings, but this later programming does make access virtually impossible.

The instruction to avoid pain seems to be a very fundamental one . . . possibly on a cellular level. Remember that we are made up of cells, each of which has its own life in its environment. And each of these cells is made up of smaller parts which, when put together assume a life. The whole body operates in a similar manner, made up of much smaller parts until it finally gains an awareness of its total self.

There seems to be little difference between physical pain and mental pain. Both are to be avoided. Both poison all other percepts that are recorded at the same time.

Therapy

Some therapists make use of this by putting patients under a light hypnosis, regressing them to painful events and going over them repeatedly, with part of their awareness in present-time. This appears to decondition the painful equations the mind has set up and erase them. (Those of you who are familiar with the method of drawing the pain out of a burned hand by holding it over a fire and getting it as hot as can be accepted have seen this system at work.)

The normal reaction to pain is avoidance . . . .to diminish the pain or ignore it. This has been found to be the worst reaction . . . which figures. Rather if you learn to make yourself aware of the pain, to feel it as deeply as possible and keep feeling it until you can't any longer, you'll find that burns will heal much more quickly, that cuts and bruises will heal quickly, etc.

Talking with a psychiatrist helps a bit. But, if instead of going over the painful memory once, you go over it again and again, until the pain is removed, that trauma will no longer have any affect on your life. The pattern of trauma starts with much feeling and emotion, gradually running through annoyance, then boredom and finally laughter. Once you reach the laughter response the trauma will be gone.

. . . no one yet has even a good hint as to where memory is?

But how can we access traumas hidden in the subconscious? It's a lot like wanting to go directly to the data base but bypass the terminal and its computer. The data is down there, but the terminal has instructions not to access it. The key to accessing the mind's data base is tied into the speed of the system. The terminal (conscious mind) runs slower than the data base. If we can get in there fast enough, we'll be able to get data out before the terminal can shut off the response.

This can be done with "flash" answers. A therapist might ask the person a question and snap his fingers demanding an immediate response, before the conscious mind can interfere. This not only works, it is infallable.

This little response mechanism may seem like a toy at first but you must have the experience and training to handle this sort of thing before you start playing around.

Wet Cobblestones

Just to give you an example of how the flash answer works, I'll tell about a doctor I met down in Bucks County Pennsylvania. I was discussing this concept with him and he thought it was rubbish. That's okay too. But I asked if he would give me five minutes to show him how it worked. He said okay.

I asked him if he had any pains which bothered him. He said no, nothing much. Hmmm, says I, what do you mean by not much? Well, on occasion he would have a very sharp pain in his chest, but it wasn't significant. When the pain came he would give himself a shot of morphine and it would go away. I asked if he knew where the pain came from. No. Would he like to know where it started? Sure, but he felt it was unlikely that it had any real roots.

So I started in. I asked him to say either yes or no each time I snapped my fingers. I then ran words by him, snapping after each word. I got a string of nos as I asked about mother, father, school and a few other obvious things. When I mentioned bicycle I got a yes. Okay, now we're getting there. Did anything come to mind about a bicycle? No? Okay, repeat the word bicycle a few times and let's go back to a bicycle accident.

Suddenly we were there. He was going to a medical class in Paris on a bicycle and he'd slipped on the wet cobblestones. The handle bar had hit him in the chest and knocked him out.

He'd managed to wipe the whole incident from conscious recall (until my flash recalled it), and was left only with the periodic pains. The next step was to go over the accident until the pain was gone in present-time and then, finally, to analyze what was restimulating the pains and remove that.

But no, he felt that the morphine was a satisfactory solution to the pains, so he didn't want to bother with the final analysis.

The whole matter probably could have been cleared up in 15 to 20 minutes, but this would have removed his excuse for the morphine. At any rate, that's how the flash answer system works . . . and I like it because it will always cut through the conscious mind to the data base.

Another hookup directly to this human data base is a physical one. If you hook a sensitive ohmmeter to the hands of a person and start asking questions you'll see dramatic drops in resistance as the hands sweat. This takes place in a fraction of a second and provides clues for a therapist even when the conscious mind is in complete control.

Why the moisture? This may be a response left over from the time our ancestors were grabbing from the trees or a stick to ward off enemies. It's part of the alarm response and tied into our data base when there are pain messages.

It is possible to write a pseudo-psychiatric program which will actually delve into our problems. The good therapist is the one who refuses to get emotionally involved, but rather acts like a computer, searching out clues to the problems. If you get in touch with a patient's data base you'll bypass all his smoke screens and get right to the heart of the problems. From there it is a matter of deconditioning the programming causing the troubles.

If you want to try to do this via psychoanalysis, you are in for years of trying to cope with the conscious mind and it is no help at all because the mind reacts automatically to programmed instructions. The conscious mind is a miracle of rationalization.

Grab Your Coat

For instance, put a person under hypnosis and instruct him that when you touch your ear he will take off his jacket. When you touch your wrist he will put it on. Awaken him and proceed to talk about anything.

When you touch your ear he will obediently take off his jacket. If you ask him why, the answer may be that it was a little warm. So, you touch your wrist and the coat goes back on. Why? Well, since you mentioned it, he thought you may have wanted him to wear the coat.

You touch your ear again and off comes the coat. Why? It was a bit binding around the arms and the material caught on the chair. And so this goes, taking perhaps a dozen offs and ons before the person becomes aware that he is behaving oddly.

A rationalization is a lie which a person believes and acts on. Its control comes from the subconscious mind. Actually, the word "mind" is out of place here because the subconscious is much more like a data base with a host computer and its own programming. It is only the conscious mind of which we are aware and yet much of our behavior is controlled by this subconscious computer that we can't contact. Worse, this damned computer twists things and lies to us, furnishing spurious excuses to explain weird instructions . . . and then prevents us from even thinking about the whole situation. No wonder we have problems.

A rationalization is a lie which a person believes and acts on.

We see these conditioned responses every day, but the reasons behind them remain a mystery. This conscious mind is what the psychoanalyst is trying to work with. It's much like trying to reprogram an intelligent terminal, but without changing the programs in the host computer. Good luck.

Any wonder psychoanalysis takes years and friends of the patients are struck by the slight changes that take place?

I know one person who was flipping out and started going to a psychoanalyst. The condition grew worse because the patient was smarter than the analyst and, as far as I know, the analyst was never able to get by the very smart terminal. They went through many years of this nonsense, investing perhaps $75,000 in this process, the patient never letting the analyst even close to the real problems. The person is still functioning, but as a smiling, treacherous fruit cake and a danger to any who comes near.

One of the problems that a therapist has is his own subconscious, that similarly reacts irrationally to events, governed by the programming built into his own system. This is where a computer may be able to outperform a person.

I'm hoping we'll have some articles on the fundamentals of writing therapy articles, particularly from a perspective that will be helpful to doctors and hospitals taking a patient's history. ■

80 ACCOUNTANT

by Michael Tannenbaum C.P.A.

One of the most desirable new financial software packages is an accounts receivable system. This sub system is customarily linked to the general ledger through the sales journal and the cash receipts journal.

In the past, many attempts have been made to improve the quality of accounts receivable recordkeeping. The bookkeeping machine and the "one write" systems are notable examples. However, these methods, while super for getting data onto an account's record, cannot rapidly analyze data.

An early attempt to mechanize information was the magnetic stripped card and associated reader. These devices looked fine in demonstrations, but as soon as the cards became worn, they usually failed.

There had to be a better way and microcomputers appear to be the answer.

Radio Shack Program

Recently, Radio Shack released its accounts receivable package (stock number 26-1555). As described in its literature, this system has several storage capacity options available.

- 300 customer accounts and 1000 transactions a month (2 disk systems)
- 100 customer accounts and 2000 transactions a month (2 disk systems)
- 500 customer accounts and 2500 transactions a month (3 disk systems)

The key features of the system are:

- complete end of the month billing
- billing statements ready for mailing
- automatic customer account record updating
- total for general ledger postings
- optional advertising message on billing statements
- full A/R analysis including aging status

With these features, the package should find a wide market among small business owners of the 80.

However, before you run out to plunk down 150 hard-earned dollars, there are a few things you should know about the system.

This system is a "balance forward" accounts receivable system, meaning that invoice and payments detail are reflected on the current month's statement only. Historical information is lost.

How does that affect the use of the system? . . . Well, it affects the system in two distressing ways. The first is that keying payments for months prior to the preceding month is difficult, unless a copy is kept of all statements.

However, the second is the most unfortunate. If a payment is received on an account with an old balance, the account status is immediately upgraded to current. This is done despite the fact that the payment may not have been for the outstanding invoice. The machine simply does not have the capability of analyzing the opening balance by due date.

In addition, since there is only one status indicator, an account that has activity in more than one period will be classified with the status of the oldest outstanding amount until a payment is received. Then everything will become current.

From an accounting point of view, a mechanical aging of a file with misclassified status will be useless. It would be necessary to age by hand to develop a legitimate accounting document. In fact, I can only recommend that the user of the system carefully review all his information before making a decision based on the status indicator.

Although these limitations are unfortunate, they need not prevent the use of the system. Aside from some distressing error messages such as "unrecognized error number n in line nnnn press @ to continue," the system has a great deal of utility.

The Set-up Program

In addition to an accounts receivable program, the diskette contains a program that sets the parameters of the system. The set-up program is an interactive procedure which prompts the owner into answering critical questions such as name, address and phone number.

During the account assignment phase of the procedure, it would be advisable for an accountant to be present. Decisions must be made that affect the subsequent recordkeeping of the general ledger.

At the conclusion of the procedure, a password may be chosen. The Radio Shack instructions properly caution the user that loss of this password could jeopardize access to files in the future. No unique passwords should be chosen unless absolutely necessary.

The system operates easily enough. Operations are menu driven, and where a sequence of operations must occur, graphic prompts are used. However, there are some things that the system will not do.

As supplied, the system is not equipped for a simultaneous billing and posting. The system assumes that invoices are prepared by a separate method and that no more than three categories of analysis are required when posting to the receivable file.

This situation is usually found in firms that do not try to analyze sales by type when posting receivables. If all that is posted is sales and postage, three categories will be sufficient. However, if an attempt is made to analyze and post sales by more categories, i.e., nuts, bolts, screws and nails, three will probably not be sufficient. If a sales analysis is desired, it may be very difficult to customize the entry routine to facilitate simultaneous billing and posting.

Transactions of all types—sales, cash, corrections, bad debt write-offs—are entered into a holding file. No attempt is made to update the accounts until a posting session is run. Therefore it's a good idea to check the system status to see if current accounts have been posted before using the account scanning routine.

When posting is done, a complete audit trail is printed out along with a summary of general ledger account activity to date. At the end of the period, a similar report is prepared during the closing routines. There is no shortage of hard copy documentation.

Referencing

Routines reference an account by number or name to increase accessibility. This is a real assist because it eliminates the need to remember the assigned account number.

It is however, achieved with some penalty. After maintaining its account files, the system pauses to sort the account by three keys: number, name and zip code. This creates an index which points to the account location in a random file. With this technique data can be rapidly accessed. A super idea, but as performed in BASIC, verrrrry slow—be patient.

Once an account is located, the routine can modify the information, search for the next account or print the contents of the screen for a hard copy reference. The customer's file as of the last date posted is provided.

As in other Radio Shack projects, the user documentation is adequate. Although the accounts receivable package is primitive, it is precisely what the small office needs.

But the technical documentation is inadequate and this is unfortunate. So much good programming is available in this package that it almost cries out for customizing. Tailoring the system will be quite a chore, however, in the absence of even the most cursory file descriptions and variable lists.

In summary, the accounts receivable system represents a good first-effort but hopefully, Radio Shack will introduce new versions of the package to improve its weaknesses. ■

If you have business software or special routines to share, please do not hesitate to write. The popularity of the 80 is its greatest benefit. I am sure that your problems and solutions can be of help to others. If you wish to contact me, the address is: 42 Bulaire Road, East Rockaway, New York 11518.

CAPTAIN 80

by Bob Liddil

Here's Captain 80, sitting in an empty microcomputer lab after a hard day's work. Still disguised in togs, jeans and a blue pocket T-shirt—the trademarks of a mild-mannered programmer-reviewer—I reflect on events from my past.

I am constantly bombarded by the chit-chat of non-computerists, each a sort of sociologist claiming to know exactly who you are out there in the micro-consumer public. Each of them, a specialist in the field of marketing or advertising, swears that you are a businessman, devoted to accounting, bookkeeping, and the simplification of the secretarial skills thereof. Another swears that you are a dyed in the wool hobbyist eternally delving, like mathematical miners, ever deeper into the mysteries of programming.

These people agree on one point: You don't like games.

Just Another 13-Year-Old

I know a computer hobbyist in Cincinnati. His name is John Lindeman Jr. He's not a math whiz or a genius or even a great programmer. As a matter of fact, when I first met him, he was just another gangling, slightly shy 13-year-old, with no general direction in life except the inevitability of growing up. Then he saw my TRS-80.

The speed of Little John's metamorphosis was awesome. Within weeks of his first introduction to my computer, this boy's agile mind had absorbed everything I could possibly teach him about BASIC on the TRS-80.

His father has not ceased to be astounded at Junior's transformation from an average student, with a 13-year-old's sense of responsibility, into an intelligent, studious, cooperative young man.

From the first, computering was a dad/lad activity with the Lindemans. Although there were business and financial programs available even then, it was Star Trek and Wumpus and the Artillery from the People's Computer Company that glued Little John to my computer. Each Saturday thousands of sacrificial Klingons littered the floor of my computer room after LJ and his dad left.

Is this an isolated situation? No.

A Case in Point

Consider a children's home in Southern California, where I worked before coming to New Hampshire. Charlie S. was a handicapped youngster about the age of 12. His attention span was very brief and his patience with devices was very limited.

When I brought my TRS-80 into his classroom, he took one look at it and demanded, as only Charlie could, to be the first one to play. I use the word play, though, frankly, it was not my intention to introduce games of any kind to this situation.

I had written a simple fairy tale with print statements onto cassette. No graphics, just plain old reading—electronic reading.

The story was about the virtues of calmness and self-control. The hero was a Quiet Apple. Whether it was the story or the computer or a combination of both that impressed this disturbed little guy is hard to say. The point is, Charlie went out of his way to be a Quiet Apple and did not hesitate to point it out whenever his behavior was appropriate and no one mentioned it.

Nor was Charlie the only child in this situation who was turned on by the computer. This same agency maintained Group Homes, domestic settings for the delinquent or disturbed. As I was assigned to each of these homes as a counselor, my computer went with me. CLOAD Magazine's Spelling Bee got more of a workout than any other single program that traveled with me. This seventh grade spelling lesson, thinly sugar-coated, was at the top of the demand list for kids who would've done *anything*, rather than read a single word.

These supposedly unteachable kids, at least for the time I was watching, fell in love with learning. When it came time for me to leave, I believe that they cried more over the loss of my computer than me.

The Gamers

What really burns me up is the way some segments of the computer software industry low-rate the intelligence and importance of gamers. They are making a big mistake.

For every single business user of the TRS-80 there is a child, or teenager, a son, a daughter, nephew, neighbor or whoever, waiting for a chance to use that marvelous computer for fun and games. It doesn't take a genius to figure out that even the lowliest of graphic drawing games still requires an increased attention span and deliberate eye-hand coordination.

Here's Captain 80, in his civies, sitting in a darkened microcomputer lab, surrounded by hundreds of thousands of dollars worth of hardware. I can't help thinking about Charlie and thousands of kids like him whose interest in computers, or in learning itself, depends on the imaginative simulations of programmers. Games, if you will. There's room in the industry for everybody. ■

80 INPUT

Future Pattern

I would like to obtain the answer to several questions about your new publication, *80 Microcomputing*.

The promotional brochure indicates that the magazine will be written for the relative newcomer to microcomputing. Does this mean that the content will be written at a significantly lower level than in the present *Microcomputing* magazine?

I am especially concerned about coverage of machine language routines and techniques and the working software "innards" of the TRS-80, disk operating systems, etc., as opposed to BASIC programs and programming techniques. It seems to me that the former area is one which receives almost zero support from Radio Shack and, consequently, one in which other help is needed the most.

What will be the future role of the current *Microcomputing* magazine? I do not know if it is just coincidence, but the December issue is virtually bare of TRS-80 coverage. Is this the pattern for the future?

I realize that magazines have to strive for a broad subscriber base in order to continue in business, but I am concerned that people like myself who are not beginners or experts will find themselves served by neither publication.

Norman H. West
7511 Valburn Drive
Austin, TX

We hope 80 Microcomputing*'s mix of articles will introduce the novice to computer electronics, while at the same time, it will capture the interest of the small businessman or professional, when applying his computer to accounting or recordkeeping problems.*

Continue to next page

80 INPUT

From previous page

80 Microcomputing *has not forgotten the hobbyist either, and we are collecting articles on expanding the performance of the machine without all the expense.*

Machine language and its usage are the subjects of this month's Applications column, as well as the "Babybug" article by the same author.

In the future, we are planning an issue devoted to programming languages, a guide outlining their strengths, weaknesses and usefulness.

We hope our readers too will let us know what they want to see in the magazine in the future.—Eds.

Why Can't They Explain!

A problem which I ran into and which had me temporarily stopped: I found I just couldn't get the recorder to operate by the TRS-80 manual's (User's Manual for Level I, Catalog No. 26-2101) instructions.

Page 5 of the manual under "Connecting The Cassette Recorder," Notes 1 through 3 refer to CTR-41. There is no explanation what a CTR-41 is and there is no CTR-41 supplied. I contacted the local Radio Shack and they said that CTR-41 was a previous recorder. It has been replaced by a CTR-80 recorder and all references to CTR-41 should be CTR-80.

More confusing was that under Note 3B which read "NOTE: Be sure you always use the Dummy Plug when loading programs onto tape (Recording)." This was followed by a picture of the Dummy Plug. I really tried hard to follow these instructions with no results.

I finally, sheepishly, contacted the Radio Shack again. They couldn't understand the problem and asked that I bring the keyboard, the power supply and the recorder down to their store. I did this and right away they said my problem was using the Dummy Plug.

They explained that the CTR-80 doesn't require this plug while the CTR-41 does. Since I had a CTR-80, I would have to omit the plug. I did and it worked.

Question: Why couldn't Radio Shack have added a revision sheet to their manual, so us novices would know about this? I sure wasted a lot of time on this—nearly gave up.

It would be interesting to know what other beginners did without the help of Radio Shack to get around this error. All the tapes were useless until I got my recorder problem solved.

Funny how one dummy plug can hold up another dummy.

L. A. Smythe, Jr.
Rt. 3, Box 650
Seguin, TX 78155

Super Utility

I would like to tell your readers about a **super** utility program for the TRS-80. It is, sadly, so poorly advertised that without actually using it a person would not realize its value.

This is the Super Graphics Editor from Level IV Products Inc., 32238 Schoolcraft, Suite F4, Livonia, MI 48154. What it allows you to do, incredibly, is directly write graphics characters into strings. You work with the actual graphics themselves, just as if they were numbers or letters. And the BASIC accepts this and will print the graphics strings just like any other string.

For example, you are now able to write directly T$ = " ", which might represent a horse. Then the BASIC command PRINT T$ would instantly print the horse! And the speed is phenomenal, much faster than even POKE graphics. If you've ever seen Android Nim, the technique is the same. In fact you can now list and edit Nim.

Best of all, the Graphics Editor need be in the machine only while you write the program. The finished program can be saved on disk or tape. The Graphics Editor is not needed to run the program.

Level IV has 16, 32 and 48K versions for disk or tape at $14.95. And they offer a money-back satisfaction guarantee on the Editor. I doubt that anyone has taken them up on it, or will.

I am so enthusiastic about the Graphics Editor that I should make clear that I have no connection whatever with Level IV. Some of their stuff (like Oregon) is junk. But the Super Graphics Editor is excellent, excellent value—a real contribution. Buy it. You will never use anything else.

Roxton Baker
56 South Rd.
Ellington, CT

CLUB 80

by Ross Wirth

Thousands of companies and individuals around the world own a TRS-80 and the readers of this magazine represent a portion of this international community. This column serves as a dialogue among TRS-80 owners. By sharing your thoughts and ideas, we can all advance in the use of our computers.

Send your correspondence to Ross Wirth, 15906 E. 96 St. N., Owasso, OK 74055. Please include a self-addressed stamped envelope for an individual reply.

Application Review

Now that winter is here, have you thought about how pale you look? A trip to a tanning center may be just what you need.

Entering a local tanning center, you are introduced to a 4K Level I TRS-80 microcomputer. After answering nine questions about your skin type, an exposure schedule is displayed. This schedule will be used to insure the proper exposure time in the ultraviolet light.

After twenty visits you are tan and healthy looking. The TRS-80 used was standard Radio Shack stock. The cost of the custom program was $75.

Sub-user Groups

The reason people attend user group meetings are as varied as the people who come. Some are there to learn Level I programming, others need to learn advanced programming techniques for DOS. Still others don't care about programming and are interested in learning about new applications or hardware.

How do you please everyone? After a year of trying I have decided you simply can't; it leads only to frustration.

I propose, instead, the formation of sub-user groups on a local scale. The TRS-80 user group will coordinate these local groups and serve as a clearinghouse for member needs. Each sub-user group will consist of three to ten individuals interested in a particular area of study. Meetings will be in homes, organized by a group coordinator. Will it work? I don't know yet.

Similar to the formation of special interest groups on a local scale is their formation on an international one. Communication may be by letters or telephone, but the function will be the same. People with specific interests can pass along information and share programs.

The following groups have already been suggested: genealogy; investment analysis; and medical office procedures. If you are interested in coordinating one of these groups or organizing a group of your own, send your name and address to me. This matchmaking may give birth to some excellent new software and hardware for all TRS-80 owners.

Short Notes

CTR-80 tape recorders with a production run earlier than 2A9 can destroy programs when the play button is pushed and the program tape is over the record/play head. To prevent this make sure the tape is completely rewound before loading the program.

Radio Shack will fix it at no charge.

The Tidewater TRS-80 Users Group publishes a free newsletter. The newsletters are distributed by mail with flyers from various companies who participate in a "cooperative mailing" to share postal costs.

While the newsletter is free, financial realities require mailings to be done with rotating lists. To insure that you receive all the issues, send a donation or extra first class stamps with your request. "Published on rainy weekends," the newsletter comes out on an irregular schedule.

For free newsletters write to TCS, PO Box 10281, Norfolk, VA 23513.■

UNLIMITED 80's

by Sherry Smythe

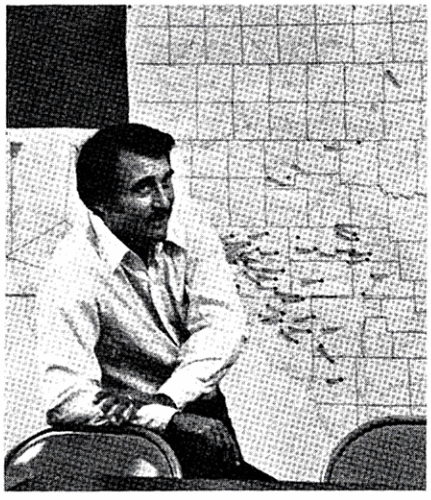
Rick Waits, manager of Radio Shack Computer Center of San Antonio, Texas, arranged for me to see a unique form of computer dating service for members of the American Sportsman's Club which matches hunter, fisherman and/or camper with the appropriate game, fish or campsite at the optimum time and season.

This complex application, utilizing a 48K TRS-80 Business System with tractor-fed printer, 4 disk drives and Rick Farr's custom programming is doing what a microcomputer does best—equalling or bettering larger computers costing ten times the price.

The primary responsibility of the American Sportsman's Club is managing hunting, fishing, and camping leases in Texas and about a dozen more states and foreign locations giving its members and their families the greatest opportunities for outdoor recreation, away from the frustrating pressures of the over-populated public camping, fishing and hunting grounds.

The ASC has many game and fish stocking programs and works closely with landowners assisting in the preservation of game and fish. Also a social club, members and their families can enjoy a wide range of exciting outdoor events with people of similar interests. Special instruction classes are offered where every member of the family can learn new sports and valuable outdoor skills.

Don Albert, southern operations director, guided us on a tour of what he called his "bookie joint." He told us that the TRS-80 and the ASC were united together almost as an afterthought.



Don Albert has ASC's hunting leases pinpointed on a map of Texas.

Someone casually mentioned the TRS-80 in a post-hunting trip discussion. After explaining its virtues and potential for handling club records, it was subsequently reviewed, found fitting for the intended applications, acquired, programmed and put into use.

Their TRS-80 microcomputer system keeps an inventory of which game is where, the census of each, the number that can safely be harvested at any one time without damage to the balance of the species. Simultaneously, it coordinates this data with the specs of the various hunting and fishing rules, regulations and seasonal restrictions for each individual state and territory, compares this with the members' preferences on hunting, fishing, and/or camping and makes the appropriate reservations.

In the pre-microcomputer days booking camping, hunting, or fishing trips and maintaining the inventory of leases for hunting, fishing or camping sites was done with cards (thousands of them). With the visual aid of several big plotting boards and lease maps, coded to show the type of game, who was scheduled to hunt, camp, or fish and when, about forty telephone operators did their best not to send too many people to the same place on the same weekend.

Don said, under the circumstances, the operators did a nice job coordinating all the data manually, but, obviously, it was costly, time consuming and subject to human error. With the microcomputer on line, fewer operators have instantly updated figures literally at their fingertips, providing a better system at a more reasonable cost for the club. ■

80 APPLICATIONS

by Dennis Kitsz

EEven critics will admit that the wworld's first popular microcomputer was designed successfully—its hardware has bbeen reliable and its software vversatiile. Other than a few areas RRadio Shack chose to ignore, or disagreements about physical or electronic aesthetics, the TRS-80 has only a few outright flaws. By now you have gguessed this month's topic . . . that ffamiliar kkeybounce.

KBEEPFIX is a short program which debounces the keyboard input, eliminating those unwanted double letters. That's no big deal, of course, as Radio Shack has been offering its own KBFIX for over a year. But it also provides an audible beep from the cassette output port at every keystroke and includes a feature which repeats a letter if a key is held down.

One advantage of a repeating key, for example, is that the fast scrolling of a LIST can be slowed merely by holding down SHIFT@; a repeating space makes editing lines of BASIC much less awkward.

Using KBEEPFIX

As written, this program sits in slightly over 100 bytes of high memory, but is completely relocatable. You'll probably want to prepare this program for loading each time you use the TRS-80.

Use a monitor such as TBUG to create a SYSTEM tape. Or you could key in the BASIC version and CSAVE it for later use. The machine-language version loads as a SYSTEM tape in a few seconds; the BASIC versions takes less than half a minute.

To use KBEEFPIX, you'll need to answer the MEMORY SIZE? prompt. (Quick question: how do you return to MEMORY SIZE? without turning off the computer? Answer: type SYSTEM [ENTER], then /0 [ENTER]. Thanks to Mike Barton of Barre, Vermont, for that hint).

Answer MEMORY SIZE? with 32654 and load KBEEPFIX. The starting address is /32655 [ENTER]. READY should appear immediately; KBEEPFIX is operational.

Connect an audio amplifier to (or in parallel with) the cassette cable's output plug, and your upgraded TRS-80 is ready to use.

How It Works

KBEEPFIX patches into the TRS-80 "switchboard"—an area of memory through which various segments of TRS-80 software-in-ROM communicate. [See "Babybug" elsewhere in this issue for background on working with machine language.]

Every keyboard search jumps off from the address specified by bytes 4016 and 4017 hex. We can go our own way by changing those bytes (in this case to point to the start of

Continue to next page

From previous page

KBEEPFIX).

The program now scans the keyboard, but instead of locking out a key continuously held down, it initiates a countdown. If the key is depressed through 256 keyboard scans (about ½ second), then the letter is repeated. Otherwise, the loop begins over.

Because muscle motion is slow, a keystroke can be validated by checking the keyboard twice in a very short time. If the same value is not found, a bounce is assumed, and the program ignores that key.

Once a keystroke is accepted, a few program loops tweak the cassette output port, causing a beep. That done, the dust is brushed off, all registers are tidied up, and KBEEPFIX steps back into ROM, hopscotching over the TRS-80's built-in bouncy keyboard scan. The keyboard byte is tucked neatly away in the accumulator, and no one is wiser but you.

Once I started using this program, I began to wonder how I put up with a silent, bbouncy keyboard for so long.■

```
7F80 *    00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 21
7F90 *    98 7F 22 16 40 C3 19 1A 21 36 40 01 01 38 16 00
7FA0 *    0A 5F A3 20 1A 77 14 2C C8 01 79 D6 80 20 F1 7E
7FB0 *    06 07 2D 86 10 FC FE 00 3E 00 C0 32 1A 40 C9 A6
7FC0 *    28 10 3A 1A 40 3C 32 1A 40 FE FF 20 D9 3D 32 1A
7FD0 *    40 7B 73 C5 01 00 02 CD 60 00 C1 0A A3 C8 C5 E5
7FE0 *    F5 06 40 3A 3D 40 E6 FD 67 F6 02 6F 7D D3 FF 7C
7FF0 *    D3 FF C5 06 40 10 FE C1 10 F2 F1 E1 C1 C3 FB 03
```

For 32K, change addresses to BF80 through BFFF, and change byte BF91 to BF.
For 48K, change addresses to FF80 through FFFF, and change byte FF91 to FF.

KBEEPFIX—Machine Version (16K)

```
10   FOR X = 32655 TO 32767 : READ A : POKE X,A : NEXT
20   POKE 16526,143 : POKE 16527,127 : M = USR(Ø)
30   DATA 33
40   DATA 152,127,34,22,64,195,25,26,33,54,64,1,1,56,22,0
50   DATA 10,95,163,32,26,119,20,44,203,1,121,214,128,32,241,126
60   DATA 6,7,45,134,16,252,254,0,62,0,192,50,26,64,201,166
70   DATA 40,16,58,26,64,60,50,26,64,254,255,32,217,61,50,26
80   DATA 64,123,115,197,1,0,2,205,96,0,193,10,163,200,197,229
90   DATA 245,6,64,58,61,64,230,253,103,246,2,111,125,211,255,124
100  DATA 211,255,197,6,64,16,254,193,16,242,241,225,193,195,251,3
```

For 32K, change DATA line 40, second value, from 127 to 191. Poke 16527 with 191.
Also, line 10 should read FOR X = -16497 to -16385 : (etc.)
For 48K, change DATA line 40, second value, from 127 to 255. Poke 16527 with 255.
Also, line 10 should read FOR X = -113 to -1 : (etc.)

KBEEPFIX—BASIC Version (16K)

80 REVIEW

Free Lance Software Publishing
by B. J. Korites
133 pages, paper bound, $14.95
1979 Kern Publications—PO Box H211
Littleton, MA 01460

by Chris Brown
80 Staff

I liked it. The book is insightful, concise and easily worth the cover price of $14.95. If you are serious about programming, sooner or later you'll want to make it pay and *Free Lance Software Publishing*, by B. J. Korites, may be the most comprehensive book on the subject available.

The book is impressive in both its scope and detailed treatment of the subject matter. Within this compact 130 pages the entire gamut of the publishing process is explored. In addition, most aspects of software salesmenship are dealt with.

Topics include: Direct Sales of Software, Service Bureaus, Selling through Intermediaries, User Groups, Case Studies (Winners and Losers), Selling Techniques, Writing the User Manual, Contracts, Pricing, Theft Protection and Tax Considerations.

No-nonsense Approach

The to-the-point, no-nonsense approach of the author is refreshing in this era of technical nonspeak. For instance, the foreword:
"This book is about money and how to make it by writing and selling computer programs. It has been written for:

- individuals who are employed full-time and want to sell their programs as a sideline
- professionals who have a marketable program idea
- established small software businesses
- owners of personal computers who want to capitalize on their investment"

The text proceeds, "Reduce the probability of breaking in on people at the wrong time by making your business calls in the early afternoon: Never between nine and twelve in the morning." Sound advice.

"Always have a name to direct your calls to in a large corporation. The higher up the managerial ladder the better." Amen.

And finally, "Never press for a decision over the phone. It is much too easy for the person at the other end of the line to say no instead of yes."

As implied in the foreword, the intended audience are savvy software authors who are after the big money. Small and medium size corporations or affluent user groups like doctors and lawyers make up the market for these semi-pros—low volume-high priced situations. The nickel and dime world of hobbyist software is not where the money is as far as Korites is concerned.

While he has no illusions about the difficulty involved in making money with low priced, mass-marketed software, Korites does devote a chapter to this facet of the business. He opens by saying: "It is easy to see that after deducting various costs, there isn't much profit left over to justify investment in producing software for this market unless it is to be sold in high volume."

This may be true, but it is also true that many authors find their game or business software does sell in quantity once published by a large marketing organization. The book gives little attention to this phenomenon.

The omission is especially unfortunate since large marketing organizations are often the first outlets fledgling software authors try in the rough and tumble world of freelancing. At this stage of their careers, authors most need guidance and usually can't get it.

Most software authors are concerned about theft. Korites offers the following advice: "Low cost software, sold through computer stores or directly to users, is probably as good as gone after you make your first sale. Time is critical, sell as much as you can as quickly as possible". A realistic attitude in view of the proliferation of disk drives on micros these days.

It seems clear that the skills necessary to write a saleable program are entirely different from the skills necessary to sell that program. Once this distinction is made, the value of a good publishing reference becomes obvious.■

80 NEWS

edited by Michael Comendul

The TinyFORTH

The Software Farm is producing the tinyFORTH 2.1 computer programming language system. A cassette and users manual makes tinyFORTH, a version of the FORTH language, compatible with Radio Shack TRS-80 computers.

TinyFORTH runs faster and uses less memory because it includes a compiler in addition to an interpreter.

The tinyFORTH language is based on a dictionary of words. Each word is a small program. The user can expand the language by defining new words. The size of the language is limited only by the available memory. The language is so compact that the 300 words supplied with tinyFORTH 2.1 occupy only 8K bytes of memory.

The tinyFORTH system also includes a powerful text editor, a Z-80 assembler and a graphics package.

The user's manual is written in a tutorial format with special chapters explaining the graphics package, the text editor, the assembler and advanced applications.

The complete tinyFORTH 2.1 system for a 16K (or larger) Level II Radio Shack TRS-80 computer costs $29.95. Shipping and handling charges are $1.50 per unit in the United States and $6.00 per unit foreign. Write: The Software Farm, Box 2304, Reston, VA 22090. ↙109

Authorware for the TRS-80

MicroGnome's CAIWARE is a software system for authoring and using computer assisted instruction on the 16K TRS-80 with LEVEL II BASIC. The author is guided by a set of well-defined prototype questions.

The range of subjects for CAIWARE includes: history; religion; auto repair; math and sales training.

The first prototype question is multiple choice with up to eight choices. The number of attempts the student is allowed is the number of choices minus 1. The second prototype provides the short answer format.

The author may specify up to three correct answers, which may be spelling variations of the same answer or totally different, but correct, answers. The student is allowed three attempts. The author may copy elements from a preceding question by depressing a single control key.

The last line of the screen is reserved for instructions to the author when in author mode.

MicroGnome's CAIWARE program is available on cassette for $24.95. Write: Fireside Computing, Inc., 5843 Montgomery Road, Elkridge, MD 21227. Tel. (301) 796-4165. ↙120

TRS-80 Grafix and Tidy

Grafix lets you design graphics directly on the display. The graphics are coded into string variables and added as lines to your BASIC program. Design tools include up, down, left and right cursor control and display of stored graphics character.

Display repeats if the key is held down. Any graphics character can be easily built from keyboard by setting or resetting graphics bits.

Tidy renumbers, deletes REMs or spaces. Renumber does not insert spaces into program and considers all Level II statements with line numbers.

Both programs are available for 16K Level II and can be loaded without affecting BASIC programs under 14K already in RAM.

Priced at $9.95 each, for further information write to Software Specialists, PO Box 845, Norco, CA 91760. ↙146

Software Turns TRS-80 Into Printing Calculator

Make your TRS-80 (Level II, 16K) into a printing calculator with Manhattan Software's new Calculator Plus.

The program works well as an on-screen calculator without a printer, providing chain and mixed calculations with a print command to record intermediate steps, if desired, and the final answer.

Entries in long add-and-subtract operations are checked with an on-screen review command or printed out for verification and a permanent record. Works with printers down to Quick Printer II size.

Significant figures are retained in a separate memory section.

An optional dollar format is available. Cost is $9.95.

Write: Manhattan Software, Inc., P.O. Box 5200, Grand Central Station, NYC, NY 10017. ↙90

GB's C-LOADER

The C-LOADER eliminates CLOAD and CLOAD? problems encountered in many TRS-80 computers. It connects between the ear jack of the cassette deck and the cable to the keyboard. It requires no tools for installation nor electrical power or batteries.

C-LOADER works with the Radio Shack CTR-41 or CTR-80 decks, as well as other decks. It allows you to CLOAD tapes and verify your dumps with CLOAD? at any volume setting above a minimum level.

Two models are available: The Model 800 is for programs generated on your TRS-80 and most commercially available tapes. The Model 810 is for the few machine language tapes which are recorded with inverted polarity.

The price is $12.95 each, postage paid. For more information write: GB Associates, PO Box 3322, Granada Hills, CA 91344. ↙147

MATH Library I for TRS-80

This 22-program package written in Level II BASIC, requires 16K of RAM and contains most of the elementary numerical methods for solving various scientific problems.

The package includes: root of equations, differentiation and integration, simultaneous equations including sparse and tridiagonal systems, matrix operations and much more.

A users manual with sample runs, program descriptions, listings, brief theories and sources is included. Two versions are available: TRS-80 disk for $35.00; and Level II cassette tape for $32.00.

For orders and further information contact: Dr. Lee, 5819 Thomas Ave., Philadelphia, PA 19143 (215) 748-4558. ↙110

Business Software Packages

Micro Architect has converted their present software products to run under the new TRS-80 Model II computer.

Most of the programs have been extensively modified to make full use of the expanded hardware capabilities, such as a bigger screen, more memory and disk storage.

Micro Architect guarantees all Model II programs to run and will provide any bug fixes and enhancements.

Initial offerings are: an advanced mailing list system at $99, a word processor at $49, A/R at $149, an inventory system at $149, and a data base manager without user programming at $199.

Future offerings will include A/P, general ledger, payroll, etc.

All packages are integrated together and also

*Reader Service number follows each entry.

can run stand-alone. Prices include 8" diskette, full documents, postage and one year newsletter. A 10% discount is allowed for the order of more than one package. Send $10 for each manual, and two self-addressed, long, stamped envelopes for catalogs.

Please mark down "MODEL-II" to: Micro Architect, 96 Dothan St., Arlington, MA 02174. ↙54

TRS-80 Time-sharing Package

ST/80 (Smart Terminal Program), enables a TRS-80 to act as a dial-up terminal on any standard time-sharing network.

Three versions of ST/80, by the Small Business Systems Group, are available on tape or disk.

ST/80-1 utilizes a 16K with the expansion interface, RS-232 board and the Interface I or Interface II with optional printer support. $39 on cassette.

ST/80-2 utilizes the same configuration as above with one disk drive. $79 on disk.

ST/80-3 includes a disk based editor to create data and program files which can be transmitted to the central facility. Data transmitted from the central computer can be directed to a disk and later edited using a text editor. Data can be stored on a disk and later transmitted from the TRS-80 disk to the central computer. $150 on disk.

Write the Small Business Systems Group Groton Road, Dunstable, MA 01827. ↙18

New Dual Drive

The Micro Squared M-250 dual drive unit is capable of single or double density and consists of two double-sided drives, a power supply, cable, and chassis, selling for $1195. It is completely assembled and comes ready to plug in and run.

The M-250 unit, in addition to quadrupling capacity, features a write protect sensor that prevents overwriting the protected diskettes.

It has time erase timing circuits internal to the disk drive and a diskette-in-place sensor used to stop the spindle drive motor.

The life expectancy of the read/write/erase head assemblies of the M-250 is over 20,000 hours with an expected media life of 3 million passes on a single track.

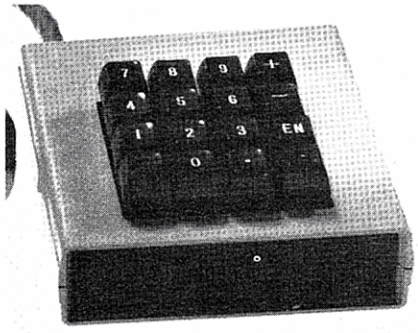
For further information contact Micro Squared Inc., Suite 5B, 5131 Owensmouth Ave., Canoga Park, CA 91303. ↙35

New Interface

The PC80 Interface comes complete with power supply and bus interface connector and houses up to four special function circuit cards.

The cards presently available are: 32-differential channel A/D, 12-bit resolution, 10uS conversion Time; 8-channel D/A, 8-bit resolution; 2-channel Serial I/O for RS-232 or 20mA current loop, software-programmable baud rate; 48-bit parallel I/O; Real Time Clock, with battery back-up for time-keeping when power is off; Floppy Disk Controller for both 5¼" and 8" drives; 2716 EPROM Programmer; Floating point arithmetic processor

The PC80 will handle up to four of any combination of these cards at one time. Prices start at $385 in single unit quantities. Write Applied Micro Technology, Inc., P.O. Box 3042, Tucson, AZ 85702. ↙148



External keypad from VR.

External Calculator Keypad

Add a calculator to your present TRS-80 system with a new external computer keypad available from VR Data.

The keypad offers 14 operations—a double-width zero key, numerals 1-9, plus, minus, and enter—in an easy-to-use calculator format.

The external keypad is available with connector for $64.95 from VR Data, 777 Henderson Boulevard, Folcroft Industrial Park, Folcroft, PA 19032. ↙31

ACCT-III for 80

ACCT-III consists of three programs that carry out the on-line account receivable functions of a small business or a medical clinic. It is specifically designed for the TRS-80 system.

Simplified accounting operations can be executed by a person who has had little accounting and computer experience. It requires DOS 32K memory and a printer.

The system is not totally invoice-oriented so that any service business, including doctors and retail stores, can also use it.

The three programs are: initialization, account manager and report generator. Account manager lets you maintain a data base. The transaction can be paid invoice, unpaid invoice, credit, debit or payment. Subcommands let you search, display, print, update, delete records. Order entry allows multiple items to be entered one after the other. Unit cost, quantity, sales tax rate and shipping cost will be asked for input. The total amount will be calculated by the system. Invoice can be printed if desired.

Reports consist of sales journal, receipts journal, aging analysis, end of period processing, data base lister, labels and statements. A consistency check is included to ensure data integrity. Batch report lets you produce three reports without operator intervention. Most reports have title, subtitle, data and automatic generated page number. The number of lines on a page can be adjusted by an on-line command.

A fast SHELL sort is used to sort names. Standard forms for statements can be ordered from the specified vendor.

Two files are used: customer file and transaction file. With a dual disk system you can store 300 accounts per disk and 3000 transactions per disk. End-of-period processing allows unlimited transactions. Each transaction is twenty-five characters long. The account record has remark fields to allow notes for the account. Examples of uses are patient history, special attention, etc.

Single disk operation is allowable, but will cut the data base size in half.

The package is priced at $69. A 24-page document is available for $5 and will be credited towards purchase. The package is about 30 pages of BASIC source listing. For more information, write Micro Architect, 96 Dothan St., Arlington, MA 02174.

TRS-80 Automatic Phone Dialer/Timer

Blechman Enterprises has made available a Telephone Dialer Program for the TRS-80 Level II. The program is written in BASIC and holds up to 500 names and telephone numbers in 16K memory or 30 names and numbers with 4K memory.

The operator ENTERs the desired name and it is displayed with the name, number and area code. The computer displays each digit as it dials.

The dialing speed, fast or slow, is accomplished through an external telephone interface circuit consisting of a Radio Shack $3 relay, SPDT switch, LED, resistor and 9V battery. The relay contacts are connected in series with the telephone line. This system can be used with all telephone circuits, rotary or tone.

The user presses ENTER when the party at the other end picks up the phone, while the computer displays the length of the call continuously by the second. The user can even determine the rate for the call.

The program is furnished on cassette with a schematic and parts list. The flip-side of the cassette has the Phone Toll-Charge Program, which keeps track of the time and charges of any call, **without** connection to the phone.

The price is $10 including shipping. For more information write Blechman Enterprises, 7217 Bernadine Ave., Canoga Park, CA 91307. ↙149

Workshops Sponsored

Virginia Polytechnic Institute and State University is sponsoring a short course on interfacing with the TRS-80.

The programs will be directed by Dr. Jona-

than A. Titus, Dr. Paul Field, Dr. Christopher Titus and David G. Larsen.

The workshop course will be held March 17 and 18.

For more information contact Dr. Linda Leffel, CEC, Virginia Tech, Blacksburg, VA 24061. (703) 961-5241.

Dungeons of Xanadu

Software Exchange announces Dungeon Explorer, by Matthew Kiriazis, an adventure game for the TRS-80 Level II computer.

Dungeon Explorer is a single player game of combat and adventure. Deep underground in the Dungeon of Xanadu there are incredible treasures.

The object is to survive these dangers and become a legendary super-hero.

Each trip into the dungeon is different. The game can last for hours.

Dungeon Explorer is available for $8.50, including cassette and instructions. Requires TRS-80 with 16K and Level II BASIC.

Write: Software Exchange, 2681 Peterboro, W. Bloomfield, MI 48033. ↙151

Patching Editor/Assembler

DISK*MOD is a machine language program for TRSDOS 2.2, NEWDOS, or VTOS 3.0, that will patch the Radio Shack Editor/Assembler to provide disk I/O for the text buffer as well as support object files written directly to disk.

Block move and global change commands are added. A directory (including free granules on a diskette) command recovers vital file specifications while within the Editor/Assembler. If necessary, files can also be killed under Editor/Assembler control.

DISK*MOD offers protected memory and pagination.

The modification includes a patch for generating sensible output of DEFM expansions on assembly listings. The symbol table is sorted and printed five to a line.

Scroll UP/DOWN holds 15 lines of source code.

Two versions of the Radio Shack Editor/Assembler are available. The MISOSYS cassette contains versions to modify either release of the Editor/Assembler.

Requiring a minimum of 32K and 1 drive to perform the modification, DISK*MOD is priced at $20. MISDSYS, 5904 Edgehill Drive, Alexandria, VA 22303. ↙152

Low Cost Expansion

The LNW Research System Expansion for the TRS-80 is a PC Board that you can build yourself.

The LNW parts list offers you the options of 32K memory, floppy disk interface, screen printer bus, parallel printer interface, serial interface, dual cassette interface and power supply, all on one board.

The low cost PC Board comes complete with a users manual for $59.95.

Write LNW Research, 8 Hollowglen St., Irvine, CA 92714. Add three dollars for postage and handling. ↙53

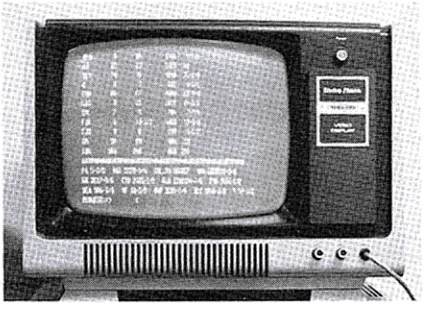
Upper and Lowercase

A kit, which includes everything you need to install lowercase on your TRS-80—wire, solder, control key, 2102 memory chip, slide switch, and mounting hardware, is now available with complete documentation.

It starts with instructions for static-proofing your workbench and ends with an explanation of how the modification works. Pictures and diagrams make the installation as easy as possible.

The low power 2102 memory chip is connected to a slide switch which allows the TRS-80 to be used with or without lowercase. To minimize the chance of damage to the memory chip the wires harnessing the chip and switch together have been pre-assembled.

The lowercase & Keyboard Modification Kit ($19.95) is distributed by Emmanuel B. Garcia, Jr. & Associates, 3950 N. Lake Shore Dr., Rm. 2310, Chicago, IL 60613. ↙153



Wire service from Max Ule.

Financial Wire Service

A financial wire service developed by Max Ule & Company Inc. and Intersystems Software Inc. has been designed to run on the Radio Shack TRS-80.

According to Jay Moskowitz, president of Intersystems Software and developer of TICKERTEC-TRS, "This system allows the individual to watch the entire New York Stock Exchange or American Stock Exchange ticker tape on a real time basis with no delay as well as monitor 48 or more stocks, keeping track of their last sale prices and volume.

"He can maintain the stocks in either of two alphabetical lists so that speculations, for example, might be kept separate from long term holdings."

The cost of this software starts at $1,000, plus a monthly service fee charged by the exchange.

The system is available for immediate delivery on cassette or disk with custom options available on request.

Write Max Ule & Company, 6 East 43rd Street, New York, NY 10017. ↙154

MMSFORTH for the TRS-80

The MMSFORTH System Diskette supplies full reliable disk I/O, virtual memory, double-precision integer math (to +/− 21,000,000.00), auto repeat keyboard, variable-character blinking cursor, sophisticated in-line editing, Level-II-BASIC-like string-handling and arrays, user-callable disk and/or tape I/O and a function to screenprint to your line printer.

MMSFORTH is compatible with versions from FORTH, INC. and the International FORTH Standards. It includes full source code which enables the user to precompile modified versions of MMSFORTH for specific applications or simply to examine it.

MMSFORTH has its own disk I/O and is run on the TRS-80 without TRSDOS.

FORTH is a structured language similar to PASCAL except that, in FORTH, the programmer defines additional commands as they are needed.

The commands can be immediately compiled into the language and subsequently deleted.

Assembler code can be inserted directly in the FORTH program using a function of the language itself. This eliminates the need to preload assembler routines as in BASIC.

A microFORTH PRIMER, for the beginning FORTH programmer, is available separately for those who wish to evaluate FORTH before buying. The microFORTH PRIMER costs $15.00 (Mass. orders add 5% tax) and $2.00 shipping/handling.

The MMSFORTH System Diskette costs $64.95 (Mass. orders add 5% tax) and $1.00 shipping/handling.

Write Miller Microcomputer Services, 61 Lake Shore Road, Natick, MA 01760. ↙112

Project Management

Charles Mann & Associates has released a new TRS-80 software program for the Engineer, Architect or General Contractor.

The program, called the Project Management System, is designed for project managers and includes elements for project planning, bid preparation, budget control and cost management.

The project management system is designed for financial control of a work assignment and can be updated by available cost data. The system includes elements for budgeting, cost recording, variance reporting and completion cost estimating.

The system requires a 32K TRS-80, Model 1, Level II processor with at least one disk drive.

Available from any of CMA's 650 dealers worldwide for $69.95, you can write Charles Mann & Associates, Micro Software Division, 7594 San Remo Trail, Yucca Valley, CA 92284. ↙155

Business Planning Package

This disk-based package contains a set of forecasting programs which can project sales, inflation and even stock prices.

THE VERY BEST.

If you're serious about your TRS-80 computer, try these disk based programs. When it comes to hardcore software, noboby does it like **TBS.**

BUSINESS MAIL SYSTEM by Dale Kuber is designed for large-scale business users. Requiring 32K, two disks and printer, this program will store up to 150,000 names in a single file spread out over multiple disks. Each data disk holds 500 names. After data entry, **BMS** automatically sorts the data by zip code and alphabetical order within the zip code. The program tells you when and which data disk to insert, expanding your files automatically until you've reached 300 disks. Data is input directly onto formatted screen display with the option to use Company Name/Attention instead of Last Name/First Name. Three numeric and one alpha code fields are provided to help you use the search and printout mode. **BUSINESS MAIL SYSTEM** allows you to program the number and spacing of your labels and then print out and read your data disks concurrently using accelerated printing. (This mode works only with Centronics printers.) With more features than can be described here, this high-powered program sells for $125.00.

ANALYSIS PAD by Del Jones is the epitome of first-class programming in business applications. Requiring 48K, and one disk with a printer recommended, this columnar calculator gives the user tremendous flexibility in data entry enabling the user to create 30 or more columns and rows. Enter your own column and row labels. Enter your data by row or column or directly onto screen display via edit mode. Move, swap, delete, and add rows or columns. Create new pads by stripping relevant data from old files. You never have to key in data twice. But more important than the powerful data manipulation provided, you can add, subtract, multiply and divide one column by another and put results in another column. You can perform up to six calculations on one column and even define one column to be a constant. The calculation routine you create can be saved and reused. Print out the entire pad in four column segments to line or serial printer. **ANALYSIS PAD** was originally advertised for 32K tape at $32.50. Since then it has been totally rewritten and expanded to its present 48K disk only form and sells for $49.50. It is easily worth twice as much. You have to see it to believe it.

DATA MANAGER by Dale Kubler starts out where INFORMATION SYSTEM leaves off. Requiring 32K and one disk, it accepts up to ten user-defined fields with up to forty characters per field and 255 characters per record. As with all TBS software, data entry and editing is professional and simple to use. What makes this program stand apart from "in-mem" data managers is that it uses up to four disks on line as memory, or as much as 320K of memory storage. Because disk sorts take more time than in-mem sorts, **DATA MANAGER** enables the user to create and maintain up to 5 "key" sort files for quick access of data. A utility program is provided to calculate the number of records possible since the amount of records you can maintain is dependent on a number of variables. This program also supports the upper/lower case modification, and printouts can be programmed to almost any format and sent to line or serial printer. For Centronic printers, accelerated printing is provided enabling the computer to search and print at the same time. If you already have INFORMATION SYSTEM, **DATA MANGER** will accept those files. (We are currently working on a program that will merge your data files with Electric Pencil files.) A necessity for organized people, this program sells for $49.50.

CHECK REGISTER ACCOUNTING SYSTEM, adapted for the TRS-80 by Dale Kubler and originally written by O.E. Dial, is the most comprehensive check-balancing program written. Requiring 32K, two disks and printer, this program does much more than just balance and reconcile your checkbook. It enables you to define up to 60 account names and will generate monthly summaries of all accounts with monthly and year-to-date totals. Single-entry input allows the user to disperse one transaction over several accounts and to make a 64-character note on each transaction. Checks can be printed out after data has been entered. Aside from the Statement of Accounts, **CRAS** also generates the following reports: Check Register for any Month, Notes to Check Register, Income/Expense Distribution, Statement of Selected Accounts, Bank Reconcile Statement and Suspense File. The Suspense file is an extra feature where you can make notes to yourself for any month in the year. **CRAS** will make both you and your accountant happy and it sells for $49.50.

TBS has other great software for your TRS-80. **BASIC TOOLKIT, SYSTEM DOCTOR & TERMINAL CONTROL** are system utilities. **CHECKBOOK II, INFORMATION SYSTEM & EXERCISER** are general applications. Don't forget the **LIBRARY 100;** 100 programs for only $49.50. **TBS** also has **DISK HEAD CLEANERS** for **TRS-80** and **APPLE** and **GRAN MASTER DISKETTES,** the best on the market.

TBS is **YOUR COMPANY,** and to you we pledge to produce quality software at a price you can afford. The above products are avaiable NOW at Computer Stores and Associate Radio Shack Stores nationwide or directly through us. For direct mail please include $2.00 for postage and handling.

TBS™ ✓6

THE BOTTOM SHELF, INC.

(404) 939-6031 • P.O. Box 49104-M • Atlanta, GA 30359

Included in the package is a short term forecasting program which utilizes both exponential smoothing and the new forecasting method of adaptive filtering.

A comprehensive, advanced version of multiple regression which contains many user conveniences along with a flexible graphing program is also integrated into the package.

The data management program allows the user to create, modify and add to any disk-based data set created.

David M. Chereb, Ph.D., has written a user manual containing over 100 pages of instructions and examples of forecasting techniques.

A second disk in the package contains over 30 of the most widely used economic data series. This data set is accessible by all programs.

For more information about the Planning Package ($97), write Applied Economic Analysis, 4005 Locust Ave., Long Beach, CA 90807. ↙47

Microsoft Bookkeeper

Data Train, Inc. is distributing a full line of accounting program products called the DTI Bookkeeper II.

The products are designed around Microsoft with a pending release on Microsoft Basic-80.

The products are a computerized return to traditional or college taught methods of accounting. Through its use of journals, ledgers and report writers, it allows the user to designate the accounting ledgers required to maintain his books.

The DTI Bookkeeper II provides the user ledger report writer modules for the accounting applications of General Ledger (Financial Statements), Accounts Receivable, Accounts Payable, and Fixed Asset Accounting.

Payroll operates stand alone, utilizing DTI Bookkeeper II conventions.

Without programming or the purchase of additional products, DTI Bookkeeper II establishes other accounting ledgers including but not limited to, job costing, simple inventory, investment accounting, route sales accounting, etc.

DTI accounting products, including the DTI Bookkeeper Payroll, are user installable with the aid of a non-technical operator's reference manual and are available to end users from $150 to $900 per module.

Write Data Train, Inc., 840 NW 6th Street, Suite 3, Grants Pass, OR 97526. ↙44

Text Editor

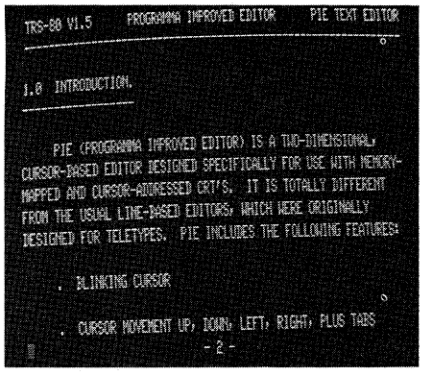
A new editor, called PIE, or Programma Improved Editor, is 2-dimensional, cursor-based and designed to operate on the TRS-80 Level II (16K) and Sphere 6800 systems.

The program generates cassettes which are compatible with the TRS-80 Editor/Assembler.

Over 25 commands control the cursor, including search for a string, append, insert, delete, backspace, set horizontal tabs, and page scroll.

PIE is sold on cassette ($19.95) and diskette ($24.95) for the TRS-80 and Sphere, complete with operating instructions.

For more information contact Programma International, Inc., 3400 Wilshire Blvd., Los Angeles, CA 90010. ↙21



Program Improved Editor.

Pre-packaged Subroutines

Johnson Associates has a Transaction Data Entry system for handling business application keyboard entry on the TRS-80.

The Data Entry system is a series of subroutines the application programmer includes in his program. The routines display a data entry form on the screen and provide a blinking cursor that guides the operator.

The cursor bypasses input field descriptions, as well as any special operator instructions.

After a complete transaction is entered another subroutine handles operator prompts for visual verification and/or individual field correction. The routines provide automatic field length checking and alert the operator if excessive characters are keyed.

A stand alone utility allows you to store screens of information.

The TRS-80 Data Entry system is available on diskette for $20.00. For more information write Johnson Associates, P.O. Box 1402, Redding, CA 96001. ↙85

Project Management And Time Accounting

National Software Marketing Inc. has two new software packages designed to run on the Radio Shack TRS-80 computer using 32K and at least one floppy disk.

The project management system maintains a record of projects within a group, providing data in up to 75 expense categories. Information can be retrieved for a specific project or group showing costs to date and comparison to budget.

The introductory price for this system is $116, including shipping costs.

The time accounting system permits recording and displaying time expended in hours for a client by job category and the hours expended for each employee by client number.

Designed to facilitate billing of clients from professional offices such as lawyers and accountants, the system costs $66.

All programs may be returned within 30 days for a refund less a $16 service charge.

The programs are shipped on diskettes in source language.

Write Elliot Kleiman, Box 6195, Hollywood, FL 33021. ↙156

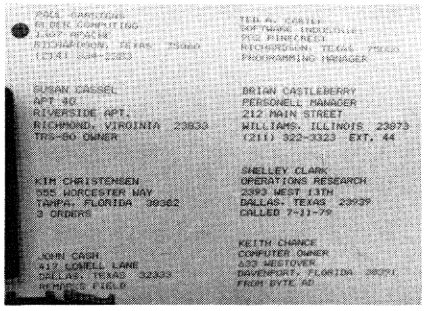
TRS-80 Functions Increased

DVR machine language, using approximately 500 bytes, repeats keys; directs screen output to screen and printer; runs programs using "LPRINT" without hanging up the system; provides a keybounce fix; allows users to LPRINT space compression codes (CHR$(191) through CHR$(255)); allows keyboard input of regular lower/uppercase characters; and generates a space, if required, after each LPRINT.

DVR comes on cassette with instructions for creating a disk file. Specify DOS or Level II and memory size.

DVR is priced for the personal consumer at $9.95, plus 75¢ postage and handling.

Write The Alternate Source at 1806 Ada Street, Lansing, MI 48910. ↙138



The mailer from Software Ind.

Mailing Label Program

Mailing List is a general purpose label program for the TRS-80 which includes: add, delete, search, sorted list, modify address or remarks, and sequential printout.

The user can allow up to 61 characters for either the name/title/company or remarks/code field to meet his particular needs.

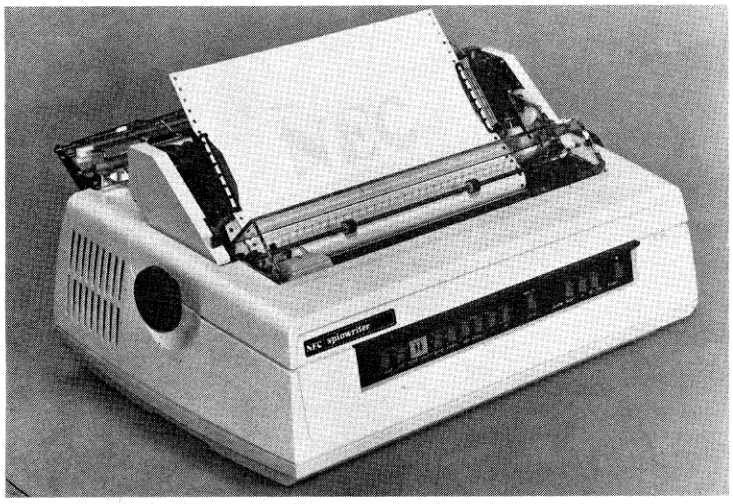
Any number of characters from the remarks/code field may be printed below the zip code, and names can be sorted on any two fields.

Over 500 names can be stored using one disk drive; over 1500 with two.

Written in Disk BASIC for a TRS-80 or Poly with one or two disk drives, the complete program comes on diskette and is accompanied by an instruction manual and hard copy listing for $39.95.

Available from Software Industries, 902 Pinecrest, Richardson, TX 75080. ↙157

DISCOVER THE MAGIC OF WORDPROCESSING AND TURN YOUR **TRS-80*** INTO A VERITABLE **"OFFICE WIZARD"** WITH A **"NEC SPINWRITER"**



NEC 5530

(CENTRONICS I/O)

What makes NEC your best typewriter quality printer value?

1. The one and only print **"thimble."** SPINWRITER'S DURABLE (up to 30 million impressions) and unique reinforced plastic print element. Up to 128 characters per print element, with a wide variety of typefaces to choose from.

2. **SPEED and RELIABILITY.** The SPINWRITER gives you up to three times the speed of IBM "Golf Ball" type mechanical printers, along with the advantages of a true electronic output printing device.

3. **HUMAN EAR COMPATABILITY.** The low pitched strumming noise made by the NEC Spinwriter won't leave your ears and nerves aching after a hard day's use, as Dot Matrix and mechanical printers tend to.

THE NEC SPINWRITER
IS FULLY SERVICED BY NEECO.

NEC 5530 SPINWRITER (plug compatible to TRS-80*) $2495.00
Word Processing Software "Electric Pencil" ..... $ 100.00

Call or Write today for NEC Specifications Sheet. (Specify TRS-80*)

***TRS-80 is a Trademark of Tandy Corp.**

✓143

NEECO
679 Highland Ave.
Needham, MA
02194

Mon-Fri 9:30-5:30
MasterCharge &
Visa Accepted

(617) 449-1760

H & E COMPUTRONICS INC.

•••EVERYTHING FOR YOUR TRS-80™•••

TRS-80 is a trademark of the Radio Shack Division of Tandy Corporation

★ *All Orders processed within 24-Hours*
★ *Free Shipping within U.P.S. areas (add $3 for orders outside of the U.S.A. or U.P.S. areas).*
★ *30-Day Money Back Guarantee on all Software (less a $3 penalty for handling).*
★ *10-Day Money Back Guarantee on Disk Drives and Printers* **PLUS** *120-Days Free Service.*

INCOME TAX PACS
TRS-80 Translations from Accountants Software Associates with Complete Users Manual.

INCOME TAX PAC A . . . $19.95
For Level II 16K Cassette
- Form 1040 • Form 1040A
- Schedule A Itemized Deductions
- Schedule B Interest and Dividends
- Schedule TC Tax Computations
- Output to Video Display

INCOME TAX PAC B . . . $49.95
For Level II 16K Cassette or 32K Disk
- All Features of Pac A **Plus**
- Output to Video or Line Printer
- Formats Form 1040 and 1040A for Tractor Feed Forms
- Schedule C Income from a Personally Owned Business
- Form 2106 Employee Business Expenses

PROFESSIONAL INCOME TAX PAC C . . . $99.95
For 32K System With Disk Drive
- All Features of Pacs A & B **Plus**
- Automatic Memory Storage for Income Tax Preparers
- Instant Line Change
- Additional Schedules and Forms
- Formats Forms for Individual or Tractor Feed Printing

BENDER'S 1980 TAX RETURN MANUAL . . . $29.95
- 400 Page Step By Step Tax Preparation Manual
- Includes Every Tax Table and Form
- All Forms Are Reproducible
- Line-By-Line Explanation of Each Form

- **TRS-80 DISK AND OTHER MYSTERIES $19.95 ($22.95 after 2/1/80).** Over 100 pages of indespensible information for disk owners. Learn to recover information from bad disks, how to make Basic programs unlistable and 12 more chapters of never published tips and information. Written by H.C. Pennington. (For all Disk Owners).

FROM ADVENTURELAND INTERNATIONAL
- **ADVENTURE #1 - #7** by Scott Adams . . . **$14.95 each** . . . available on Cassette or Disk.

- **HORSE SELECTOR II** by Dr. Hal Davis . . . **$50.** The TRS-80 version updated for the TRS-80 and originally reviewed in Systems and Methods.

FROM APPARAT
NEW DOS + $99.95
35, 40 and 77 Track Versions available

FROM RACET COMPUTES
- **REMODEL-PROLOAD** - Renumbers program lines, combines programs. The only renumber program that will renumber the middle of a program. Specify 16K, 32K or 48K. Works with Cassette or Disk . . . **$34.95**
- **GSF** - Use in your Basic Programs for Instant Sorting (will sort 1000 items in 9 seconds). Other commands include Compress and Uncompress Data, Duplicate Memory, Display Screen Controls and Fast Graphic Controls . . . **$24.95** (For Cassette or Disk, specify 16K, 32K or 48K).
- **DOSORT** - All G.S.F. commands plus special Multiple Disk Sorting Routines . . . **$34.95** (Specify 32K or 48K).
- **INFINITE BASIC** - Adds 70 commands to your TRS-80 including Instant Sort, Matrix Commands, String Commands, Left and Right Justification, String Centering, Simultaneous Equations, Upper and Lower Case Reverse and more . . . **$49.95.** (For Cassette or Disk).
- **INFINITE BUSINESS** (Requires Infinite Basic) - Eliminate Round-off error, 127-Digit Calculation Accuracy, Insert New Elements in Sorted Arrays, Automatic Page Headings, Footings, and Pagination, Multiple Precision Arithmetic and more . . . **$29.95.** (For Cassette or Disk).
- **COPSYS** - Copy Machine Language Programs . . . **$14.95** (For Cassette only).

FROM SMALL SYSTEM SOFTWARE
- **RSM-2** Machine Language Monitor . . . **$26.95**
- **RSM-2D** Disk Version of RSM-2 . . . **$29.95**
- **DCV-1** Converts Machine Language Programs from tape to disk . . . **$9.95**
- **AIR RAID** - The ultimate TRS-80 game converts your TRS-80 into a real time shooting gallery . . . **$14.95**
- **BARRICADE** - A fast pong style game . . **$14.95**
- **CPM** - $150 (for Disk only)
- **TRS-232 INTERFACE** - Interface with Software driver RS-232 printers to your TRS-80 . . . **$49.95**
- **TRS-232 FORMATTER** -Additional (optional) Software for TRS-232 owners. Ads many printer commands to your TRS-80 . . . **$14.95** ($9.95 with purchase of TRS-232).

- **MAIL PAC** - For Model I or Model II Disk Systems only . . . **$99.95.** Quick-sorting full user control over mailing list from Gallactic Software.

MICROSOFT FORTRAN (DISK)
~~$350~~ **NOW $150**

SARGON II
THE CHESS CHAMP
$29.95

FROM THE BOTTOM SHELF
- **CHECKBOOK II** (for Cassette or Disk) . . **$18.50**
- **INFORMATION SYSTEM** (for Cassette or Disk . . . **$24.50**
- **SYSTEM DOCTOR** (a complete diagnosis of your TRS-80 . . .checks memory, video, cassette, disk, ROM and all other parts of your system) - for Cassette or Disk . . . **$28.50**
- **CHECKBOOK REGISTER ACCOUNTING SYSTEM** (requires 2 disk drives) . . **$49.50**
- **LIBRARY 100** - 100 established business, game and educational programs plus **FREE** Tiny Pilot all for . . . **$49.50**
- **BASIC TOOL KIT** - lists all variables, GOTO's and GOSUB's in your program . . . **$19.80**

- **SOUNDWARE** - Ads sound to your TRS-80. Just plus it in . . .**$29.95.** Sample programs included.
- **TING TONG** - Can be used with Soundware for a **Sound** version of pong . . . **$9.95.**

DISK BUSINESS PROGRAMS
MODEL I AND MODEL II
GENERAL LEDGER/CASH JOURNAL . . $99.95
ACCOUNTS RECEIVABLE . . $99.95
ACCOUNTS PAYABLE . . $99.95
INVENTORY CONTROL . . $99.95
PAYROLL . . $99.95

VISTA V80 DISK DRIVE
110 K OF STORAGE
$395
Add $29.95 for Cable
(Free with Purchase of Two Disk Drives).
— 10 Day Money Back Guarantee —
8" Drives also available

FROM HOWE SOFTWARE
MON-3 - Machine Language Programming for Beginners. **MON-3** is a Complete System Monitor with Users Manual . . . **$39.95**
MON-4 - Disk Version of **MON-3** . . . **$49.95**

LEVEL III BASIC . . . $49.95 FROM MICROSOFT - Now Cassette owners can add Disk Commands to their TRS-80 without owning a Disk Drive.

- **BRAND NEW OLIVETTI PRINTER . . .$2495** Business Letter quality print, Automatic Line Justification (on request), Quick Printing, can be used as a Memory Typewriter, plugs right into your TRS-80 without any modification or software.

THE ELECTRIC PENCIL
Cassette . . . $99.95
Disk . . . $150.00

H & E COMPUTRONICS INC.
MATHEMATICAL APPLICATIONS SERVICE™

Box 149 New City, New York 10956

✓9

36-Page Catalog $2 **FREE** *With Any Order*

Order by Phone or Mail
No Shipping Charge
Add $3 for C.O.D.
Add $3 for all Foreign and non-U.P.S. shipments
Add $3 for U.P.S. Blue Label

VISA

24
HOUR ORDER LINE
(914) 425-1535

master charge

✓ Reader Service—see page 131

Three Different Mailing List Programs

by Reese Fowler

The **Business Mail System** by The Bottom Shelf (TBS) is not the easiest of these programs to use. It comes in a nice folder with an instruction book and two diskettes. The instructions say that the program will handle a mailing list of 150,000 names and addresses. This may be true, but you can put only 500 on a single disk.

The two diskettes supplied with the programs are the Security diskette and the Program diskette. Before you can use the program, you must transfer TRSDOS 2.2 to the program disk. This requires the following procedure:

1. Insert the program disk into drive 0 and press (ENTER).
2. You will then be prompted to remove the program disk, insert a DOS diskette into drive 0 and press (ENTER).
3. You will then be told to remove the DOS disk, re-insert the Program disk and press (ENTER). This transfers the DOS to the Program disk.
4. You then insert a formatted disk into drive 1 and initialize a DATA DISK. At this point, your program is ready, but make several data disks, if your list has over 500 names.

To run the program insert the Program disk into drive 0, a data disk into drive 1 and press (RESET). The prompt asks that you remove the Program disk, insert the Security disk into drive 0 and press (ENTER).

This Security diskette is not used to protect your data, but to protect the TBS software from unauthorized copying.

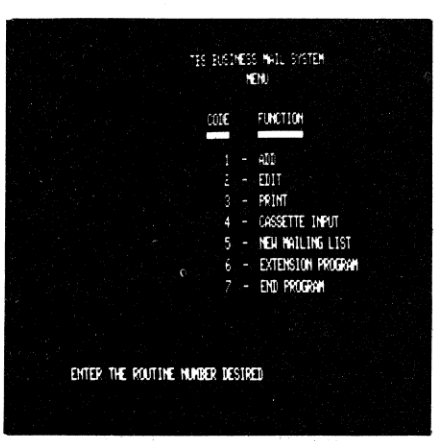


Photo 1

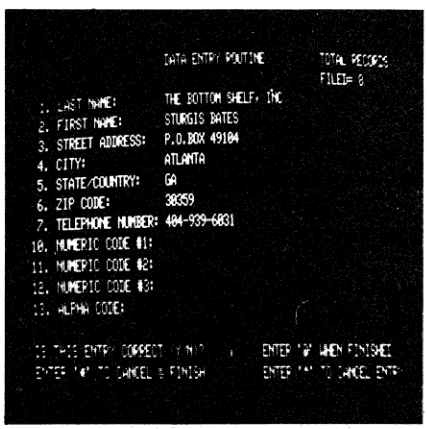


Photo 2

After you remove the Security disk, re-insert the Program, press (ENTER) and the menu will appear on the screen (Photo 1). You are now ready to enter your names and addresses.

Because this is a new mailing list, you must use option five from the main menu to enter your first batch of names. Presorting the list into zip codes will make loading much faster though the program can do it for you. You can enter up to 100 names and addresses before saving them. (See Photo 2 for format.)

When the first disk is full, the program will tell you to insert the next one, and you can enter 500 more names and addresses.

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

To add more names to an existing list, use option one on the menu. The program will expand your list and insert the additions, in order, to the proper data disk.

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

The **Mail/File List** by Galactic Software Ltd. is possibly the most unusual of the mailing label programs. While this program will support a maximum of only 600 names and addresses on a two-disk system, it maintains both an alphabetic and a zip code file under constant sort. A list entry is automatically placed into proper sequence in both files.

This program comes in a loose leaf notebook with an instruction book and a program diskette. You must transfer the program to a TRSDOS 2.2 or 2.3 diskette. It will not run under TRSDOS 2.1 or NEWDOS+. Once you have the program on a DOS diskette, it is ready to run. A single disk holds 300 addresses.

After you have typed RUN"MAIL", the main menu (see Photo 3) appears on the video screen. Option A starts a new list, and the first thing the program asks is the name of the file. This can be any eight-character TRSDOS filespec. The computer checks for any other file by that name, and if this is a new file it asks on which drive you wish to place it.

The program then asks for NAME #1, which should be a company name or the last name of an individual (Photo 4). This is the field that is sorted for the alphabetical list. Name #2 should be the first name of an individual or the ATTN line for a company. Next the usual address information is added ending with a five-character alphanumeric that gives you up to 6 million different selection codes for your list.

The program then writes the data to the disk and asks for the next entry. You can exit this

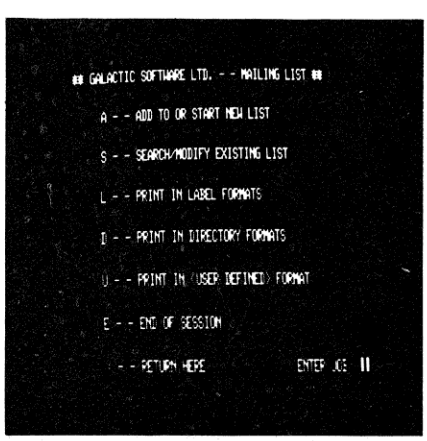


Photo 3

Reviewed for Your Disk Based System

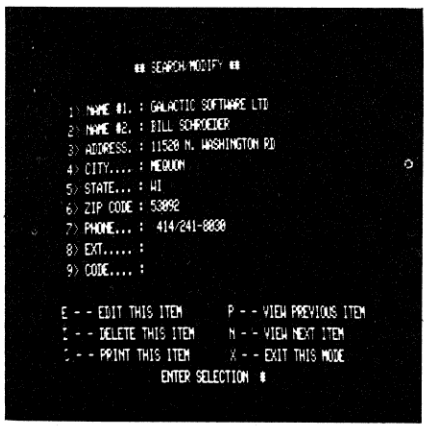


Photo 4

routine at any time by typing an X as the only entry for NAME #1.

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

In the standard three-line label the print format is NAME #2 and NAME #1 on the first line. A name entered as DOE, JOHN T., will be printed as JOHN T. DOE. In the four-line label NAME #2 is on the first line and NAME #1 is on the second line. This is the normal format for company addresses and allows you to print:

Attn: William Smith
ABC Computer Company

In addition, you can print a 30-character message as the first line on a three-line label making it in effect a four-line label. This will allow you to print labels with such messages as ATTN: STORE MANAGER on the first line. In addition to labels the program will print a directory in either alphabetical or zip code order. The program also has the usual edit routines.

The **Name and Address System** by Small Business Systems Group (SBSG) is available in two versions. One is a normal label program, and the other has a form letter routine. When used with the Electric Pencil, you can type individual letters for everyone on your mailing list. We'll be reviewing the form letter program, though the two programs are the same except for this routine.

This program is the only one of the three that is ready to run when you receive it. It has its own DOS which automatically boots the program. To run this program, you insert the diskette into drive 0 and press (RESET). The program then asks for the date, and after you enter it the main menu (see Photo 5) will be displayed on the screen.

SBSG has included a sample data file on the diskette so that you can familiarize yourself with all the program's functions before you enter your own data. A sample entry is shown in Photo 6. Your data is written to the disk as an INPUT file in the order in which it is entered. You must create an OUTPUT file to print your labels.

This can be done by sorting your INPUT file. You cannot add names to an existing file. To do this you must create a second INPUT file and merge this with the first. You then sort the merged files to create a new OUTPUT file. This is the major fault of the program, as it uses too much disk space for the number of records in the file.

The label printing routine is straightforward and prints up to three labels across in a five-line format. Two select fields allow you to print any part of your list. One is alphanumeric and the other is alphabetic.

One thing to remember is that this program uses dynamic memory allocation, so keep your select codes short. Between the two you have an almost unlimited selection when you print your labels.

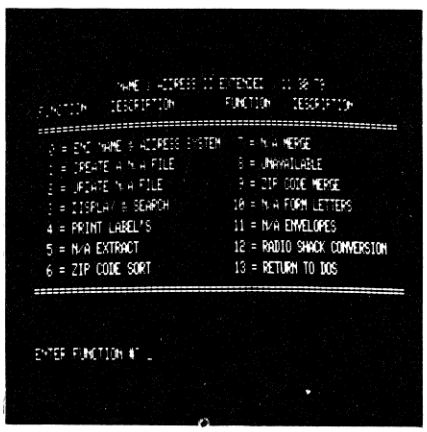


Photo 5

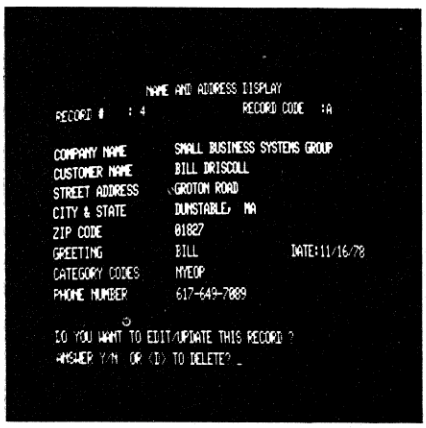


Photo 6

We now come to the best feature of this program, the Form Letter routine. If you like to send personalized form letters, this is the routine for you. To use it you must have a copy of Electric Pencil by Michael Shrayer.

Once you've written your letter, use special variables for the name and address fields which the label program will change to the appropriate entries when the form letter routine is run.

All of the other label printing options are still functional in the form letter mode, so you can send the letter to a selected few recipients.

Conclusion

By now you should have a good idea which of these programs is the one for you. If you have a very large list, the TBS Business Mail should be your choice. Once you have entered your data the printing routine is very fast, over 100 labels per minute.

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

Without its Form Letter routine, the SBSG Name and Address System is not the best of the three, but that routine is worth the price of the program. ■

Good documentation but poor performance!

Radio Shack's Mailer

Dr. Duncan A. Buell
Assistant Professor
Dept of Computer Sciences
102 Nicholson
Louisiana State University
Baton Rouge, LA 70803

The key to a businessman's computer system is not the hardware, but the software. Beyond the minimal requirements of reliability, the real test of quality in a business system is the software package.

The software must be totally "idiot-proofed." That is, the user must be protected from erroneously entering the wrong kind of data and any defaults must be clearly stated during execution of the program.

Further, the software must present to the businessman the information he wishes to see in the manner he wishes to see it. And finally, the program must be sound; not only must the program **look** correct, it must **be** correct under the surface.

Early Impressions

The first day I had my system marked my first negative impression of a Radio Shack program. I loaded and ran their backgammon tape and after a few minutes of play, I entered an incorrect number. The program died.

Being new to the micro game, I assumed that the Radio Shack programs would be idiot-proofed in the same manner that professional programs are. Alas, I was mistaken.

Months later, I encountered a similar problem when playing with the Personal Finance Program cassette in a local store. At one point the program prompted me for a choice of activity, asking for either a 1 or 2. Feeling capricious at the time, I pressed ENTER. Instead of being prompted again, the program continued, defaulting (without telling anyone) to one of the two options. Since I had no idea where I was, I quit. There was no echo of my input and no header message telling me what was now going on.

These flaws could be considered minor. After all, the backgammon tape is free and the Personal Finance tape is quite cheap, and both are designed for personal use. What does concern me, as a professional in computer science and as a consultant, is the cavalier attitude with which Radio Shack has marketed business software with the same sorts of inherent flaws in design.

While doing a programming job for a small businessman, I used Radio Shack's disk mailing list program, which my client had obtained without consulting me. Installing the system, I found a great many major and minor errors. Unfortunately, the problems are not only in the cosmetics of the program but go deeper into its basic organization.

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

The first flaw I found was that there was no exit from the program. The menu listed the above five options, but no "quit" option.

By reading the manual I found that the normal termination for the program was to hit the BREAK key. The what? Whoever had written the program had included a nice menu, idiot-proofing it against selections not on the menu, and had left off the two lines of code that would allow a clean termination.

A naive user, should not use the BREAK key, unless it is absolutely necessary. Sooner or later he will get halfway into a program, find that there is a phone call waiting for him, hit the BREAK key and walk away with his files still open.

The User's View

The user's view of the add-names and edit modules is generally good. Both modules use menus for input and the editing is easy. This is a credit to Radio Shack.

The listing of names to the screen is good, as far as it goes. The module lists a screenful at a time and allows the user to ask for another screenful. There are only two problems with this listing. Firstly, it would have been nice to have name and address or name and company listed, instead of just the name. This would help the user who was unsure about listing several members of the same household, for example.

Secondly, one finds when printing labels that they are **not** printed in sorted (alphabetical) order, but in the physical order in which they reside on the disk. Someone concerned with checking the disk list against another list, or checking for duplications, would find himself with an unnecessarily tedious task. The code required to get names and addresses listed either to the screen or to the printer is a negligible addition to the list module and should have been included in the system.

But these are only superficial flaws. Flaws derived from an unwillingness to program the tedious code that provides "pretty" output. The deeper structural problems reflect a fundamental lack of understanding of how to write software. Were a freshman to hand the Radio Shack system to me as a programming assignment, I would have given him an A. But had the student been a senior, I would have been forced to fail him (or her). The flaws are, I believe, that serious.

Always Rewritten

The edit module allows one to search through the list of names and addresses one by one. In this search the records are examined in order and the user has the option to change the record in several ways.

However, the records are **always rewritten** to disk, whether or not any information has been changed. The reliability of mini-floppy disks is always in question and the constant rewriting of good records seems to me to be unwisely asking for trouble.

A similar problem exists with the recover space module. The disk is overwritten in the same order in which the good records currently reside with the flagged deletions now actually deleted. Not only does the rewriting of all records beyond the first deletion require great confidence in the disks, but it takes longer.

Perhaps, however, this strategy was adopted because the labels are printed in physical order

TRS-80* OWNERS

- *Do you want to write BASIC programs faster?*
- *Do you want your programs to have that "professional look"?*
- *Are you tired of erasing and redrawing screen formats on paper?*
- *Do you want faster graphics than "poke" will allow?*

Then you may never write programs again without using:

*Trademark of Radio Shack, a Tandy Co.

IMAGINE!!!!!!!

- Letting the computer write your BASIC programs.
- Creating a library of your own personal displays.
- Up to thirty powerful commands.
- Data Entry and Display on Video Forms.
- No more scrolling data entry.
- Instant screen displays with data.

The **Magic Cursor** is a **Revolutionary Family of Products** which provides, even the novice, a dramatic new method of creating professional grade software (in Basic) at a fraction of the time previously required.

If animated games is your interest, you'll write no more print @ or poke statements.

If you want to create personalized programs for home use, you can now do it faster, create outstanding screen displays and have more fun doing it.

If your business is software creation, soon you will not be able to adequately market the antiquated scrolling displays of the past. And what's more, you will spend up to 40% more time writing programs than your competitors. You can't afford to waste another minute writing in the tedious, cumbersome, old outdated method of the past. You need to be one of the first to market these new screen oriented programs.

In addition, it is available for any level 2, 16K (or larger) system without a disk drive or with one, two or more drives. Your only limitation is with your own system capability. And as your system grows, you may upgrade your original copy to a larger program by paying the difference and a moderate service charge.

Be sure to pick out the version that fits your present needs and order it today, so you can take advantage of the remarkable new product as soon as possible. Remember, the **Magic Cursor** is unique and there is presently no other product available on the market anywhere that will provide you this capability.

THE BABY CURSOR allows you to easily create screens (including graphics) on your video. A powerful command then generates the BASIC instructions to recreate the screen. For the first time, a program for automatic generation of video display forms. (16K Tape or 16K Disk) **$24.95**

THE MAGIC CURSOR I additionally makes sophisticated Data Entry and Display easy. With the **Magic Cursor I** you define the Data Entry or Display fields directly on your screen. The definition commands generate the BASIC instructions to implement the Data Entry and Display. **The Magic Cursor I** has commands which move, center and duplicate blocks of graphical or alpha/numeric displays. You can even justify text .. **$79.95**

THE MAGIC CURSOR II adds the power to write animated games easily in BASIC. The **Magic Cursor II** allows you to reload previous screens either from memory or from disk. You can then modify them and store either the modified screen or only the changes. (32K Disk only) **$99.95**

THE MAGIC CURSOR III will be available soon for the new Model II Computer. (32K one or more disks) **$149.95**

BUSINESSMEN: *It will make your programmers more productive!!*

Use our toll free number, or order by mail. Master Charge, Visa, Certified Check or Money Order accepted. Personal checks require 14 days to clear. C.O.D. or collect calls not accepted. Software guaranteed for replacement only. Prices subject to change without notice. Programs supplied on cassette tape. For disk versions, the cassette supplied will automatically create a disk file.

ORDER FROM:

Send check or money order payable to:

CUSTOM COMPUTER CENTER, INC.

P.O. Box 58042 / Dept. 8-B / Houston, Texas 77058

QUAN.	DESCRIPTION	SYSTEM (size / tape or disk)	PRICE
		TAX & HANDLING	
		TOTAL	

Handling Charge: $1.50 / Texas Res. add 6 % Sales Tax

Name _____

Address _____

City _____ State _____ Zip _____

MASTER CHARGE / VISA

CARD # _____

EXPIRES _____

✓7

CUSTOM COMPUTER CENTER, INC.

P.O. BOX 58042 / HOUSTON, TEXAS 77058

Phone Orders: (800) 621-5199 / Illinois (800) 972-5855

and the user, it can be assumed, entered the records in more-or-less sorted order. (More on this point later.)

We come finally to the most serious problem, the sort routine and its usage. Not only are there flaws, but these flaws indicate that some major design errors occurred in writing the program.

Consider sorting a mailing list file. It cannot be assumed by the programmer that the records are already sorted. True, there will probably be several sorted segments; at any given input session, the records may well be sorted when entered. But the file itself will not be sorted.

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

The Radio Shack program uses an insertion sort to order the records. This implies that the designers felt that the file would be reasonably sorted to begin with and that the ratio of new to old records would be small.

An insertion sort has an 0(N↑2) worst-case running time. This is the same as a bubble-sort and essentially the worst possible time of any of the standard sort methods. Only if the file is largely sorted and few additions are made, is the insertion sort effective.

Unfortunately, it appears that the insertion sort method decision was made independently of other design decisions. For its effectiveness the insertion sort relies on a large almost-sorted file with only a few changes to be made. Why does Radio Shack choose to sort from scratch each time? The current sort index is ignored in re-sorting, so that the heavy initial cost is paid each time the file is sorted.

More Problems

Having uncovered problems of this magnitude, it is perhaps not surprising to find that the sort routine not only has the potential for making 0(N↑2) comparisons, but 0(N↑2) disk accesses. All comparisons are made between one record currently in memory and one record brought in off disk.

Performance is poor. With 210 records in the file, sorting took (nearly) an hour. The disk can hold about 990 records, says Radio Shack. I shudder to think of how long it would take to sort them all.

I cleared up my client's sort problems by first bringing all the names into core. (On a 32K system this meant that I had to cut back to a maximum of 900 names per disk.) I then did an in-core heap sort. Sorting 210 records took about three and a half minutes.

With a 16K system you cannot sort completely in core. Various options still exist, however, any of which significantly speed up the sort. You could sort half the file at a time and then merge the two. Or, you might sort all the records on the first 8 characters and since the records would be nearly sorted at that point, use a bubble-sort routine to finish the task.

As a final comment, you must count bytes and consider the possibility of filling the disk. The add-names module does not refuse new records when a fixed number has been entered, but when the "disk full" error message is trapped by the program.

This does not, however, guarantee that there will be space for the index file. If you have added a great many records when the disk-full error is trapped, the index file will require more sectors than it currently does and that extra space simply won't exist.

Conclusion

Though I have not examined the other Radio Shack packages in depth, and cannot say that they exhibit similar problems, I would have to recommend that a prospective buyer of any of the packages get a professional to examine the code in detail before buying. My early impressions of Radio Shack's software are negative, and I must remain skeptical until shown that my suspicions are groundless. ■

GET WORLDLY...GET ISI

Enter the world of "80" programs from Instant Software. A world where you and your TRS-80 can rule duchies, play poker, learn speed reading, explore caves, design antennas, launch rockets, or simply pay your bills. When you've explored all of these possibilities, you will have surveyed only about 1/15 if ISI's world of "80" programs . . . .

LEVEL II

MODEL ROCKET ANALYZER AND PRE-FLIGHT CHECK Let your TRS-80 help you enjoy the fast-growing hobby of model rocketry. The complementary programs included are:

•**Model Rocket Flight History Prediction** – This program will compute the flight characteristics for almost any model rocket. Engine and body tube data included covers Estes, Centuri, Flight Systems, A.V.I. Astroport, C.M.R., and Kopter products.

• **Weather Forecaster** – Before you launch your rocket, get an up-to-the-minute weather forecast. Just enter your location, elevation, average temperatures for January and July, and barometric pressure. You'll be the short-range weather forecaster for your area.

For a successful launch, you'll need TRS-80 Level II 16K. **Order No. 0024R $7.95.**

RAMROM PATROL/TIE FIGHTER/KLINGON CAPTURE Buck Rogers never had it so good. Engage in extraterrestrial warfare with:

•**Ramrom Patrol** – Destroy the Ramrom ships before they capture you.

•**Tie Fighter** – Destroy the enemy Tie fighters and become a hero of the rebellion.

• **Klingon Capture** – You must capture the Klingon ship intact. It's you and your TRS-80 Level II 16K battling across the galaxy. **Order No. 0028R $7.95.**

TRS-80 UTILITY II Let Instant Software change the drudgery of editing your programs into a quick, easy job. Included in this package are:

•**CFETCH** – Search through any Level II program tape and get the file names for all the programs. You can also merge BASIC programs with consecutive line numbers into one program.

•**CWRITE** – Combine subroutines that work in different memory locations into one program. This works with BASIC or machine-language programs and gives you a general checksum.

This package is just the thing for your TRS-80 Level II 16K. **Order No. 0076R $7.95.**

CARDS This one-player package will let you play cards with your TRS-80 – talk about a poker face!

•**Draw and Stud Poker** – These two programs will keep your game sharp.

•**No-Trump Bridge** – Play this popular game with your computer and develop your strategy.

This package's name says it all. Requires a TRS-80 Level II 16K. **Order No. 0063R $7.95.**

SPACE TREK IV Trade or wage war on a planetary scale. This package includes:

•**Stellar Wars** – Engage and destroy Tie fighters in your attack on the Death Star. For one player.

•**Population Simulation** – A two-player game where you control the economy of two neighboring planets.

You decide, guns or butter, with your TRS-80 Level II 16K. **Order No. 0034R $7.95.**

FINANCIAL ASSISTANT Compute the figures for a wide variety of business needs. Included are:

•**Depreciation** – This program lets you figure depreciation on equipment in five different ways.

•**Loan Amortization Schedule** – Merely enter a few essential factors, and your TRS-80 will display a complete breakdown of all costs and schedules of payment for any loan.

•**Financier** – This program performs thirteen common financial calculations. Easily handles calculations on investments, depreciation, and loans.

•**1% Forecasting** – Use this simple program to forecast sales, expenses, or any other historical data series.

All you need is a TRS-80 Level II 16K. **Order No. 0072R $7.95.**

PERSONAL BILL PAYING

NOTE: This package can take the headaches and/or penalties out of paying your bills.

In a business office the accounts payable (bills) are usually paid on or immediately before their due date. That way, the payer gets the fullest use of his money without incurring penalties for being behind in paying his debts. Now you can take advantage of this system for your monthly bills, letting your TRS-80 do all the drudgery and record keeping.

This useful package provides a computerized list of all your bills and payments. You can access as many as 22 accounts, all of which can be named – up to 15 characters per name. Each account is listed by number, amount owed, due date, and present activity.

Don't confuse this system with a "checkbook" program. The functions of this package are threefold: (1) to monitor your bills; (2) to order payments most effectively; and (3) to make historical comparisons of individual accounts or specific months.

After you load the program, it displays a menu of 11 activities. They include:

Build and Maintain Files
List All Accounts
List Current Accounts
Make Payment(s) to Account
Enter New Bill to Account
Display Payment History of Individual Account (includes date paid, check number, and 12-month total)
Display Payment History of Selected Month
Delete Account
Delete Prior Month's Payment
Save File on Tape
Input File from Tape

After you have updated the records by entering new bills, paying bills, or changing the accounts, you can save all the information on data tape. This data tape will then be input for the next time you use the package. Maybe it can't make paying bills all fun and games, but it should relieve some of the agony. Level II 16K required. **Order No. 0103R $7.95.**

DOODLES AND DISPLAYS II Wait until your children get hold of this package:

•**Doodle Pad** – Draw pictures and save them on cassette tapes.

•**Symmetrics** – An electric kaleidoscope that changes from black to white and back again. It's almost hypnotic!

•**Drawing** – Like Doodle Pad, but for the serious artist. Over 40 user commands!

•**Random Pattern Display** – The computer does the drawing, but those with itchy fingers can tamper.

•**Mathcurves** – Bring those geometry lessons to life. Six different geometrical curves on the screen of your TRS-80.

•**Rugpatterns** – Yes, it does design rug patterns; and with a choice of user or computer control, it can do a whole lot more.

For the Level II 16K TRS-80. **Order No. 0042R $7.95.**

BOWLING LEAGUE STATISTICS SYSTEM This package is the answer to the prayers of harried bowling league scorekeepers. The Bowling League Statistics System will keep a computerized list of league data, team data, and data for each bowler. It is extremely flexible and has a total of 16 different options to let you modify the program to suit your league's rules. The program is very easy to use and has extensive "built-in" aids to help you along. Requires TRS-80 Level II 16K. **Order No. 0056R $24.95.**

DEMO III This is the big one: big on value, big on fun – the perfect package for the beginner and old hand alike. Check out this list of programs!

•**Race 1** – It's you against the clock as you careen around the track in this simulation of a high-speed car race.

•**Target UFO** – Rack up a big score by destroying all the UFO's in the shortest possible time.

•**Life** – Experiment with population density factors in this simulation of the life cycle of a colony of bacteria.

•**Phone Number Converter** – Let your computer figure out clever words for all those hard-to-remember phone numbers.

•**Biorhythm** – Fact? Or fantasy? Who can say for sure? In any case, you and your friends will be able to plot your biorhythmic curves whenever you want!

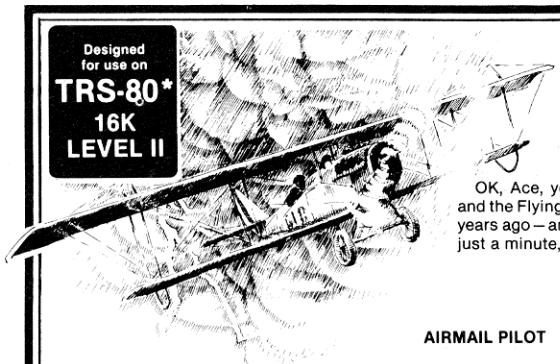
•**Graphics Program** – No user commands, just sit back and enjoy as your TRS-80 demonstrates its artistic abilities.

•**Race 2** – After you've mastered Race 1, you'll be ready for this more advanced version – with a choice of five different tracks!

•**Horse Race** – Up to nine players can place their bets and watch the ponies run. The computer will keep track of the winnings.

•**Drawing Board** – Your TRS-80 supplies the "pencil and paper," and you supply the ability. Messages or drawings may be stored on cassette for later unveiling.

•**24-Hour Clock** – That's right, this program allows your computer to act as a digital timepiece. Perfect for sporting events. For the TRS-80 Level II 4K and Level II 16K. **Order No. 0055R $7.95.**



EVERY FLIGHT IS A SPECIAL DELIVERY

OK, Ace, you survived everything that von Richthofen and the Flying Circus threw at you. Well, that was four long years ago – and yesterday's medals don't pay the rent. But just a minute, here's an ad:

"Airmail Pilot wanted . . ."

AIRMAIL PILOT

You can almost smell the gasoline as the ground crew fuels your J-4 Jenny biplane to her 26-gallon limit. Precious mail is loaded into the cargo area, tagged for Chicago. The weatherman reports severe icing above 6,000 feet, so you know you have to keep the plane low. It will be a dangerous flight, but you knew that when you took the job. The mail must go through. So, in the tradition of Lindbergh and a hundred unsung heroes, you bravely turn your plane into the wind. The engine roars. Suddenly you're aloft on the first leg of your journey. Dayton's socked in by fog. You change your course for Lucasville. Lightning zigzags the sky. A massive, fast-moving thunderstorm forces you to land in a cornfield. As the weather clears, your plane leaps once more into the sky. But even clear skies can cause problems – violent air currents buffet your fragile wooden aircraft. Your fuel is down to two gallons as Lucasville comes into sight. You make it! Refuel and head for Chicago. But you're not out of trouble yet. There's a wind shear at the Chicago airport. You have to land in a shifting crosswind. Can you make it? AIRMAIL PILOT from INSTANT SOFTWARE. Unlike any other computer simulation you've ever experienced. Challenging. Difficult. But never impossible. An event in a cassette. Crash or fly, it's so realistic, you can almost feel the wind. **Order No. 0106R $7.95.**

✓ *Reader Service—see page 131*

LEVEL II

DEMO II The company that brings you more programs for the dollar is proud to offer Demo II, an extraordinary package that contains programs to suit your every mood:

•**Tic-Tac-Toe**—Fun for the whole family! You all know the rules, and this version gives you three different levels of difficulty, one of which is sure to suit everyone in the family.

•**Time Trials**—You won't have to leave your house to experience "pre-race" excitement. It's you against the clock as you maneuver your car through the curves, chutes, and chicanes of the computerized course.

•**Maze**—Somewhere within a ponderous maze, the secret home square waits for you to uncover its presence. If you're not alone, your computer will be happy to offer a two-player version so that you can both race to uncover your respective squares first! Amazing.

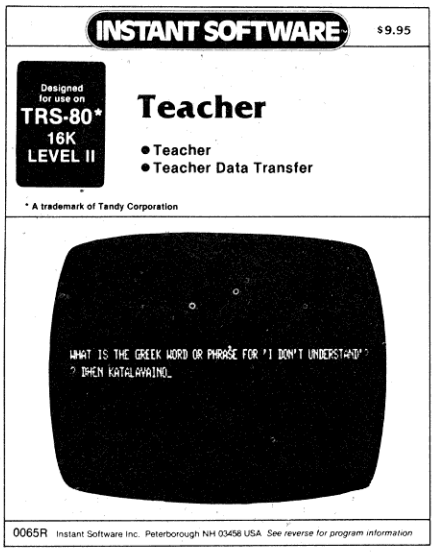
•**Hangman**—How many people have grown up playing different versions of that old grade school standby, hangman? If you've never played the game on a computer, you're in for a special treat! You or the computer will supply the word. With each wrong guess, the poor hangman's figure grows—and only you can spare him!

•**Wheel of Fortune**—"There's one born every minute," or so goes the old saw. In this simulation of the carnival wheel of fortune, you have your choice of the regular or the "crooked" version, where you can't help but win. The casino will even give you the keys to the place!

•**Hurricane**—OK, all you disaster buffs, here's a program that will let you chart the path of oncoming hurricanes anywhere in the world, using data available from the National Weather Service.

•**Bugsy**—Sure, everybody talks about computer bugs, but how many people have ever seen one? It's you against the computer in this game, with the computer rolling the dice, to see which one gets to add another part to his Z-80 Bug. The first one to complete his "curse of computing" wins.

•**Horse Race**—It isn't every day you get to see a horse race . . . at least not until now! Up to 100 bettors can cheer their horses to the finish line with their choice of win, place, or show. For the TRS-80 Level II 16K. **Order No. 0049R $7.95.**

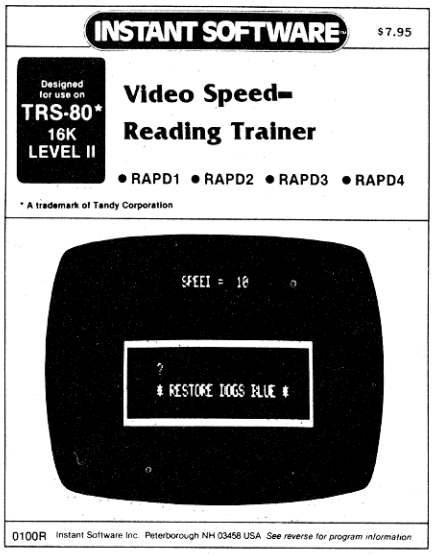


TEACHER

What do you need to learn? Would you like to know all of the cranial nerves? Electronic color codes? Civil War battles? Signs of the zodiac? Whatever your subject matter, the Teacher package can help you learn it. You simply input up to twenty questions and answers at one time. Next, review the material, and then take the test until you have your lesson down pat. The program gives you up to three hints per question and even offers graphic rewards for children, all at your discretion. All the information can be saved on cassette tape for reuse.

This package also contains the Teacher Data Transfer program, which allows you to combine several tests on one tape. That means you can learn a number of lessons sequentially without changing tapes.

Teacher is an effective instrument for anyone who needs to learn a lot of material in the shortest possible time. For decades leading educators and computer scientists alike have been promoting the future role of computers in education. Now you and your family can reap the benefits of computer-assisted instruction in your own home. The program is furnished with a blank data cassette tape. You'll need a TRS-80 Level II 16K. **Order No. 0065R $9.95.**



VIDEO SPEED-READING TRAINER

You can increase your reading speed and comprehension with this package. It uses the principle of the tachistoscope, a device that teaches by displaying images for a fraction of a second. These programs can train you to recognize words and phrases quickly, so that your everyday reading becomes an uninterrupted process.

With this three-part package, you can learn to recognize numbers, letters, words, and phrases. You start at your present level of skill by choosing the number of prompts to be shown and the length of time they will be visible. After the cue is flashed on the screen, you type back what you saw. If your response is incorrect, there is immediate feedback as the correct answer is displayed for several seconds. All cues are presented randomly so that the exercise questions cannot be learned in sequence.

The computer will monitor your progress and will automatically speed up presentation as you improve. Conversely, should you miss more than half of the questions, the speed will drop to an easier level.

The speed range is from 10 to 100, so you can readily determine your rate of progress. To increase your throughput, you'll need a Level II 16K. **Order No. 0100R $7.95.**

LEVEL I AND II

GOLF/CROSS-OUT Have fun with these exciting one-player games. Included are:

•**Golf**—You won't need a mashie or putter—or a caddie, for that matter—to enjoy a challenging 18 holes.

•**Cross-Out**—Remove all but the center peg in this puzzle, and your neighbors will call you a genius.

You'll need a TRS-80 Level I 4K, Level II 16K. **Order No. 0009R $7.95.**

HAM PACKAGE I This versatile package lets you solve many of the problems commonly encountered in electronics design. With your Level I 4K or Level II 16K TRS-80, you have a choice of:

•**Basic Electronics with Voltage Divider**—Solve problems involving Ohm's Law, voltage dividers, and RC time constants.

•**Dipole and Yagi Antennas**—Design antennas easily, without tedious calculations.

This is the perfect package for any ham or technician. **Order No. 0007R $7.95.**

ELECTRONICS I This package will not only calculate the component values for you, but will also draw a schematic diagram. You'll need a TRS-80 Level I 4K, Level II 16K to use:

•**Tuned Circuits and Coil Winding**—Design tuned circuits without resorting to cumbersome tables and calculations.

•**555 Timer Circuits**—Quickly design astable or monostable timing circuits using this popular IC.

•**LM 381 Preamp Design**—Design IC preamps with this low-noise integrated circuit.

This package will reduce your designing time and let you build those circuits fast. **Order No. 0008R $7.95.**

BEGINNER'S BACKGAMMON/KENO Why sit alone when you can play these fascinating games with your TRS-80?

•**Backgammon**—Play against the computer. Your TRS-80 will give you a steady, challenging game that's sure to sharpen your skills.

•**Keno**—Enjoy this popular Las Vegas gambling game. Guess the right numbers and win big.

You'll need a TRS-80 Level I or II. **Order No. 0004R $7.95.**

AIR FLIGHT SIMULATION Turn your TRS-80 into an airplane. You can practice takeoffs and landings with the benefit of full instrumentation. This one-player simulation requires a TRS-80 Level I 4K, Level II 16K. **Order No. 0017R $7.95.**

OIL TYCOON Avoid oil spills, blowouts and dry wells as you battle to become the world's richest oil tycoon. Two players become the owners of competing oil companies as they search for oil and control their companies. Requires a TRS-80 4K Level I or II. **Order No. 0023R $7.95.**

BOWLING Let your TRS-80 set up the pins and keep score. One player can pick up spares and get strikes. For the TRS-80 Level I 4K, Level II 16K. **Order No. 0033R $7.95.**

SPACE TREK II Protect the quadrant from the invading Klingon warships. The Enterprise is equipped with phasers, photon torpedoes, impulse power, and warp drive. It's you alone and your TRS-80 Level I 4K, Level II 16K against the enemy. **Order No. 0002R $7.95.**

BUSINESS PACKAGE IV Business Package IV gives you, the businessman, a superb tool to help you make those important decisions. This package includes:

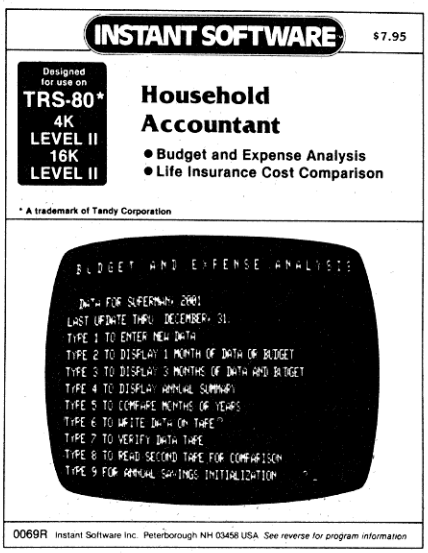
•**Business Cycle Analysis**—This program isn't a crystal ball, but it can show you your business's expansion and contraction cycles. You can plot any aspect of your business on a graph and see, in black and white, just what's happening. This program will give you access to information you couldn't get before.

•**Financial Analysis**—Would you like a financial assistant who could instantly give you the figures for almost any kind of investment? Financial Analysis can handle annuities, sinking funds, and mortgages, and compute bond yield and value. You'll have the facts you need at the tips of your fingers with this program.

Included in the package is one specially marked blank data cassette for use in storing essential business data.

Business Package IV, with its combination of analytic functions and convenience features, is an invaluable asset for any businessman. All you need is a TRS-80 Level I 4K or Level II 16K. **Order No. 0019R $9.95.**

BASIC AND INTERMEDIATE LUNAR LANDER Bring your lander in under manual control. The BASIC version is for beginners; the Intermediate version is more difficult, with a choice of landing areas and rugged terrain. For one player with a TRS-80 Level I 4K, Level II 16K. **Order No. 0001R $7.95.**

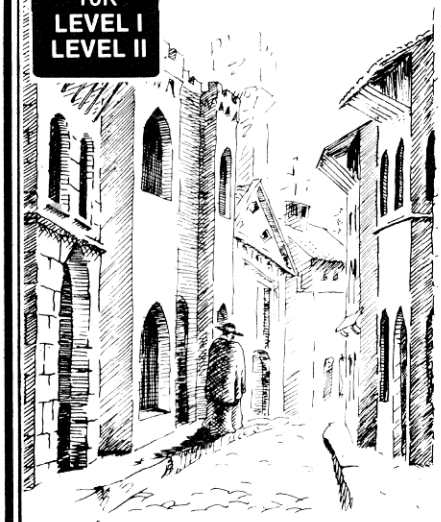


HOUSEHOLD ACCOUNTANT

•**Budget and Expense Analysis**—Impose order on your tangled financial affairs. This program makes provision for twenty-seven expense categories and three income sources. Budgets can be reviewed monthly, quarterly, and yearly. Dollar and percentage comparisons can be made between budgets, months, and year-end totals, so you can see where your money is going.

•**Life Insurance Cost Comparison**—Learn how your choice of life insurance policy can save you money. This program can help you decide between term or dividend-paying whole life policies. It will also compare within categories, when, for example, two whole life plans vary in dividends and in cash values. You can store and display up to six different value comparisons. Requires a Level II 16K. **Order No. 0069R $7.95.**

Designed for use on **TRS-80*** **16K LEVEL I LEVEL II**



SANTA PARAVIA AND FIUMACCIO

The year is A.D. 1400, and you are the ruler of a tiny Italian city-state. You are ambitious by nature and intend to build your little city-state into a powerful kingdom.

So begins Santa Paravia and Fiumaccio, where you and your fellow players compete as rulers of neighboring cities. You control the grain harvest, feed your people, set tax rates, exercise justice, invest in public works and, of course, try to stay on the good side of the church.

Life was short back then, and you'll have only a limited amount of time in which to build your kingdom. The lives of your serfs will depend on your decisions. If they are wise, then your city-state will grow and you will acquire loftier titles. If your rule is incompetent, your people will starve and your city-state may be invaded by your neighbors.

You can play the game yourself or set up the tournament version, which allows up to six players to compete at one time. Either way, you're sure to find your route to the throne a challenging road.

How will you rule your kingdom? Will you be a benevolent ruler – an iron fist in a velvet glove – or will you become unscrupulous and follow the example set by Niccolo Machiavelli in his book on government, **The Prince?**. Only you can answer that question—with the Santa Paravia and Fiumaccio. Requires a Level I or II 16K. **Order No. 0043R $7.95.**

LEVEL I

HEXPAWN/SHUTTLE CRAFT DOCKING/ SPACE CHASE/SEA BATTLE This four-game package is sure to provide hours of fun for the whole family.
- **Hexpawn** – Turn your TRS-80 into a model of artificial intelligence by playing a simple game.
- **Shuttle Craft Docking** – Land your shuttle craft on the starship – even through varying gravity fields!
- **Space Chase** – Seek out and destroy the enemy delta that's hidden in the star field.
- **Sea Battle** – You must find and destroy the enemy fleet.

This package requires a TRS-80 Level I 16K. **Order No. 0041R $7.95.**

BUSINESS PACKAGE III This package can change a TRS-80 into a full-working partner for any businessman:
- **Inventory** – Maintain a computer-based inventory for a constant inventory system.
- **Commissions and Percentages** – Let your computer figure out markup and discount calculations, sales tax and more. This is a perfect time-saving package for any small business.

For the TRS-80 Level I 4K. **Order No. 0061R $7.95.**

SPACE TREK III Let yourself go to the far ends of the solar system – and beyond. This package includes:
- **Stellar Wars** – Shoot down the Tie fighters and destroy the Death Star.
- **Planetary Lander** – Land your spacecraft and plant your flag across the solar system.

These one-player games require a TRS-80 Level I 4K. **Order No. 0031R $7.95.**

FUN PACKAGE I Why call it "Fun Package"? Judge for yourself! This entertaining package includes:
- **Rocket Pilot** – Flying it is easy – it's the landing that's tough!
- **Paper, Rock, Scissors** – It's the time-honored game just as you remember it, played against your TRS-80.
- **Hex I** – Just when you master this puzzle game, the computer will increase the difficulty.
- **Missile Attack** – Use your missiles to protect your city from jet attack.

Requires a Level I 16K TRS-80. **Order No. 0037R $7.95.**

TYPING TEACHER Tired of being a two-fingered wonder at the keyboard? Do you tell your friends that you use the "Columbian" method, because you discovered it yourself? Well, you can change from that awkward hunt-and-peck approach and start typing like a professional with this practical instructional program from Instant Software.

This seven-part program turns your TRS-80 into both typewriter and instructor as it takes you from initial familiarization with the keyboard through the typing of words and phrases and on to complete mastery of the keyboard.

On-screen diagrams show you the correct placement for fingers, and the program's practice sessions will quickly improve your coordination. In no time at all you'll be typing in that smooth, free-flowing style you've always admired. For the 4K Level I TRS-80. **Order No. 0099R $7.95.**

DOODLES AND DISPLAYS I Here's a mixed bag of programs that's sure to entertain:
- **Doodle Pad** – Draw pictures and save them on cassette tapes.
- **Symmetrics** – Turn your TRS-80 into a kaleidoscope.
- **Video Display** – Follow the bouncing cursor as your TRS-80 draws its own pictures.
- **Mathcurves** – Bring those geometry lessons to life as the computer draws six different geometrical curves.
- **Rugpatterns** – A never ending stream of symmetrical patterns that's sure to spark your imagination.

All you'll need is a 16K Level I TRS-80. **Order No. 0030R $7.95.**

CAR RACE/RAT TRAP/ ANTIAIRCRAFT Enjoy these challenging, fun-filled programs:
- **Car Race** – You and a friend can race on a choice of two tracks.
- **Rat Trap** – Trap the rat in his maze with your two cats. For one player.
- **Antiaircraft** – Aim and shoot down the enemy airplane. Requires a Level I 4K TRS-80. **Order No. 0011R $7.95.**

KNIGHT'S QUEST/ROBOT CHASE/HORSE RACE This varied package of one-player games will give you hours of fun.
- **Knight's Quest** – Battle demons to gain treasure and become a full-fledged knight.
- **Robot Chase** – Destroy the deadly robots without electrocuting yourself.
- **Horse Race** – Place your bet and cheer your horse to the finish line.

These programs require a TRS-80 Level I 4K. **Order No. 0003R $7.95.**

CAVE EXPLORING/YACHT/MEMORY These three programs are not only fun, but stimulating as well:
- **Cave Exploring** – Search for fabulous treasures as you explore the magic cave. For one player.
- **Yacht** – A two-player game of strategy and chance. The computer rolls the dice and keeps score.
- **Memory** – Two players can pit their memories in this program based on a popular television show. You'll need a TRS-80 with Level I and 16K. **Order No. 0010R $7.95.**

DESTROY ALL SUBS/GUNBOATS/BOMBER This package of three programs is fun for the whole family. Included are:
- **Destroy All Subs** – Hunt down enemy subs while avoiding mines and torpedoes. A one-player game.
- **Gunboats** – Try to blow the enemy's ship out of the water. For one or two players.
- **Bomber** – Carefully release your bomb to destroy the moving submarine. A one-player game.

To enjoy these programs, you'll need a TRS-80 Level I 4K. **Order No. 0021R $7.95.**

PERSONAL FINANCE I Let your TRS-80 handle all the tedious details the next time you figure your finances:
- **Personal Finance I** – With this program you can control your incoming and outgoing expenses.
- **Checkbook** – Your TRS-80 can balance your checkbook and keep a detailed list of expenses for tax time.

This handy financial control for the home requires only a TRS-80 Level I 4K. **Order No. 0027R $7.95.**

DEMO I This package is just the thing to show your friends what your TRS-80 can do. Included are:
- **Computer Composer** – Compose and play music using only a standard AM radio.
- **Baseball** – Play baseball with your computer while it does the scorekeeping.
- **Horse Race** – Place your bet and cheer your pony to the winner's circle.
- **ESP** – Test your powers of extrasensory perception.
- **Hi-Lo/Tic-tac-toe** – Guess the secret number or get three in a row.
- **Petals Around the Rose** – Can you figure out the secret behind the five dice?
- **Slot Machine** – Turn your computer into a one-armed bandit. These programs require a TRS-80 Level I 4K. **Order No. 0020R $7.95.**

BUSINESS PACKAGE I Keep the books for a small business with your TRS-80 Level I 4K. The six programs included are:
- **General Information** – The instructions for using the package.
- **Fixed Asset Control** – This will give you a list of your fixed assets and term depreciation.
- **Detail Input** – This program lets you create and record your general ledger on tape for fast access.
- **Month and Year to Date Merge** – This program will take your monthly ledger data and give you a year to date ledger.
- **Profit and Loss** – With this program you can quickly get trial balance and profit-and-loss statements.
- **Year-End Balance** – This program will combine all your data from the profit-and-loss statements into a year-end balance sheet.

With this package, you can make your TRS-80 a working partner. **Order No. 0013R $29.95.**

*A trademark of Tandy Corporation

SEE INSIDE BACK COVER FOR LIST OF DEALERS NEAR YOU.

Name _____

Address _____

City _____

State _____ Zip _____

☐ Check ☐ Money order

☐ VISA ☐ AMEX ☐ Master Charge

Card No. _____

Expiration Date _____

Signed _____ Date _____

Order your Instant Software today!

Quantity	Order No.	Unit Cost	Total Cost
		Handling	$1.00
		Total Order	

Instant Software Inc. **Dept. 70B0**
✓2 Peterborough, N.H. 03458 USA

TRS-80 Disk & Other Mysteries

by Harv Pennington

Computers are addictive. Harv Pennington is a certified, grade A1, computer addict. To support his habit he has been forced into writing the definitive work on the TRS-80 disk system. In less than two years, Harv has become an expert on the darkest secrets of the DOS. His book is overflowing with useful information, most of it unavailable elsewhere.

Rather than a normal review, we have arranged to extract sections for your enlightenment.

The whole 132 page book was produced on a TRS-80 with the aid of Michael Shrayer's Electric Pencil. The TRS-80 Disk & Other Mysteries—don't boot up without it!—Eds.

Just by reading this book, one might get the impression that the only thing the TRS-80 is good for is to fix errors that are created on the machine! Not true! Not a single day goes by that I do not use my TRS-80 for some useful and productive purpose. Occasional errors are just a small part of the day to day experience. It is only when you cannot fix those errors that they begin to dominate the "computer experience."

It is my wish that you will, as a result of this book, be able to make your TRS-80 one hundred percent productive and enjoyable.

No doubt you have been told that you cannot do certain things with the TRS-80—like "BOOT" a "BASIC PROGRAM" because you need "BASIC" to load a program—or that you cannot lock out the break key without messing up the I/O routines—or that you cannot defeat the "LIST" and "LLIST" commands. You have been told wrong. All of these things can be done! I have been able to do all of the above with little or no trouble. The ONLY limitation you have is your own imagination.

. . . Oh, yes. This couldn't have been done without an incredible program called "SUPERZAP." It is a product of APPARAT Inc., of Denver, Colorado. You may purchase a copy of this program with the NEW DOS operating system from your local software dealer. Ordering information is in the appendix at the end of this tome. You will find that "SUPERZAP" is indispensable if you are going to take the voyage to the bottom of the disk.

"SUPERZAP" is unique in several ways. First it has its own disk I/O routines and does not require that a DOS be in drive zero to perform miracles. Second, it will read ANYTHING that is readable, regardless of its "PROTECT" status. Third, it will recover almost every imaginable type of error condition.

In addition, it has a "BACKUP" routine that will make umpty-ump tries when it encounters an incorrect or electrically damaged sector before it gives up. Then, it allows you to try again as many times as you want!

Wait! There's more . . . it will copy disk sectors, relocate disk sectors, allow modification of any byte or combination of bytes on the disk or in memory, move data from one disk sector to another and "ZERO" disk sectors.

Operating Systems

This will be a brief review of the various operating systems that are available as of this writing. I will not dwell too long on the pros and cons of each and you must remember that the following is an OPINION, mine.

TRSDOS 2.1

Except for the few unfortunate souls that started with 2.0 this is the operating system that most of us developed our first, genuine love-hate relationship with. For all practical purposes, due to the short life of 2.0, this was the "FIRST" operating system generally available for the TRS-80.

2.1 has many problems. Of course, Radio Shack never came out and admitted, in plain English (at least to me—did they tell you?), that the problems existed. TRSDOS 2.1 is adequate for most trivial programming requirements and a few serious applications IF you are prepared to tolerate an occasional lost file.

TRSDOS 2.2

TRSDOS 2.2 is a huge improvement over 2.1. Most of the errors are corrected. However, it will still create errors. Most of the complaints I have about the system are that they still have not given the user any of the utility that is available with NEW DOS.

As far as data recovery goes, there is one major point. When you "KILL" a file with 2.2, it ZEROS THE ENTIRE DIRECTORY ENTRY. There is not a single clue as to what was there or where it was! Since Radio Shack has no utility for looking at the disk, I presume it was to prevent all you "SUPERZAPPERS" out there from finding out too much! However, if you need to recover something, this makes it not impossible but a genuine bitch because you have to go "mucking around on the disk" looking for the file.

VTOS 3.0

This is Randy Cook's version of 2.2 with quite a few bells and whistles. Cook is the author of Radio Shack's 2.1 and, I have reason to suspect, most of 2.2. This system has some nice features but is, in my opinion, VERY AGGRAVATING to use because of its "BACKUP" protection feature. In the version that I used for evaluation, some of the commands did not work entirely as advertised. I'm sure that this will be corrected in a later release. On the whole, the system is good and the concepts are excellent. I have not used it enough, at this time, to have detected any errors, if it has any.

NEW DOS 2.1

It works! The current release has no known bugs and will do everything Radio Shack says cannot be done. It corrects every KNOWN error in TRSDOS 2.1. All in all, there are over 200 additions, corrections, and enhancements to TRSDOS. Many of the "improvements" in TRSDOS 2.2 are poor "implementations" of NEWDOS 2.1. (That's an opinion, and I cannot verify it, but from the looks of things, I'd give better than even odds that it's true.)

NEWDOS 2.1 is oriented to the programmer as well as the user. Included in the NEW DOS+ package, are utilities such as "SUPERZAP," "DIRCHECK," "LMOFFSET" and others. These utilities are especially designed to assist the user and are very necessary if you need to recover data.

Disk Organization

In the TRS-DOS 2.1 MANUAL we are told that we have 67 GRANULES of free space on

a formatted disk and somewhat less on a disk with a DOS. Fig. 1 is a "MAP" of a typical "SYSTEM DISK" (TRS-DOS 2.1). You will notice that the system programs are grouped together. It is not absolutely necessary that this always be the case. In fact it is possible to put the SYSTEM programs anywhere except for "BOOT/SYS," SYS0/SYS" and "DIR/SYS." NEWDOS requires that "SYS13/SYS," when it is resident on the diskette, to be specifically located also.

Other programs such as FORMAT/CMD and BASIC/CMD may not be in the same location on your disk, especially if you have "COPY"ed these programs from another disk.

Disk allocation is handled in groups of 5 sectors at a time. For this reason every program or file is allocated disk space in 5 sector chunks called "GRANULES."

TRS-DOS 2.1 and 2.2 assign a minimum of two GRANULES at a time. That is why you run out of disk space so quickly when you have a bunch of small files or programs. NEW DOS assigns only one GRANULE at a time.

Actually there are only a few areas on the disk that **must** contain specific object code material. There are "BOOT/SYS," "SYS0/SYS" and "DIR/SYS." "BOOT" must always be located on Track 0, beginning at sector zero. 'SYS0/SYS" must be located on track "0," sector 5 and "DIR/SYS" must be located on Track 11 (HEX) beginning at sector zero.

TRS DOS 2.1 DISK MAP (35 TRACK)

TRACK NUMBER HEX/DECIMAL	GRANULE NUMBER (HEX)	:<-------- TRACK CONTENTS --------->: SECTORS 0 - 4 :<----GRANULE---->	SECTORS 5 - 9 :<----GRANULE---->
0 - 0	0 & 1	:<----BOOT/SYS--->	:<----SYS8/SYS--->
1 - 1	2 & 3	:<----SYS0/SYS--->	:<----SYS9/SYS--->
2 - 2	4 & 5	:<--FORMAT/CMD--->	:<--FORMAT/CMD--->
3 - 3	6 & 7	:<--FORMAT/CMD--->	:<--BACKUP/CMD--->
4 - 4	8 & 9	:<--BACKUP/CMD--->	:<--BACKUP/CMD--->
5 - 5	A & B	:<------FREE------>	:<------FREE------>
6 - 6	C & D	:<------FREE------>	:<------FREE------>
7 - 7	E & F	:<------FREE------>	:<------FREE------>
8 - 8	10 & 11	:<------FREE------>	:<------FREE------>
9 - 9	12 & 13	:<------FREE------>	:<------FREE------>
A - 10	14 & 15	:<------FREE------>	:<------FREE------>
B - 11	16 & 17	:<------FREE------>	:<------FREE------>
C - 12	18 & 19	:<------FREE------>	:<------FREE------>
D - 13	1A & 1B	:<------FREE------>	:<------FREE------>
E - 14	1C & 1D	:<------FREE------>	:<------FREE------>
F - 15	1E & 1F	:<------FREE------>	:<------FREE------>
10 - 16	20 & 21	:<----SYS1/SYS--->	:<----SYS2/SYS--->
11 - 17	22 & 23	:<----DIR/SYS---->	:<----DIR/SYS---->
12 - 18	24 & 25	:<----SYS3/SYS--->	:<----SYS4/SYS--->
13 - 19	26 & 27	:<----SYS5/SYS--->	:<----SYS6/SYS--->
14 - 20	28 & 29	:<----SYS6/SYS--->	:<----SYS6/SYS--->
15 - 21	2A & 2B	:<---BASIC/CMD--->	:<---BASIC/CMD--->
16 - 22	2C & 2D	:<---BASIC/CMD--->	:<---BASIC/CMD--->
17 - 23	2E & 2F	:<------FREE------>	:<------FREE------>
18 - 24	30 & 31	:<------FREE------>	:<------FREE------>
19 - 25	32 & 33	:<------FREE------>	:<------FREE------>
1A - 26	34 & 35	:<------FREE------>	:<------FREE------>
1B - 27	36 & 37	:<------FREE------>	:<------FREE------>
1C - 28	38 & 39	:<------FREE------>	:<------FREE------>
1D - 29	3A & 3B	:<------FREE------>	:<------FREE------>
1E - 30	3C & 3D	:<------FREE------>	:<------FREE------>
1F - 31	3E & 3F	:<------FREE------>	:<------FREE------>
20 - 32	40 & 41	:<------FREE------>	:<------FREE------>
21 - 33	42 & 43	:<------FREE------>	:<------FREE------>
22 - 34	44 & 45	:<------FREE------>	:<------FREE------>

Fig. 1.

The directory may be moved (it's a hassle) to another location. It MUST also be read protected. If the directory is moved, "SAVE" has a bitch of a time trying to figure out where to put the directory information since it expects the directory to be on track 11 (HEX). Eventually it will find it and deposit its data in the right places. This can be speeded up a bit by changing relative byte "02"(HEX), in the "BOOT" (track "0," sector "0") to the HEX value of the track you have moved the directory to.

The "BOOT" is not actually a program but rather a machine language "TABLE" that is automatically loaded on power-up or reset—sometimes referred to as "IPL." (Initial Program Load. "IPL" is computer jargon for, "Push the button, Hilda!")

The Directory

The key to finding anything on the disk is the directory. Even the operating system can't find anything without the directory.

The directory is located on track 17 (11 HEX). It is composed of 10 sectors of 256 bytes per sector. This gives the directory 2,560 bytes in which to store data. There are no unused bytes in the directory. Fig. 2 is a "MAP" of the DIRECTORY.

The minimum space allocated for storing any type of file, is one "GRANULE." (No, Virginia, I do not know where the word "GRANULE" came from. Perhaps it describes the size brain of the person who thought of inventing another "computer jargon" term.) At any rate, the overall scheme for representing free space is as follows:

5 sectors = 1 granule 2 granules = 1 track

Recovering A "Direct Statement In File"

I must confess that the first hundred times I encountered this error it nearly drove me crazy. (Nearly?) Before I got the TRS-80, I had never laid hands on a computer in my life and the cryptic messages from this magic machine, without explanations, were completely baffling.

What made it doubly worse is the fact that neither the LEVEL II manual or the disk manual gave the slightest clue as to what a "DIRECT STATEMENT IN FILE" was or how it got there.

This little cutie may occur in one of two ways. It is usually the result of a very minor "bug" in LEVEL 2 BASIC. It happens when you "SAVE" a program that has a statement line that is longer than 240 bytes.

How can that happen? Easy. It happens when you "EDIT" a long line and insert more characters than the disk operating system can handle. Normally the system checks line lengths and will not allow you to make a line too long. In the "EDIT" mode however, the checking does not function quite correctly.

The other condition is very similar to the "EDIT" condition, in that you "SAVED" a file *with the ASCII option,* and it had statement lines that were longer than 240 bytes *when the BASIC tokens were expanded to their full englih equivalent!*

In the TRS-80 LEVEL II manual, Appendix A, page 16 it clearly states:

Program Line Length: Up to 255 characters.

Actually BASIC will only "LOAD" 240 characters of program material at a time! An assembly language "OBJECT CODE MODULE" will load 256 characters of program material. A random file record or an ASCII data record, on the other hand, will load up to 255 characters with TRSDOS 2.1 and NEWDOS 2.1 and 256 characters with TRSDOS 2.2.

For a BASIC program, each statement line must have a line number. The condition that exists with a "DIRECT STATEMENT IN FILE" is that the computer loaded a line with a line number and 240 characters and there

were some characters left over. These are the "DIRECT STATEMENTS" that are in the file. Since they don't have line numbers, BASIC doesn't know what to do with them!

What does that have to do with the ASCII mode? Well, Level 2 BASIC actually uses "TOKENS" to store program statements in memory. For instance, when you type "PRINT" the machine does not store the actual characters that you typed or that it is displaying on the video. It is actually storing a "?" in memory. This "?" takes only one byte to store. The word "PRINT" would take 5 bytes to store. (See Appendix A for a complete listing of the LEVEL II "BASIC TOKENS")

When you are writing a program, the system keeps track of how many characters each "TOKEN" would take if it were completely spelled out. This would NORMALLY prevent you from getting a direct statement in file when you "SAVE" a program file with the ASCII option. In the "EDIT" mode, LEVEL 2 will allow you to insert a few extra characters—just enough to put you over the legal limit. There you have it, friends and neighbors—the Secret of the Shifting Whispering Sands.

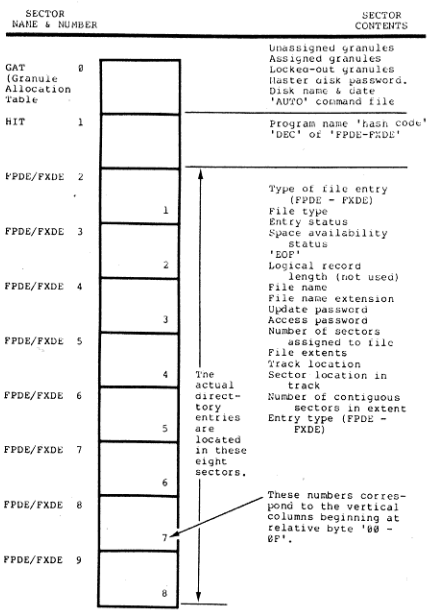


Fig. 2.

Now, what to do about it. Actually this is a fairly easy condition to fix. All we need to do is insert a line number in front of the offending "DIRECT STATEMENT" that's in the file.

ASCII File With Direct Statement Error

(1) Determine the last line number that "LOAD"ed.

(2) Determine the last characters that "LOAD"ed. (Use "LIST" to determine (1) and (2).)

(3) Locate the file on the disk, using the previously described methods in 10.0.

(4) Scan the sectors of the file until the sector with the error is found. This will be easy with as ASCII file because everything, including line numbers, are in readable form.

(5) Now, "ZAP" a line number anywhere in the offending line that is LARGER than the preceding line number and SMALLER than the next line number. You will lose a few characters of your program. (A small price to pay.)

Electric Pencil Goodies

Here are a couple of things that might make your day brighter, for what it's worth.

To make "ELECTRIC PENCIL" compatible with NEWDOS 2.1 all we have to do is change 3 bytes in relative sector 0 of "PENCIL" to "00 00 00." Find relative sector 0 then, at or near relative byte "AE," you will see the following code:

F332 9B46 C36F

"ZAP" this so it reads:

F300 0000 C36F

Another thing you might like to do is speed-up "PENCIL's cursor—a simple one byte change. In relative sector 10 (HEX) on or about relative byte "7B" you will see the following code:

0600 10FE 1116

"ZAP" this so it reads:

0664 10FE 1116

My cursor is set at "50." The "00" that is in there now, is a value of 256—this is as SLOW as it can possibly go. A little experimentation will tell you what value to put into this byte. A word of caution . . . "50" really makes that thing zip along.

In addition to all the before-mentioned "goodies" you can do with the "PENCIL," here are a few more.

Write BASIC Programs in PENCIL

Wouldn't it be neat to be able to write programs in BASIC and have the editing features of "PENCIL"! It's not only possible but I do it all the time.

There is no secret, all you have to do is just do it. No tricks, no special things to do, just write the program like you would normally do, only use "ELECTRIC PENCIL" to write the program. When finished, write the text to disk; exit "PENCIL," go to BASIC, and "LOAD" and "RUN." There are only two things to watch for. (1) Your file name will have "/PCL" on the end of it and, (2) only put carriage returns at the end of a statement line. Now go do it and see how easy it is.

"Load"ing a BASIC Program or ASCII Data File into "Electric Pencil"

You will only have difficulty in doing this, if you have "packed" your BASIC program or data file, i.e., eliminated all spaces between words, statements, and characters.

One of the really neat things about using "PENCIL" with a BASIC file as text, is the global search and replace. You can replace every single "PRINT" with an "LPRINT" in less than a couple of seconds! You can also use it to make translations from one dialect of BASIC to another. Using "PENCIL" enter the text for a BASIC program out of a magazine. Don't try to make all of the statement conversions. At the end of the magazine version, enter the subroutines that replace the non-"RUN"able statements. Now "SEARCH AND REPLACE" these statements with a "GOSUB" to your subroutine. In a few minutes you can make a translation that would normally take hours or even days!

"PENCIL" MUST have at least one space every 30 or so characters for its video display management routine. Now that you know what can go wrong, let's give it a whirl.

"You can replace every single "PRINT" with an "LPRINT" in less than a couple of seconds!"

Enter any simple program but when you "SAVE" it, use this or a similar name:

SAVE"FILETEST/PCL",A

Remember, "ELECTRIC PENCIL" only loads files with the "file name extension" of "/PCL." The ",A" at the end of the "SAVE" statement, will cause the program to be "SAVE"ed in ASCII format.

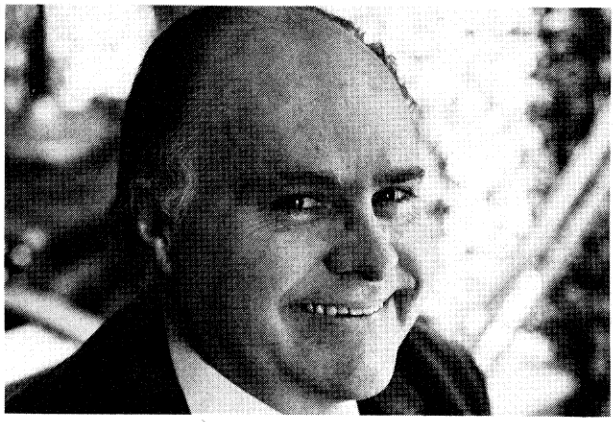
Now, with "SUPERZAP," locate the end of the "FILETEST/PCL" file and "ZAP" the last carriage return ("0D" HEX) with a "00." Execute "PENCIL" and load "FILETEST."

Another way to do the same thing, if you are using NEW DOS, is to use the "OPEN'E' " function. After you have "SAVE"ed the program, type "NEW" and enter and run the following:

```
100 OPEN"E",1,"FILETEST/PCL"
200 PRINT #1, CHR$(0)+"      "
300 CLOSE
```

This will open the file at the end and write the "EOF" marker for "ELECTRIC PENCIL." The blanks between quotes will guarantee that the file will load into "PENCIL" and not give you that "DOS ERROR 22" crap. You may do nearly the same thing with TRSDOS except you will have to read the file and write it to another file, then when you get to the end of the original file, write the "EOF" marker to the new file. Actually, you should save yourself a lot of grief and aggravation, get NEWDOS and be done with it!

The above techniques can be used with data files as well as program files. ■



Okay, now you've had a chance to see what I have in mind for you with *80 MICROCOMPUTING*. Oh, I admit that we're just getting started and that the magazine will be improving a lot as we go along. We have some interesting ideas in the works for you.

With the TRS-80 (or 90 . . . etc.) being the most popular microcomputer in the entire world, you are going to benefit from this in many ways. The more computers there are out there of one kind . . . the more good programs you are going to have for this system. I hope that is obvious. You may be sure that *80 MICROCOMPUTING* will be packed with the shorter programs and reviews of the larger ones. You can waste an awful lot of money on stuff that looks great in the ads, but fizzles out when you try to use it. You need our reviews.

The wealth of programs will also mean that there will be much better programs for the TRS-80 than any other system. Put yourself in the seat of a computer programmer and you'll understand this. If you are going to spend several months developing a comprehensive program, and it takes all of that to write and debug a big program, would you write it for a system which has sold one hundred units or one which has sold over 300,000 systems? The answer is obvious . . . and this is why we are already seeing programs coming out for the "80" which are far better than anything for any other system on the market. This is tough for other systems . . . the law of the computer jungle.

Between our connections with Instant Software, the largest publisher of microcomputer programs in the world, and Kilobaud MICROCOMPUTING, you know that *80 MICROCOMPUTING* is going to be your most important link with software for the TRS-80.

With Instant Software being sold and promoted in every country in the world where the TRS-80 is being sold, our input of programs is also the best in the world. We get programs submitted from everywhere . . . often from 50 to 100 a week! You'll get the cream of the crop either published or reviewed in "80".

HARDWARE TOO

The same law of the computer jungle holds for hardware. Would you, as a manufacturer, market an accessory for a system which has sold 100 units or would you go first for the one which has sold hundreds of thousands. It is, as with software, self-evident why the great bulk of the hardware accessories for computers are for the TRS-80 these days.

80 MICROCOMPUTING has the advantage of the use of the largest and most complete microcomputer lab in the world . . . the one developed for Instant Software and Kilobaud MICROCOMPUTING. This means that most new pieces of equipment are tested and in use by our staff . . . and this means that we can tell you what we think is outstanding . . . and where we find ripoffs. This lab is important to you.

SUBSCRIBE

If you are not already a subscriber to *80 MICROCOMPUTING*, please get signed up right now. The yearly rates are $15, and that is a bargain. Just one single program of use to you can be worth much more than that. One review of an accessory could save you many times that much investment. I would appreciate it if you would appoint yourself a committee of one to get more subscribers for the magazine. You will benefit even more than we do here at the magazine . . . because the more readers we have, the more ads we will be able to attract . . . and the more ads, the more pages of articles you will get every month.

The "80" market can, I think, support a couple of hundred pages of ads . . . and that would mean a magazine of nearly 500 pages a month. That should hold you. You may not have time left to use your computer.

ENCYCLOPEDIA

If you've read Kilobaud MICROCOMPUTING, you know that I try hard not to duplicate published material. My concept is that every reader should save every issue (we sell inexpensive boxes for this so they can sit on your library shelf) and treat the magazine as a continuing encyclopedia of computing. I make sure that much of the material in each issue is written in simple language so it will be understandable by even the rawest newcomer to computers. Oh, I have articles for the more advanced users too, so you'll have something to look back over later and use as your understanding of your system grows.

Try to think of *80 MICROCOMPUTING* as more of a large club newsletter than an ivory tower high-level publication. I'll leave the pomp to other publishers . . . the ones with the well-deserved inferiority complexes who cater to their inadequacies by publishing esoteric baloney. This magazine is written by the readers and edited by people whose aim is to help you enjoy your TRS-80.

SAVE

With each issue costing $2 at your computer store, that's $24 a year. For $15 a year you can subscribe . . . at least for now. As the magazine expands, please do not be surprised if the cover price increases, along with the subscription price. I started 73 Magazine for radio amateurs twenty years ago with a cover price of 37¢ (two for 73¢) and it is up to $2.50 a copy now (and it is the largest of the ham magazines).

For you bargain hunters . . . and those who find that one year goes by all too rapidly, the three year rate for "80" is $36. This, too, will be going up . . . reflecting the inflation, paper increases, postage increases, and a short vacation for me in Hong Kong next year. Someone has to pay for *that*.

The Bottom Shelf: Software For All Seasons

by Dan B. Shuford

Anyone who has bought a microcomputer in the past few years—a relatively new consumer phenomenon—and expected to find a wide selection of useful software has undoubtedly been disappointed.

While companies with the capital assets necessary for software development chose to wait for their market to clearly emerge, most computer owners waited for support.

Despite the risks, some programmers or freelancers began to peddle home-grown software while several cottage industries sprang up around the country. One such company that has grown from a partnership and a fondness for programming is The Bottom Shelf, Inc. of Atlanta, Georgia.

Founded from Need

When Delbert Jones and Sturgis Bates first met at a meeting of Atlanta computer users, Radio Shack's TRS-80 had been on the market about a year. Both men had purchased TRS-80's for their own business and personal use.

Bates, 42, a former radio engineer and an attorney for 11 years, had found a calculator too limited for making the complicated computations required of estate law.

Jones, 35, whose varied background had included data processing, federal procurement and business management, sought to alleviate the endless repetition of business budgeting and correspondence involved in office management.

Both Bates and Jones felt that only the scarcity of good software prevented the TRS-80 from serving small business as a versatile, cost efficient device. In addition, each shared a love for programming. The two men pooled their experience and resources and formed The Bottom Shelf.

For their initial entry into the software market, they conceived and developed the Library 100—a five-tape BASIC collection of 102 individual programs for the TRS-80. By spreading the selection of programs across business, educational, graphics, entertainment and home applications, Bates and Jones felt they not only could insure the marketability of the product, but just as importantly, could provide a general introduction for new owners, many of whom were likely to be computer novices. The programs were intended to serve as examples of good programming techniques and to illustrate the range of what the computer could be made to do.

As both a philosophic outlook and a marketing strategy, this consumer-supportive approach has become a hallmark of The Bottom Shelf's orientation in the microcomputer industry.

Neither Bates nor Jones, however, had envisioned just how successful the Library 100 would be. Bates reported that when Library 100 was released in November of 1978, orders rose to almost ten times the anticipated sales and that not until the following February could production catch up with demand.

The Library 100 turned out to be a primer not only for the computer user, but also for TBS itself. At that time, the field of mass-production of software was almost totally unexplored. Moreover, as far as Bates and Jones knew, no one had ever attempted to produce an album-type collection containing such a large quantity of programs.

Starting from scratch, TBS worked for almost three months just to get a reliable signal on tape.

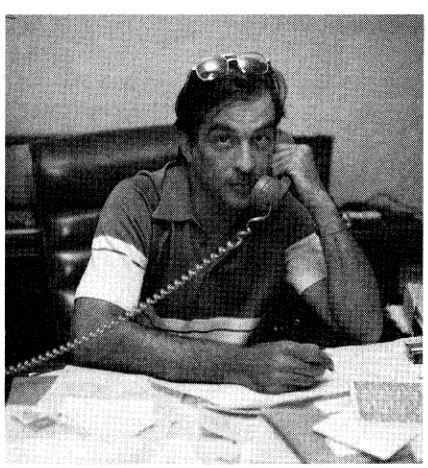
Initial Success

Library 100 also proved to be a hard taskmaster with regard to the development of systems utilities. These utility programs are used to facilitate the production of other programs or enhance them to allow more aesthetic screen-formatting, to accelerate the speed of various functions or to increase the amount of data space available to the computer user through the use of more compact coding.

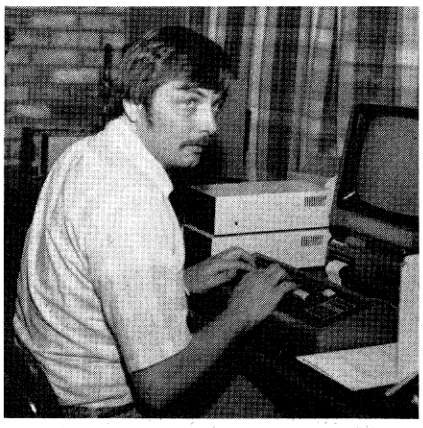
"A lot of it is trial and error," Bates explains, "building experience with the machine itself and building a set of utilities to pull in when necessary. We're getting away from doing things in BASIC, and we've developed a whole library of machine language routines."

The financial success of Library 100 (Jones reports sales in 44 countries), enabled Bates and Jones to hire additional staff members.

Expanding beyond the limitations of a two-man organization, TBS embarked upon its second major project in January of 1979, the pro-



Sturgis Bates



Delbert Jones.

duction and publication of *Systems Extensions*. Released in June, the publication continued the company's orientation towards consumer education.

Systems Extensions contains articles aimed at providing the novice with a basic computer background.

Although the articles vary in quality and the appearance of the text is somewhat stark, the publication does provide the persevering reader with a unique compendium of information on the computer.

The second half of *Systems Extensions* is a catalog of TBS support items for microcomputers, including a small but intriguing selection of software for the TRS-80.

With a disproportionate number of software companies targeting games toward the consumer market, TBS has chosen to place its emphasis on making the computer useful to its owner. The majority of their new software offerings are utilities, business or general applications. Utilities, such as The Basic Toolkit and System Doctor, a diagnostic program, are aimed at the serious user.

Avoiding the traditional class of business programs such as payroll, accounts payable and receivable, inventory, etc., TBS has developed programs of a more specific type. Programs such as the Business Mail System, Check Register Accounting System (for small businesses), Analysis Pad (a computerized columnar pad), have been constructed to allow the user to initialize the program as much as possible, adapting it to fit his own needs.

Bates and Jones believe that the experience they are gaining from the development of these smaller scale business programs will serve them well when the time is right to market a larger business package.

Security

TBS has devoted much time and effort to developing effective security for their software. With their most recent device, TBS believes they have solved the problem of diskette security. The technique involves the use of a separate security disk which is necessary to run the main program. While the program itself can be readily copied, the security disk cannot. This allows the buyer to make as many backup copies of the program as he wants, but prevents their being used by anyone who doesn't have the security disk.

A different technique has been developed for tapes, but cannot yet provide backup copies.

TBS has devoted its efforts not only to marketing new software, but also hardware, firmware and accessories, including paper and printer supplies, cassettes, diskettes, programming aids and storage units for almost everything. A few items (such as the Lightning Buster which protects the system against heavy electrical surges) were developed through research by TBS, but in most cases The Bottom Shelf serves as a distributor.

According to Bates, TBS's relationship with Radio Shack has been mixed at best. Radio Shack Associate Stores, Bates says, have been more than happy to get TBS's support products since this gives them an entree into a market from which they would otherwise largely be excluded. The relationship with Radio Shack company stores has been much cooler. Tandy's corporate policy, Bates says, restricts the company stores from buying any items over $25 from outside businesses and has prohibited these stores from displaying any items other than those marketed by Radio Shack. He adds, however, "We would like very much to work with Tandy. We feel that quite often they are not doing the things they should in order to properly support their own computer. We would like to see them do more."

Anticipating the success of Radio Shack's Model II, TBS has already developed two software packages for the market. In addition, TBS is adapting several products to meet the needs of the Apple or Pet computer owner.

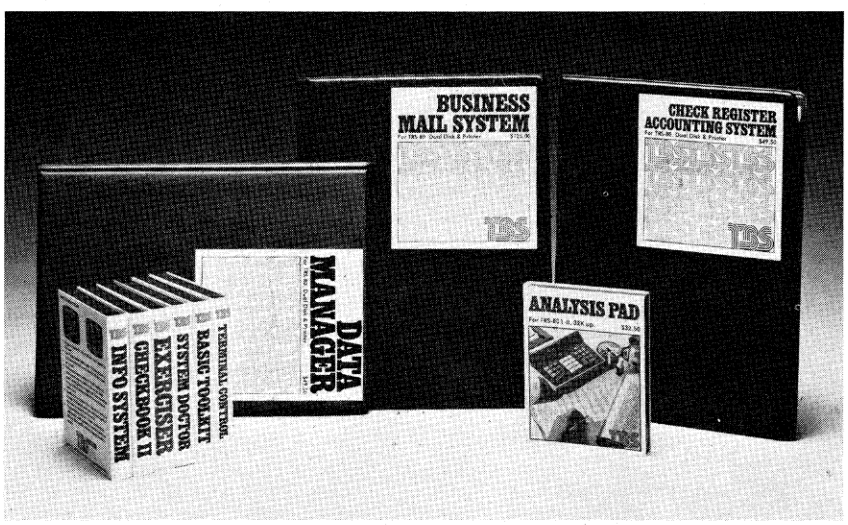
Product Expansion

Although the administrative duties of running the company have kept Bates and Jones away from the computers, both men look forward to a chance to get back to programming.

While Bates is hoping to find time to develop a real estate package, Jones recently took time off to rework and finalize the production of Analysis Pad—a direct outgrowth of his experience in business management.

Projects are also in the works for educational packages aimed primarily at the lower grade levels. These programs are to be used in conjunction with a text and to supplement home study.

While the internal utilities for such programs are almost completed, TBS feels that much of the market software will have to wait until the sales volume of microcomputers can support its mass-production. ■



Four more TRS-80 publications and an update on last month's review.

Rival Publications II

Rod Hallen
Road Runner Ranch
PO Box 73
Tombstone AZ 85638

TRS-80-related publications fill a definite need: They pass along information, advertising and programs that are of interest to the TRS-80 owner.

In Part 2 we will continue our review of these publications following the format of Part 1, and listing the prices of the publications.

TRS-80 Publications

Name: *80-US* (formerly *80-NW)*
Publisher: 80-NW Publishing Co., PO Box 7112, Tacoma WA 98407
Interval: Bimonthly
Price: $16 per year
Format: 40 pages—8½ × 11
Comments: After reading the November-December and January-February issues, I think that *80-US* is heading in the right direction. The publisher seems determined to put out a good, useful magazine and to keep to a regular bimonthly schedule. Some advertising, all TRS-80 related, is contained in each issue.

This magazine is software oriented and it contains many BASIC programs. For instance, the January-February issue has a bowling game, a text editor, a math tutorial and two others. Most of these programs are also available on cassette at very reasonable prices.

Hardware and software reviews, instructional and informational articles, letters and editorial comments give *80-US* a well-rounded outlook.

Name: *Guide to TRS-80 Information*
Publisher: F.E. Huebner, PO Box 37206, Oak Park MI 48237
Interval: Unknown
Price: $2.30
Format: 36 pages—5½ × 8½
Comments: I have the second edition, which is a gold mine of information. Included between its covers are the following chapters: Software articles list, Software list, Hardware articles list, Hardware list, General programming information, Users' group information, Bugs, Ideas, Tips, Special comments and predictions. Also hidden in here are 21 useful tips to better cassette reliability.

If your computer magazine library is growing, you need something like this to point you to articles and programs for your TRS-80. Its references are up-to-date.

Name: *Insiders*
Publisher: Computer Cablevision, Inc., 2617 42nd St. NW #2, Washington DC 20007
Interval: Bimonthly
Price: $7.50 for six issues
Format: 16 pages—5½ × 8½
Comments: This little flier will interest mainly the TRS-80 user who is into machine-language programming and hardware. It explains how to use the subroutines in the BASIC ROM, how to interface peripherals and how to get around Level II program bugs. The only issue that I've seen (8.0) contains one page of advertising. Those of us who like to dig into our computers will find this useful.

Name: *TRS-80 Monthly Newsletter*
Publisher: H&E Computronics, Inc., Box 149, New City NY 10956
Interval: Monthly
Price: $24 per year
Format: 32 pages—8½ × 11
Comments: This software-oriented magazine consists mainly of program listings. It does, however, contain a few advertisements and short notes on various happenings in the TRS-80 field. The January 1979 issue has a complete (13 page) BASIC listing of a Federal Income Tax program. It looks like it might be quite useful, but I am adverse to hand-typing programs this long.

Other programs include an Expansion Interface test, Graphics demo, RAM test and Line Renumber. I entered this last one in a hurry because it is quite short and I've been looking for a good renumber routine. It turned out to be a big disappointment, however. It renumbers the lines OK, but not the GOTOs and GOSUBs. It does give you a list of line numbers to change, but for a fairly large program that is an awful long list. Many hobbyists will find enough here to warrant spending $24 a year, but I didn't.

Other Info

The following publications are not devoted exclusively to the TRS-80, but they all contain information that should be of interest.

Name: *The SSI Microcomputer Software Guide*
Publisher: SSI, 4327 E Grove St., Phoenix AZ 85040
Interval: Annual?
Price: $7.95
Format: 124 pages—5½ × 8½
Comments: This is a general software guide. Listed in its 124 pages are over 2000 programs arranged in 236 different categories. Since a great many of these programs are written in BASIC, they can easily be converted to Level I or II by the budding personal programmer. Examples of some of the categories are: games, music, DOS, simulations and graphics. Games listings alone take up 19 pages and consist of approximately 475 references of books, magazines and vendors.

THERE IS A DIFFERENCE IN TRS-80 DISK DRIVES CAPACITY

Expansion interface – gives your TRS-80 the disk capacity it needs, and much, much more!

10 to 40 MByte, 8" Winchester drive – expands capacity far beyond Model II storage.

Single sided minifloppy – up to 150 KBytes of storage capacity.

Single or double sided 8" floppies – up to 2.5 MBytes in dual drive cabinet – for the serious TRS-80 user.

LOBO DRIVES' new family of disk memory products provides you with a choice of memory capacities you need to effectively execute the complex business software you've developed for your TRS-80*. LOBO DRIVES' selection of readily available, software compatible drives permits you to expand your inventory, payroll, customer list, and accounts receivable files as your business grows.

And LOBO DRIVES brings you more . . . a new plug-in expansion interface that provides an easy way to add hardware enhancements, communications capability, and programmable features . . . and it comes with the LOBO DRIVES famous 1 year, 100% parts/labor warranty.

Call or write for the complete LOBO DRIVES story. Find out just how competitively priced a family of high capacity drives can be . . .

✓15

LOBO drives INTERNATIONAL

935 Camino Del Sur Goleta, California 93017.
(805) 685-4546

"CAN YOU REALLY AFFORD TO PAY LESS?"

Quantity discounts available – Dealer inquiries invited

TR-2

Yes, I want to know more about LOBO Drives and what they can do for my TRS-80. Send me information on:

- ☐ 5 1/4-in. Floppy drive
- ☐ 8-in. Winchester hard disk, 10 Mbyte drive
- ☐ 8-in. Floppy drive Single sided Double sided
- ☐ Double density expansion interface

Name _____

Company _____

Address _____

City _____ State _____ Zip _____

Phone No. _____

If dealer, provide resale no. _____

*TRS-80 is a registered trademark of Radio Shack, a Tandy Company.

✓ Reader Service—see page 131

My copy of the Software Guide is the second edition. According to the publisher, the third edition—greatly expanded and improved— should be out by the time you read this. Improvements include listing software by chip (8080, 6800, Z80, etc.) and by machine (TRS-80, PET, SOL, etc.), which will make it even more useful.

Name: *Periodical Guide for Computerists*
Publisher: E. Berg, 1360 S.W. 199th Ct., Aloha OR 97005
Interval: Annual
Price: $5.00
Format: 72 pages—8½ × 11
Comments: This is a reference list of magazine articles that will be of interest to the personal computer owner. It is divided into 120 categories that span alphabetically from amateur radio to weather. Many articles are listed in more than one category. Included in the references are articles, book reviews, editorials, letters and miscellaneous items. Also provided are the addresses of all of the periodicals that were used to make up the lists and an index of authors that refers you to where the categories appear.

I have the 1975-76 and the 1977 editions, and I'm looking forward to the 1978 edition, which should be out by the time you read this. Since each edition pertains to a specific period of time, you might want all of these if you have an extensive library. I believe that all editions are still available.

Name: *Software Buyer's Guide*
Publisher: Wallace Electronics, 4921 N. Sheridan Rd., Peoria IL 61614
Interval: Unknown
Price: $.50
Format: 74 pages—8½ × 11
Comments: This is really a catalog, although the advertisement that describes it calls it a TRS-80 and Apple buyer's guide to software, accessories and supplies. Apple information overwhelms that for the TRS-80 about 10 to 1. The TRS-80 listings are exclusively software oriented. While the $.50 cost for mailing and handling isn't particularly excessive, I don't feel that I should have to pay to let someone tell me what they have for sale. Offered in the back of this catalog is a one year *Buyer's Guide* update service for $25. Since almost everything offered in this catalog is advertised in the magazines, I can't see paying $25 for more of the same.

Name: *Recreational Computing*
Publisher: Peoples's Computer Company, PO Box E, Menlo Park CA 94025
Interval: Bimonthly
Price: $10 for six issues
Format: 64 pages—8½ × 11
Comments: *Recreational Computing,* named *People's Computers* until recently, is one of the oldest magazines in the personal computer field. The name of the magazine was changed to more properly indicate its editorial slant. While it is not concerned strictly with the TRS-80, the popularity of that machine is reflected in the large amount of TRS-80 material that it contains.

This is a magazine for those of you who are interested in the recreational aspects of the personal computer. It contains games, tutorials, science-fiction stories, how-I-did-it and general-interest articles. The educational possibilities of the computer are also stressed. At the present time it does not carry any advertisements but apparently will start to do so in the near future. If you look on your computer as a source of recreation then you should subscribe to *Recreational Computing.*

Conclusion

I have just received the latest issues of three publications that I reviewed a while back and think that an update is called for. *TRS-80 Bulletin* and *TRS-80 Computing* seem to be sticking to their publishing schedules and continue to contain much of interest to the TRS-80 hobbyist. The *Bulletin* is given away free at computer stores. *TRS-80 Computing* #3 contains the results of a survey among its subscribers. Lowercase, denser graphics and better cassette operation head the list of features that owners would like to see. I hope a copy of these survey results finds its way to the Radio Shack Computer Engineering Department. TRS-80 Computing is available $15 for 12 issues.

The latest edition of "Robert Elliot Purser's Reference List of TRS-80, PET and Apple Computer Cassettes" (#5) has doubled in price, but since it contains four times as many pages as its predecessor, it is still a bargain. Hundreds, if not thousands, of programs are listed. It costs $4.

Although I wrote to *Level I* magazine two months ago because I was unable to load any of the issues of that cassette magazine that I had received, I have yet to get a reply or replacement cassettes. I can't therefore, review *Level I.*

There are three references in my library that I use more than any others: the *SSI Software Guide,* the *Periodical Guide for Computerists* and *Purser's List,* which was reviewed in Part 1 of the TRS-80 publications review.

How many times have you looked for a magazine article or program that you recalled seeing but couldn't remember where you had seen it? I usually find what I'm looking for in one of these three references. ■

MAKE EVERY BYTE COUNT

IN YOUR TRS-80® MODEL I OR MODEL II DISK SYSTEM

PROGRAMMING TOOLS

Buy 4, get 1 Any 5, $79.80* **FREE**

SUPERSEDE $19.95
For Model II $29.95
Write programs in shorthand - change variable names - generate program documentation - use with REBUILD and MINGLE to build new programs from old ones.

REBUILD $19.95
For Model II $29.95
Reorganize programs for adding program code, faster execution, readability. Rearrange groups of program statements - line numbers automatically updated. Use with SUPERSEDE and MINGLE for maximum effect.

SHRINK $19.95
For Model II $29.95
Makes Every Byte Count! Make programs smaller and faster! Combines lines & removes unnecessary code including remarks, without altering program operation.

DIVERGE $19.95
For Model II $29.95
Compares two BASIC program files, showing the differences between them. Identifies & lists lines which have been inserted, deleted, & replaced. Use for version control.

SIFTER $19.95
For Model II $29.95
Nine in-memory high-speed sorts for use in any BASIC program: stable, non-stable, with/without tags, for numeric or string data. Random Access Sort included. Relocate as needed with REBUILD.

MINGLE $19.95
For Model II $29.95
Combine sorted files into one file. May be used to combine program or data files.

*For Model II $119.80

AILING INFORMATION?

Doctor it up with AIDS-II. This Automated Information Directory System is user-defined, features user-specified fields and print/display formats, conditional record selection, updating of fields within records, sorting by any combination of fields, and more! Unique "windowing" capability allows directories of unlimited size. Window size is typically 200 or more records in 32K. Greatly expanded product based on the very popular AIDS system.

MTC AIDS-II $49.95
For Model II $79.95

PEEK-A-BOO MODEL II?

Frustrated because PEEK and POKE have been removed from MODEL II BASIC? Satisfy your curiosity with PEEK&POKE from MTC. Included are 8-bit and 16-bit (LSB,MSB) self-relocating machine language routines, instructions, and demo program.

PEEK&POKE $14.95

TIRED DISKS?

MAILSORT replaces existing sort in Radio Shack Disk Mailing List® system. Sort by any combination of fields. Hi-speed in-memory routine sorts at appx. 5 records per second. Minimum 32K required.

MAILSORT $19.95

FILE OUT-OF-SORTS?

Sort ASCII files by any combination of user-specified fields; ascending or descending, numeric or non-numeric keys. Optionally retains original file.

FILESORT $29.95
For Model II $49.95

SAME OLD B.S.?

Not from MTC! Our exclusive Technical Bulletin Service reveals the inside story on the TRS-80® I & II. Sent by first class mail, bulletins are issued as the new breaks, not just once a month. Expensive but worth it! Cancel any time -unused balance refunded. Free year-end subject index with 12-month ($36) subscription.

MTC Tech B.S. $3/month

Single sided, Single density, Soft-sectored

DISKETTES

Verbatim 5¼-inch

$24.95 Box of 10

With plastic Library Case $28.44
8-inch FLOPPIES
Single-density, Box of 10 ..... $29.95
With plastic Library Case $33.65
Double-density, Box of 10 ..... $39.95
With plastic Library Case $43.60
Library Cases available separately.
Factory Fresh, First Quality
Minimum order 1 box, NO order limit..

Complete with all utilities

NEWDOS +

$69.95 by Apparat

includes REF, RENUM, SUPERZAP, EDITOR/ASSEM., DISASSEM., DIRCHECK, and more!

Transfer PROGRAMS and DATA from MODEL I to MODEL II

TRAN-SEND

$49.95 by MTC

Requires MODEL II and MODEL I with disk & RS-232. Simple to use, not a kit -nothing else to buy. Complete with custom cable, 5¼" & 8" floppies, instructions. May be used over phone lines.

Custom Cable only $19.95

Suitable for use with Radio Shack® transfer program (ACT-0131)

MTC-SHIRTS

HIGH-QUALITY, POLY-COTTON BLEND T-SHIRTS. WHITE with Navy Blue neck and sleeve "ringers". MTC logo on sleeve. Top quality transfer of your choice on front.

mTc-SHIRTS $5.95

Specify size (S, M, L, XL) and Transfer:
- META TECHNOLOGIES MAKES EVERY BYTE COUNT!
- DON'T TOUCH MY BITS!
- RAM it!
- MICRO LOVERS TAKE SMALLER BYTES!

All products guaranteed for replacement only. Prices, Specifications & Offerings subject to change without notice.

MOST ORDERS SHIPPED WITHIN 24 HOURS

DEALER & USER GROUP INQUIRIES INVITED

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

- Add $2.50 for shipping & handling
- $2.00 EXTRA for C.O.D.
- Ohio residents add 5½% sales tax.

TO ORDER CALL TOLL FREE

1-800-321-3552

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

META TECHNOLOGIES CORPORATION ✓20

26111 Brush Avenue, Euclid, Ohio 44132

master charge

VISA

791215

® TM TANDY CORP.

Frightened by machine code? Use this elegant monitor to get to know the inside of your 80, gently.

Babybug 1

Dennis Bathory Kitsz
Roxbury, VT 05669

BASIC is a powerful, useful language, easy for the TRS-80 user to understand. On the other hand, machine-language tapes, which are entered into the TRS-80 in the SYSTEM mode, have a mystery about them. They do not LIST, they refuse to acknowledge standard keyboard commands, and they often load into some etherial area bounded by MEMORY SIZE.

These are programs which, because of their complexity, repetitive nature, or need for speed, must be written as lists of commands that can direct the central processor without the time-consuming electronic drudgery of interpretation through BASIC.

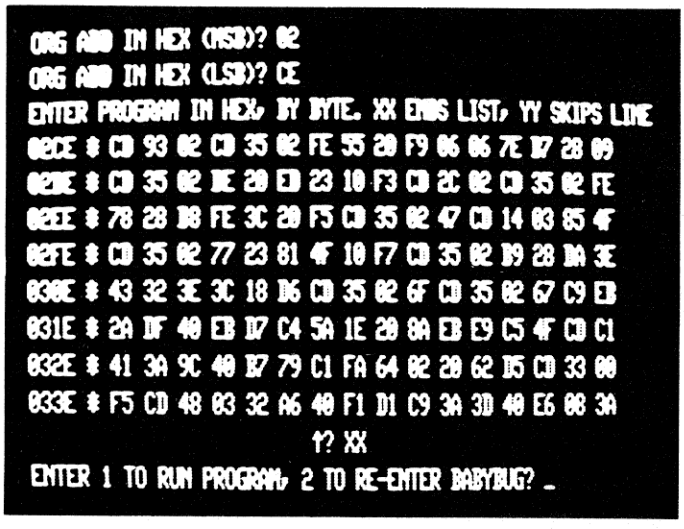


Photo 1. The Babybug Monitor, starting from address 02CE. 128 memory locations are displayed on eight lines of 16 bytes each.

What's Babybug?

Babybug is a program which allows you to create and run programs written in machine language. It is a good tool for learning to create small, useful machine-language modules, and is also useful to those with experience in assembling programs by hand.

If you have used an editor/assembler for creating small programs, you know the frustration of loading and reloading the editor/assembler tape and the source and object codes before you can successfully complete a module of code.

On the other hand, if you have never written a byte of machine code, the anonymous-looking strings of characters can be confusing, and the melange of letters and numbers called hexadecimal notation makes little sense at first glance.

Babybug is a compromise between painful hand-assembly and the convenient yet time-consuming editor/assembler.

Part of what makes computers both exciting and difficult to program at the bit level is their stubbornness. As electronic devices, they lack the ability to conceptualize, to adjust their point of view. Only through great effort and cross-referencing can they be made to accept and execute instructions in a human-oriented language like BASIC.

The computer can only comprehend blocks of on and off pulses and since processors have been developed as number-crunching accessories to our lives, the best way to interpret such blocks of pulses is in a numerical format. Thus, an "on" signal becomes one, and "off" becomes zero. There is no useful third condition in this binary system. All calculations and instructions are complex patterns of on-off, one-zero.

Since we use the decimal number system every day, computer operations had to become more manageable to our "decimal eyes." If we make a block of binary digits, four-bits (signals) wide, it makes counting easy; with two such blocks we can represent a good deal of information; and with four blocks we can number over 65,000 places in memory.

A lot of forethought has gone into the development of these electronic circuits, because

nearly everything we need to use is done in blocks of 4, 8 and 16 binary digits. And there is a good way to represent one block of four by a single "number."

Binary "Block"	Decimal Conversion	Hexadecimal Equivalent "Number"
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	10	A
1011	11	B
1100	12	C
1101	13	D
1110	14	E
1111	15	F
10000	16	10

Because hexadecimal "carries the one" at the same time a block of four binary digits does, it follows the electronic pattern much better than decimal. It takes some getting used to, but if your goal is to produce some interesting and often very fast programs, machine language can put both you and your TRS-80 to the test.

Babybug is a BASIC program that allows you to create these machine-language programs in hexadecimal form. It is compact, easy-to-use, and loads in less than 20 seconds. If you blow up your program, you can reload it and the chances are good that your machine program will be waiting where you left it, ready to revise and run again.

To write these programs you will need a list of the Z80 processor's instruction set, which is available with the Radio Shack Editor/Assembler package, or separately in the *Z80 Technical Manual* (Zilog, Inc., 10340 Bubb Road, Cupertino, CA 95014).

The best source for a beginner learning the instruction set is *The Z-80 Microcomputer Handbook*, by William Barden, Jr., and available from Howard W. Sams Co. Although marred by frequent typographical errors that could temporarily waylay a novice, its explanations are lucid and its solutions creative and interesting.

How Babybug Works

Set up the two BASIC programs in Listing 1.

These programs are the only ways of "painting" the screen white in BASIC. If you have been using your TRS-80 for awhile, or have purchased SYSTEM-loading tapes that used TRS-80 graphics, you may have wondered how to produce spontaneous-looking graphics.

We can apply Babybug to this problem. First, let's see how Babybug works.

A starting address is entered from the keyboard in hexadecimal form, which the program accepts as a string. A subroutine converts this into decimal pieces and POKEs them into the USR(0) locations. This allows the program to be run at any time.

The entire hexadecimal address is also converted to decimal form. A PEEK is performed into this and the next fifteen memory locations and a second subroutine converts these PEEKs back into hexadecimal form. The address and the contents of those memory locations are then displayed on the screen.

An arrow and a prompt appear under the first byte of code presented and the program is ready to be given some input: either a hexadecimal number to load into the location pointed to by the arrow, or one of a few commands. The commands can skip the line, leaving the contents undisturbed; can execute the program starting at the origin address; or can restart at any address.

A memory location is loaded by typing the hexadecimal number and pressing ENTER; the conversion subroutine once again translates this string into decimal format and POKEs it into that memory spot. The program advances the arrow and with it the address. After the contents of a line have been changed, the new listing is written in place of the former one; the address display is dropped from the screen to indicate that the line has been altered.

The pointer may then be advanced from line to line. At the completion of eight lines, the screen clears and the listing continues. Entering XX will return you to a Babybug command level at any time.

```
10 FOR X = 0 TO 127
20 FOR Y = 0 TO 47
30 SET (X,Y)
40 NEXT Y,X
50 GOTO 50
```

```
10 FOR X = 15360 TO 16383
20 POKE X, 191
30 NEXT X
40 GOTO 40
```

Listing 1.

```
21 00 3C 11 01 3C 01 FF 03 36 BF ED B0 21 FF FF  2B 7C B5 C2 10 4C C9 xx
```

Listing 2.

```
3E 00 CD 12 02 CD 96 02 21 00 4D E5 CD 35 02 E1 77 23 18 F7 xx
```

Listing 3.

"Painting" the Screen

First, power up the TRS-80 and respond to MEMORY SIZE? by entering 18432; this leaves plenty of room for the Babybug program to operate and lots of memory to play with. The next step is to load Babybug and RUN.

One of the goals in developing this program was to have it load quickly for constant re-use; because of this, the prompts it provides for you are minimal.

The first question you will be asked, then, is to enter a location for the machine program to start. Unlike BASIC's renumberable first line, this starting (origin) address is fixed; it is one of the numbered memory cells of the TRS-80. The purpose though, is the same. Hence,

```
ORG ADD IN HEX (MSB)?
```

is an abbreviated way of asking for the most significant byte (eight binary digits) of the starting address, written as a two-digit hexadecimal number.

You would want to choose any place that is out of the way of BASIC (higher than the protected memory of 4800 hex = 18432 decimal). For convenience, this example starts at location 4C00. Enter the most significant byte, 4C, and after the prompt

```
ORG ADD IN HEX (LSB)?
```

enter 00. The machine next prompts you with

```
ENTER PROGRAM IN HEX BY BYTE
XX ENDS LIST, YY SKIPS LINE
```

and displays the origin address and the contents of the 16 memory locations beginning there.

Enter the program in Listing 2, pressing ENTER after each byte (every two hexadecimal characters). Note: Do NOT use the shift key.

Following XX, the program will return to command level, and the screen will read:

```
ENTER 1 TO RUN PROGRAM,
2 TO RE-ENTER BABYBUG
```

Double-check your work; if you need to make corrections, enter 2 to re-start the program. Key in the same address (4C00) as before, then insert the correct program bytes. Now you are ready to try a test run. Enter 1—flash! a white screen—zip! the screen clears and reads:

```
ENTER 1 TO RUN PROGRAM,
2 TO RE-ENTER BABYBUG
```

Aha! An instantaneous white screen, much faster than BASIC, yes . . . but where did it go? Let's look again at Listing 2.

The first byte the machine reads is 21. This is an instruction that says, "load your internal registers (memory slots) identified as H and L with the

next two numbers from the program." It looks for the next two bytes and finds 00 and 3C, loads them into that internal storage, glances at its electronic checklist to find if there is anything else to do, and, finding nothing, goes on to look for the next program byte.

It discovers 11, which tells it, "load another set of internal registers, this time D and E, with the two bytes that follow." It dutifully picks up the two bytes 01 and 3C, and puts them in place. A look around for other things to do, then back to your program for orders.

01 follows, which commands, "take two more registers, B and C, and fill them with the two program bytes that follow this instruction."

Again, the processor runs back to the program, finds the two bytes (FF and 03) and stores them in the B and C register pair. After some electronic housekeeping it's back to your program again.

At this point, the various processor registers have contents that look like this:

H L	D E	B C
00 3C	01 3C	FF 03

Now it sees 36, which says, "check back to that H and L register pair. Find out what number is there, then take a walk to the program, look for the next byte you find and drop it into a TRS-80 memory location with the same number as H and L." So it finds 00 and 3C in the registers. It looks up address 3C00, returns to the program, finds the next byte (BF) and stores BF in memory slot 3C00.

A Day's Work

Back to the program again: it sees ED, which says, "this is a partial command—keep looking for more." ED has told the processor to fetch the next program byte, which happens to be B0.

ED B0 represents a single command, but what a day's work it is. It says, "okay, line up those three register pairs HL, DE and BC. Find out whose address is in HL and go to that address; get what you left there earlier, then come back; find out whose address is in DE, and go there; take what you've gotten from HL and put it in the place DE indicates. Increment by one the address numbers of both HL and DE. Now turn around and take a look at BC. Decrement BC by one; is it zero yet? Well, if not, repeat your routine with HL and DE until nothing remains in the B and C registers. Do a little bookkeeping, and only then come back to the program."

By this time, the screen is white. Maybe you didn't follow that through to come up with a white screen, so here's a better look:

The program loaded H and L with 3C00, or 15360 in decimal form. If you glance back at the program which painted the screen white with POKE statements, you will find the first POKE location is 15360. This is the first spot in video memory, the first location on the upper left corner of the screen.

The machine program next loaded D and E with 3C01, the adjoining place on the screen. The third command loaded B and C with 3FF (1023 in decimal), the number of places remaining from D and E to the end of the screen.

The pattern should become clear: it filled up the 3C00 (15360) screen position with hexadecimal BF (191 decimal)—a full graphics block—which is exactly what POKE 15360,191 does. It followed by putting the block in 3C01 and incrementing both sets of registers, meaning that H and L now addressed location 3C01 (15361) and D and E pointed to 3C02 (15362).

Once more the program took what was in HL and put it in DE, again and again all the way down the screen. When BC reached zero, the process stopped, leaving a screen filled with contiguous white blocks.

The only thing left to do was to keep it white for a time like this:

✓ Reader Service—see page 131

21 FF FF 2B 7C B5 C2 10 4C C9

21, as before, commands the processor to load H and L, this time with FFFF (65535 decimal). 2B is an instruction that tells it to decrement by one the number in the H and L pair of registers.

7C orders it to load its accumulator, a special-purpose register where arithmetic can be performed, with the H byte of the HL pair. B5 instructs it to OR its contents (the H byte) with the L byte.

In an OR situation, any true condition gives a true result. The only time we can get a zero (false) result is if both bytes H and L are zero.

The next instruction, C2, is a conditional jump instruction. It can be flagged down by the above onrushing zero. It will jump every time, unless the zero flags it down.

The next two bytes tell it where to jump, 4C10. If you count carefully (in hex!) from the beginning of the program (address 4C00), you will find address 4C10 is the instruction 2B—decrement HL. It jumps back to do this over and over, FFFF times, until H and L are both zero; then the zero flag is raised and the jumping ceases. This entire procedure constructs a simple loop, very much like

```
FOR X = 65535 TO 0 STEP -1 : NEXT
```

in which a loop is maintained until X reaches 0.

But discover one very important thing about this machine language loop: It has counted from FFFF (65535) to 0, executing 262,141 total instructions, in the fraction of a second the screen remained white!

The program's final instruction, C9, is a return, bringing you back to line 19 of the Babybug program.

More Applications

Many short programs are of interest, but only two more will be developed here. The first reads in data tapes and displays their contents in hexadecimal form and the second is an entertaining BASIC Bounce program.

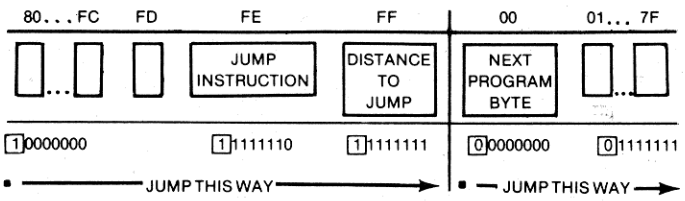


Fig. 1.

Run the Babybug program again, set the organizational address to 4C00 and enter the bytes in Listing 3.

You will be prompted once again with

```
ENTER 1 TO RUN PROGRAM,
2 TO RE-ENTER BABYBUG
```

Place any SYSTEM cassette you have into your recorder, adjust the volume as you would normally, and set it to play. Place a radio near the TRS-80 so that you can tune in on the computer's operation. Enter 1 and wait until you can hear that the data has stopped flowing. Press the Reset button on the back of the TRS-80 and run again. Babybug will begin, but this time you should originate your program at address 4D00.

Enter YY and play out eight lines of hex numbers onto the screen. What you are seeing is a list of the first 128 bytes of data on the tape you have just loaded. You may continue to skip through memory a line at a time until you reach the point where the data flow ended.

Here is a summary of how this was accomplished: 3E, the first instruction, clears the accumulator by setting it to zero. CD means call, somewhat the equivalent of GOSUB in BASIC. Following CD is 12 02, the address of the subroutine, written least significant byte first. This calls the part of the BASIC language located at 0212 that turns on the tape recorder.

Next CD 96 02 calls a subroutine internal to BASIC that searches through the tape leader buzz for the symmetrical group of eight bits which can synchronize the computer to the subsequent incoming data. When the subroutine at 0296 finds this sync byte, it returns to your program, where the next instruction is 21.

You will recall that 21 loads the H and L pair of registers, in this case with 4D00 (written, as before, 00 4D).

What follows fairly well defines what makes a computer very different from a calculator. That next instruction, E5, tells the processor to PUSH the number now resident in H and L (4D00) onto the top of a memory area called the stack.

This way the information can be saved intact for later use, out of the way of other operations you may need to perform. Although this program's stack is only two bytes deep, complex programs may require a stack of thousands of bytes.

The next operation is again a call, CD, and the place called is another internal part of the BASIC language found at location 0235. This particular routine reads the tape, assembles each arriving group of eight bits into a byte, and stores that byte in the accumulator. Then it returns to your waiting program.

The processor is now ready to use the number it stored earlier, so it will POP the number setting atop the stack back into the H and L register pair. Now the program can tell the computer why the number 4D00 is sitting in the HL registers.

It says 77, and the computer responds by taking the byte waiting in the accumulator (where it was just read in from the tape), and stores it in the memory location numbered the same as H and L—that is, 4D00.

The next directive is 23, which increments the HL registers by one (to 4D01). 18 says jump, and F7 tell how far—backwards seven steps, where it sets aside HL, reads the tape, finds HL again, stashes in 4D01 the byte from the tape, and jumps back again. You have to press Reset to get back the READY because this machine program is an endless loop.

But wait! That 18 F7 jump was backwards? Seven steps? But F7 is—calculating quickly, converting to decimal—247 steps ahead. What is happening here?

More Compromises

Do you recall the comment about computers being both exciting and difficult?

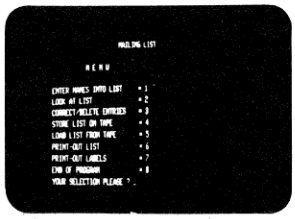
Here is another compromise effected to make the computer a useful tool. Some conditions at first appear undigitalizable, as this backwards, or negative, jump. The only way to force a negative condition is with a little electronic and mathematical trickery. There are many ways of thinking of this, whether visually or mathematically.

Mathematically, this jump is accomplished by adding the value of the jump to the next program byte—but disregarding the carry over into the most significant byte. Like this:

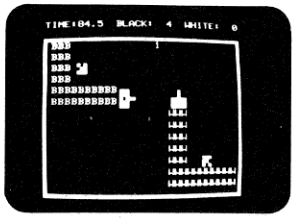
Address	PROGRAM	Action	Mnemonic**
4C00	11 00 3C	Load the D and E registers with 3C00	LD DE,3000
4C03	21 00 00	Load the H and L register with 0000	LD HL,0000
4C06	0E 20	Load the C register with 20	LD C,20
4C08	46	Load the B register with what is stored at the memory location named by H and L	LD B,(HL)
4C09	10 FE	Decrement B, and if B is not zero, jump back one step (to 4C09)	DJNZ FE
4C0B	0D	Decrement the C register	DEC C
4C0C	3E 00	Load the accumulator with zero	LD A,0
4C0E	B1	OR the accumulator contents with C register	OR C
4C0F	C2 08 4C	Jump to 4C08, unless the result of the previous operation is zero	JP NZ,4C08
4C12	23	Increment the HL register pair	INC HL
4C13	7E	Load the accumulator with what is stored in the memory addressed by HL	LD A,(HL)
4C14	12	Load the memory locations named by the DE registers with accumulator contents	LD (DE),A
4C15	13	Increment the D and E registers	INC DE
4C16	C3 06 4C	Jump to 4C06	JP 4C06

Listing 4: The BASIC Bounce

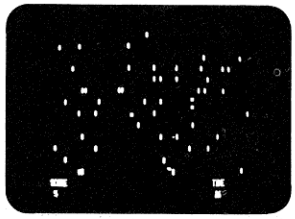
Software for the TRS-80



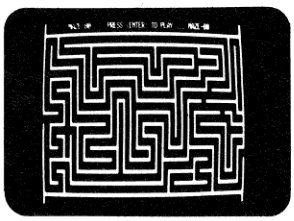
MAILING LIST $ 9.95



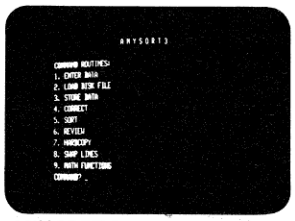
TANK $ 6.95



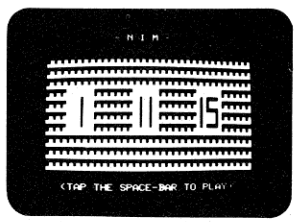
MINEFIELD $ 6.95



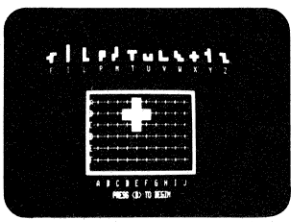
MAZE-80 $ 9.95



ANY SORT 3 $29.95



NIM $ 6.95



PENTOMINOES $ 9.95

TRS-80 TIELINE

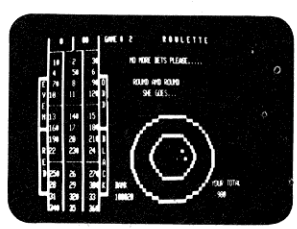
TRS-80 TIELINE is an extended smart terminal program. Functions supported with this package are the ability to send and receive BASIC data and programs. A fully supported set of timeshare ASCII control keys are software selectable. 'ESC' and a 'BREAK' key function do not require any hardware modifications. Smart functions make it possible to jump from mode to mode with communication prior to program transmission or reception. Half and full duplex modes as well as line feed transmission or suppression, baud rate, parity, word length, stop bits are software selectable and can be changed while running. A printer can be connected for hard copy of communications as well as LLIST at baud rates that include 134.5 baud for certain serial printers. A special host or source mode allows other computers to use the TRS-80 TIELINE as a timeshare style computer. Programs can be run as well as disk files loaded, saved or transmitted by control from the distant computer. Character echo-back is supported. Host override of forbidden commands is possible. An additional feature allows testing of the TRS-80/RS232 hardware. The program is available for DOS 2.0 - 2.2 machines with 32K memory, RS232 board and modem. Additionally, the package includes a free copy of 'THE TRS-80 DATA COMMUNICATION HANDBOOK' by Stephen Gibson. The handbook is a compilation of terms and in-depth explanations of data communication from the standpoint of the user, the hardware, and the phone company. Various phone line services are detailed. Baud rates, bits and codes are explained at hobbyist level. How a Bell 103 modem works and how to connect it is covered with emphasis on originate and answer frequencies, duplex, half-duplex and RS232 conventions. The EIA standards are given with ASCII control character set information.

TIELINE	$24.95
TRS-80 DATACOMM HANDBOOK	$ 5.95

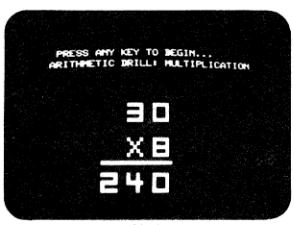
All orders include 3% postage and handling with a minimum of $1.00. California residents include 6% Sales Tax.

VISA MASTERCHARGE

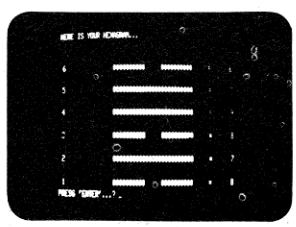
TRS-80 IS A REGISTERED TRADEMARK OF THE TANDY CORP.



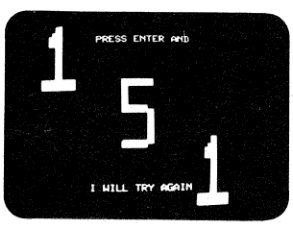
ROULETTE $ 6.95



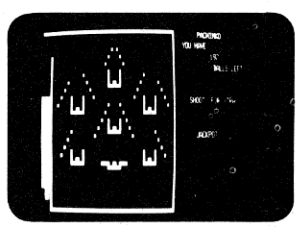
ARITHMETIC $ 6.95



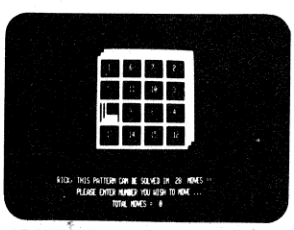
THE I CHING THING $ 6.95



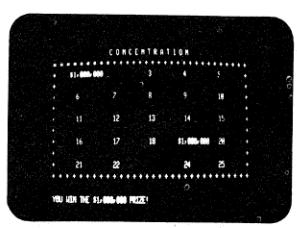
COMPUTER MINDREADER $ 6.95



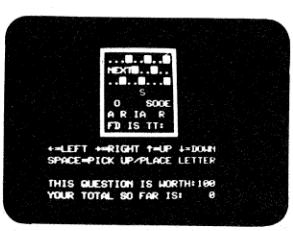
PACHINKO $ 6.95



15 NUMBERS $ 6.95



CONCENTRATION $ 6.95

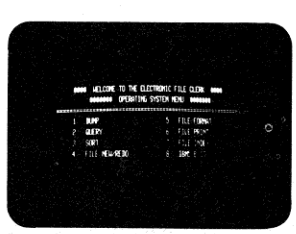


QUIZMACHINE $ 6.95

PROGRAMMA INTERNATIONAL, Inc.
3400 Wilshire Blvd.
Los Angeles, CA 90010
(213) 384-0579
384-1116
384-1117

✓21

Dealer Inquiries Invited



DBM/5 $49.95

PROGRAMMA Software Program Products

✓ Reader Service index—page 241

```
 2 CLS:PRINT"BABYBUG":FORX = 1TO500: NEXT
 3 CLS:INPUT"ORG ADD IN HEX (MSB)";R$:GOSUB20:POKE16527,M:L = M
 4 INPUT"ORG ADD IN HEX (LSB)";R$:GOSUB20:POKE16526,M:K = M
 5 Q = K + 256*L:PRINT"ENTER PROGRAM IN HEX, BY BYTE. XX ENDS LIST, YY SKIPS LINE"
 6 FORY = 1TO8:GOSUB50:FORW = 1TO16
 7 P = 260 + 3*W + (Y-1)*64:PRINT@P,CHR$(91);
 8 R$ = "":INPUTR$:IFR$ = "XX"THEN17ELSEIFR$ = "YY"THEN13ELSEIFR$ = ""THEN10
 9 GOSUB20:POKEQ,M
10 Q = Q + 1:PRINTCHR$(27);CHR$(30);:K = K + :IFK<256GOTO12
11 K = 0:L = L + 1
12 NEXTW:GOSUB60:GOTO16
13 PRINTCHR$(27);:FORX = QTOQ + (16 - W):K = K + 1:IFK<256GOTO15
14 K = 0:L = L + 1
15 NEXTX:Q = Q + (17 - W):GOSUB70:GOSUB60
16 NEXTY:CLS:PRINT:PRINT:PRINT:GOTO6
17 INPUT"ENTER 1 TO RUN PROGRAM, 2 TO RE-ENTER BABYBUG";Z
18 IFZ = 2GOTO3
19 PRINTUSR(0):CLS:GOTO17
20 R = ASC(RIGHT$(R$,1)):S = ASC(LEFT$(R$,1))
21 IF(R>47)AND(R<58)THENT = R - 48
22 IFR>57THENT = R - 55
23 IF(S>47)AND(S<58)THENU = 16*(S - 48)
24 IFS>57THENU = 16*(S - 55)
25 M = T + U:RETURN
40 A = FIX(V/256):B = V - (A*256):C = FIX(B/16):D = B - (C*16)
41 IFC>9GOTO43
42 PRINTCHR$(48 + C);:GOTO44
43 PRINTCHR$(55 + C);
44 IFD>9GOTO46
45 PRINTCHR$(48 + D);:RETURN
46 PRINTCHR$(55 + D);:RETURN
50 GOSUB70:FORX = QTOQ + 15:V = PEEK(X):GOSUB40:PRINT" ";:NEXTX:PRINT:RETURN
60 PRINTCHR$(27);TAB(7);:FORX = Q - 16TOQ - 1:V = PEEK(X):GOSUB40:PRINT" ";:NEXTX:
   PRINT:RETURN
70 V = L:GOSUB40:V = K:GOSUB40:PRINT" • ";:RETURN
```

Babybug Monitor.

```
10 FOR X = 1 TO 9 : READ A
20   FOR Y = 1 TO 3*(10-A)
30     FOR Z = 0 TO A : NEXT
40   NEXT Y,X
50 RESTORE : GOTO10
60 DATA 7,7,5,3,2,1,2,3,5
```

When you tire of this (which should be in short order), reload Babybug and enter Listing 4 starting at address 4C00.

Listing 4 has been described in three ways. The first is the hexadecimal machine code which you enter in Babybug. The next column is a verbal explanation similar to others in this article. The last column is a shortened way of representing these explanations, a method called mnemonics.

You may know the program you want to write, but the several hundred numerical codes are difficult to recall. Mnemonics are a way of tweaking your memory banks into remembering the functions of each machine command.**

There is only one more thing you need to know to understand how this program works: Address 0000 is the beginning of the BASIC language in the TRS-80.

In Part Two of this article, a simple yet powerful hardware modification to the TRS-80 will be described, along with some additional programs to illuminate the potential of this modification.

The listing of Babybug presented in this article is compressed. It uses few lines and packs many commands per line. The user's prompts are few and comment lines nonexistent. In order to get a very short program load, it was necessary to compact the information this way. If you wish to explore the detailed workings of the program, expand it to one statement per line.

The hex to decimal conversion subroutine is found at lines 20-25; the reverse process is found from 40 to 46. Line 50 prints current memory contents; line 60 the revised memory contents. ■

L = MSB origin address (in decimal)
K = LSB origin address (in decimal)
Q = The current address (in decimal)
Y = The current line on the screen
X = The current byte in the line
P = Current position of the prompt arrow
V = The current byte at Q (in decimal)

NOTES TO THE TEXT:

*I always leave a small portable radio on when using the TRS-80, because it acts as an audible trace, telling me when a key has bounced, when the computer is searching through a long array, or if it is caught up in a loop. It is particularly helpful with machine-coded programs, because there are no BASIC error messages ready to let you know your program is about to erase itself. Sometimes it can be like writing NEW into a BASIC program!

**I have chosen to use the Z-80 mnemonics rather than those for the 8080 because I find them a better visual representation of the procedure taking place. I can always be sure that HL is a number or address and (HL) the hidden contents; parentheses help me feel stored inside some place. The machine codes are the same, so you 8080 fans don't *need* to learn new mnemonics.

Next logical program address:	4C14
Value of jump:	+ F7
Result, disregarding carry over into 4C:	4C0B

The result is, of course, the address we are looking for, 4C0B, where the program finds instruction E5.

If you prefer to think in signals or bits, it is possible to conceive of the most significant binary bit of the jump's magnitude as isolated; it is assigned not a numerical value, but a kind of locational function.

All jumps are forward jumps, but they can originate from one of two possible places: either from the next logical byte of the program, or from a point 128 bytes earlier.

If the most significant bit of the jump is 0, the jump begins from the next byte; if the most significant bit of the jump is 1, the jump begins 128 bytes earlier. Fig. 1 outlines this process.

A relative jump is a very desirable computer operation and it was important to devise a way to accomplish that sort of movement. Although the method described is very bit-oriented, it will become second nature with the experience of use.

You may ask, of course, why C3 0B 4C (jump to address 4C0B) was not used instead of such a complicated procedure. Indeed, C3 0B 4C is just as valid; but in complex programs where thousands of jumps are written, each relative jump takes one less byte than an absolute jump—a 33 percent saving of memory.

But there is another important reason: You can start this program from *any address in memory* you choose. It is completely relocatable. You might use this short program as a cassette load module in some other program. Whenever you use programs with absolute jumps (such as the white-screen program), you must *rewrite every jump* to include the proper new address.

Imagine having a BASIC that forced you to number lines consecutively, to put just a single instruction on a line and to use every line. Even the addition of a single CLS might require renumbering every GOTO and GOSUB statement in the program!

The BASIC Bounce

By now you have noticed that your TRS-80 emits signals that can be heard on a nearby AM radio.* This short BASIC program demonstrates a little organized interference; try it:

Photo 2. The author's system with newly installed hexadecimal keyboard and an AM radio.

**** BUSINESS/ACCOUNTING SOFTWARE ****

We didn't INVENT the wheel ***
We just made it better -------

—but then we're not really talking about wheels, **ARE** we. We're talking about excellent, low cost, full-function, fully integrated business or accounting software—and **NO ONE** does that better than we do!!!

SO—what can you buy for **$3000** that would cost you **$5000+** elsewhere—Aaron Associates Businessmaster X naturally.***

We offer you **EVERYTHING** that Peachtree offers you plus Mailing List and Fixed Asset Accounting. We offer you a choice of Microsoft or CBASIC. We offer you over 75 pages of comprehensive **USER** documentation per module plus over 100 pages of implementor documentation for a total of almost 700 pages generously illustrated with line drawings and flow charts. **WE ALSO OFFER AN AUDIO CASSETTE WITH EACH MODULE** giving you step by step audio instructions on the use of each package. And finally, we offer a ten day trial period. If the software does not function, your money will be refunded. We believe this offer is unprecedented.

Businessmaster X adds the new dimension of Sequentially Keyed Random Access Method (SKRAM) files to our Businessmaster II.

	Businessmaster		Manuals	
	II	**X**	**II**	**X**
General Ledger	400	1200	40	120
Accounts Receivable	150	450	15	45
Accounts Payable	150	450	15	45
Payroll with Reports	200	600	20	60
Inventory	100	300	10	30
Mailing List	100	300	10	30
Fixed Asset Accounting	100	300	10	30
	----	----	----	----
All for	1000	3000	100	300
Demonstration Package with Audio cassette	100	300	10	30
Federal Tax Preparation		1000		100
Financial Planning Pack prepares 12 month projections		1000		100

Specify System: TRS-80, Apple, Pet, Dynabyte, Chromemco, North Star, Micropolis, etc.
Also specify CBASIC2 or MICROSOFT

For further information—Write or better yet call—
AARON ASSOCIATES
102 Ave de la Estrella #208
San Clemente, CA 92672
Phone (714) 492-7633 ✓8

Dealer Inquiries Invited—We accept MasterCharge and VISA

Are your programs taking too long to run? Speed them up for less than a dollar

Faster! Faster!

Dennis Bathory Kitsz
Roxbury, VT 05669

Attention courageous solder manipulators, circuit snippers and other upstarts!

Speed up your TRS-80 for a buck! No switches! No fumbling!

Here's how: get two integrated circuits, one 74LS02 and one 74LS367. Open up the TRS-80 and locate the following ICs: Z7, Z22, Z43, Z44, Z52, Z53, Z54, Z56, Z59 and Z72. Turn the main board so the solder side faces up.

Following Carefully?

Now locate Z7. Take a *sharp* Exact-o knife and cut through the solder and traces that connect together pins 7, 10, 11, 12 and 13. What this does is free all the inputs of Z7, the spare half of a 74LS74 flip flop.

Next step. Find pin 8 of Z56. Cut the foil trace that leads from this pin to the hole that is plated through the circuit board. Remember that hole. By cutting that trace, we have separated the Z-80's clock input from the 1.77 MHz output of Z56.

Now heat up the soldering iron. The next steps will require fine wire (preferably wire wrap size), good soldering technique and neat layout of wires. This is the master clock we're messing with, and sloppiness just won't do—take it from a connoisseur of sloppiness.

Back to Z7. Solder a wire connecting pins 10 and 13, and from there to +5 volts (available at pin 14). This enables the flip flop for our use. Now turn the board with the component side up. The rest of the work will be done on this side of the board.

Find Z59, which has a convenient data 0 line. Run a wire from pin 4 of Z59 to pin 12 of Z7 and solder it in place. This step makes one line of data available at the input of flip flop Z7.

Prepare the 74LS02 by bending all the leads *except* 7 and 14 so that they are parallel with the body of the integrated circuit. Find Z53. Seat the 74LS02 directly atop Z53 and solder pins 7 and 14 to it. These are the power pins on the 74LS02, which we can now call ZGATE.

Locate Z52. Run a wire from pin 6 of Z52 to pins 2 and 3 of ZGATE and solder. Now locate Z54. Run and solder a wire from pin 8 of this IC to pin 11 of ZGATE.

Next, run a short wire between pins 1 and 12 of ZGATE.

```
10  CLS : OUT 254,0 : FOR X = 1 to 100 : PRINT X; : NEXT
20  CLS : OUT 254,1 : FOR X = 1 to 100 : PRINT X; : NEXT
30  GOTO 10
```

Program Listing 1

These steps decode eight address lines into FE (decimal 254), which is the output port we will use.

Now we have to connect the output signal. Locate Z22; pin 3 has the OUT signal. Run a wire from this pin to pin 5 of ZGATE. Now solder together pins 8, 9 and 13 of ZGATE.

Finally, run a short wire between pin 6 and pin 10 of ZGATE. The remaining pin (pin 4) contains a fully decoded output signal that goes high whenever the instruction OUT254 . . . is executed. Connect a wire between this pin (pin 4) and Z7's pin 11. When activated, this line will clock the signal present at data 0 through to the outputs of Z7.

One Down, One to Go

Now pick up the 74LS367. As with the previous IC, bend all leads, except power pins 8 and 16, parallel with the body. Place the 74LS367 atop Z44, and solder pins 8 and 16 to it. Now it too will have power, so we can call the 74LS367 ZSPEED.

Locate pin 8 of Z56. (The trace to this pin was cut on the underside of the board). Run and solder a wire from this pin to pin 2 of ZSPEED. Now run a wire from pin 3 of ZSPEED to the plated-through hole, the trace to which was previously cut. Take care that this is the correct hole before soldering the wire in place.

Finally, run a wire from pin 9 of Z7 to pin 1 of ZSPEED and solder. These previous steps have reunited the normal clock speed to the CPU, but through a gate which is activated only when the signal OUT254,1 is executed.

We now need to put the higher speed hardware in place. Locate Z43, pin 2. This is a clock output running at 5.32 MHz. Run a wire from this pin to pin 14 of Z56. That is the input of an unused divide-by-two circuit, with the resulting 2.66 MHz now available at pin 12 of Z56. Run a wire from Z56 pin 12 to pin 14 of ZSPEED.

Next tie pins 13 and 3 of ZSPEED together. Last, run a wire from pin 8 of Z7 over to the second gated section of ZSPEED, pin 15. Solder.

By making these connections we send the 2.66 MHz frequency through ZSPEED to the CPU clock. Because both outputs of flip flop Z7 can never be on at the same time, only one clock frequency is gated through ZSPEED to the CPU. To get the higher frequency, OUT 254,0 must be executed.

Four gates of ZSPEED remain unused and their inputs should be tied off. Solder a wire connecting pins 4, 6, 8, 10 and 12 of ZSPEED.

This finishes the high-speed modification. Double check all the connections with the circuit diagram and with these instructions. When you are satisfied all is well, check for solder blobs and splashes; make certain that all traces that should be cut are cut and that no other traces have been damaged. Put every-

✓ Reader Service—see page 131

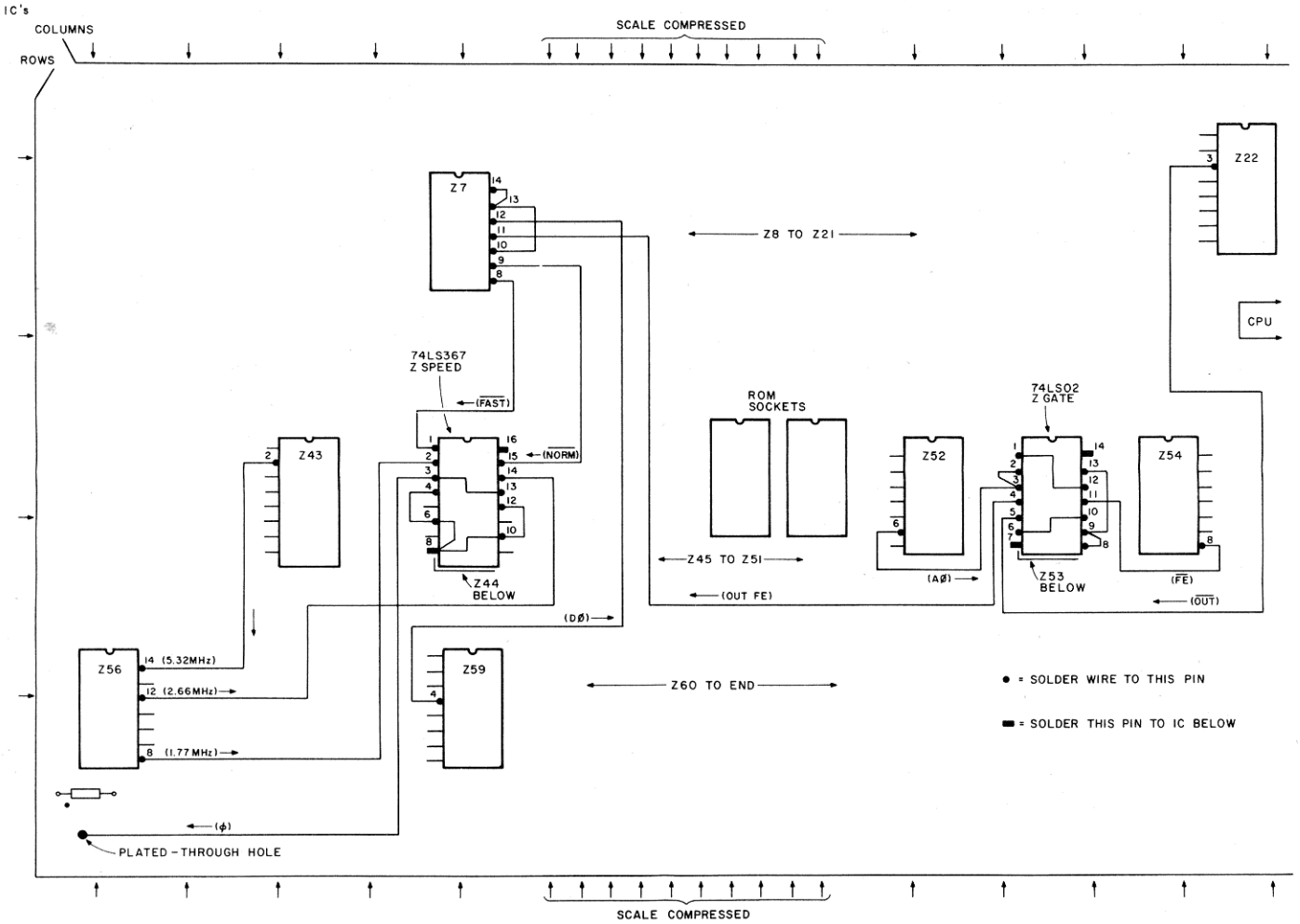


Fig. 1. Parts Layout

thing back together, power up and run Program Listing 1.

Operation

First thing to note: When loading programs from cassette, use the slower speed (OUT 254,1). If your programs load well with your present cassette system, then you should be able to dump at the higher speed without difficulty. Just command OUT 254,0 before your CSAVE command. This will provide programs that load 50% faster—but do remember which they are!

Second thing to note: You will be able to run any BASIC program at higher speed, but don't forget that all your timing loops will be faster. I would suggest making the speed selection (OUT 254,1 or OUT 254,0) the first line of any BASIC program.

Third thing to note: SYSTEM (machine-language) programs don't allow you to dump them on tape and in some cases you wouldn't even be able to command an OUT statement before running them.

To run them at the higher speed, press BREAK after the tape has loaded completely (if it's a single-load program—not like MICROCHESS, which loads in sections). Command OUT 254,0, then type SYSTEM and the appropriate starting address as instructed. ■

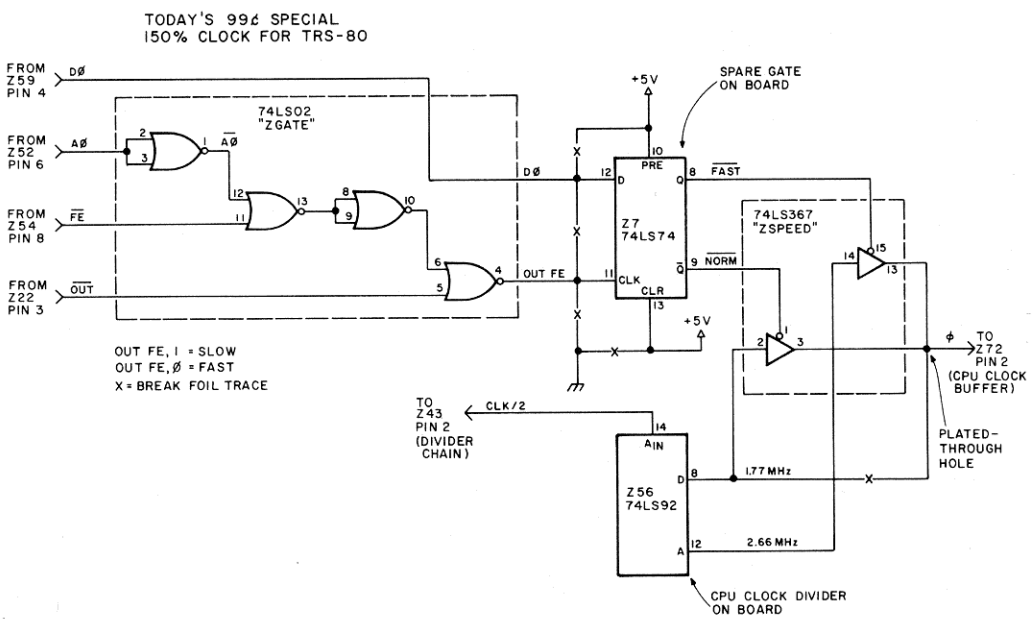


Fig. 2. Clock for TRS-80

You read what it can do last month, so why not build one!

Cassette Problems II

Donald L. Stoner
Dick Barker
The Peripheral People
Box 524
Mercer Island WA 98040

In Part 1 of this series, we examined some of the problems encountered loading and running program tapes with the TRS-80 system. Part 2 shows you how you can eliminate these problems with the Data Dubber. It virtually eliminates the critical nature of CLOADing tapes by regenerating the original CSAVE pulses. You can run the CTR-41 volume up and down or flip the tone switch back and forth while CLOADing, and your TRS-80 will never blink an asterisk!

You can also use the Data Dubber to make copies of *any* tape, even those from vendors who say their tapes can't be copied. If cassettes are your principal storage media, you can probably save the cost of the Data Dubber in a month or two. If you have written some crafty programs, you can become an entrepreneur by duplicating and distributing your software. All it takes is a couple of CTR-41s from Radio Shack, the Data Dubber and a smidgen of operating capital.

How It Works

The schematic diagram is given in Fig. 1. Starting at the input plug (P1 from the cassette), there are several proprietary features that ensure loading and reproduction of poor-quality tapes. The input circuit, consisting of C1 and T1, forms a bandpass filter network to minimize hum modulation and noise on the tape. Furthermore, ground loop hum (a notorious problem with the CTR-41) is eliminated by floating the primary of the transformer. This isolates the CTR-41 from the TRS-80.

The full wave rectifier (D1-D4) ensures that all data pulses are positive going, regardless of whether positive or negative pulses predominate on the tape. These pulses turn on the data switch transistor, Q1.

The stream of data pulses feeds a series of Schmitt trigger stages in IC1, a CMOS-type 74C14. The time constant of each stage is chosen to produce pulses of exactly the same width as the TRS-80 supplies when CSAVEing a program. The pulses at pins 4 and 8 of IC1 are combined at P2 to provide an exact reproduction of the CSAVE pulses (see Fig. 2d). Pulses are also applied to Q3 and Q4 to drive the light-emitting diode (D10) that indicates the Data Dubber is operating. The pulses applied to this lamp are also differentiated by CB (.1 uF) and the 100 Ohm input impedance of the TRS-80 CLOAD circuit.

One of the craftiest features of the unit is the Q2 circuit. During the design phase of the

Duplicating tapes is easy with this setup. Pulses from the right cassette are regenerated by the Data Dubber. These replicas of the TRS-80 CSAVE pulses are recorded by the left cassette.

BA1	9 volt battery
C1	.47 uF, 50 volt disk ceramic capacitor
C8	Mfd, 50 volt disc ceremic
C2	4.7 uF tantalum capacitor
C3	100 uF, 16 V electrolytic capacitor
C4,C5,C6,C7	470 pF disk ceramic
C9	.01 uF Mylar capacitor
C1,C5,D7-D9	1N4148 silicon diode
D6	1N4001 silicon diode
D10	Light-emitting diode
IC1	Hex Schmitt Trigger-CMOS 74C14
J1	Jack, open circuit
P1,P2	Plug, molded with shield cable (Calrad 55-904)
Q1,Q3,Q4	MPS5172 NPN silicon (see text)
Q2	MPS3638 PNP silicon (see text)
R1,R8,R9	10k 1/4W resistor
R2	47k, 1/4W resistor
R3	47 Ohm, 1/4W resistor
R4,R7	1k, 1/4W resistor
R5	15 Ohm, 1/4W resistor
R6	470k, 1/4W resistor
R10,R11	270K, 1/4W resistor
R12,R13	4.7k, 1/4W resistor
R14,R15	270 Ohm, 1/4W resistor
R16	5.6k, 1/4W resistor
R17	2.7k, 1/4W resistor
T1	Transformer, 1k to 8 Ohms (reverse connected)

Parts List

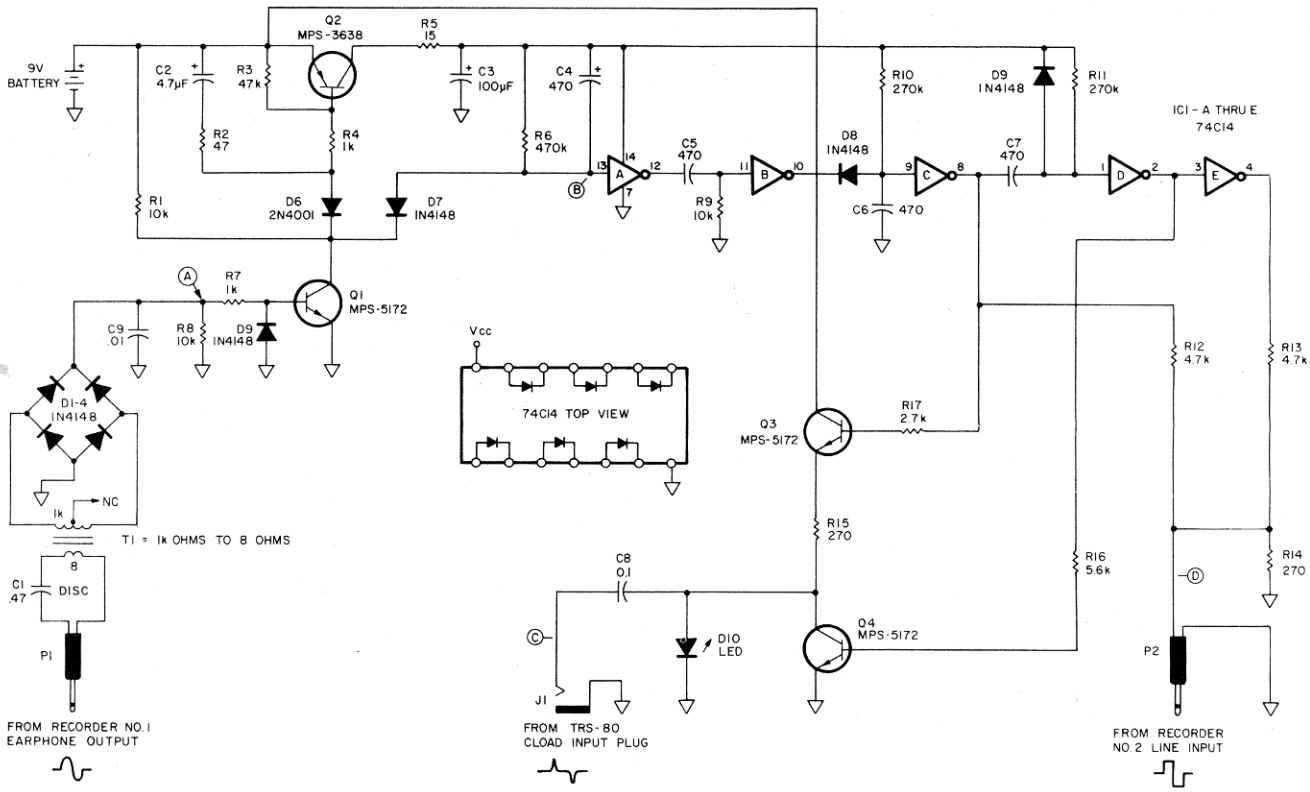


Fig. 1. Schematic diagram for the Data Dubber. Transistor Q2 is the electronic switch that turns the unit on.

Data Dubber, someone was always forgetting to switch the contraption off. A 9 volt battery was usually needed at the worst possible time (after the Radio Shack store closed), and the local drug store wouldn't accept a Radio Shack battery card! Thus, an electronic switch was added to the Data Dubber. As long as there is no signal input, the Dubber contentedly draws a fraction of a microampere from the 9 volt battery. However, as soon as there is a signal input (the beginning of the leader), the electronic switch springs to life and connects the battery to R15. The battery consumption is 7-10 milliamperes while data is coming in. As soon as the input ceases, the Data Dubber turns off automatically and goes back to sleep.

Thus, there are no controls to set or switches to throw. In fact, the only indication that the Dubber is working is the light-emitting diode on the front panel. It illuminates whenever data comes in from the cassette.

Construction

Probably the easiest way for the home constructor to duplicate the project is to build it on a piece of perfboard. The layout is totally noncritical, although you should observe the usual good construction practice of short leads.

The transistors used were selected because they were locally available at low cost, but any general-purpose PNP silicon transistor can be used for Q2. The other transistors are general-purpose NPN silicon. Use good-quality, 470 pF capacitors, since these determine the pulse width. Electrolytic capacitors should *not* be used for C1 and C8, although they are less expensive than the disk capacitors. Capacitor C2 (4.7 uF) must be a tantalum type for low leakage. The diodes are also not critical. The 1N4148 specified is a common type, but most low-leakage computer-grade silicon diodes will work. Note, however, that D6 is a 1N4001 power diode.

There is one assembly precaution regarding the integrated circuit. Since this is a CMOS type, it is subject to damage from static electricity. It is a good idea to use a socket and only insert the IC after wiring is complete. Also, don't touch the IC after you scrape across the carpet unless you touch a grounded object first.

Testing

Before applying power to the circuit, you should make a couple of pretests. First, inspect your wiring to ensure there are no shorts or bad connections. Then, measure the resistance across C3. It should measure several hundred Ohms. Finally, connect a milliammeter in series with the positive terminal of the 9 volt battery and the battery clip. If the meter pointer moves visibly, it indicates something is wrong with the circuitry of Q2. There should be no leakage through this stage.

So far, so good? Connect the battery and apply data to P1. If the LED lights up, you have probably successfully completed the construction of the Data Dubber.

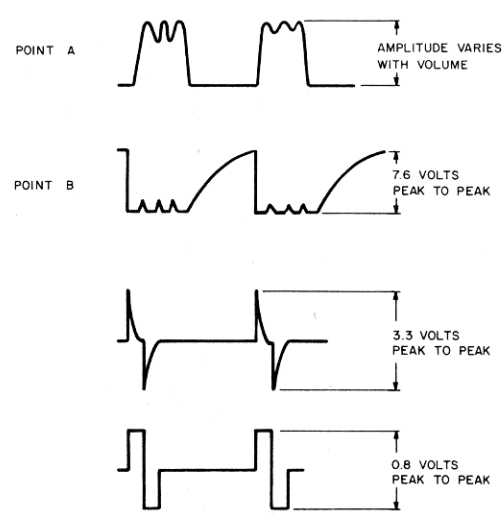


Fig. 2. Oscilloscope waveforms found in the Data Dubber. The number of glitches on A and B are determined by waveform distortion on the tape.

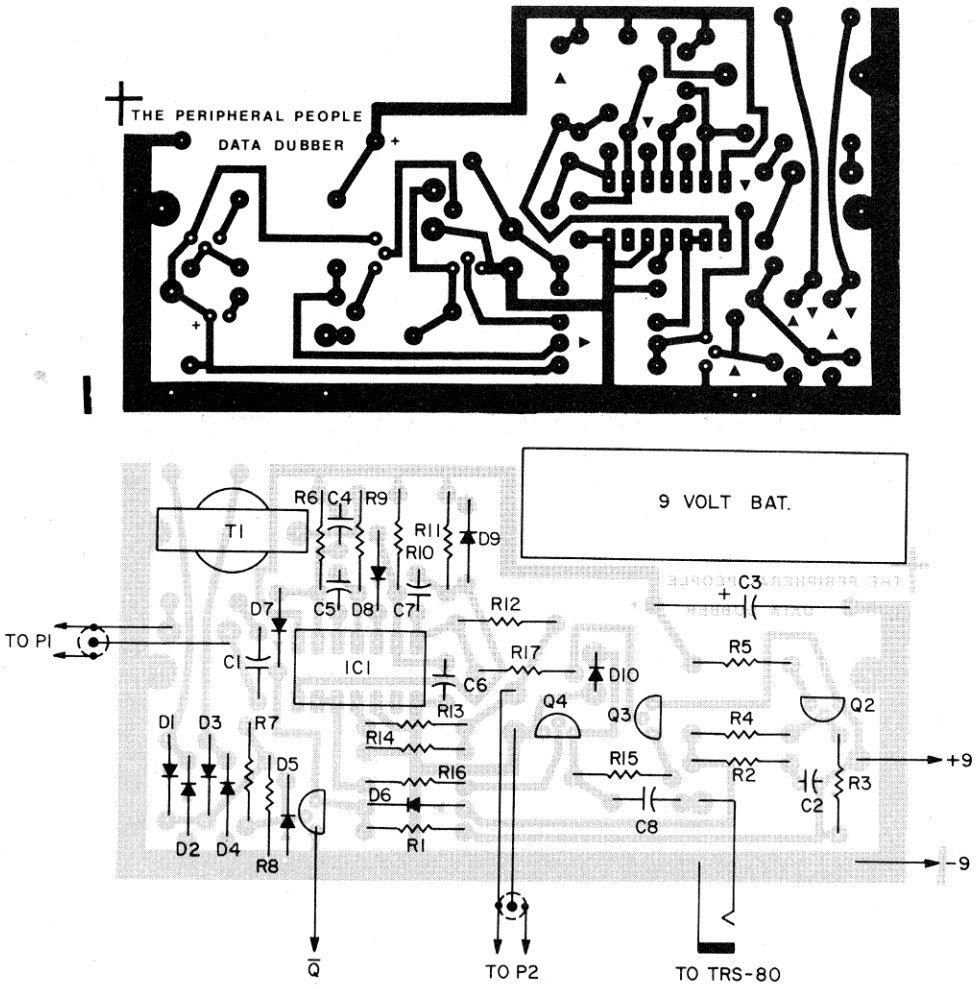


Fig. 3. Artwork and component layout for the circuit board. Note—this drawing is actual size.

If you have an oscilloscope, you should check the waveforms at points A, B, C and D in Fig. 1. The waveform at point B is particularly critical. If the glitches along the bottom of the waveform stick up past the center, they can cause double triggering of the output signal. Note that as you run the level higher, the glitches are more suppressed.

Using the Data Dubber

While the Data Dubber can be a godsend to the TRS-80 owner, it cannot resurrect life. If you have a tape that has never been loadable under any circumstances, there is a good chance you won't be able to load it with the Dubber either. Very likely, pulses are missing, and it will never load.

However, if one of your tapes loads after much fiddling with the volume and tone, it will likely load the first time with the Data Dubber. Normally, you will set the volume somewhere between 4 and 6 for a good-quality tape. Particularly troublesome tapes may require a volume setting between 6 and 10 to get a load.

Most of the time, advancing the volume until a good load is obtained should prove satisfactory. However, some tapes may have garbage on the baseline that is increased when the volume is advanced. In this case, a compromise setting of the volume might be required.

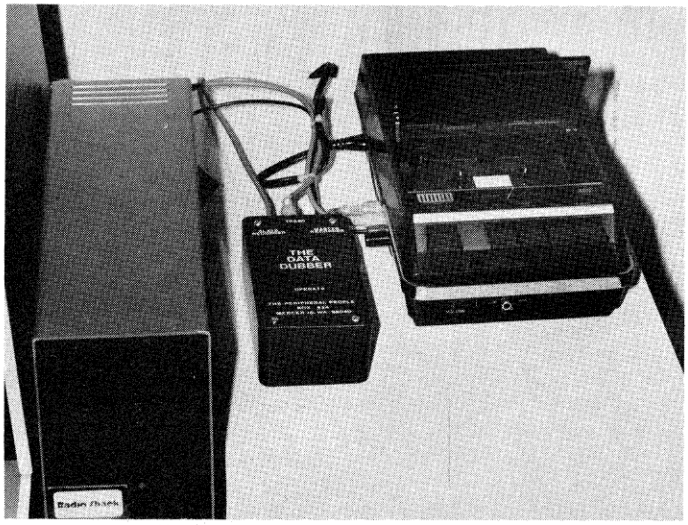
In testing more than 100 tapes, we have never found one that would load directly into the TRS-80, but not through the Data Dubber. However, since the forms of distortion added by duplication can be varied and fiendish, it is conceivable that this situation could occur. We would like to obtain such a tape for testing and design improvement of the Data Dubber.

The LED can be used as a level indicator. A good data stream is indicated by a steady light on the leader and a barely perceptible flicker when data is coming in. As you advance the volume from zero, the light will suddenly come on. Keep advancing the level two numbers on the volume knob. The optimum point for the volume seems to be slightly above the point where illumination occurs.

Once you have successfully loaded a troublesome tape, you can copy it one of two ways. Probably the easiest route is to simply CSAVE the program on a new cassette. However, if it is a machine-language program, connect a second recorder (slave) to the Data Dubber and use the same volume level (master) that produced the successful load. The duplicated tape from the slave recorder should load perfectly with or without the Data Dubber.

For those of you who despise building electronic gadgets, the wired and tested Data Dubber is available from The Peripheral People, Box 524, Mercer Island WA 98040, for $49.95 postage paid. For those of you who despise buying wired and tested electronic gadgets, the circuit board or a kit of parts is also available for $4.95 and $24.95, respectively.

Readers of *80 Microcomputing* are welcome to construct the Data Dubber for their own personal use. However, any duplication of the Data Dubber or conceptual variation for commercial sale is expressly prohibited by law. Not only is it uncouth, legally uncool and morally reprehensible, but our ill-tempered attorney is gonna getcha! ■



The Data Dubber can be connected in line between the cassette recorder and the TRS-80 CLOAD input.

CRYPTEXT

THE DAWN OF SECURITY

CRYPTEXT

DATA PRIVACY FOR TRS-80 !

FEATURES: KEY LENGTH=80 bits (more than DES!)
DATA RATE: Exceeds 15,000 chars./sec.
NEW! Pocket-size encryption device
Plugs into TRS-80 ready to use

IS YOUR DATA SECURE ?

Write for brochure & price list. **APPLE II Interface available**
Users Manual $ 4.95

✓51

VISA master charge THE INTERBANK CARD

CRYPTEXT CORPORATION
P. O. Box 425
Northgate Sta.
Seattle, WA 98125
(206) 364-8585

DATA BASE MANAGER IDM-IV $69
You can use it to maintain a data base & produce reports without any programming. Define file parameters & report formats on-line. Features key random access, fast multi-key sort, field arith., label generator, audit log. MOD-II version with more than 50 enhancements $199.

ACCOUNTS RECEIVABLE ACCT-III $69
One or more drives. Order entry calculates sales tax, shipping, amount for multiple items. Credit checking, aging, sales analysis, invoices, statements and reports. As opposed to most other A/R, ours can be used by doctors, store managers, etc. MOD-II version $149.

WORD PROCESSOR 16K $39. 32K $49. MOD-II $49.
First word processor specifically designed for the TRS-80 that uses disk storage for text. Written in BASIC. No special hardware and text limit. Use for letters, manuals & reports. 32K version features upper/lower case without hardware change and multiple input text files.

MAILING LIST advanced MAIL-V $59.
Fast sort by any field. Multiple labels and reports. 4-digit selection code, new zip code ext., screen input, live keyboard, powerful report writer. MOD-II $99.

INVENTORY INV-V $99.
9-digit alphanumeric key for fast key random access. Reports include order info, performance summary, etc. Calculate E.O.Q. Powerful report writer. MOD-II $149.

All programs are on-line, interactive, random access, virtually bug free, documented and delivered on disks. MOD-I requires 32K, DOS. We challenge all software vendors to offer low cost manuals so you can compare and avoid those high-priced undocumented, 'on-memory' programs. Send $5 for a MOD-I manual and $10 for MOD-II.

MOD-II programs are extensively modified, guaranteed to run with 1 year newsletter & updates. 10% off for ordering more than 1 MOD-II programs.

MICRO ARCHITECT ✓54
96 Dothan St., Arlington, MA 02174

UP TO 25% OFF

YOUR OWN TRS-80 SYSTEM AT TREMENDOUS SAVINGS

❶TRS-80 Complete System
Includes: CPU/Keyboard, Power Supply, Video Monitor, Cassette Recorder, Manual, and Game Cassette.
❷Line Printer
❸Mini Disk System
❹C-10 Cassettes
❺Verbatim Diskettes

DISK DRIVES IN STOCK!

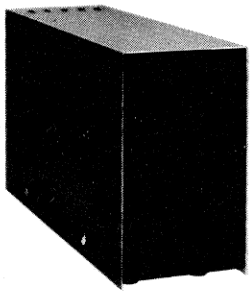
ITEM		REG. PRICE	OUR PRICE
Level II—4k		$619.00	$575.70
Level II—16k		$849.00	$789.60
Expansion Interface		$299.00	$278.00
Mini Disk Drive		$ 495.00	$ 385.00
Centronics 779 Printer		$1599.00	$1175.00
Centronics 101 Printer		$1595.00	$1400.00
Anadex DP-8000 Printer		$1295.00	$ 995.00
Memory Kit-(16K) FREE INSTALLATION		$ 149.00	$ 98.00
Verbatim Diskettes ea.		$ 5.95	$ 4.95
	3	$ 17.89	$ 12.00
	10	$ 59.00	$ 37.50
C-10 Cassettes	5	$ 4.95	$ 4.50
	25	$ 24.75	$ 18.75
Paper (9½" x 11" fanfold, 3500 sheets)		$ 35.00	$ 29.95

SALE
Centronics 730 $850.00
Same as Line Printer II

TRS—80 MODEL II $3626.00
- 64K RAM
- ½ MEG DISK

VR Data's 1st Drive $899.00
ADDITIONAL DISK DRIVE (1ST) $1069.50
ADDITIONAL DISK DRIVE (2ND + 3RD) $ 540.00

MINI DISK DRIVES NOW $385
Over $100 less than Radio Shack's !



There are new developments every day—write or call for the latest information.

VR Data corp
777 Henderson Boulevard N-6
Folcroft Industrial Park
Folcroft PA 19032
(215) 461-5300 ✓13

TOLL FREE
1-(800) 345-8102 *Orders only!

FOREIGN and DOMESTIC DISTRIBUTORSHIPS AVAILABLE . . .

A look at some interesting (and not so interesting) programs.

Software Review II

Rod Hallen
Road Runner Ranch
PO Box 73
Tombstone AZ 85638

It must be fairly obvious by now that the Radio Shack TRS-80 is one of the more popular, if not the most popular, personal computers. It is hard to overlook the large number of advertisements offering hardware, software and newsletters pertaining to the TRS-80.

This is not a review of the TRS-80. I have found it to be an extremely useful computer and feel that it is one of the best buys on the market. What I want to do here is to look at some of the software that is being offered for it.

I have assembled here an evaluation of all of the TRS-80 applications software that I have at the present time. This review will be from the user's point of view rather than the programmer's.

The Software

Name: AUTOK and QEDIT
Vendor: Discovery Bay Software Co., PO Box 464, Townsend WA 98368
Price: $15 for both
Purpose: AUTOK provides an automatic keyboard repeat feature; QEDIT is a screen editor for BASIC program statements
Documentation: Very good
Loading: OK—Level 7
Implementation: AUTOK will cause any key that is held down for more than half a second to repeat itself about eight times per second until it is released. This is a useful feature that replaces the repeat key, which is not on the keyboard.

QEDIT is the TRS-80 BASIC text editor that I have been looking for. I grew up (computerwise) with the SOL, which allows you to move the cursor to any location on the screen and add, change or delete characters as required. The remaining text opens or closes up as necessary, without code letters. Level II BASIC and other BASICs that were written by Microsoft are hard-copy-oriented in the sense that they were designed to be used with a printing terminal instead of a video screen.

To edit a line, bring it to the screen with LIST, hit CLEAR to get into the QEDIT mode and use the four arrow keys to position the cursor where you want to make changes. To insert you just start typing; to delete you hit the CLEAR for each letter to be eliminated, and then ENTER to cause the edited line to be entered into the program. By using the BREAK key, you can abort QEDIT after making changes without the changes taking effect. I especially like its ability to edit line numbers. Now you can move lines around within a program without having to retype them.

You cannot use QEDIT on a line that is more than 64 characters long or while the normal Level II text editor is in use, but these are minor disadvantages.
Suitability: In Part 1, I indicated a need for more sophisticated applications software. These are two very good examples of what I had in mind.

Name: Moving Signboard
Vendor: Circle Enterprises, PO Box 546, Groton CT 06340
Price: $9.95
Purpose: To display messages
Documentation: Very good
Loading: OK—Level 6
Implementation: Signboard is a machine-language program that stores a message and then displays it in a horizontal line on the screen. The text slowly moves from right to left like the electric signboard in Times Square. A message of up to 1024 letters is first typed on the screen and then stored in memory. Next a second message is typed on the screen leaving at least one horizontal line vacant. When Signboard is activated, the second message is displayed constantly on the screen while the first crawls across any chosen line.
Suitability: While this program might have some novelty appeal to many personal computer owners, I can see much more potential as a window display for a computer store or other business. It is an attention-getter!

Other software available from the same vendor: Day of the

Week, File Handling, Loan Payment, Prime Numbers, Amway Distributor systems and many more.

Name: MON-2
Vendor: Hubert S. Howe, 14 Lexington Rd, New City NY 10956
Price: $20
Purpose: An assembly-language monitor with built-in Z-80 disassembler
Documentation: Very good
Loading: OK—Level 7

Implementation: MON-2 satisfies most of the needs of the assembly-language programmer. It works well and is easy to understand and use. A block of memory of any size can be displayed in either ASCII or hex, locations can be changed, and programs executed. FIND BYTE and FIND WORD features allow memory searches for a match. The disassembler will dump to the screen, to a hard-copy device or to cassette tape. The resulting tape is then compatible with the Radio Shack Editor/Assembler. Zilog Z-80 mnemonics are used.

Suitability: This is a very effective disassembler. I am not using the standard TRS-80 hard-copy device, so I had to modify the printer output subroutine somewhat. This was easy to do by letting the disassembler disassemble and modify itself! If it included an assembler that wrote object tapes that could be read with the SYSTEM command, it would be perfect.

Other software available from the same vendor: Life, Level I BASIC, Disk Editor/Assembler and more.

Name: OBJREL
Vendor: Hubert S. Howe
Price:$10
Purpose: Object Code Relocator
Documentation: Very good
Loading: OK—Level 6 1/2
Implementation: This relatively simple program is designed to move a block of object code from one memory location to another. It will change memory references within a program as long as they are easily discernible as addresses and fall within the boundaries of the program being moved. Tables of addresses are not changed.

Suitability: This is another useful program for the assembly-language programmer. It would be even better if it were combined with MON-2 above.

Name: Step-By-Step
Vendor: Program Design, Inc., 11 Idar Court, Greenwich CT 06830
Price: $29.95
Purpose: Teaches how to program a TRS-80 using BASIC
Documentation: Outstanding
Loading: OK—Level 6, not critical

Implementation: This is a case of a BASIC program that teaches BASIC programming. It starts out with the assumption that the student only knows how to turn the TRS-80 on. Three cassette tapes are mounted in the cover of a loose-leaf notebook that also contains supplementary information frames. The course is divided into ten two-part lessons. From a simple PRINT "HI" through arrays and graphics to complex programs, all of the Level II commands and statements are exercised.

The instruction method consists of explanation, example, trial and testing. Commands and statements are presented and explained, examples are shown both on the screen and in the notebook, and then the student is presented with some problems to solve using the BASIC elements under discussion. If an incorrect answer is given, two more tries are allowed, and then the correct answer is displayed. Each lesson ends with a test that is administered and scored by the computer. The results are then entered into the student's progress chart. More comprehensive examinations are given at the end of Lesson 5 and at the end of the course.

Suitability: This is the kind of educational programming that personal computing needs more of. The student (my teenage son) learned much more quickly than I could have taught him, and at his own pace. However, this course isn't just for youngsters but for anyone who wants to be able to program effectively using the BASIC language. In a household where there isn't anyone to do the teaching, this course would be especially useful. I'd like to see a similar course for assembly-language programming.

Other software available from the same vendor: IQ Builders (four different kinds), Memory Builder and Story Builder.

Name: ESP-1
Vendor: Small System Software, PO Box 483, Newbury Park CA 91320
Price: $29.95
Purpose: Editor, assembler, monitor and debugger
Documentation: excellent
Loading: OK—Level 7

Implementation: ESP-1 is an old program that has been rewritten for the TRS-80. That doesn't mean that it isn't a good one. There seems to be quite a bit of interest in assembly-language programming the TRS-80. Source files are generated with ESP-1 using the editor, and each line has a line number just like a BASIC program. Lines can be inserted or deleted as required. The assembler portion of ESP-1 is a full-blooded assembler with all of the standard features. The system monitor and debugger has over 20 different commands to control your TRS-80. These include cassette read and write, memory display and edit and breakpoint features.

Suitability: One drawback to ESP-1 is that the assembler recognizes Intel 8080 mnemonics instead of Zilog Z-80 mnemonics. This is great for the 8080 programmer moving already written programs to the TRS-80, but it does restrict the person who wants to learn to use Z-80 code.

Other software available from the same vendor: Monitor/disassembler, Monitor, Electric Pencil Word Processor, Disk Conversion and more.

Name: SORT-II
Vendor: Northeast Microwave, PO Box 6153, Syracuse NY 13217
Price: $19.95
Purpose: Provide sophisticated sorting capability
Documentation: Excellent
Loading: OK—Level 7

Implementation: SORT-II is not just a simple sorter. It will sort alphanumeric as well as numeric data. SORT-II looks at data as a series of multi-field records. It will handle and sort records containing up to 20 fields and can sort key on any five of them. Input can be from the keyboard, cassette tape or from a merged combination of the two. Output can go to the screen, to tape or both. A bypass feature allows writing directly from the keyboard to tape without sorting.

On entry certain specifications—input, output, record size and sort keys and tape header skip—are set by default, and you are given the chance to change them. After any modifications of the specs, they are displayed and you are given another chance to correct them. Information is also given on how to change the default values so that SORT-II will automatically come up in the configuration that you normally use whenever it is loaded. The same goes for

Games are all right but I bought my computer to help me in my work as well as to entertain me.

special input/output routines such as printer drivers.

Suitability: At first I thought that I didn't really have any need for a sorting program; I didn't seem to be doing much sorting. However, working with SORT-II for a while revealed some interesting possibilities. For instance, this is great for maintaining a mailing list that can be sorted (or searched) by name, phone number, ZIP code or whatever. How about a computerized list of your phonograph record library that could be sorted by name, composer, type of music, performer or location? How about personal records, a stamp collection, the contents of your pantry or anything else that you want to keep track of and update regularly? SORT-II will handle all of these.

Other software from the same vendor: SORT-I, Blackjack Simulator, Discsort and more.

Name: Android NIM
Vendor: 80-NW Publishing Co., PO Box 7112, Tacoma WA 98407
Price: $8
Purpose: Game
Documentation: Self-documenting
Loading: OK—Level 7

Implementation: This is an animated version of the game NIM. In the usual game the player and the computer take turns removing objects from the three piles. The one who takes the last object loses. In this version the piles of objects are replaced with rows of Androids (robots).

Suitability: I had not intended to include games in these reviews but I have made an exception here because of this program's very imaginative use of graphics. The Androids turn their heads, change the expressions on their faces and move their arms. They even shake their heads yes or no to indicate a valid or invalid move. This program has value just for its use of graphics.

Conclusions

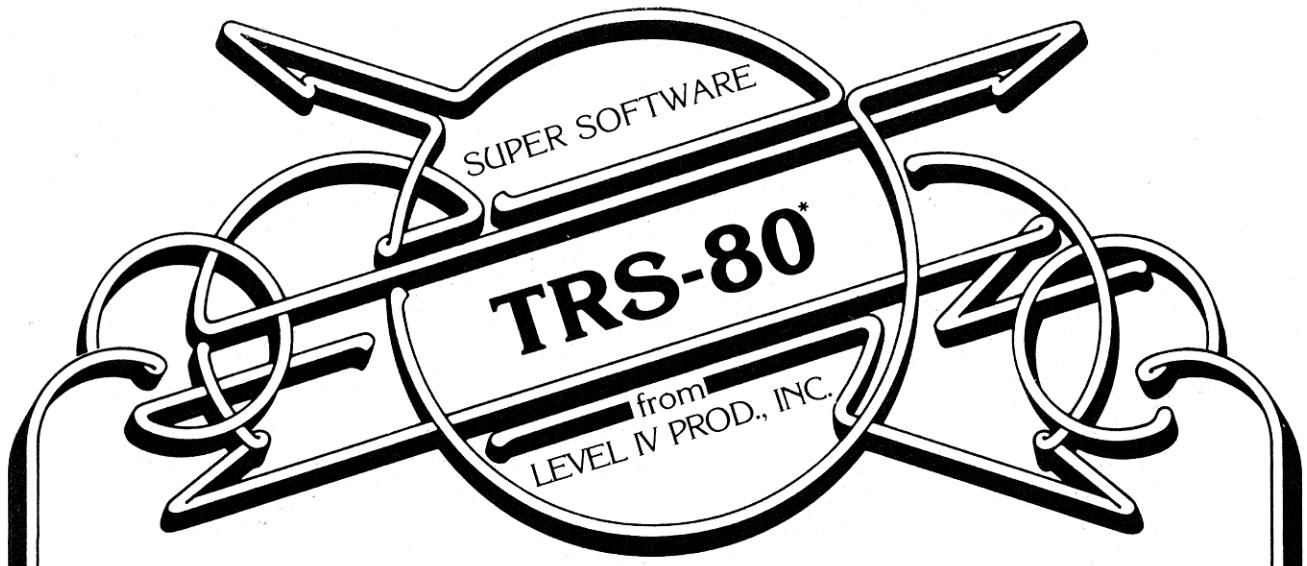
There are many useful, well-written programs in this group. This is the kind of software that I feel is needed and the kind that will sell. Games are all right but I bought my computer to help me in my work as well as to entertain me.

I have still not indicated whether I think the prices of the individual programs described here are too high for the value received or not. Value will depend on the user's interest and needs. I do feel that it is worthwhile to spend $10, $20 or $30 for a program that I can use if it would take me a couple of days or more to write something similar myself. After all, my time has value, too! I think that all of the above programs fall into that category. ■

IF YOU CAN READ, YOU CAN WRITE !

80 is mainly written by people like you, not an elite group of intellectuals. A lot of the readers think that it is difficult to write, but it is remarkably easy if you sit down and try. Other budding authors think that rejection slips are the rule, at 80 we accept about 70% of all submissions. There is a lot to be said for creative writing, including FAME & a small FORTUNE. There is even a helpful leaflet, "How to write for 80," available on request. So why not drop us a line and ask if we would like to see your article idea written up, it can't hurt to ask.

Send all correspondence to Jim Perry, Managing Editor, 80 Microcomputing, 80 Magazine Street, Peterborough, NH 03458.



Cylon - L IV	$7.50	Payroll Disk - TSE	$59.95	Sargon - Hayd.	19.95
Animated Hangman - L IV	7.50	Checkers Disk - TSE	24.95	Sargon II - Hayd.	29.95
Space Battle CS - L IV	14.95	Personal Finance - TSE	9.95	Poker Pete - Qual.	9.95
Level IV Graphics CS- L IV	14.95	GSF - RACT	24.95	Fastgammon - Qual.	19.95
Typing I - L IV	15.00	Remodel & Proload - RACT	34.95	Fastgammon Disk - Qual.	19.95
Typing II - L IV	15.00	Remodel - RACT	24.95	Advent Cass - Adve.	14.95
Space Battle Disk - L IV	19.95	Dosort - RACT	34.95	Advent 2 Cass - Adve.	14.95
Typing I & II - L IV	22.50	Comproc - RACT	11.97	Advent 3 Cass - Adve.	14.95
Warfare 1 - TSE	7.95	Infinite Basic - RACT	49.95	Advent 4 Cass - Adve.	14.95
Final Approach - L IV	10.00	Infinite Basic - RACT	29.95	Advent 5 Cass - Adve.	14.95
Cribbage - TSE	7.95	Copy System - RACT	14.95	Advent 6 Cass - Adve.	14.95
Tycoon - TSE	7.95	Autok/Qedit - DISC	14.95	Advent 7 Cass - Adve.	14.95
Concentration - TSE	7.95	Autok/Qedit Sk - Discovery Bay	19.95	Advent 8 Cass - Adve.	14.95
Amazin Maze - TSE	7.95	Morse Code - Discovery Bay	14.95	Advent 10 Cass - Adve.	5.95
Time Bomb - TSE	4.95	Win 21 - Discovery Bay	19.00	Advent 1&2 Disk - Adve.	24.95
9 Games/Pre School - TSE	9.95	Gomoco - Discovery Bay	15.00	Advent 3&4 Disk - Adve.	24.95
Slalom - TSE	7.95	Level 1 - Apparat	15.00	Advent 5,6&7 Disk - Adve.	39.95
Ten Pin - TSE	7.95	New Dos - Apparat	49.00	Advent 8 Disk - Adve.	19.95
Endzone- TSE	7.95	New Dos + - Apparat	99.00	Bowlstats - RLS	29.95
Star Trek 3.4 - TSE	14.95	KVP - L. Micklus	29.95	Forth Disk - Mill.	64.95
Adventure Mission IM - TSE	14.95	KVP-232 - L. Micklus	29.95	Primmer - Mill.	15.00
Dog Star - TSE	9.95	Library 100 - TBS	49.95	Space War - Mill.	9.95
Safari - TSE	7.95	Electric Pencil Cass. - Shay	99.00	Starfleet Orion - Asim.	19.95
Treasure Hunt - TSE	7.95	Electric Pencil Disk - Shay	150.00	Invasion Orion - Asim.	19.95
Round the Horn - TSE	9.95	Snake Eggs/SND - 80US	14.95	Temple of Apsh - Asim.	24.95
Pork Barrel - TSE	9.95	Android Nim/SND - 80US	14.95	Misosys Tape U - Acro.	14.95
Kamikaze - TSE	7.95	Life Two/SND - 80US	14.95	Misosys Disass - Acro.	19.95
ST80D - TSE	79.00	Bee Wary - 80US	14.95	System Savers - Acro.	14.95
Advance Personal Finance - TSE	24.95	Owl Tree W/S - 80US	9.95	Opera Theater - Acro.	9.95
Atlantic Balloon - TSE	9.95	Great Race W/S - 80US	9.95	Disk Utility - Acro.	19.95
Tarot Cards - TSE	9.95	Lying Chimps - 80US	9.95	Video Pager - Acro.	9.95
Ham Radio - TSE	14.95	Concentration W/S - 80US	9.95	Tshort - Web.	9.95
Breakaway - TSE	6.95	Scrabble W/S - 80US	9.95	Wordo - Micr.	14.95
X-Wing Fighter - TSE	9.95	Level III Basic - Micr.	49.95	Stock Exchange - Micr.	15.95
Simple Simon - TSE	4.95	Micropoly - D.P.	7.50	Whereami - Micr.	10.95
Sinkum - TSE	4.95	Airaid - SSS	SPECIAL 10.00	Othello - L IV	9.95
Educational Assistant - TSE	9.95	Barricade - SSS	SPECIAL 10.00	Invaders w/sound - L IV	14.95
Home Financial - TSE	9.95	RSM-2 - SSS	26.95	Creature Tic-Tac-Toe	7.50
Math Drill - TSE	4.95	RSM-2D - SSS	29.95	Creature Tic-Tac-Toe w/ sound	9.95
Object System - TSE	9.95	DCV-1 - SSS	9.95	XREF Cass - L. Micklus	19.95
Text 80 Disk - TSE	59.95	Basic P1 - SSS	14.95	Renumber Cass - L. Micklus	7.95

LEVEL IV PRODUCTS, INC. ✓14

32238 Schoolcraft • Livonia, MI 48154 • 313-525-6200 outside Michigan 1-800-521-3305

10% Off All Software Listed

If you don't see what you want call, find out why! We carry nothing but the BEST. We also sell TRS-80* 10% off list price.

We ship C.O.D. - Add $1.50 for shipping and handling **Send 25¢ for Catalog**

* Product of Tandy Corporation

Get rid of the fuzzy edges on the characters and make your monitor electrically safe.

Video Tune-up

David F. Miller
7462 Lawler Ave.
Niles IL 60648

The video display used with the TRS-80 microcomputer (as it comes from Radio Shack) is a modified RCA KCS-201K monochrome TV chassis. The tuner, as well as the rf and if stages, has been removed, and the video stages have undergone substantial modification to improve the set's bandwidth and to enable the transformerless chassis to be safely interfaced to the computer (via an optoisolator).

The overall performance of this scheme is adequate for the average purchaser of the system, but improvements are possible for the knowledgeable TRS-80 hobbyist. These improvements require cutting into the existing circuitry, and even a bit of experimentation, but the rewards are well worth it for the technically able.

Modifications

The first modification has to do with the installation of a 120 V ac to 120 V ac isolation transformer in the input power circuitry of the video display. The addition of this component makes the display considerably safer to work in and around, and also makes it more adaptable to standard troubleshooting techniques using grounded test equipment (if and when that need arises).

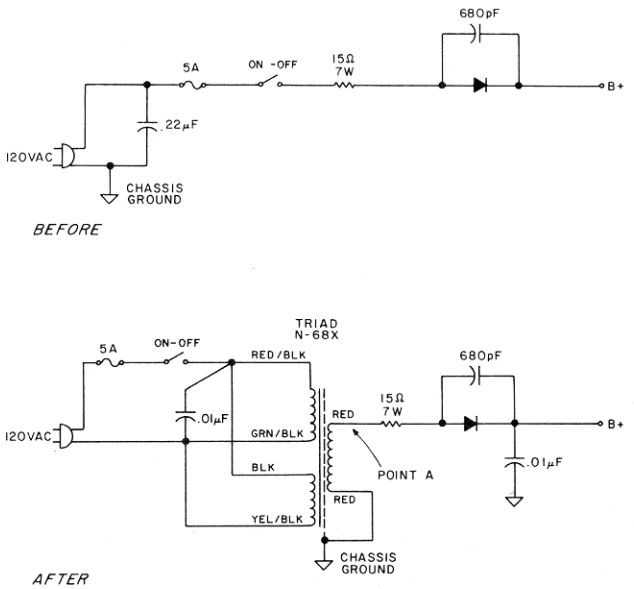


Fig. 1. Triad N-68X transformer available from Jameco Electronics, 1020 Howard Ave., San Carlos CA 94070.

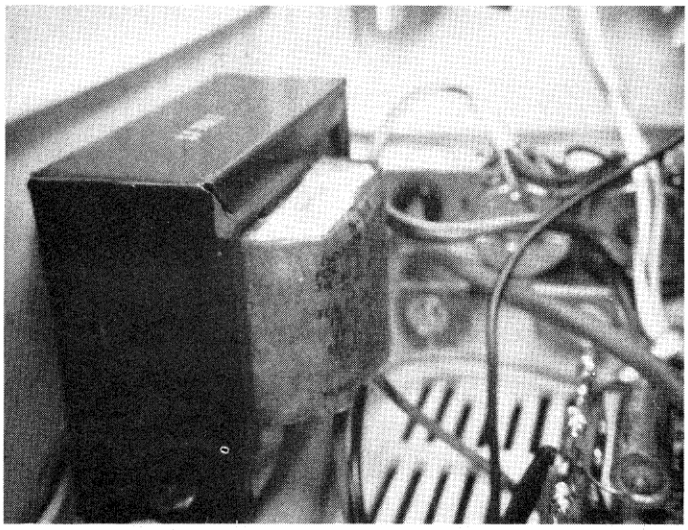


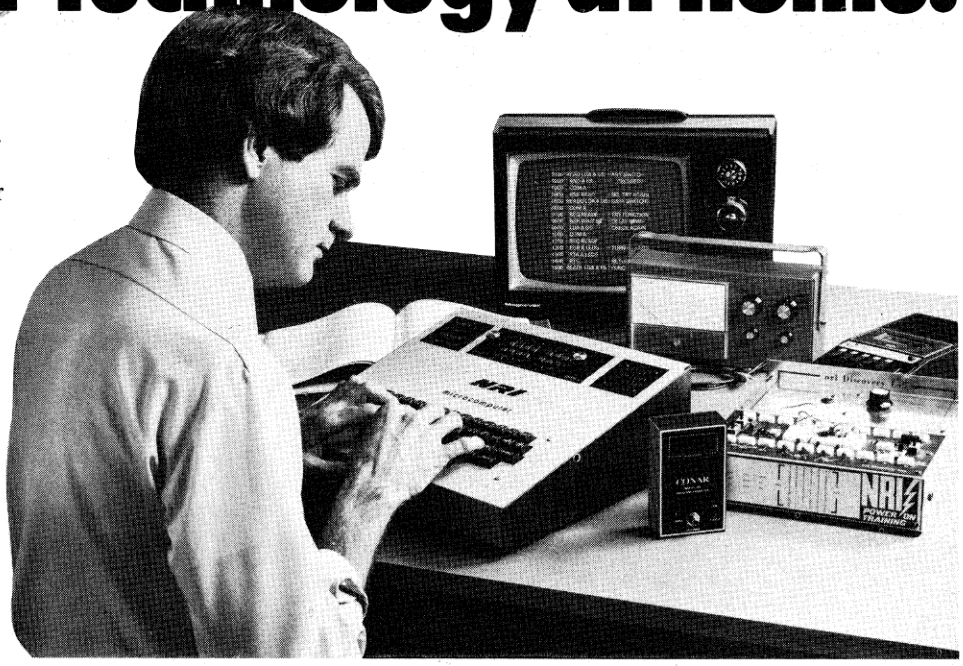
Photo 1. The physical placement of the Triad N-68X line isolation transformer within the video display cabinet of the TRS-80.

Fig. 1 schematically shows the video display input power circuitry before and after the transformer installation. The isolation transformer itself is physically located in the open area to the left of the chassis and picture tube (when viewed from the rear). Installed in this location, the transformer's magnetic field does not affect the linearity of the display or cause any other undesirable side effects.

If you decide not to install the isolation transformer, be extremely careful not to come in contact with the display chassis anytime it is plugged into the 120 V ac line. This is a "hot chassis" set, which means that you have a 50-50 chance of having full line voltage on the metal chassis anytime the unit is connected to the wall outlet!

The second modification will require perhaps a bit more effort and a little experimentation on your part. The focus electrode of the 12YBNP4 picture tube (pin 7) is normally tied to chassis common in production-run TV sets, and this procedure usually results in reasonably good overall picture focus (depending on the parameter of the individual picture tube). There is generally room for some improvement, however.

Build your own microcomputer as you learn computer technology at home.



New from NRI! The First Interdisciplinary Home Study Course Ever Offered

As the microprocessor revolutionizes the computer world and microcomputers appear almost everywhere, NRI brings you a new, convenient, and effective way to keep up with this expanding technology. It's NRI's courses in Microcomputers and Microprocessors, created and designed exclusively for learning at home in your spare time.

Designed for the New Breed of Computer Technician

It's no longer enough to be just a programmer or technician. With microcomputers moving into the fabric of our lives as low-cost, easily available tools for business and home, both the programmer and technician must become total professionals. With practical knowledge of hardware, the programmer can design simpler, more effective programs. And with advanced programming skills, the technician can test and debug systems quickly and easily. The NRI course gives you simultaneous training in both skills...makes you one of this rare new breed.

Build Microcomputer, Test Instruments

NRI goes far beyond book learning to give you practical, "hands-on" experience. As you learn, you actually assemble NRI's designed-for-learning microcomputer. It performs like the finest of its kind, and features both assembly and basic language capabilities.

Every assembly step's a learning step. Using the NRI Discovery Lab® plus the NRI transistorized volt-ohm meter and CMOS digital frequency counter you also build, you perform meaningful experiments throughout your course...trace circuitry, interface components, introduce and correct problems, design your own programs, and more.

The Proven Way to Learn at Home

You don't have to worry with travel, classes, or time lost from work when you learn the NRI way. As they have for more than 60 years of teaching technical subjects, NRI brings the material to you. You study in your spare time, at your convenience, using "bite-size" lessons that program material into logical segments for easier assimilation. You perform experiments and build equipment using kits we supply. And your personal NRI instructor is always available for consultation should you have questions or problems. Over a million students have already shown the effectiveness of NRI training.

Choice of Courses

Several courses are available, depending upon your needs and background. NRI's Master Course in Microcomputers and Microprocessors starts with the fundamentals, explores basic electronics and digital theory, the total computer world, and the microcomputer. The Advanced Course, for students already versed in digital electronics, concentrates on software and the world of the microprocessor and microcomputer. In both courses, you build all instruments and your own computer.

Send for Free Catalog... No Salesman Will Call

Get the details on these exciting new courses in NRI's free, 100-page catalog. Shows all kits and equipment, lesson outlines, and full information, including facts on other electronics courses. Mail the coupon today and we'll rush your catalog. No salesman will ever call. Keep up with the latest technology as you learn on your own computer. If coupon has been removed, write to NRI Schools, Computer Department, 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
- ☐ Amateur Radio • Basic and Advanced
- ☐ 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 **179-020**

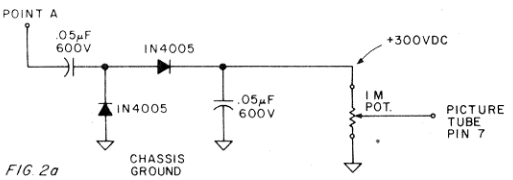
The focus electrode can have anywhere from −300 V dc to +300 V dc applied to it and still remain within manufacturer's specifications. The KCS-201K chassis does not provide for selectable focus voltage tap-off points (as some sets do), so you must provide for them yourself if you wish to see if any improvement in this area is possible.

It's not as bad as it sounds; in fact, it's harder to explain it than it is to actually do, so at this point you might take a look at Fig. 2 for an idea of what must be done.

The two circuits shown in Fig. 2 are positive (Fig. 2a) and negative (Fig. 2b) voltage doubler circuits. These two circuits will output approximately 300 V dc with 120 V ac input. The reason for this is that the capacitors charge to the peak of the 120 V ac root mean square (RMS) line voltage, which is actually 150 V ac peak. The doubler then multiplies this value by two and outputs the 300 V dc figure.

The current requirements for the focus electrode are very low, so the capacitor values and their physical sizes can be kept relatively small.

Whether your particular video display will require a positive or a negative potential, and at what exact value, will have to be determined experimentally by you. The purpose of this article is to help guide you to that point. In the case of my own video display, a value of +300 V dc yields optimum focus.



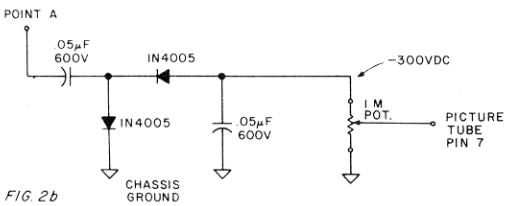


Fig. 2. Circuit modification.

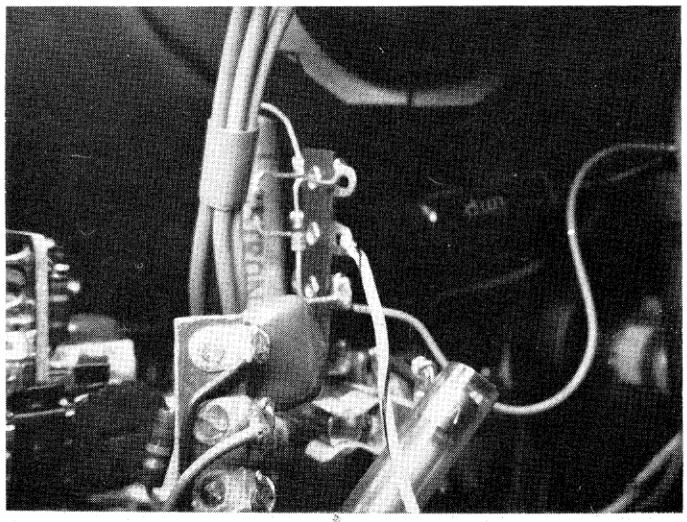


Photo 2. The terminal strip used to mount the voltage doubler components. It is mounted on the existing power terminal strip just below the picture tube neck in the TRS-80 video display.

To determine the requirements of your own display, construct the circuits in Fig. 2 one at a time (in breadboard fashion) and vary the 1 megOhm potentiometer while observing the effect that it has on the sharpness of the characters on the video display. You should, of course, have your computer connected to the display in the normal fashion and look for the best focus on the individual scan lines. Take your time at this and avoid contacting any exposed high potentials.

When you have determined the correct polarity and value of focus voltage needed, simply remove the potentiometer and measure the final values on either side of the slider with an Ohmmeter, substituting fixed resistors of the same value in the final circuit.

Now simply install the proper doubler on a separate terminal strip located at a convenient point on the chassis and get set to enjoy a sharper picture. Your eyes will be forever grateful! ■

✓ Reader Service—see page 131

Why pay for the Radio Shack disk drives when you can add your own—cheaper.

Percom Disk Drives

David Buffington &
Cindy Wagner
c/o Lerch's R.V. Center
South Main & Charles Streets
Lewistown PA 17044

On one hand, our TRS-80 has served us well at our travel trailer sales center, handling two major functions: Calculating price quotes for customers and keeping our daily cash and sales records.

The TRS-80 has carried out these tasks day after day in a harsh environment with sawdust in the air and a five horsepower air compressor on the same line; seven days a week often twenty-four hours a day. The only hardware failure that we've experienced was a video display failure. Yet we were back in operation the next day.

```
100 DEF FIELD #1, 4 AS BF$, 4 AS PY$, 4 AS CH$, 4 AS NB$
110 INPUT"OLD BALANCE"; OB
120 LSET BF$=MKS$(OB)
130 INPUT"PAYMENTS"; P
140 LSET PY$=MKS$(P)
150 INPUT"CHARGES"; C
160 LSET CH$=MKS$(C)
170 NB=OB-P+C
180 LSET NB$=MKS$(NB)
190 PUT#1, DS          DS-drive and sector
```

Fig. 1. To store a balance forward. Payments, charges and new balance using a field buffer:

For a "hobby" computer, that is an impressive performance.

A Weak Link

On the other hand, one incredibly weak link in the TRS-80 system is the tape recorder. First, even at the faster Level II baud rate, data transfer to and from cassettes was slow. The daily books program took over five minutes to load.

Also, the slow rate inhibited the general ledger program which handles large amounts of data. Such tape data loads could run thirty minutes or more making the program impractical.

More severe than the speed problem was its reliability. Long program loads were often filled with errors which caused the program to crash midway through its operation. Furthermore, even short data loads had to be repeated again and again to get the proper data into the computer. Frequent head cleaning and using the best quality tapes we could buy helped little.

The night we loaded the daily books program six times convinced us that something had to be done.

The speed and reliability of a disk drive seemed to be the answer, but frankly, Radio Shack didn't stand up to the competition.

The Radio Shack drive recorded on thirty-five tracks (10 sectors/track), while most competitors recorded on forty tracks (10 sectors/track). The track-to-track stepping time of the Radio Shack drive, a major factor in determining speed, was double that of some competitors. To compound matters, the Radio Shack drive cost $50 to $150 more than the others, while dealers were quoting four, six, or more months delivery time.

But the clincher was TRSDOS, Radio Shack's disk operating software. After mastering Level I and Level II in a fairly short time, the TRSDOS manual floored us. The entire system seems more complex than necessary to handle our needs.

However, we did decipher the part of the manual that said that a diskette containing TRSDOS must be in one drive at all times. This cuts the storage capacity of a single drive and we could not afford dual drives.

Worse than that were the reports of TRSDOS's lack of reliability. The fact that several firms were selling patches to make TRSDOS work properly disturbed us.

Whose Disk Drive

But then, whose disk drive and whose software do we choose? There were several. We hunted for one that had shown both hardware and software success with other computer systems, a firm able to back up their product. We kept reading ads and articles until we came up with a name.

Percom Data Company's TFD-100 drive offered forty-track operation, 20 millisecond track-to-track stepping time and recorded on both sides of a diskette without punching extra holes. The price was about $75 less than Radio Shack's, depending on the connecting cable ordered and delivery time was set at four to six weeks!

Best of all, Percom offered its own disk software, MicroDOS, which could be loaded into memory leaving the entire disk free for data and program storage.

We ordered one TFD-100 drive ($399), a two drive cable ($30) to allow for future expansion and MicroDOS ($30).

In two weeks we had a copy of MicroDOS. The manual was a loose leaf mimeographed affair, but it seemed clear, something that the TRSDOS manual isn't.

Essentially, MicroDOS is a straightforward system for storing data and programs. There are no disk directories, passwords or clock functions as with TRSDOS. However, it does support the range of commands and statements necessary to get the job done. Furthermore, MicroDOS is completely resident in less than 7K of memory.

MicroDOS is highlighted by the following commands and statements:

LOAD, SAVE & MERGE: These three commands allow you to store (SAVE) programs on diskettes, to retrieve (LOAD) programs from diskettes and to combine (MERGE) several programs and subroutines.

DEF FIELD #N, N1 AS A$, N2 AS B$: This statement sets up a field buffer, the key to disk data storage. MicroDOS allows four buffers with up to 255 characters in each.

LSET or **RSET A$ = B$:** These statements are used to move data into the buffer either by left justifying the data within the field or by right justifying the data.

PUT and **GET:** These statements either read (GET) or store (PUT) data in individual diskette sectors.

MKI$(I), MKS$(S), & MKD$(D): These statements convert integers, single precision and double precision values into strings of two, four and eight bytes respectively. This is necessary since only string data can be placed into a buffer.

CVI(I$), CVS(S$), &CVD(D$): These statements perform the reverse function of the above.

Swift Response

There is much more to MicroDOS, but these are the building blocks of the system. Yet, how do you put them all together? The manual is not complete in this area and being novices we could not figure it out. So a letter full of questions was sent to Percom.

Not long afterwards we received a reply from Percom. It was a handwritten response from James Stutsman, author of MicroDOS. He answered my questions and gave examples of MicroDOS programming. For instance, Fig. 1 shows how to use field buffers to store four data items while Fig. 2 shows how to retrieve them. Armed with this information we converted our programs to disk data storage even before the drive itself arrived.

When the drive did arrive, six weeks after ordering it, we were ready to go. There was some minor assembly required to get the drive and cable together. A few minutes after opening the box we had everything plugged in. It is important to remember that once you have the RS Expansion Interface, which is necessary with any drive, you cannot simply press a button to turn on your TRS-80. First, turn on everything except the keyboard. Then, hold down the BREAK key and press the keyboard on/off button. Failure to do that will result in a screen full of garbage.

Now MicroDOS can be loaded. Insert the diskette into the drive with the long oval cutout going into the drive first. Close the drive door and press the RESET button on the keyboard. MicroDOS will then load itself and, if you are using the MicroDOS system diskette provided by Percom, a four program menu will be displayed. These programs are not part of MicroDOS and should be cleared out of memory in normal use. However, they are provided as examples of what MicroDOS can do.

One is a file management program which shows how complex disk systems like TRSDOS can be emulated by MicroDOS. Another tests the operation of the disk system. The third is the Percom 5¼ Inch Notebook. This program allows you to enter pages of information, to leaf through them and to index the pages in a table of contents.

The notebook comes preloaded with sample pages concerning MicroDOS statements. These pages contain useful information on MicroDOS missing from the manual, including information on special functions such as INSTR, used to pick out a string within a string and DEF FN, used to create user-defined functions. These are not even mentioned in the manual.

However, the remaining program is the most important. The Disk Utilities program allows the user to call up functions like:

FORMAT: This is used to write the necessary control and timing information onto a blank diskette.

BACKUP and **COPY:** These two functions are used to produce copies of all or part of a diskette. Notably these functions did not work with our version of MicroDOS. All that I could get out of them were "Out of memory" errors. The problem is a typing flaw in the program itself. If you run into this problem, edit line 100, changing 300 to 500 in the second statement. (Thanks to Jim Stutsman for that fix.)

FREE: This determines the number of blank sectors on a diskette.

VERIFY: This checks a diskette for defective sectors.

ERASE: With this you can erase all or part of a diskette.

DUMP: This function allows you to display the contents of a single sector. This is very useful in debugging programs.

The bottom line with any system is how it operates. Our converted programs ran with only minor debugging and our disk based general ledger program is well on its way. We have had no hardware problems at all, but we have had many software questions.

Percom and Mr. Stutsman have been happy to answer them all. However, they ask that you write and not call in your questions.

In short, we have been very pleased with Percom, their drive and their software. The only lingering problem is the MicroDOS manual which is clear but incomplete. However, Percom has promised a new manual, the perfect addition to a fine disk system. ■

```
100 GET#1, DS
110 DEF FIELD #1, 4 AS BF$, 4 AS PY$, 4 AS CH$, 4 AS NB$
120 PRINT"BALANCE FORWARD"; CVS(BF$)
130 PRINT"PAYMENTS"; CVS(PY$)
140 PRINT"CHARGES"; CVS(CH$)
150 PRINT"NEW BALANCE"; CVS(NB$)
```

Fig. 2. To read that data back:

Modify your RSM-1 or other monitor.

CLOAD Machine Language

Mark A. Schimelman, M.D.
255 Townhouse
Hershey PA 17033

Machine-language monitors, such as T-BUG (Radio Shack) and RSM-1 (Small System Software), have been created for Radio Shack-based computers to allow machine-language programming. These monitors each possess features that are desirable in a monitor.

I purchased RSM-1 because of the extensive command table and its ease of implementation, but it did have deficiencies. For instance, there is no command to allow display of the microprocessor registers when you're debugging programs. Also, the monitor does not permit the production of machine-language tapes that can be loaded and run directly from BASIC (via CLOAD).

The Routines

Therefore, I embarked on a project to develop routines that (1) display registers in a format that could be included in the monitor and (2) produce tapes that were CLOAD readable. This would let me modify RSM-1 or develop self-loading machine-language programs that did not require monitor support to be loaded or run. The results of my work are shown in Listings 1 and 2.

The register display routine (Listing 1) pushes values from each register onto the stack in two iterations. One iteration is used to feed values to the output routine (48A7 to 48B3), while the second is used to restore the computer's environment to that which existed before the command was given.

A title string is defined and output to delineate each register's contents. The routine calls up a number of RSM-1 subroutines to save memory space. The routine as shown is located at 4880 and uses 108 bytes. I define this routine as command letter P by placing 50 80 48 at location 440D-440F in the RSM command table.

Now the next problem was how to write my modified RSM-1 monitor to tape so that it could be preserved as is for future use. After consulting with Jim Orleff, a member of the TRS-80 Users Group of Onalaska WI who provided the Radio Shack BASIC Cassette ON and CSAVE ON routine locations (Listing 2a), and Tom Nussmeyer, the creator of RSM-1, who provided the CLOAD tape format (Listing 2b), I realized that CLOAD tapes could be easily made if RSM-1 were relocated to a different RAM address.

This is necessary because CLOAD readable tapes must have two features:

(1) The tape leader must have an A5 sync byte plus two start address bytes and two end address bytes after at least 128 00s (NOPs).

(2) The tape as it is fed in must fill location 41FE-41FF with the address of the program's entry point.

This was easy for me since I had already relocated RSM-1 to 3000-37FF and burned it into PROM at the unused addresses.

So, all you have to do is set the starting address of the program at 41FE and load these first two bytes with the entry point address, then the next two bytes with the end address. The routines in Listing 2 will automatically provide the tape leader and sync byte.

An alternate method for producing these tapes that does not require RSM-1 relocation is accomplished by writing the

```
4880  FD E5        PUSH IY
  82  DD E5        PUSH IX
  84  E5           PUSH HL
  85  D5           PUSH DE
  86  C5           PUSH BC
  87  F5           PUSH AF
  88  08           EX AF,AF'
  89  D9           EXX
  8A  E5           PUSH HL'
  8B  D5           PUSH DE'
  8C  C5           PUSH BC'
  8D  F5           PUSH AF'
  8E  FD E5        PUSH IY
  90  DD E5        PUSH IX
4892  E5           PUSH HL'
  93  D5           PUSH DE'
  94  C5           PUSH BC'
  95  F5           PUSH AF'
  96  08           EX AF',AF
  97  D9           EXX
  98  E5           PUSH HL
  99  D5           PUSH DE
  9A  C5           PUSH BC
  9B  F5           PUSH AF
  9C  21 C3 48     LD HL,NN
  9F  CD AB 42     CALL STRNG
48A2  CD B6 42     CALL CRLF
  A5  16 0A        LD D, 09
  A7  E1           POP HL
  A8  CD EB 42     CALL HLOUT
  AB  CD E6 42     CALL SP1
  AE  CD E6 42     CALL SP1
48B1  15           DEC D
  B2  20 F2        JP NZ
  B4  F1           POP AF'
  B5  C1           POP BC'

48B6  D1           POP DE'
  B7  E1           POP HL'
  B8  08           EX AF',AF
  B9  D9           EXX
  BA  F1           POP AF
  BB  C1           POP BC
  BC  D1           POP DE
  BD  E1           POP HL
  BE  DD E1        POP IX
  C0  FD E1        POP IY
48C2  C9           RETURN
  C3  3A           STRNG LEN
  C4  41 46 20 20 20 20
  CA  42 43 20 20 20 20
  D0  44 45 20 20 20 20
  D6  48 4C 20 20 20 20
48DC  41 46 27 20 20 20
  E2  42 43 27 20 20 20
  E8  44 45 27 20 20 20
  EE  48 4C 27 20 20 20
48F4  49 58 20 20 20 20
  FA  49 59
48FC  ZZZ

Subroutine Called

STRNG  42AB  Output a string
HLOUT  42EB  Output HL regist.
SP1    42E6  Output a space
CRLF   42B6  Carriage Return
             Line Feed
```

Listing 1. Register display routine.

program and debugging it at a high memory location, then altering the addresses relative to 4200. Put the 3-byte code A5 FE 41 in front of the program and fill the preceding 256 bytes with 00s.

These operations are easily done with RSM-1 commands, after which the RSM-1 Write command is used to write the tape. Write also produces a leader, sync byte and address format that must be avoided on playback either by not recording until after the RSM leader has passed or by not ENTERing CLOAD until it has passed the tapeheads on playback.

With this method, when CLOAD is complete and the program jumps to 41FE to find the entry point, it finds 41FE to be that entry point and begins program execution at 4200.

I enjoyed solving these problems. I hope these routines and concepts will help other Radio Shack computerists get the most out of their machines. ■

```
5000    CD    E9    0F    CALL Cassette ON
5003    21    FE    41    LD, 41FE (start addr.-this location should be filled with
                           the prog. entry pt.)
5006    11    ( )   ( )   LD, END end address
5009    CD    4B    0F    CALL CSAVE ON
500C    C9                RETURN
```

(a) Tape production routine

00 00 00 00 128+NOP's 00 00 A5 XX YY PROGRAM CODE — — —

A5-SYNC BYTE
XX-PROGRAM START ADDRESS
YY-PROGRAM END ADDRESS
N.B.-Location 41FE must be filled by the tape with the program entry point.

(b)CLOAD tape format

Listing 2.

Want to keep track of donations?
Try this successful Sunday service program.

Passing the Plate

Henry G. Riekers
208 Phelps Ave.
Glen Burnie MD 21061

Recently, I was asked to look into the possibility of keeping track of Sunday church collections using a computer costing less than $1000. I selected a 16K Level II TRS-80 with cassette interface. This article explains the problem, rationale and program developed for this project.

The Problem

The church has about 400 parishioners, most of whom make weekly contributions in individual envelopes. In addition to the weekly collections, there is a monthly collection for the building fund and an occasional special collection. Individual donations usually do not exceed $5 per week. It is necessary to know the amount collected from each person at each collection, as well as the total amount collected from each person for all collections for the year.

The Rationale

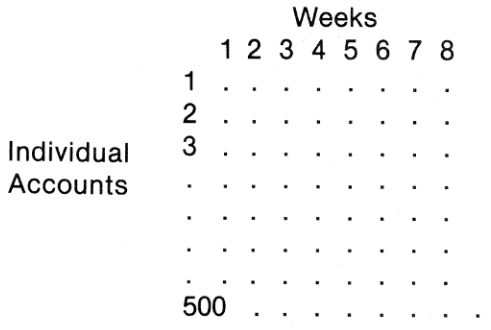
The 16K Level II TRS-80 has 15,572 bytes of usable memory available. About 2000 bytes are required for the program instruction. This leaves about 13,000 bytes for organization of data. A two-dimensional numeric array will have to be defined. One dimension will be the account number; the other dimension, the number of the week.

In a Level II array, variables can be integer, single precision or double precision. Normally, the single-precision mode, consisting of seven significant digits, is used for most applications and requires three bytes of memory for each array value stored. Double precision, consisting of 16 significant digits, requires four bytes of memory for each array value stored, but is too versatile for this application.

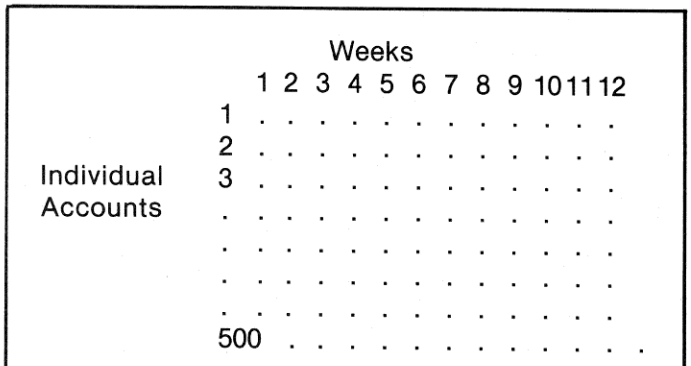
If we divide the 13,000 available bytes by 3, we can create an array that can hold approximately 4000 values. If we assume that congregation growth will increase to 500 by the year's end, then we can store eight week's worth of data (see Example 1). But let us examine the size required of each data element in the problem.

If we multiply a $5 weekly contribution figure times 60 collections per year, we will find a maximum contribution of $300 per account. With a little ingenuity, we can construct an array with integers, thus increasing the storage capability. In Level II, integers consist of whole numbers up to 32767 and only require two bytes of memory for each array value stored. Therefore, if we divide our 13,000 available bytes by 2, we will obtain a new array that can store 6500 2-bit array values (see Example 2).

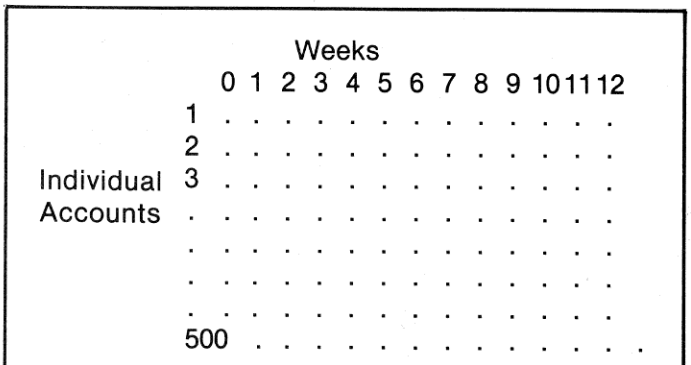
The integer array presents two problems: how to store values less than one dollar, for instance, 50 cents, and how to



Example 1. Array consisting of 4000 single-precision values.



Example 2. Array of 6000 integer values.



Example 3. Array of 6500 integer values. This array is used in the Church Collection program.

store a full year's collections. The solution to the first problem is to store the amount contributed in cents. In other words, $5.23 is stored as 523. Data is entered and displayed in dollars and cents format; only the storage is done in integer format.

The second problem is more difficult. Since we want to accumulate a full year's worth of data, a method is required to carry individual account data from one 12-week period to another. Also, it is apparent that we have used only 6000 of the 6500 two-byte locations for the array of Example 2. In order to accomplish the summary of individual account data, week 0 is added to the array (see Example 3).

Since week 0 will only be used for summing data, how is an entire year's data collected? Each array carries two month's worth of data, so one data tape for each two-month period is saved. In total, there will be six data tapes at the end of the year.

The Program

The program for this project is relatively short and consists of less than 70 lines. Line 5 describes the array limits, where H is the maximum number of accounts and G is the maximum number of weeks. There are two limitations on G and H: Their product will not exceed 6000, and H must always be a number evenly divisible by 20 in order for TAPE to work properly.

The menu consists of nine different selections:

1. DISPLAY will ask for the week number and then allow you to interrogate any number of individual accounts displaying the amount for the given week. DISPLAY is terminated by entering 0 for the account number.

2. ENTER allows you to put data into the system. ENTER asks for the week number and then will permit any number of data entries for the given week by asking for the account number and the amount. ENTER is terminated by entering 0,0.

3. ACCT SUM will display the total to date for a particular account number after access control is returned to the menu.

4. WEEK SUM will display the total given from all accounts in a particular week after access control is returned to the menu.

5. TAPE allows data to be transferred to and from tape by creating and reading data files. It is essential that the best tapes are used since transfer of a 6500 valued array will take approximately 30 minutes. The tape should be loaded in the cassette recorder, which should be placed in the proper mode before entering the TAPE mode. As transfer occurs, numbers appear on the CRT. Each number represents the block transfer of 20 array locations after access control is returned to the menu.

6. SUM TO 0 will collect each weekly value including week 0 for each account and put the total into week 0 for each account. This option permits you to carry balances forward for each account to a new data tape. An option also permits you to zero all weeks except week 0 after SUM TO 0 is executed in order to ready the array for the next two month's data.

7. ACCT DETAIL will display the week and amount given for each that is greater than zero after access control is returned to the menu.

8. WEEK DETAIL will display the account and amount given for each that is greater than zero after access control is returned to the menu.

9. RANGE will display the range of account numbers and week numbers available after access control is returned to the menu.

It should be noted that all entry points are protected so that an error cannot occur because an out-of-limits number was attempted. Either a note that says TOO LARGE will appear or the question will be asked again.

A few additional notes are probably needed concerning the tape transfer process and how it is used. The following steps could be followed:

For week #1.

1. Enter data for week #1.
2. Transfer to new tape.

For week #2 to week #11.

1. Load tape from above and then erase tape.
2. Enter data for new week.
3. Transfer to new tape.

For week #12.

1. Load tape from above and then erase tape.
2. Enter data for week 12.
3. Transfer to new tape and save this tape.
4. SUM TO 0 and then zero all accounts.
5. Transfer to another new tape.

For second week #1.

1. Load tape from above, erase tape and follow instructions from week #1 above.

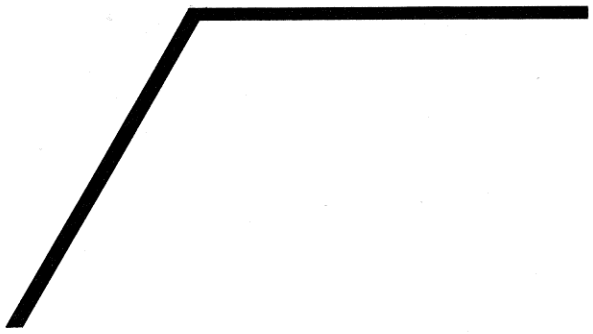
Therefore, as each additional week's data is generated, a new tape that has all the previous week's data as well as the new data is created. After 12 weeks, the data is summed and a new tape is created. The new tape will have the sum of the accounts of the previous tape in week 0.

Conclusion

Some will argue that this project should be done on a disk-based system. However, my task was to record the collections data on a computer selling for less than $1000. Because of the inherently low cost of the system it is easily within the grasp of many churches. This project may only be the beginning of microcomputer applications for churches.■

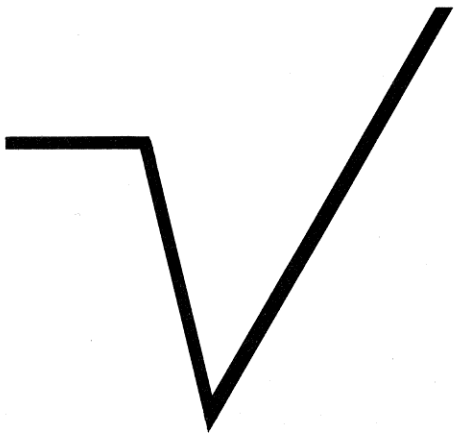
```
1 REM // CHURCH COLLECTION
2 REM BY HENRY G. RIEKERS
3 REM // GLEN BURNIE MD 21061
4 DEFINT A,G,H,I,J,N,M,Q
5 G=12:H=500:I=1:J=1
10 DIMA(G,H):GOSUB900
90 PRINT"* * * *"
91 PRINT"DISPLAY=1,ENTER=2,ACCT SUM=3,WEEK SUM=4,TAPE=5"
92 INPUT"SUM TO 0=6,ACCT DETAIL=7,WEEK DETAIL=8,RANGE=9";X
93 IFX=9THEN800
94 CLS:IFX=7THEN600
95 IFX=3THEN200
96 IFX=8THEN700
97 IFX=4THEN300
98 IFX=5THEN400
99 IFX=6THEN500
100 INPUT"WEEK ";N:IFN>GTHEN100
101 IFX=1THENINPUT"ACCT";M:IFM>HTHENPRINT"TOO LARGE":GOTO101
102 IFX=2THENINPUT"ACCT,AMT";M,L:IFM>HORL>300THENPRINT"TOO LARGE"
    :GOTO102
103 IFM=0THEN90
105 IFX=1THEN130
110 A(N,M)=L*100:GOTO101
130 PRINT"WEEK ";N,"ACCOUNT ";M,"AMT $";A(N,M)/100
140 GOTO101
200 INPUT"SUMMARY OF ACCOUNT";M:IFM>HTHEN200
210 FORN=ITOG
215 B=A(N,M)+B:NEXTN
225 PRINT"SUM OF ACCOUNT ";M;" IS $";B/100:B=0
230 GOTO90
300 INPUT"SUM OF WEEK ";N
305 IFN>GTHENPRINT"TOO LARGE":GOTO300
310 FORM=JTOH:C=A(N,M)+C:NEXTM
340 PRINT"SUM OF WEEK ";N;" IS $";C/100:GOTO90
400 INPUT"CREATE TAPE 1, LOAD TAPE 2 ";Y
401 INPUT"PRESS ENTER WHEN READY";Z$:IFY=1THEN405
403 IFY=2THEN450
404 GOTO90
405 FORN=I-1TOG
407 FORM=J-1TO(H-20)/20:Q=20*M
410 PRINT#-1,A(N,1+Q),A(N,2+Q),A(N,3+Q),A(N,4+Q),A(N,5+Q),A(N,6+Q
    ),A(N,7+Q),A(N,8+Q),A(N,9+Q),A(N,10+Q),A(N,11+Q),A(N,12+Q),A
    (N,13+Q),A(N,14+Q),A(N,15+Q),A(N,16+Q),HN,17+Q),A(N,18+Q),A(
    N,19+Q),A(N,20+Q)
415 PRINTM;:NEXTM:PRINTN,:NEXTN
430 GOTO90
450 FORN=I-1TOG:FORM=J-1TO(H-20)/20:Q=20*M
460 INPUT#-1,A(N,1+Q),A(N,2+Q),A(N,3+Q),A(N,4+Q),A(N,5+Q),A(N,6+Q
    ),A(N,7+Q),A'N,8+Q),A(N,9+Q),A(N,10+Q),A(N,11+Q),A(N,12+Q),A
    (N,13+Q),A(N,14+Q),A(N,15+Q),A(N,16+Q),A(N,17+Q),A(N,18+Q),A
    (N,19+Q),A(N,20+Q)
465 PRINTM;:NEXTM:PRINTN,:NEXTN:GOTO90
500 PRINT"SUM EACH ACCOUNT TO DATE"
510 FORM=JTOH:FORN=ITOG
530 A(0,M)=A(N,M)+A(0,M)
535 NEXTN:T=A(0,M)+T
545 IF A(0,M)>0 THEN PRINT"ACCT ";M;" $";A(0,M)/100
550 NEXTM
555 PRINT"TOTAL IN ALL ACCOUNTS $";T/100:T=0
557 INPUT"TO ZERO ALL ACCTS ENTER 1";Z:IFZ=1GOSUB900
560 GOTO90
600 INPUT"DETAIL OF ACCOUNT ";M
605 IFM>HTHENPRINT"TOO LARGE":GOTO 600
610 FORN=ITOG
620 IF A(N,M)>0THEN PRINT"WEEK ";N;" $";A(N,M)/100
630 NEXTN:GOTO90
700 INPUT"DETAIL OF WEEK ";N:IFN>GTHEN700
710 FORM=JTOH
720 IFA(N,M)>0THENPRINT"ACCT ";M;" $";A(N,M)/100
730 NEXTM:GOTO90
800 PRINT"WEEK ";I,G,"ACCT ";J,H:GOTO90
900 FORN=ITOG:FORM=JTOH
910 A(N,M)=0
920 NEXTM:NEXTN:CLS:RETURN
```

Program listing



How fast can you compute square roots in Level I BASIC?

Root Routines



Curtis F. Gerald
980 West St.
San Luis Obispo CA 93401

In many microcomputer implementations of BASIC, such as Level I BASIC on the Radio Shack TRS-80 system, there are few functions built into the interpreter. When this is the case, subroutines must be written that can be called to evaluate the function. Taking square roots, a frequently used function, is typical.

While Radio Shack's manual gives a subroutine for square roots and while alternative sources for such programs are not hard to find, it is important that the efficiency of the software be good. After all, the subroutine will be called hundreds of times, so it is worthwhile to spend the effort to make it as efficient as possible. Then, too, a study of the timing relationships, a determining factor in the efficiency, is important in its own right to give insight into the function of the language.

```
500  IF N = 0 THEN X = 0: RETURN
510  IF N>0 THEN 530
520  PRINT "ROOT OF NEGATIVE NUMBER?": STOP
530  X = N*.5: Z = 0
540  W = (N/X – X)*.5
550  IF (W = 0) + (W = Z) THEN RETURN
560  X = X + W: Z = W: GOTO 540
```

Fig. 1a. Algorithm for the square root of N from Radio Shack's Level I manual, p. 216.

```
500  X = N/2
510  IF X*X = N THEN RETURN
520  D = N – X*X
530  IF ABS (D)≤.001 THEN RETURN
540  X = 1/2*(N/X + X): GOTO 510
```

*Fig. 1b. Algorithm for the square root of N by William Colsher (*Kilobaud, *July 1978, p. 68).*

Some idea of the difference in the performance of two alternative subroutines can be gained by a comparison of the subroutine suggested by Radio Shack and that described by William Colsher in the July 1978 issue of *Kilobaud* (p. 68). Fig. 1 gives the code for the two square root programs and Table 1 shows the times for typical calls to each of them. These times were for 50 repetitions of the call, using the short BASIC program listed in Fig. 1c. There are moderate differences in the execution times, but, more important, the Colsher algorithm does not converge for values of N greater than 64 x 10^{10} and for N less than about 0.0005. It also has serious errors for N less than 0.005.

The variation in execution time with N shown in Table 1 and the differences between the two algorithms indicate that a study of the various factors that influence the execution time would be valuable. We can separate the time factors into four categories: the choice of method, the starting value used, the individual computation steps that are involved and the criterion for ending the computational steps.

```
10  REM DRIVER TO TEST SQUARE ROOT ALGORITHMS
20  REM WHICH ARE IN A SUBROUTINE AT 500
30  INPUT N: FOR I = 1 TO 50: GOSUB 500: NEXT I:
40  PRINT "DONE", N, X
50  END
```

Fig. 1c. Program to test execution times.

Newton's Method

Most square root subroutines used in computers are based on Newton's method. The usual derivation of Newton's method is based on drawing a tangent to a curve at a point corresponding to an initial estimate. The point where this tangent line crosses the x-axis is taken as a second estimate, as illustrated in Fig. 2. The technique is repeated until sufficient precision is obtained. Since the derivation of Newton's method based on the tangent involves calculus, it may be easier for some readers to follow a different approach that leads to the same result.

Suppose we make a guess X for the square root of the number N. In general, X will not be the correct value, and, if X is too small, N/X will be too large. (For example, if N = 10 and X = 3, N/X = 3.3333.) However, the correct value for the square root of N will lie between X and N/X. (This is true for both X too small and too large.) An improved value for the square root, then, will be the average of X and N/X. We repeat this until the average doesn't change. This rule can be expressed by the formula:

$$X(\text{improved}) = (N/X + X)/2$$

While other methods than Newton's could be used, this one has a special advantage in that the error is reduced about equal to its square in each repetition of the formula. The net result is that the accuracy improves very rapidly. Table 2 illustrates this by giving some computations for $\sqrt{10}$ using Newton's method. We see that the number of correct decimal places about doubles each time as a result of the rapidly decreasing error.

Newton's method, like a wide variety of similar techniques, needs a starting value to initiate it. Obviously, that choice has a strong influence on how many times the formula must be applied. So, the accuracy of the starting value is directly related to the execution time of the calculation. If our subroutine is to handle any value of N, we have to make some arbitrary choice. It is customary to begin with N/2. (Both procedures shown in Fig. 1 use this value.) There is an obvious improvement that can be made, however. If X = N/2 is substituted into the formula, the first improved value of X is readily seen to be N/4 + 1. If we were to start with that value, we save one *iteration*. (Iteration is the technical name given to one ap-

Value of N	Time, seconds, for 50 computations	
	Radio Shack (Fig. 1a)	Colsher (Fig. 1b)
1	15	13
100	21	19
1 E 4	29	31
1 E 8	49	53
1 E 16	85	overflows
1 E 32	160	"
.01	27	22
1 E −4	37	doesn't converge
1 E −8	58	"
1 E −16	92	"
1 E −32	168	"

Table 1. Timing comparison of square root algorithms using TRS-80, Level I.

Estimate number	Estimate of $\sqrt{10}$	Number of correct digits	Error
1	5	0	1.8377
2	3.5	1	0.3377
3	3.17857	2	0.01629
4	3.16232	4	4.176×10^{-5}
5	3.162278	9	1×10^{-9}

Table 2. Decrease of error in Newton's method.

plication of the computational scheme.) Unfortunately, we can't do this again to jump one more step because the next result requires us to solve an equation in which we must take the square root.

Other Speed Factors

The other speed factors require more study. We have to know the relative times to execute the various instructions. For example, if an IF comparison is quite slow, some other means of ending the computations might be preferred. We also need the same kind of information in order to decide which individual computational steps are to be avoided and which are to be preferred. This

Case	Statement	Time, sec, for 1000 repetitions
1	X = 1	5.5
2	X = 20	15
3	X = 20.	15
4	X = 20.0	21
5	X = 20.02	27
6	X = 20.020	32
7	X = 20.0202	38
8	X = 20.02020	45
9	X = 1.23456	21
10	X = 1.234567890	51
11	(Y = 1.234567890) X = Y	5
12	(Y = 1.234567890) X = 1*Y	17
13	(Y = 1.234567890: Z = 1) X = Z*Y	9
14	(Y = 1.234567890 Z = 1.234567890) X = Z*Y	9
15	(Y = 1.234567890 Z = 1.234567890) X = Z/Y	9
16	(Y = 1.234567890 Z = 1.234567890) X = Z + Y	8
17	(Y = 1.234567890 Z = 1.234567890) X = Z − Y	8
18	(N = 50) Y = N/2: X = N/Y	12
19	(N = 50) Y = N/2: X = (N/Y + Y)/2	18
20	(N = 50) Y = N/2: X = .5*N/Y + Y)	21
21	(N = 50) Y = N/2: X = 1/2*(N/Y + Y)	23
22	(N = 50) Y = N/2: LET X = (N/Y + Y)/2	16
23	(N = 50) Y = N/2 LET X = (N / Y + Y) / 2 (statement was spread out to maximum line)	17
24	(X = 25) IF X<.001 THEN 40	16
25	(X = 25) IF X≤.001 THEN 40	16
26	(X = 25) IF X>.001 THEN 40	18

Table 3. Execution times for statements using TRS-80, Level I.

✓ Reader Service—see page 131

Starting value: X = N/4 + 1

N	Interations Needed: Absolute error of 1 E – 4	Interations Needed: Relative error of 1 E – 4
1	3	3
100	4	4
1 E 4	8	7
1 E 8	12	11
1 E 16	15	14
1 E 32	42	41
.01	6	6
1 E – 4	9	9
1 E – 8	11	13
1 E – 16	13	16
1 E – 24	14*	29
1 E – 32	– *	56

*Absolute error of 1 E – 4 is insufficient accuracy

Table 4. Number of repetitions of Newton's method to achieve errors of less than 1 E – 4.

is dependent on the computer system being used, as well as on the particular BASIC interpreter. Table 3 shows the times in seconds for 1000 repetitions of some representative statements in BASIC for a Radio Shack TRS-80 computer with Level I BASIC. Some of the results are quite surprising.

The results in Table 3 were obtained by putting each of the statements into a loop that repeated itself 1000 times. The recorded times are the measured times minus the time for the loop itself (the loop run with the subject statement deleted).

The data in Table 3 shows that a simple assignment can vary in timing by over nine to one (cases 1-10). Increasing the number of digits in the constant assigned to a variable, even by adding trailing zeros, adds to the execution time by about six seconds (per 1000 executions) for each digit. Digits before and after the decimal point are not equal in their effects on timing, as shown by cases 7 and 9. Equating one variable to another, as indicated by case 11, is rapid, however. The explanation for this is that BASIC, which is an interpreter rather than a compiler, must repeat the conversion of constants in character form to internally represented binary floating-point numbers every time a statement is executed. When X = Y is executed, the already converted constant that Y equals is only moved to the location of X.

It is somewhat surprising to note that multiplication and division operations involving floating-point numbers are only slightly slower than addition and subtraction (see cases 13-17). The time-consuming conversion process, even when the constant is unity, is again illustrated by case 12.

Cases 19-21 explore the timing of statements that are normal in a Newton square root algorithm. Observe that dividing by 2 is quicker than multiplying by .5, and that multiplying by ½ is still slower. From the earlier cases, multiplying by H, where H has previously been set equal to .5, should be still better.

Case 22 illustrates something that is not obvious in advance. Omitting LET from an assignment slows the execution. (Not illustrated in Table 3 is a similar effect of using the abbreviations that are permitted in Level I.) Case 23 shows that spacing out a statement by inserting blanks has a small

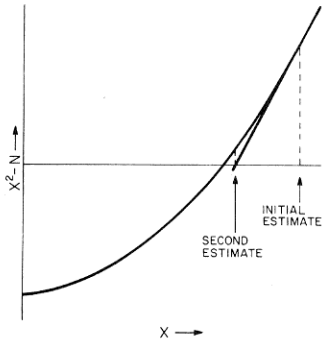


Fig. 2.

negative effect on speed. Along this same line is a minor acceleration when multiple statements are stacked on the same line. This is presumably because less parsing time is needed when the computer reads the instruction.

A simple IF condition is quite time-consuming as shown by cases 23-25. When the branch is taken, slightly more time is needed than when it is not. A test for completion using an IF condition takes as much time as one additional Newton-type calculation as shown by comparing case 22 with case 24.

We need to interpret the data of Table 3 in the light of the Newton computational scheme. These conclusions seem warranted:

1. The form of case 22 is best for getting the next estimate of $\sqrt{N}$.
2. Testing for convergence, as in case 24, should be minimized. One way this might be done is to find good enough starting values to permit doing only a known number of iterations for the desired precision.
3. Variables should be used instead of constants when the value is used repeatedly.
4. Factors such as stacking statements and avoiding abbreviations will be of help.

Choosing a Starting Value

There is one other important issue that must be addressed before a square root subroutine can be optimized. This is how the starting value can be obtained. We have previously mentioned that using X = N/4 + 1 instead of X = N/2 will save one iteration. This is a reasonable way to get the starting value for N between 0.6 and 20, but outside this range it is poor. Table 4 gives the number of iterations to reach four-decimal-place accuracy, starting with X = N/4 + 1. Also shown is the fact that a relative error criterion (error/true value) reduces the number of iterations required by at most one, even for N = 1 E 32.

The major reason for the variation in the required number of repetitions of the Newton-method computation is the lack of good starting values when N is greater than 100 or less than 0.1. This factor causes the time to calculate the square root to increase by up to 19 to 1! This also suggests how the subroutine could be improved best: Find a better starting value, which must take into account the size of N.

When N is expressed in floating-point format, it is readily apparent that the exponent of N determines its size and that N will have an exponent about half as large in absolute value. Unfortunately, Level I BASIC does not have a PEEK instruction to permit us to easily get the exponent of N, nor a POKE instruction to let us adjust the exponent of our starting value. If we can't manipulate exponents directly, we will have to do it by indirection.

Suppose we use a starting value for X of N/4 + 1 when N is not too large, say between 1 and 200. (In this range of N, the starting value is not more than 3.6 times the correct square root. When we are this close to begin with, only four iterations are needed for an accurate square root.) If N is larger than 200, up to 20,000, we use a starting value that is one-tenth as

```
500  REM THIS SUBROUTINE COMPUTES SQUARE ROOTS BY
510  REM NEWTON'S METHOD. VARIABLES USED ARE:
520  REM N = VALUE WHOSE ROOT IS DESIRED, PASSED TO ROUTINE
530  REM X = RETURNED VALUE OF ROOT
540  REM M, O = TEMPORARY VARIABLES
550  LET O = N: IF O<1 THEN LET N = 1/O
560  LET M = N: LET X = 2.5•N + 10
570  LET X = X/10; LET M = M/100: IF M>2 THEN 570
580  FOR I = 1 TO 4: LET X = (N/X + X)/2: NEXT I
590  IF O<1 THEN LET N = 1/N: LET X = 1/X
600  RETURN
```

Fig. 3. Subroutine for square roots.

N	Time, sec
1	7
100	8
1 E 4	9
1 E 8	12
1 E 16	16
1 E 32	26
.01	9
1 E −4	10
1 E −8	13
1 E −16	17
1 E −32	27

Table 5. Time to compute 50 values of square root using subroutine in Fig. 3.

great. For N between 20,000 and 2,000,000, we divide by ten again and continue to divide the starting value by ten for each hundred-fold increase in N.

This scheme must be modified for fractional values of N. For very small fractions, a starting value of N/4 + is too large and should be divided by ten for each hundred-fold *decrease* in N. We can use the previous computations for fractional N, however, if we apply them to the reciprocal of N and then take the reciprocal of the answer at the end.

This strategy spends time in getting a good starting value in order to save time doing the iterations. Experiments show that if the starting value is between one-fourth and four times the correct square root, four iterations will give four to five digits of accuracy. Five iterations would double the number of correct digits.

When a program incorporating the idea was tested, the times for 50 computations of the square root were shown in Table 5. Comparison with Table 1 shows greatly improved speed, by as much as a factor of six or more. Fig. 3 gives the code for the algorithm, written as a subroutine. A very minor improvement in speed would result if some constants were replaced by variables, but so many more temporary variables would be needed that it did not seem worthwhile considering the rather severe limitation to variable names in Level I on the TRS-80.

In Conclusion

A machine-language routine would be even faster than any subroutine written in BASIC. Some idea of the difference between machine code and a higher-level language that is interpreted is obtained by noting that Level II in the Radio Shack computer computes 50 values of SQR(N) in under four seconds. This is faster by a factor of about seven for N = E 32 than the time for the subroutine of Fig. 3. It is, at the same time, faster by a factor of over 40 in comparison to the subroutine given in the Level I manual.■

In the market for a cheap printer? See how one of our readers fared with the Anadex 8000.

Anadex Printer Review

Walter C. June
18216 Ginaville Lane
Eden Prairie MN 55344

The day finally came when a line printer became a necessity rather than a luxury purchase. I was looking for a printer that could handle my heavy documentation. Also, having installed Electric Pencil and the lowercase mod, I was not about to spend time on uppercase only printers. At the National Computer Convention in New York, I took the time to visit every printer manufacturer at the show. That took two full days and I have to admit, I came away totally confused. Too much data input in too short a time had thoroughly overloaded my circuits.

When I returned to my office I started going through the handouts from the printer reps. One caught my eye right off and continuing through the others I found myself comparing them to this one, the ANADEX DP-8000. Others had more impressive specs in one area or another, but the ANADEX had all that I required and that most important spec, low cost.

ANADEX DP-8000

The ANADEX DP-8000 handout received was a printout. The printer attaches to the TRS-80 through either of two connectors. One provides EIA std. RS232C and 20/60 milliamp current loop interface and the other provides a Centronics plug compatible interface.

The device prints 80 columns at 112 characters per second bi-directionally, giving you a throughput of 84 lines per minute. The 96 characters (upper/lowercase) are formed by a 9 × 7 dot matrix (no descenders).

Additional features include top-of-form control, skip-over-perforation control, eight programmable vertical tab positions, double width printing, and a 1024 character buffer that can be optionally expanded to 2048.

All that with a list price of $995 sounded great. I found my local representative, ordered one and, surprisingly, had it in my office within a week.

Plugging It In

The Anadex hardware manual includes two notes regarding the TRS-80. Unfortunately both of them are negative. The first informs you as to how misleading the term Centronics plug compatible is. I'm starting to believe this term literally means the plugs are compatible, it's just the wires inside that change. Anadex instructs you to relocate the wire on pin-11 of your Centronics cable to pin-36. This is not difficult and is accomplished in a matter of minutes, however if you are unfortunate enough to have purchased a printer manufactured before January 4, 1979, your work is not finished. For these early models you must disassemble the entire printer and make a wire change on the circuit board. Incidentally, the note does not tell you how to determine the manufacture date of the printer, you will have to trust the word of your salesman or plug it in and see what happens.

The second note warns that the printer may not operate with certain models of the expansion interface. The problem revolves around the length of the DATA STROBE signal generated by the interface. Anadex printers require a minimum of 2 micro-seconds and they claim some interfaces generate signals as small as 1.5 micro-seconds. But this is confusing as well since the Centronics printers specify a minimum DATA STROBE length of 1.0 micro-second.

Anadex recommends a capacitor change in the expansion interface to eliminate the problem. I escaped this alteration too, for my printer, but having made other changes within my expansion interface I would advise the average buyer to seek professional help.

Some Drawbacks

With only one cable modification the printer was up and running. DIP switches are provided permitting you to simulate the Centronics printer so no software changes are required and the standard LPRINT statement works fine.

Two idiosyncracies of the printer should be noted. Top-of-form control is implemented using DIP switches to set page length. Normally this works fine, however, while at top-of-form, all line-feed commands are ignored. I even set the form length to zero, hoping this would disable top-of-form, but this had no effect. All line feeds were still ignored. Depending on how you use your printer this could be extremely inconvenient.

The Anadex incorporates a 1024 byte buffer. The TRS-80 places characters in the buffer until a carriage return is received. After receiving a carriage return, the line is printed, while the next one is entering the buffer. With this buffer you will find that the TRS-80 is finished with your LPRINT and is continuing with your program while the printer is still running. Anadex decided to place what is called a dead man timer within the printer so that if nothing comes from the TRS-80 in 10 seconds any residue left in the buffer is erased.

Remember: An LPRINT list followed by a semicolon does not send a carriage return and thus is sitting in the buffer. Remember also that Level-II BASIC will go off and clean up the string area whenever it chooses. Put these two facts together and they equal missing lines in your printouts.

Conclusion

Though this article pointed out possible problems and may have appeared negative, the only honest complaints I have are with the ribbon and DIP switches for setting page length and skip-over-perforation.

The ribbon is only available, as far as all the suppliers in Minneapolis are concerned, from Anadex and only available in the one color, Intense Purple. The ribbon price is reasonable, but sometimes I prefer formal black ribbon.

The DIP switches are inside the case and require removal of paper and printer cover for changes.

Overall, the 80-column DP-8000 is a reliable printer at a reasonable price. ■

✓ Reader Service—see page 131

How to convert the Scelbi Personal Information Management System to run on disk BASIC.

Floppy PIMS

Morris Herman
503 Rosario Drive
Santa Barbara CA 93110

This article deals with the conversion of the Personal Information Management System (PIMS from Scelbi Publications) from a cassette-based program to a disk version. In addition to the I/O changes, some Disk BASIC features are used to enhance the system.

PIMS is a good introduction to the subject of data base management. The program is available in a listing format at the end of an 84-page book that details some of the applications and uses for the system. The program is written in Microsoft BASIC and can be run on a TRS-80 without any changes. The book also contains notes on changes required for running on a Commodore PET. I own a TRS-80 and will confine my enhancements to the TRS-80 since I am not familiar with PET BASIC, but I presume the same changes can be made to accommodate a PET user.

The book is informative and written in a very readable form. The program was written by Madan L. Gupta and contains quite a few remarks, which makes the program easy to follow.

A database system is a program that enables an operator to use a computer in an application that involves data without doing any programming that might be required for the specific application. The user has the capability to define the format of the database, which generally consists of alphanumeric or numeric fields in records within a file. For example, a record within a checkbook file might consist of fields defining date of check, payee, amount of check, expense category, and whether the item is tax deductible or not.

The user defines the fields, their format and their number within a record. The PIMS TRS-80 version is dimensioned for ten fields within each record and 99 records in a file. I have modified these limits because I have 48K of RAM in my system.

Commands

With PIMS, a user can generate a database and operate on that database with commands such as SEARCH, SUM, LIST, ADD, CHANGE, RUBOUT, SORT and SAVE. The commands describe the operation involved quite adequately.

The SEARCH command will find any one or group of records that contains a field equal to a desired expression. For the previous example, the user can search for all checks written for medical expenses or all checks written to a particular doctor. SUM will sum a numeric field in all or part of the records in the file. The sum operation can be made contingent on an expression in another field of the records. This operation can be used to get the sum of all checks written to a particular doctor.

LIST will display one, a range or all records in the file. ADD is used to add records to a current file. The format of the file has been previously defined by the user. CHANGE will change any field in one or more records. RUBOUT will delete one, a range or all the records in the file. SORT will sort the records in the file in ascending order of any field in the records, alphanumeric or numeric. SAVE will save the file on the cassette in the original version (on the disk in the version I am presenting in this article).

LABELS is a command that can be used to generate address labels from a mailing list file. This command allows label alignment and prints only the first five fields in the specified records. This allows the other fields to contain qualification data to be used for deciding which records are to be printed. HELP will list all valid PIMS commands. Finally, END will terminate the program with a timely reminder about saving the file before exiting.

Modifications

The first modification of the system involves expanding the number of records allowed within a file. With 48K of RAM, I could afford to allocate more space for string variables by changing CLEAR 6000 to CLEAR 22000 at the very least. In addition, I changed the dimensions of the R$ and D$ arrays from 100 to 500. I saw no need to enlarge the number of fields within a record from the preset ten fields.

The second modification involves the use of the LINE INPUT command in place of the INPUT command. This change allows the use of commas or other punctuation in an alphanumeric input stream. This negates the need to warn users about separating last and first names or city and state by commas. The lines involved in the change are 850, 1320, 1710, 1910, 1950 and 3190.

The next modification involves the use of a printer for some of the output. I will assume that a TRS-80 with disks will most likely have a hard-copy device too. Otherwise, for a long file, it is impossible to list a large number for records. The current program scrolls all output on the screen, and you can only stop it with a SHIFT@ command. This

can be annoying after a while. . .after all, didn't you get a printer for hard copy? This modification allows printing for the LIST and SEARCH commands. The changes for printer usage are as follows:

1. Add lines 1122 and 1325 as follows:

```
1122 IF T = 2 GOSUB 3450
1325 IF T = 7 GOSUB 3450
```

2. Add lines 1650 through 1667 as follows:

```
1650 IF LP = 1 GOTO 1665
1655 PRINT "(";I;")"
1662 GOTO 1670
1665 LPRINT TAB(20);"(";I;")"
1667 LPRINT TAB(20);"-----"
```

3. Add lines 1685 through 1695 as follows:

```
1685 IF LP = 1 GOTO 1695
1692 GOTO 1700
1695 LPRINT TAB(20);N$(J);": ";B$(J)
```

4. Add lines 3450 through 3490 as follows:

```
3450 REM HARD COPY SWITCH SETTING
3460 LP = 0
3470 INPUT "DO YOU WISH HARD COPY
(Y OR N) ";A$
3480 IF LEFT $(A$,1) = "Y" THEN LP = 1
3490 RETURN
```

The last and most significant modification involves changing from cassette files to disk files. The task is quite easy if you don't mind having sequential access instead of random access records within a file. Since the program only operates on memory resident files, I saw no need to have random access files.

The one problem with sequential access files is that when a file is opened for output, its entire contents are lost. This could be disastrous if the SAVE operation is interrupted by a disk failure and TRSDOS is booted up. Since the file only exists in memory at that point, a bootup means lost data. To safeguard against such an occurrence, a backup diskette should always be made. In addition, I have changed the name of the file for the data output. That way, the old file is still there if the output operation fails. The last letter in the name of the file before the extension, password or drive number will be changed to a Z, regardless of what letter is there when you input the file name.

The only disadvantage to this scheme is that if there is not enough room on a diskette for a second copy of your file, then the scheme itself will cause the problem you're trying to avoid. If you suspect that this will be the case, you can defeat this scheme by either using Z as the last letter for all your files or deleting lines 3191 through 3196 from the changes below.

The changes are as follows:

1. Delete lines 1914 through 1942.

2. Add lines 1920 through 1940 as follows:

```
1920 OPEN "I", 1, F$
1930 INPUT #1, T$
1935 PRINT T$; "FILE LOADING"
1940 GOTO 1950
```

3. Change line 2070 to

```
2070 CLOSE :GOTO 540
```

4. Insert lines 3191 through 3198 as follows:

```
3191 FOR I1 = 2 TO LEN(F$)
3192 IF MID$ (F$,I1,1) = "/" GOTO 3196
3193 IF MID$ (F$,I1,1) = "." GOTO 3196
3194 IF MID$ (F$,I1,1) = ":" GOTO 3196
3195 NEXT I1
3196 MID$ (F$,I1-1,1) = "Z"
3197 PRINT F$;"FILE
CREATED—RENAME IF DESIRED AFTER
EXIT
3198 OPEN "O",1,F$
```

5. Change lines 3200, 3280, 3300, 3320 by deleting the "-" character after each PRINT# command.

6. All line 3325 as follows:

```
3325 CLOSE
```

In addition to the above changes, the June 1979 issue of *Kilobaud Microcomputing* has a review of PIMS (p. 13, by Curtner B. Akin, Jr.) and a list of corrections and enhancements to improve operation. I was mystified by the first four changes, since my PIMS book has the corrected lines already. There is no note in the book of a second edition, so I don't know who made the mistake. The other changes outlined by Akin are very sound.

Let me state again that the program is a very good value for $9.95 (I got it for $6.95 at the West Computer Faire in San Francisco). It might just get you to finally automate your household chores and justify your expenses in purchasing the microcomputer system. ■

✓ Reader Service—see page 131

How to add additional BASIC statements to your program, from cassette.

APPEND It!

Curtis F. Gerald
980 West St.
San Luis Obispo CA 93401

When programs are loaded from tape by the CLOAD command in your TRS-80 (and similarly in other systems), programs or data that are in memory are overlaid. This means that you cannot add a subroutine called from a main program that is already in memory, nor can you add additional lines of DATA statements to supply values for the READ statements of your program. The net result is that main programs and all the subroutines they might wish to call must be written as one composite program, and the use of alternative subroutines or different data is impossible without entering them by hand.

In some more elaborate computer systems, an APPEND command that adds the saved program to the end of the current one rather than replacing it is available. This is a most useful facility; the program described here provides the TRS-80 owner with the ability to append additional BASIC statements from cassette storage to a program already in the memory.

A second purpose of this article is to show how the program was developed. If you are like me, you often learn more by seeing how a problem is solved than by looking only at the final result. In the process of explaining the development, I will show how BASIC statements are stored, how they are chained together during the execution phase, how variables are located in memory and how certain utility routines (which are located in ROM) are exposed. The technique used to locate these utility routines is also of interest, since the addresses where these are to be found are not given in Radio Shack's documentation. (Some of these entry points are given in the T-BUG documentation for Level I BASIC, but not for Level II*.)

There are several questions that must be answered before you can write the APPEND program, if you want to understand the internal workings of BASIC in the TRS-80. Even if your orientation to computing is purely from the high-level language point of view, your ability to use the machine and to correct some of the more subtle bugs will be enhanced by this information. (I guess that means that everyone should read this piece!) The program relates to Level II, although the changes needed to use the program in Level I are pointed out in the text.

The program listing gives the machine-language listing of the APPEND program. The accompanying assembler-language code and the comments should help you to understand the sequence of steps that are

```
10  REM TEST PROGRAM
20  X=2
30  FOR I%=1 TO 4:PRINT I%,X↑I%:NEXT I%
```

Location	Contents	Significance			
42E9	FC	Next line	6	1E	Line 30
A	42	at 42FC	7	00	Null
B	0A	Line 10	8	81	FOR
C	00	Null	9	20	Space
D	93	REM	A	49	I
E	20	Space	B	25	%
F	54	T	430C	D5	=
42F0	45	E	D	31	1
1	53	S	E	20	Space
2	54	T	F	BD	TO
3	20	Space	4310	20	Space
4	50	P	1	34	4
5	52	R	2	3A	:
6	4F	O	3	B2	PRINT
7	47	G	4	20	Space
8	52	R	5	49	I
9	41	A	6	25	%
A	4D	M	7	26	,
B	00	Null	8	58	X
C	04	Next line	9	D1	↑
D	43	at 4304	A	49	I
E	14	Line 20	B	25	%
F	00	Null	C	3A	:
4300	58	X	D	87	NEXT
1	D5	=	E	20	Space
2	32	2	F	49	I
3	00	Null	4320	25	%
4	22	Next line	1	00	Null
5	43	at 4322	2	00	End of
			3	00	program.

Table 1. Hexadecimal codes that are stored when the test program 1 is entered.

*This statement was true when the article was written, but more recent versions of T-BUG do give some entry points for Level II.

taken by the program. The discussion below shows how the program was developed and also provides a more detailed explanation of the program and how it is used.

Storing BASIC Statements

Look at the machine-language (hexadecimal) codes that are stored when a statement is entered either using PEEK or through the M command of T-BUG. The latter was used in this study because most of the documentation for the Z-80 processor is written in hexadecimal. To illustrate this, see the simple program below:

```
10 REM TEST PROGRAM
20 X=2
30 FOR I%=1 TO 4:PRINT I%,
   X↑I%:NEXT I%
```

The Level II manual gives a memory map that shows that programs always begin at location 42E9 (hex, equal to decimal 17129). Using T-BUG, the bytes, in hexadecimal, are shown in Table 1, which also gives the meaning of each byte. We see that special codes are used to represent each of the standard operations, such as 93 for REM, D5 for = and B2 for PRINT. This is how TRS-80 BASIC is able to compress program statements into a smaller space than the actual entry. The format in storage of a line in the BASIC program is also apparent:

1. Each line begins with a two-byte pointer that gives the address at which the next line begins.
2. This pointer is followed by the hexadecimal value of the line number.
3. The line number is followed by a zero byte. When a program is listed, this null byte prints a space, even if no space was entered after the line number.
4. Some of the special codes that are used to represent operations are given in Table 1. A more complete list is shown in Table 2.
5. Variables are encoded as the ASCII values of the name.
6. Constants are just the ASCII characters for the digits.
7. Remarks are stored verbatim as their ASCII codes.
8. Spaces within the line are also stored as ASCII codes.
9. A null byte is inserted at the end of each line, coming just before the next address pointer.

Determining the End Address of a Program

By following the address pointers from one line to the next line, you can discover that two zero bytes occur where an address pointer would otherwise be after the last line of the program. Including the terminating null at the end of the last line, it appears that the end of a program is signaled by three zero bytes.

REM	93
=	D5
FOR	81
TO	BD
PRINT	B2
↑	D1
NEXT	87
LET	8C
IF	82
GOTO	8D
THEN	CA
STEP	CC
STOP	94
END	80

Table 2.

By following through the bytes after the lines of the program, you find that the values of variables are stored in successive locations, with bytes that indicate the kind of variable (integer, single precision, double precision or character string), together with the name of the variable and its value. (String variables are stored as an address pointer to the locations that define them, together with their length.)

Continuing System Execution

Nothing in Table 1 indicates this, but it is reasonable to expect that if the two bytes after the last line were replaced by another address pointer, indicating the start of another line, the execution would naturally continue until two zeros are encountered. This speculation is confirmed by the performance of the program. There is another more subtle issue, however, that is explained below.

Utility Routines Available for Reading a Cassette Tape

The T-BUG booklet gives entry points for subroutines to turn on the cassette and to read or write tapes. These are not necessarily the same as the CLOAD and CSAVE routines, however. Unfortunately, this information is not given for Level II, only for Level I.

In addition to routines in ROM, T-BUG has subroutines to do these chores. However, to find them, you must disassemble T-BUG. This is a tedious chore when done by hand, for T-BUG is about 1000 bytes long. Looking for the routines within ROM, which is about 12K, is even more tedious.

However, there is a clue that allows the task to be shortened. The T-BUG booklet explains that each tape has a leader consisting of 128 zero bytes followed by a hexadecimal A5 that is used for synchronization. If this is so, a loading routine should have, near its beginning, these assembler statements:

```
CP  A5          FE  A5
JR  NZ, displ 20    xx      hex codes
```

If we can locate the string of hexadecimal bytes, FE A5 20, we will have found a candidate for the tape input subroutine. Fortunately, the Z-80 instruction set has a block search command, so the short assembler program in Example 1 ought to find the string.

After the return to the T-BUG monitor produced by the last statement, the register display command will show where the sequence is located. Two such points exist; their hexadecimal addresses are 029D (in ROM) and 464B (in T-BUG). Example 2 shows the program in the ROM area.

By looking at the code for the subroutine at 0241H, you can see that this reads bits from the tape until a one bit is found (this is the "start bit" for a group of eight that comprises a byte of information). To get a full byte, the bit input routine must be called eight times. Hence, we suspect that a set of instructions like this will be in the routine that inputs a full byte:

```
LD B,08         06 08
CALL 0241       CD 02 41        hex codes
DJNZ –5         10 FB
```

The sequence of hex digits 06 08 CD will probably be enough to locate the routine. Sure enough, when the previous search routine was modified, the sequence was found at two locations, one in ROM at 0237 (hex) and the other in T-BUG. The program beginning at hexadecimal location 0235 is shown in Example 3.

Hexadecimal	Assembler language	
21 00 00	LD HL,0000	;begin at start of memory
01 00 00	LD BC,0000	;look forever!
3E FE	LD A,FE	;load A with FE so we
ED B1	CPIR	;can compare and repeat
3E A5	LD A,A5	;see if next byte
BE	CP (HL)	;is A5
20 F7	JR NZ, – 9	;if not, look some more
23	INC HL	;see if
08	DEC BC	;next byte
3E 20	LD A,20	;is 20
BE	CP (HL)	;in hex
20 F0	JR NZ, – 10H	;if not, repeat
CD 80 43	Break Point	;we found it!

Example 1.

Hex address	Hex codes	Assembler code	
0298	CD 41 02	CALL 0241	;subr call
029B	FE A5	CP A5	
029D	20 F9	JR NZ, – 7	;repeat the call
029F	3E 2A	LD A,"*"	;display
02A1	32 3E 3C	LD (3C3E),A	;an*
02A4	32 3F 3C	LD (3C3F),A	;and another
02A7	E1	POP HL	
02A8	C9	RET	;return

Example 2.

The subroutine we have found will enable us to input one byte from tape. Some additional searching, similar to the above, located a routine, entered at location 01FE, to initiate cassette motion.

In summary, the utility routines that are needed to input a program from tape are shown in Example 4. (These are Level II routines. There are corresponding routines within the Level II monitor, but they have different addresses.)

Format on Tape from CLOAD

With the above information about utility routines, it is not hard to write an assembler program to read in a sample program on tape and put it at any desired location in RAM. The one I used is shown in Example 5. It writes the tape, beginning with the first byte after the A5 synchronization code, into successive locations after 4B00.

Hex address	Hex codes	Assembler code
0235	C5	PUSH BC
0236	E5	PUSH HL
0237	06 08	LD B,08
0239	CD 41 02	CALL 0241
023C	10 FB	DJNZ −5
023E	E1	POP HL
023F	C1	POP BC
0240	C9	RET

Example 3.

CALL 01FE	Initiates cassette motion
CALL 0241	Read one bit, shift into A register
CALL 0235	Read one byte, return in A register

Example 4.

```
CD FE 01    CALL 01FE     ;turn on cassette
CD 41 02    CALL 0241     ;get a bit into A
FE A5       CP A5         ;is (A) = A5?
20 F9       JR NZ, -7     ;repeat until it is
21 00 4B    LD HL,4B00    ;point HL to loc 4B00
CD 35 02    CALL 0235     ;get a byte from tape
77          LD (HL),A     ;and store it
23          INC HL        ;increment HL
18 F9       JR -7         ;repeat for next byte
```

Example 5.

This program is an endless loop, of course, but we can stop the action with the reset button on the keyboard.

Using this assembly-language program (translated into hexadecimal by hand, since I don't have an assembler program), I put a sample tape that had a program written by the CLOAD command into memory. The major new information this gave was that four bytes are placed before the program itself. The first three of these are each D3 (in hex); the fourth byte is the ASCII code for the name under which the program was saved.

Following these four bytes come the bytes of the program, each line beginning with the pointer to the address of the start of the succeeding line, and the three successive zeros that indicate the end of the program. A checksum byte follows all this, but that is not of principal concern in this study.

It appears that we now have all the information needed to write the APPEND routine. The strategy to be employed is composed of these steps:

1. Beginning at location 42E9 where the first of next line address pointers is located, jump from one address pointer to the next until two zeros are found. This gives the end point of the program already in memory and the point where the appended portion should be placed.

2. After saving the above address, read in the program to be appended from tape, writing it into successive locations after the root program.

3. Adjust the next line pointers within the appended portion by adding the displacement from 42E9 (where these originally resided) to the new starting location.

4. Transfer back to BASIC by the HALT command (hex code 76), which, in some mysterious way, has been transformed by Radio Shack into a call to the READY prompt.

When I follow this strategy, all seems to go well. The program from tape is properly appended, the address pointers are adjusted to their correct values, the LIST command correctly displays the combination of the root program and the appended part, but when the RUN command is entered, queer results occur. In some cases, gibberish is displayed on the CRT; while other trials go into endless loops.

Requesting a LIST of the program after such unusual behavior shows that one more thing is needed to APPEND a new program. When the program is executed, the variables referred to in the statements of the root program are inserted into memory on top of the appended portion. When these statements are reached during the execution, they have been replaced by the memory representation of variables, and this is certainly not a valid BASIC statement. It is no wonder that the results are weird.

Diagnosing the problem is a big step towards its cure. The system must maintain a pointer to the address where variables begin. If we can locate where that address pointer is stored, it will not be hard to make it point to a new location after the end of the appended portion.

By entering a variety of short BASIC programs into memory and then determining where the variable is stored, it is not difficult to find that the required address pointer is stored by the system at locations 40F9 and

Program listing. Assembly-language program to append a BASIC program on tape to the end of an old one in memory.

```
Location   Hexadecimal    Assembler code

                          ;Each line of BASIC begins with a pointer
                          ;to the beginning of the next line and
                          ;ends with a zero byte.  The first line
                          ;begins at 42E9.  Using these pointers
                          ;in HL, trace through to find the end of
                          ;the present program which is signalled
                          ;by two zero bytes after the last line.
4F80       2A E9 42       LD HL,42E9     ;point HL to first addr ptr.
           3E 00          LD A,00        ;to test for zero bytes.
           BE             CP (HL)        ;does next line begin wth zero?
           28 06          JR Z,06        ;if so, look for second.
4F88       5E             LD E,(HL)      ;not the end, so load
           23             INC HL         ;  the pointer
           56             LD D,(HL)      ;    into DE.
           EB             EX DE,HL       ;and then back to HL.
           18 F5          JR -11         ;repeat for next line.
           23             INC HL         ;is therea second
           BE             CP (HL)        ;  zero byte?
```

40FA. I modified the first version of APPEND accordingly to change the address stored there to the next location after the two zero bytes, which is the normal end of the program indicator.

Using the APPEND Program

The machine-language coding shown in the program listing is loaded into memory. I used the T-BUG program to do this by modifying memory with the M command. It could also be done with the POKE command, but addresses and hex bytes would first have to be converted into decimal. The program is then put onto tape for storage and to make it easy to reload at a later time. (This requires the T-BUG program.)

To add a program that has been put onto tape with the CSAVE command to follow a root program already in memory, follow these steps:

1. Enter the root program in the usual way.

2. Using the SYSTEM command, load the APPEND program. To prevent overwriting by the BASIC monitor, MEMORY SIZE should be set at 20,350 or less.

3. Before initiating execution of APPEND, put the tape holding the program to be appended into the tape player, positioned at the start of the program and readied for input.

4. Initiate execution of the APPEND program by '/ENTER'. The tape recorder will begin to operate, and when the appended portion has been loaded, the READY prompt will appear. You can now LIST or RUN the composite program.

5. To append two or more programs at the end of those already in memory, repeat steps 3 and 4. In this case, execution of APPEND is initiated by this sequence: POKE 16526,128: POKE 16527,79: X = USR(0). (This is the technique used to transfer control to a machine-language program whose entry point is at 4F80 hex.)

If your system has more memory than 4K, you will probably want to locate the APPEND program near the end of it. The only changes that are

```
4F90    2B          DEC HL        ;move HL back.
        20 F5       JR NZ,-11     ;if not two zeros, continue.
                    ;we have now found two zeros at end of
                    ;the root program and its address is
                   ;in HL.  Store this into the stack, then
                    ;load the new material from cassette.
        E5          PUSH HL       ;save end address.
        CD FE 01    CALL 01FE     ;turn on cassette.
4F97    CD 41 02    CALL 0241     ;get a bit, shift it
        FE A5       CP A5         ;  into A and repeat
        20 F9       JR -7         ;    until the sync byte.
        06 04       LD B,04       ;skipping over the
4FA0    CD 35 02    CALL 0235     ;  next four bytes,
        10 FB       DJNZ -5       ;    (three D3, then name).
        E1          POP HL        ;get the start addr for
                                  ;  the new segment.
        E5          PUSH HL       ;but also save for later.
4FA7    CD F0 4F    CALL TRANS    ;transfer one byte.
        A7          AND A         ;see if it was a zero.
        20 FA       JR NZ,-6      ;continue if not.
        CD F0 4F    CALL TRANS    ;one zero found. Transfer
4FB0    A7          AND A         ;  a second, see if zero.
        20 F4       JR NZ,-12     ;continue if not.
        CD F0 4F    CALL TRANS    ;two zeros found,
        A7          AND A         ;  see if three.
        20 EE       JR NZ,-18     ;only two, so continue.
                    ;the new segment is now loaded at the
                    ;end of the old one, but the line pointers
                    ;must be adjusted by adding the length
                    ;of the old program to each of them.
                    ;Compute this length and put into DE.
4FB9    AF          XOR A         ;be sure CY is zero.
        E1          POP HL        ;get loc of end of old
        E5          PUSH HL       ;  program but also save it.
        11 E9 42    LD DE,42E9    ;start addr of old into DE
        ED 52       SBC DE        ;  and subtr from (HL).
4FC1    EB          EX DE,HL      ;put length into DE.
                    ;We now have the correct value to adjust
                    ;the line pointers in DE.  The stack
                    ;holds the location of the first one
                    ;to be adjusted.  Use BC as pointer
                    ;to the next line address to be
                    ;adjusted.  Use HL to add the length
                    ;value to them, then replace each pointer
                    ;with the adjusted value.
        C1          POP BC        ;BC points to the line pointer.
        0A          LD A,(BC)     ;put first byte of
        6F          LD L,A        ;  addr into L and also
        A7          AND A         ;    test to see if a zero.
        03          INC BC        ;bump the pointer in BC
4FC7    20 04       JR NZ,04      ;if first byte not zero,
                                  ;  move on.
        0A          LD A,(BC)     ;we have one zero, If
        A7          AND A         ;  there is a second, we
        28 0C       JR Z,12       ;    are done loading.
        0A          LD A,(BC)     ;continue by putting
        67          LD H,A        ;  second byte of addr
                                  ;    into H.
        19          ADD HL,DE    ;adjust the value.
4FD0    7C          LD A,H        ;put the adjusted
        02          LD (BC),A     ;  next line pointer
        0B          DEC BC        ;    into where BC
        7D          LD A,L        ;      points so that
        02          LD (BC),A     ;        old one is replaced.
        44          LD B,H        ;move addr of next line
        4D          LD C,L        ;  pointer into BC from HL.
4FD7    18 EA       JR -22        ;repeat for next addr ptr.
```

```
                        ;at this point, the new segment is loaded,
                        ;and its address pointers have been
                        ;adjusted.  We still must make a
                       ;correction to the system first variable
                        ;pointer that is located at 40F9.
          03            INC BC          ;bump BC to give loc for
                                        ;  start of variables.
          21 F9 40      LD HL,40F9      ;point HL to system location
          71            LD (HL),C       ;  and store the
          23            INC HL          ;     the address from
          70            LD (HL),B       ;       BC there.
          76            HALT            ;really is a return to BASIC.


                        ;This subroutine is called to transfer
                        ;one byte from the tape to the location
                       ;whose address is contained in HL.
4FF0      CD 35 02      CALL 0235       ;get the next byte.
          77            LD (HL),A       ;store it.
          23            INC HL          ;bump the pointer.
          C9            RET             ;return.
```

needed are to modify the subroutine calls that occur at locations 4FA7, 4FAD and 4FB3. Since only JR transfers are used within the main body of APPEND, it is in relocatable form except for these three subroutine calls.

If you have Level I instead of Level II, the subroutine calls to monitor routines will have to be changed. The program will have to be loaded using T-BUG and control passed to it from that same utility package. While this, in principle, allows you to use the APPEND program with Level I, it would hardly be a practical thing to do because the BASIC program will frequently overlay T-BUG. ■

Move your Machine Code programs around with this BASIC idea.

Relocate with PEEK POKE

Irwin Rappaport
24 Hemlock Hill Road
Upper Saddle River NJ 07458

If you are already well acquainted with machine code and microprocessor instruction sets, please excuse the rest of us during some of the discussion that follows. The BASIC program resulting from this, however, may be of interest to anyone who has had an urge to work on a general method for relocating machine-language programs.

The relocation routine described here can, for example, move Microchess 1.5 so that it will be fully compatible with the TRS-80 mini-disk system. And if you use T-BUG and have not yet moved it to high memory (see "Getting T-BUG HI" *80 Microcomputing*, January, 1980, p. 118), this general address changing system will just about do it for you.

While the procedure refers to the Z-80 and TRS-80, the concept is adaptable to other processors and machines. Try it. After all, it's your computer, and programs should go where you want them to go!

Introduction

In relocating machine-language programs, all instructions containing specific intra-program addresses must be found, and the addresses must be changed to correspond to the new location. For certain, there will be an abundance of jumps, calls and address loads in a commercially available machine-code program. After all, if it was short, simple and direct we would have written it ourselves.

Unless you have a lot of time and patience, the detour addresses are too numerous to search out one by one. Instead, that's where the TRS-80, or any other computer, can be put to work. The only requirement is that the machine have software capable of altering the "object code" under program control. . .in other words, PEEK and POKE capability. Then work can be done in BASIC, so it's understandable.

Since addresses in our microcomputers are contained in two successive bytes, giving us the 64K (actually 65,536) different address possibilities, we can keep "moving day" simpler by dealing with the first digit of the high-order byte only. That is, if a move from the 4000H area to the 7000H area is desired, all we need to do, for example, is redirect a 430A address to 730A, and be done with it. So basically, we will PEEK, evaluate and decide either to POKE a new value or go on to the next PEEK.

To design a routine with this capability, we must become just a little more than casually familiar with the instruction set for our particular microprocessor. Actually, working on this project turned out to be a painless way of getting to know more about the Z-80, and the whole machine-language process in general. It was fun, almost. So get hold of an instruction set and look at it and the PEEKPOKE program listed here as we go through this.

Examining the Instruction Code

First, we will divide the instruction set into groups that are meaningful to the problem at hand.

Input	Hex Value	Decimal Val	TRS-80 Val
Start PEEKing	913F	37183	−28353
Stop PEEKing	9FA1	40865	−24671
High Order Start, Original	40	64	64
High Order End, Original	50	80	80
High Order Start, New	90	144	144

On Screen Results

# of Changes	1 Byte load
684	−24804 (9F1A)

PEEKPOKE specifications.

Group I—a complete instruction contained in one byte; no address or values follow.
Group IIA—a one byte load instruction followed by one byte value.
Group IIB—a one byte instruction followed by a two byte value.
Group III—a one byte non-load instruction followed by a one byte value.
Group IV—a two byte instruction where the first byte is hex CB.
Group V—a two byte instruction where the first byte is hex DD.
Group VI—a two byte instruction where the first byte is hex ED.
Group VII—a two byte instruction where the first byte is hex FD.

The eight groups cover all possibilities for the Z-80. Now we must design a routine that will logically step through a machine-language program, categorize what it finds, decide when a change is needed and make the change. Well, certainly Group I won't be of concern to us, since no values, and therefore no addresses, are associated with those instructions.

Group III contains arithmetic, compare, relative jump instructions and values, as well as port I/O addresses and values. While it is conceivable that some program may go through a convolution wherein the value byte of an instruction in this group is indeed based upon a high-order portion of an address, it is most unlikely. Therefore, we will have our routine ignore all of Group III also. As for Group IV, none of the instruction codes beginning with hex CB have value bytes associated with them, so these manipulators will have nothing to do with our project either.

So far it has been easy. We haven't had to do a thing. But now for the real action, which will not be much more difficult. When our routine comes upon Group IIB instructions, we will have it take a look at the value bytes following.

Since only the high-order address byte is of interest, our routine skips the first value byte and examines the second one. It compares that byte with the intra-program high-order address byte test range (which we will have input into the routine at the start), and if a favorable compare is determined, a new address value will be POKEd in. (Again, we will input the information needed to determine the desired new address.)

The DD's of Group V have to be examined. First, the byte right after DD is checked to determine the nature of the in-

New Address	Contents	Purpose
8C00 to 8FFF	Transferred video screen contents	Instructions for game
9000	21	Addresses
9001	00	9000
9002	8C	to
9003	11	900D
9004	00	are
9005	3C	block
9006	01	transfer
9007	00	to
9008	40	video
9009	ED	screen
900A	AO	
900B	EA	
900C	09	
900D	90	
900E	C3	Jump to
900F	FD	chess
9010	91	entrance
9011 to 913E		Extra space and stack area
913F	Start of chess main body	
91FD	Entrance point in program	
9FAO	End of chess main body	

Microchess relocation map.

✓ Reader Service—see page 131

struction. We can ignore it if the second byte is CB. However, if it is found to be an instruction that is followed by a value, and if the value is not just a one byte indicator to increment the program counter, our routine will check it out just as it did for Group IIB potential address bytes. Otherwise, we will skip it.

Group VI and VII are examined in a similar manner, checking and changing as necessary.

I have saved the discussion of Group IIA for last. Here we have an instruction to load a one byte value, which, we assume is a candidate for an address change. Our routine will check it out and act upon it as necessary. (Note that there is also a one byte loader in the DD's of Group V and in the FD's of Group VII. These are handled by PEEKPOKE as two-byters since there are actually two value bytes, the first one being an indicator.)

One byte loaders need special attention. While we treat them as high-order address bytes, it is easily possible that they have nothing whatever to do with addresses even though they may fit our address range perfectly. So to be safe, our routine will list on screen the location of all one byte load values that have been changed. The listing can be a helpful debugging tool later.

Once an instruction code has been examined, and any necesary change made, the routine must pick up its PEEKing at the first byte of the next instruction. This synchronization with the beginning of each instruction is critical. If the routine begins a read in the wrong place, some very strange new machine code might be produced!

The Program

The PEEKPOKE routine listed here will run on the TRS-80 and, with some minor adaptations, on most any PEEKPOKEable machine. Prior to running, block-transfer the machine code to be relocated so that it is at the desired new place in RAM. Monitor-debugging software, such as T-BUG for the TRS-80, is useful for this. (While a block transfer can be POKEd in, some method will be needed to save the relocated program.)

Upon running, the routine asks for the start and stop PEEKing addresses. These are the beginning and ending addresses of the new location to which you have transferred the program. Also input at this time are the high-order bytes of the starting and ending addresses of the original program, forming the test range referred to earlier. Finally, the high-order byte of the new start address, which the routine uses to calculate address changes, is input.

Run time can take two minutes or more per kilobyte, depending on the number of changes necessary and the nature of the code sequence. While it is working, the routine will print on screen the current location being PEEKed and the number of changes made, in addition to the one byte load-change locations.

At the completion of a run, you will have accomplished in a matter of minutes what would otherwise have taken hours and hours of concentrated work. T-BUG, for example, requires 191 changes, while Microchess takes almost 700 by the PEEK-POKE count. That's a lot of alteration to do by the eyeball method.

Bugs

Unfortunately, even after PEEKPOKEing, we may still have to do some debugging by hand. Most direct addressing will be captured by our routine, but there could be occasional misses caused by certain programming techniques utilized in the original program. Methods used to generate an instruction or its address value are not always as direct as we would like.

A program may, for example, have a value "buried" within it, to be picked up at some point for addressing purposes. Use of buried data in a machine-language program, even when not used for addressing, can cause PEEKPOKE some grief. Should a PEEK mistake data for an instruction, an inadvertent POKE may be made or missed, if

the conditions are right.

With all these potential bugs, why PEEKPOKE? For one thing, hundreds of needed changes are completed for you with no strain. The remaining debugging work, although frustrating at times, is more like puzzle solving than drudgery. And take heart. A lot of debugging may not be necessary at all.

If you move T-BUG with PEEKPOKE, a convenient location for it is 7380H through 7980H. After the run, you will find one of the bugs caused by buried data. The troublesome value is a hex 20 at 7642H in T-BUG HI. It is mistaken for instruction code, and PEEK thinks the next byte is the associated value for the instruction. The routine becomes temporarily unsynchronized, missing needed changes at 7645H and 7648H. Change these two by hand, and T-BUG HI becomes operational.

One might imagine that Microchess, then, would require plenty of debugging. In fact, it takes hardly any at all! A few tricks are needed to operate on it, though, as described below.

By devising and using your PEEKPOKE, you will derive a true and well-deserved feeling of accomplishment, regaining perhaps some lost mastery over your computer. Try it on T-BUG or Microchess, and then some others you may have around. In the process, and almost without realizing it, you may learn a lot more about machine language than you ever thought you would want to know. And you'll love it! Does that mean you have changed? Of course not, unless you are the type whose head gets turned just because you've come upon new knowledge and power.

Putting Microchess on Disk with PEEKPOKE

Microchess may be saved on disk by transferring it to a clear area of RAM and adding a block transfer that will restore it to its original location upon loading. However, in the process of transferring back to low RAM, it destroys some needed system code, causing a hang-up before execution can take place. Therefore, execution via a jump from T-BUG HI is required. While this method may be acceptable, a true relocation by PEEKPOKE will be both practical and personally satisfying.

In any case, it will be necessary to regain control of your computer after loading Microchess in order to be able to work on it. Microchess is loaded in two segments. The first one is a short program that begins at 4FA1H and goes to 4FEFH. This little segment is used to load the Microchess main body.

```
10 CLS:INPUT"START PEEK-ING";A:INPUT"STOP PEEK-ING";A1:INPUT"HIGH ORDER BYTE OF START ADDRESS OF CURRENT PROGRAM ";K1:INPUT"ENDING
ADDRESS , AS ABOVE";K2:INPUT"HIGH ORDER BYTE OF DESIRED NEW START ADDRESS";K3:I=K3-K1
20 CLS:A9=118:PRINT @ 85,"# OF CHANGES";:PRINT @ 48,"1 BYTE GROUP ";:PRINT @ 65,"NOW PEEK-ING";
30 P2=0:P1=PEEK(A):IF P1=0 THEN GOSUB 1000:GOTO 30
35 IF P1=203 OR P1=221 OR P1=237 OR P1=253 THEN 50
40 GOSUB 300:GOSUB 1000:GOTO 30
50 GOSUB 1000
60 P2=PEEK(A)
70 GOSUB 500:GOSUB 1000:GOTO 30
300 REM *** 1 BYTE CODE GROUPS , LINES 300-400 ***
305 REM *** SINGLE BYTE CODE ONLY ***
310 IF P1>=63 AND P1<=193 THEN RETURN
315 REM *** LOOK FOR 1 BYTE CODE & LOAD, JUMP, OR CALL ***
320 IF P1=1 OR P1=17 OR P1=33 OR P1=34 OR P1=42 OR P1=49 OR P1=50 OR P1=58 OR P1=194 OR P1=195 OR P1=196 OR P1=202 OR P1=204 OR
P1=205 OR P1=210 OR P1=212 OR P1=218 OR P1=220 OR P1=226 OR P1=228 OR P1=234 THEN 330
325 IF P1=236 OR P1=242 OR P1=244 OR P1=250 OR P1=252 THEN 330:ELSE GOTO 340
330 GOSUB 2000:RETURN
335 REM *** 1 BYTE CODE FOR 1 BYTE LOAD ***
340 IF P1=6 OR P1=14 OR P1=22 OR P1=30 OR P1=38 OR P1=46 OR P1=54 OR P1=62 THEN 350:ELSE GOTO 360
350 GOSUB 2100:RETURN
355 REM *** 1 BYTE + DISPLACEMENT  -  IGNORE AND START AGAIN ***
360 IF P1=16 OR P1=24 OR P1=32 OR P1=40 OR P1=48 OR P1=56 OR P1=198 OR P1=206 OR P1=211 OR P1=214 OR P1=219 OR P1=222 OR P1=230 OR
P1=238 OR P1=246 OR P1=254 THEN A=A+1:RETURN
395 REM *** DID NOT FIND ANY 1 BYTE CODE - LOOK FOR 2 BYTE ***
400 RETURN
500 REM *** 2 BYTE GROUPS ***
505 REM *** CB GROUPS - IGNORE AND START AGAIN ***
510 IF P1=203 THEN RETURN
520 REM *** CHECK DD GROUPS - LINES 520-580 ***
530 IF P1=221 THEN 540:ELSE GOTO 600
535 REM *** IF DDCB IGNORE, SKIP NEXT 2 ADDRESSES, START AGAIN *
540 IF P2=203 THEN A=A+2:RETURN
545 REM *** DD 2 BYTE OP CODE, NO LOAD OR ADDRESS ***
550 IF P2=9 OR P2=25 OR P2=35 OR P2=41 OR P2=43 OR P2=57 OR P2=225 OR P2=227 OR P2=229 OR P2=233 OR P2=249 THEN RETURN
555 REM *** ADDRESSES FOLLOW THESE ***
560 IF P2=33 OR P2=34 OR P2=42 OR P2=54 THEN 570:ELSE 580
570 GOSUB 2000:RETURN
575 REM *** DEFAULT LINE - MUST BE 2 BYTE OP+1 BYTE IND ***
580 A=A+1:RETURN
600 REM *** ED ***
610 IF P1=237 THEN 620:ELSE GOTO 650
615 REM *** ED OP CODE WITH ADDRESSES TO FOLLOW ***
620 IF P2=67 OR P2=75 OR P2=83 OR P2=91 OR P2=115 OR P2=123 THEN 630:ELSE RETURN
630 GOSUB 2000:RETURN
650 REM *** FD ***
665 REM *** FDCB - SKIP 2 ADDRESSES  AND START AGAIN ***
670 IF P2=203 THEN A=A+2:RETURN
675 REM *** 2 BYTE FD OP CODE - NO ADDRESS ***
680 IF P2=9 OR P2=25 OR P2=35 OR P2=41 OR P2=43 OR P2=57 OR P2=225 OR P2=227 OR P2=229 OR P2=233 OR P2=249 THEN RETURN
685 REM *** ADDRESSES WILL FOLLOW ***
690 IF P2=33 OR P2=34 OR P2=42 OR P2=54 THEN 700:ELSE 710
700 GOSUB 2000:RETURN
705 REM *** FD DEFAULT - 2 BYTE OP+1 BYTE IND ***
710 A=A+1:RETURN
745 REM *** SOMETHING WRONG BUT WILL CONTINUE ANYWAY! ***
1000 REM *** UPDATE ADDRESS AND REPORT WHERE AT ***
1010 A=A+1:IF A>A1 THEN PRINT @ 644,"CHANGES COMPLETE";:END:ELSE PRINT @ 129,A;:RETURN
1500 REM *** UPDATE AND REPORT NUMBER OF CHANGES ***
1510 C=C+1:PRINT @ 154,C;:RETURN
1700 REM *** REPORT ADDRESSES OF 1 BYTE LOAD CHANGES ***
1710 PRINT @ A9,A;:A9=A9+64:RETURN
2000 REM *** CHECK AND CHANGE ADDRESS IF IN RANGE ***
2010 A=A+1:GOSUB 1000:P=PEEK(A):IF P<K2+1 AND P>K1-1 THEN 2020:ELSE RETURN
2020 POKE A,P+I:GOSUB 1500
2025 REM *** THERE IS 1 DD & 1 FD WITH A 1 BYTE LOAD (DD&FD36) *
2030 IF P2=54 THEN GOSUB 1700
2040 RETURN
2100 REM *** 1 BYTE LOAD , CHECK AND CHANGE ***
2110 GOSUB 1000:P=PEEK(A):IF P<K2+1 AND P>K1-1 THEN 2120:ELSE RETURN
2120 POKE A,P+I:GOSUB 1500:GOSUB 1700:RETURN
```

PEEKPOKE program.

Start by loading in both T-BUG HI and the short segment of Microchess. Reenter T-BUG HI and examine addresses 4FED, 4FEE and 4FEF. You will find a jump instruction to 41FD, which is the entrance point in the Microchess main body. Enter a breakpoint at 4FED and then begin execution of the short program at 4FA1. The main body will begin to load, and, upon completion, control will be regained via the breakpoint. A portion of the game instructions will be wiped away when T-BUG HI reenters, but you can reconstruct them on the right side of the screen by loading ASCII directly into the appropriate video addresses.

Now you can block-transfer the whole section, from the beginning of the video screen to the end of the chess main body (3C00 through 4FA0), to the location that you have chosen for the new home of Microchess, or HICHESS, as we'll call it. (A convenient place for the transfer routine is 6000H.)

Let's assume that your HICHESS will begin with the video screen contents at 8C00. At 9000H add, as a permanent part of your HICHESS program, a block transfer similar to the one just used at 6000H, but with reversed addressing so that it sends the video screen ASCII portion of your program back to the screen when HICHESS is loaded. You will therefore be arranging to send 400H (1024 decimal) bytes from 8C00 to 3C00.

Immediately following this block transfer, add an instruction to jump to the HICHESS main body entrance point. Since you are moving Microchess 20K higher in RAM, the entrance point will be 91FD and the instruction will be C3 FD 91. If you are wondering about the breakpoint we put in before, it is not necessary to remove it since that portion of the program will no longer be needed or executed. HICHESS will load in one operation.

With the program now positioned where you want it, go into BASIC and load PEEKPOKE. Have it begin PEEKing at the starting address of the main body, which is 913F, and stopping at 9FA1. Input the high-order starting bytes information as if the move was from 4000H to 9000H to catch the stack pointer change of address, which is in the neighborhood of 4064H in Microchess.

On the TRS-80, all values entered from BASIC must be in decimal, and check your Level II manual for special instructions on PEEK/POKE values above 32767. Get a cup of coffee or something, because the run will take 10 minutes or so.

Now take a look at 9EBE, 9F1A, 9F20 and 9F29. The contents of these locations have been changed by PEEKPOKE because they seem to be high-order address bytes, according to the code immediately preceding. However, the "environment" around these locations suggests that this is an area of ASCII code/program data rather than executable code. So change these four locations back to their original values.

I confess that I cannot find any difference in the game whether these four spots are left high or changed back low. That may be a reflection of two weaknesses: my chess ability and my understanding of the program. Nevertheless, it works.

Save HICHESS starting at the beginning of the transferred video screen contents (8C00). Define the entrance point as 9000H, the beginning of the block transfer that sends the ASCII to the screen.

When you load HICHESS, the instruction will come on screen, and pressing ENTER brings the chessboard in as usual. All regular Microchess instructions still apply.

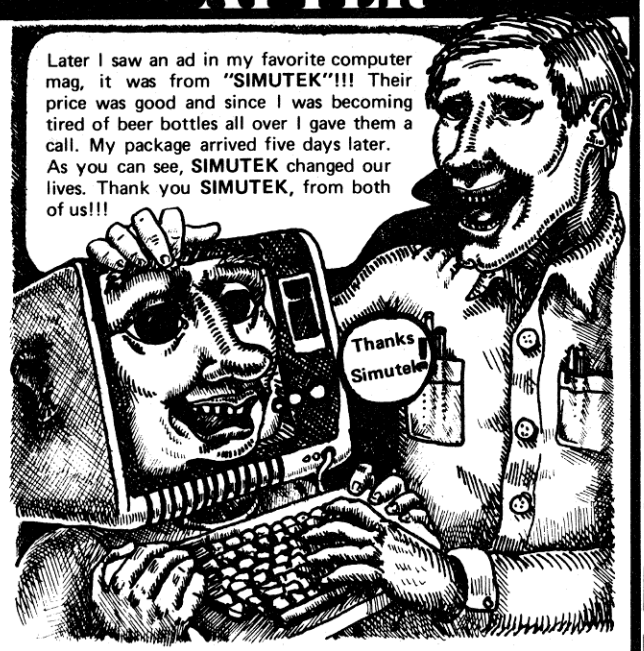
It should be noted that although we have, for personal convenience, relocated and renamed this fine game, it remains Microchess 1.5, and is protected by copyright. ■

"A TRUE STORY"

My **TRS-80** was becoming a degenerate. He was upset because of the tasteless, (although expensive), software I had been feeding him. When he started smoking and drinking I put my foot down. "What do you want from me?," I screamed in desperation. **"SIMUTEK"** he said in his usual monotone, over and over again. "What was **SIMUTEK**"? I wondered.

BEFORE

AFTER



Later I saw an ad in my favorite computer mag, it was from **"SIMUTEK"!!!** Their price was good and since I was becoming tired of beer bottles all over I gave them a call. My package arrived five days later. As you can see, **SIMUTEK** changed our lives. Thank you **SIMUTEK**, from both of us!!!

SIMUTEK SOFTWARE

Exceptional Software thru Research & Imagination

SIMUTEK SIMUTEK SIMUTEK

SIMUTEK PRESENTS

★ TRS-80 ★

.....GAMES.....

!!! WHOLESALE !!!

★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ PACKAGE ONE ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★

GRAPHIC-TREK "2000" — This full graphics, real time game is full of fast, exciting action! Exploding photon torpedoes and phasers fill the screen! You must actually navigate the enterprise to dock with the giant space stations as well as to avoid klingon torpedoes! Has shields, galactic memory readout, damage reports, long range sensors, etc! Has 3 levels for beginning, average, or expert players! ★ **INVASION WORG** — Time: 3099, Place: Earth's Solar System Mission: As general of Earth's forces, your job is to stop the Worg invasion and destroy their outposts on Mars, Venus, Saturn, Neptune, etc! Earth's Forces: Androids — Space Fighters — Lazer Cannon — Neutrino Blasters! Worg Forces: Robots — Saucers — Disintegrators — Proton Destroyers! Multi level game lets you advance to a more complicated game as you get better! ★ **STAR WARS** — Manuever your space fighter deep into the nucleus of the Death Star! Drop your bomb, then escape via the only exit. This graphics game is really fun! May the Force be with you! ★ **SPACE TARGET** — Shoot at enemy Ships with your missiles. If they eject in a parachute, capture them — or if you're cruel, destroy them! Full graphics, real time game! ★ **SAUCERS** — This fast action graphics game has a time limit! Can you be the commander to win the distinguished cross! Requires split second timing to win! Watch out!

★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ PACKAGE TWO ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★

CHECKERS 2.1 — Finally! A checkers program that will challenge everyone! Expert as well as amateur! Uses 3-ply tree search to find best possible move. Picks randomly between equal moves to assure you of never having identical games. ★ **POKER FACE** — The computer uses psychology as well as logic to try and beat you at poker. Cards are displayed using TRS-80's full graphics. Computer raises, calls, and sometimes even folds! Great practice for your Saturday night poker match! (Plays 5 card draw). ★ **PSYCHIC** — Tell the computer a little about yourself and he'll predict things about you, you won't believe! A real mind bender! Great amusement for parties. ★ **TANGLE MANIA** — Try and force your opponent into an immobile position. But watch out, they're doing the same to you! This graphics game is for 2 people and has been used to end stupid arguments. (And occasionally starts them!) ★ **WORD SCRAMBLE** — This game is for two or more people. One person inputs a word to the computer while the others look away. The computer scrambles the word, then keeps track of wrong guesses.

★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ PACKAGE THREE ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★

POETRY — This program lets you choose the subject as well as the mood of the poem you want. You give TRS-80 certain nouns or names, then the mood, and it does the rest! It has a 1000-word + vocabulary of nouns, verbs, adjectives and adverbs! ★ **ELECTRIC ARTIST** — Manual: draw, erase, move as well as, Auto: draw, erase and move. Uses graphics bits not bytes. Saves drawing on tape or disk! ★ **GALACTIC BATTLE** — The Swineus enemy have long range phasers but cannot travel at warp speed! You can, but only have short range phasers! Can you blitzkrieg the enemy without getting destroyed! Full graphics — real time! ★ **WORD MANIA** — Can you guess the computer's words using your human intuitive and logical abilities? You'll need to, to beat the computer! ★ **AIR COMMAND** — Battle the Kamikaze pilots. Requires split second timing. This is a FAST action arcade game.

★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ PACKAGE FOUR ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★

LIFE — This Z-80 machine language program uses full graphics! Over 100 generations per minute make it truly animated! You make your starting pattern, the computer does the rest! Program can be stopped and changes made! Watch it grow! ★ **SPACE LANDER** — This full graphics simulator lets you pick what planet, asteroid or moon you wish to land on! Has 3 skill levels that make it fun for everyone. ★ **GREED II** — Multi-level game is fun and challenging! Beat the computer at this dice game using your knowledge of odds and luck! Computer keeps track of his winnings and yours. Quick fast action. This game is not easy! ★ **THE PHARAOH** — Rule the ancient city of Alexandria! Buy or sell land. Keep your people from revolting! Stop the rampaging rats. Requires a true political personality to become good! ★ **ROBOT HUNTER** — A group of renegade robots have escaped and are spotted in an old ghost town on Mars! Your job as "Robot Hunter" is to destroy the pirate machines before they kill any more settlers! Exciting! Challenging! Full graphics!

★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ PACKAGE FIVE ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★

SUPER HORSERACE — Make your bets just like at the real racetrack! 8 horses race in this spectacular graphic display! Up to 9 people can play! Uses real odds but has that element of chance you see in real life! Keeps track of everyone's winnings and losses. This is one of the few computer simulations that can actually get a room of people cheering! ★ **MAZE MOUSE** — The mouse with a mind! The computer generates random mazes of whatever size you specify, then searches for a way out! The second time, he'll always go fastest route! A true display of artificial intelligence! Full graphics, mazes & mouses! ★ **AMOEBA KILLER** — You command a one man submarine that has been shrunken to the size of bacteria in this exciting graphic adventure! Injected into the president's bloodstream, your mission is to destroy the deadly amoeba infection ravaging his body! ★ **LOGIC** — This popular game is based on Mastermind but utilizes tactics that make it more exciting and challenging — has 2 levels of play to make it fun for everyone. ★ **SUBMARINER** — Shoot torpedoes at the enemy ships to get points. Fast action graphics, arcade type game is exciting and fun for everybody!

★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ PACKAGE SIX ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★

20 HOME FINANCIAL PROGRAMS — Figures amortization, annuities, depreciation rates, interest tables, earned interest on savings and much, much more. These programs will get used again and again. A must for the conscientious, inflation minded person.

★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ PACKAGE SEVEN ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★

BACKGAMMON 5.0 — 2 different skill levels make this game a challenge to average or advanced players. (Not recommended for beginners). Looks for best possible move to beat you! FANTASTIC GRAPHICS. Plays doubles and uses international rules. ★ **SPEED READING** — Increases your reading speed. Also checks for comprehension of material. Great for teenagers and adults to improve reading skills. ★ **PT 109** — Drop depth charges on moving subs. Lower depths get higher points in this fast action graphics game. ★ **YAHTZEE** — Play Yahtzee with the computer. This popular game is even more fun and challenging against a TRS-80! ★ **WALL STREET** — Can you turn your $50,000 into a million dollars? That's the object of this great game. Simulates an actual stock market!

NOT AVAILABLE AT RETAIL STORES ANYWHERE

INSTRUCTION BOOK WITH EACH PKG.

ONLY 12.95 EACH!!!!

ALL PROGRAMS GUARANTEED TO LOAD

CASSETTE PACKAGES REQUIRE 16K LEVEL II

PACKAGES ON DISKETTE (32K) $5.00 EXTRA

Send check, Money Order or Bank Card #

TO: SIMUTEK, P.O. Box M35298
Tucson, Arizona 85740
(602) 882-3948

✓19

PHONE ORDERS WELCOME!

PLEASE ADD $2.50 POSTAGE & HANDLING PER ORDER

3 OR MORE PACKAGES GET 10% DISCOUNT

✓ Reader Service—see page 131

This circuit will allow you to interface switches to your system. One chip & eight resistors is all it takes.

A Simple Interface

Arthur T. Mullin, Jr.
Rt. 3, Box C-9
Beaumont TX 77706

Want a simple device that will interface switches to your TRS-80 Level II machine? Fig. 1 shows you just how simple it is—one chip and eight resistors. What kind of switches? The choice is yours; I have two cadmium sulfide cells—one inside, the other outside.

Now my computer knows when it's daylight and when the room light is on. There's a magnet glued to the inside bottom of the door to this room and a magnetic reed switch attached to the door sill. Now the system knows if the door has been opened, and the DOS activated clock tells me when. That's three inputs with five more to go. My next step is to connect an analog to digital converter.

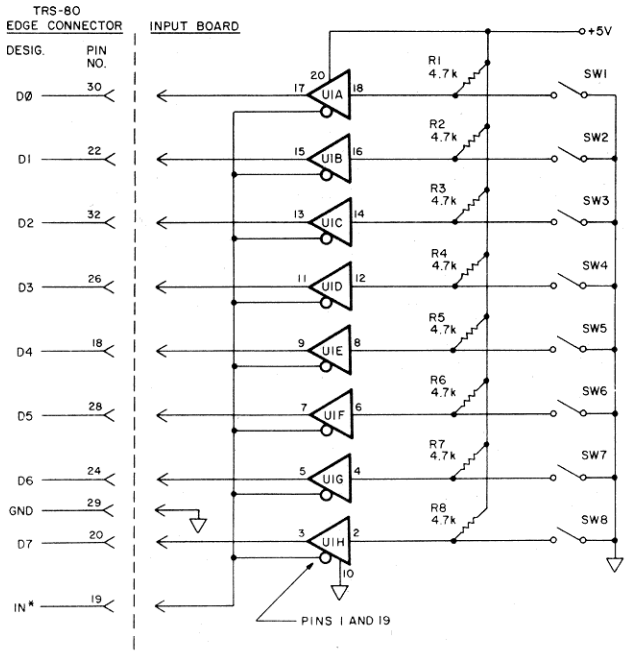


Fig. 1. Full schematic for input interface for the TRS-80 Level II.

The Circuit

Fig. 2 explains the operation of the circuit using just one section of the chip. The rest work exactly the same. The schematics are oriented to the TRS-80, and the part numbers are Radio Shack numbers.

Some of you will recognize Fig. 2 as a Tri-state non-inverting buffer. For those of us who are not strong on the subject, the key is the pin marked "1&19" (refer to data sheet for details on the 81LS95). If this pin is high (close to +5 volts), the computer will be totally unaware that U1A even exists. If this pin is low (close to zero volts), the logic at pin 18 will appear at pin 17.

The line marked IN* is normally kept high, and goes low only when an INP command is executed. The INP command calls for an address, doesn't it? Don't worry. Ask for any address within the 0 to 254 limitation and IN* will still go low. And that's all we require to sense the switch at pin 18. Since the computer is trying to read whatever is on the data line at this time, it will read the condition of all the switches in one fell swoop.

If a switch is open, the pull-up resistor associated with its particular buffer will cause a "one" to be transferred into the computer. If the switch is closed, the +5 volts that was previously on the input to its buffer will be shorted to ground and a "zero" will be transferred to the computer.

So it becomes apparent that upon request by the user to " PRINT INP(100) ", a number reflecting the condition of all the switches will be printed on the monitor. If the number eight is printed on the monitor, what does it mean? Using basic binary, if the number is eight, then all switches must be closed except the one associated with U1D (Fig. 1).

Although this method of inputting the outside world to the computer is primitive, it will enable those of us who are interested to gain first-hand experience with both hardware and software applications. ■

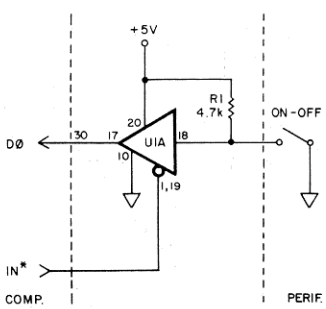


Fig. 2. Partial schematic.

✓ Reader Service—see page 131

WE MEAN BUSINESS!

BUSINESS SOFTWARE, THAT IS

USE YOUR TRS-80* FOR MORE THAN FUN AND GAMES

THE DATA DUBBER $49.95

Duplicates any program tape to TRS-80 quality. Reconstructs date pulses to ensure accurate CLOADs. Permits easy loading of even poor quality commercial tapes without constant volume adjusting. Money-back guarantee if not satisfied.

THE ELECTRIC SECRETARY $75.00

A powerful word processor to turn your TRS-80 into an automatic typewriter. Features page numbering, movable margins, headers, variable page length, and title centering. Enter text, revise, correct, and output to printer page formatted, justified, even hyphenated as required. Cross-coupling files permits individually addressed form letters. Complete with upper/lower case conversion information on diskette. Specify if RS-232 adapter is installed in interface.

MAILROOM PLUS $75.00

A versatile and powerful mailing program to print labels by sequential coding: zip, city, state, customer ID code, even last name. Sorts by any code in minutes and stores sequentially in a single string (approx. 1500 records per diskette). Includes AUTOPRINT. Supplied on diskette.

MINIMAIL $50.00

A compact version of MAILROOM PLUS but without customer coding. Features alpha-lookahead for duplicates. Supplied on diskette.

FORMLET $35.00

Generates form letters from MINIMAIL records. Prepare your letter, bulletin, notice, advertisement, etc., then load the MINIMAIL files. Your printer will print the inside address, letter, and repeat for each name in the file—all properly spaced and justified. Supplied on cassette.

AUTOBOOT $15.00

Simplifies automatic BASIC program loading from your DOS. Permits sequencing through your choice of DOS commands, selects files and memory size you specify, and loads or runs selected program. Allows user to see directory and free space before program runs automatically. Supplied on cassette.

SIR ECHO $10.00

A handy program to make your printer work like an electric typewriter. Use alone or merge with your programs to make what appears on the screen echo to the printer. Supplied on cassette.

TELEFON $20.00

Make your TRS-80 a smart terminal. Communicate with time-share and other computers, bulletin boards, etc. Transfer programs over the phone. For disk systems with modem.

UPPER/LOWER CASE CONVERSION $20.00

Reprint of KILOBAUD article explaining how to modify the TRS-80 to display both upper and lower case characters. Kit contains step-by-step instructions, parts, and necessary software on cassette for case reversal, echo, and automatic line feed routines.

User group discounts available.
Dealer inquiries invited.

*TRS-80 is a trademark of the Tandy Corp.

VISA

TERMS: Check, money order, Visa, Mastercharge. Washington residents add 5.3% for tax.

master charge

THE PERIPHERAL PEOPLE

P.O. Box 524, Dep't. M
Mercer Island, WA 98040

✓43 (206) 232-4505

THE SMART TERMINAL

SOFTWARE PACKAGE FOR THE

TRS-80

"TRS-80 is a registered trademark of TANDY CORP."

- TRUE BREAK KEY
- AUTO REPEAT KEYS
- PROGRAMMABLE 'SOFT' KEYS
- MULTIPAGE SCROLLING DISPLAY
- FULLY CONFIGURED FROM KEYBOARD
- TWO SELECTABLE LINE DIAGNOSTIC MODES DISPLAY CONTROL CHARACTERS
- MULTI PROTOCOL CAPABILITY
- GRAPHIC CHARACTERS GENERATED FROM KEYBOARD
- TRANSMIT SCREEN, PRINT SCREEN
- TRANSMIT FILE, RECEIVE & CREATE DISK FILE
- FLEXIBLE I/O LINKAGE CAPABILITY
- DISPLAY AND KEYBOARD DRIVERS CAN BE USED UNDER DOS

● $79.95

✓73

MICRON, INC.
10045 Waterford Dr.
Ellicott City, MD 21043
(301) 461-2721
MC/VISA accepted

Now Available: SPOOL-80
•PRINT YOUR LISTINGS WHILE RUNNING OTHER PROGRAMS. (DISK SYSTEMS ONLY.)
$39.95

QUALITY TRS-80 SOFTWARE

KEYWORD Indexing System

A series of programs that will create a data file on disc, build an index of all occurrences of "Keywords" in the text of the data file and allow inquiries or searches into the file using the indexed keywords. The system features:

- •flexible record lengths with location pointers
- •deletion of non keywords from index by system
- •"and" "or" "not" logic for inquiries
- •interface for user written inquiries

KEYWORD INDEX—2 disc 32K DOS system **$39.95**

SORTS for HOME and BUSINESS

No computer user should be without a versatile, easy to use sort program. The Northeast Microware in memory sort programs are written in Level II BASIC and have the following features:

- •Sort ALPHA or NUMERIC data
- •Sort on up to 5 fields simultaneously
- •In ascending or descending sequence
- •Supports kb, video or tape I/O
- •Supports seq. disk and printer I/O (SORT-IID)
- •Supports user I/O routines
- •User exits (SORT-IID only)

SORT-II—16K Level II in memory sort **$19.95**
SORT-IID—32K DOS in memory sort **$29.95**

FOR the SERIOUS GAMBLER

BLACKJACK SIMULATOR. Allows you to simulate the playing of thousands of hands of BJ and analyze the results on tape in Level II BASIC. **$19.95**

Manuals for all programs available for $3.00 ea.
(price deductible on purchase of program)

"TRS-80 is a registered trademark of TANDY CORP."

Northeast MICROWARE
BOX 6153, ✓74
SYRACUSE, N.Y. 13217

With these disks, I can turn your TRS-80 into a serious computer.

I'm Irwin Taranto, and I've put the first computer into more than 300 different businesses.

It's taught me that the TRS-80 is an elegant piece of hardware despite its low price. Given the right programs, it can jump through hoops.

Put simply, I have the right programs. Four of them are the genuine Osborne & Associates systems, originally designed for the $30,000 Wang computer. I've made a few minor modifications, and they now work on a $4000 TRS-80. The other two programs I added myself.

These programs are fully-documented, and you can buy the books locally or from me. I made them work on the TRS-80, and if you buy them from me, I'll make them work for you.

If you're not sure about that, call the number below and get the names of some of the people who've bought all over the world. Then ask them.

These programs cost $99.95 each. (The Cash Journal option on the General Ledger adds another $50.) That gets you the disk, all the instructions you need and my phone number. If you call, we answer all your questions. If your question's tough enough, I'll talk to you personally.

Because I plan to turn that TRS-80 of yours into a serious computer.

THE OSBORNE PROGRAMS

Accounts Payable: invoice-linked, it reports, does checks and links to general ledger.
Accounts Receivable: invoice-linked, it tracks invoices and aging, prints statement and links to general ledger.
General Ledger: handles 1750 transactions each on 200 different accounts. Cash journal option available.
Payroll: figures the pay, does the checks and all the bookkeeping.

AND MY OWN PROGRAMS

Inventory Control: gives immediate readout on any inquiry. It has many existing versions or can be individually tailored.
NEW! Invoicing: linked to accounts receivable. Prints invoices and feeds data into receivables.

✓45 **Taranto** & ASSOCIATES

P.O. Box 6073, 4136 Redwood Highway, San Rafael CA 94903 · (415) 472-1415. Add $3 per order for handling, 6% sales tax in California only. If you don't already have the books, add $15 each (invoicing book, $10). Mastercharge, Visa OK.

Four different games from software houses, checked out for your entertainment.

Games Review

Rod Hallen
Road Runner Ranch
PO Box 73
Tombstone AZ 85638

All of the software reviews that I've written up until now have pertained to applications programs. I haven't done anything with games, but I recently came across some that you might be interested in. Keep in mind that these are my personal opinions, which reflect my view of each of the games covered. You might feel completely different about them.

Win21

Win21 is a Blackjack program from Discovery Bay Software Co., PO Box 464, Port Townsend WA 98368. It sells for $29.

Before you dismiss this as just another of the many Blackjack games available, read on a little further. While Win21 can be played as a game pitting you (and up to six of your friends) against the dealer, there is much more to it than that. It is intended to be used as a tutorial in the art of winning at a real Blackjack table.

Included with Win21 is a 200 page paperback book by Edward O. Thorp, PH.D., titled "Beat The Dealer." This book presents a theory and system for regularly winning at Blackjack. One chapter describes how the author, with the help of two wealthy friends, tested his system in Nevada and won $11,000. He also discusses the problems that he ran into as a winner, such as dealers who cheat and casinos that wouldn't let him play.

Win21 is designed to be used along with the book to improve your card skills. It allows you to set up the rules in any manner you like in order to simulate 21 as it is played in various casinos. The casinos most often mentioned are in Nevada, Puerto Rico and London.

You can also decide whether you will make your own moves, follow a computer-suggested course of action or let the computer play for you. When you play your own cards, a grade is kept. This is in addition to the scoring according to the amount of money you've won or lost.

You also have a choice of playing with from one to four decks. If you choose more than one deck, the shuffle time becomes quite long. This is one of the drawbacks to writing complex programs in BASIC.

The graphics employed are not dramatic, but they are effective. An outlined square is provided in front of each player for his bet, and then the cards are dealt face down or face up as required. One interesting feature is used when face-down cards are turned up. You first see the back, then the edge and then the face. The cards actually look like they are being turned over!

It is claimed that the theory is sound and that you can win if you follow it exactly. I haven't tried it in Las Vegas and probably won't. If I set the program to play automatically and let it run for long periods of time, I always lose. The losses are small but steady. Right at the moment it has been running for three hours, playing seven hands for me, and I've lost $2690 betting $10 chips.

It is a very interesting program, and anyone who plays Blackjack for money might get some benefit out of it. In any case, it's fun to play.

Gomoku

This is another program from Discovery Bay Software; it sells for $15. Gomoku, which is an ancient Japanese game, is a variation of Tictactoe. It is played on a 9 × 9 multi grid, and the object is to place five of your markers in a row, either horizontally, vertically or diagonally.

I have little use for computerized Tictactoe or for the paper and pencil version either. Once you have discovered the key to the game, the results are predictable. Such is not the case with Gomoku . . . at least up until now. It is obvious that the computer has a strategy, and it is even becoming clear to me what that strategy is, but I have yet to win a game. I am sure that I will in time.

I do not like games based completely on chance or games that are easy to beat. I judge a game program, or any game for that matter, by the way that I feel when the game has ended. If I immediately want to play another round and find the time slipping away as I rise to the challenge again and again, then I rate it high. If I play it a few times, slide the cassette back into its box and seldom take it out again, then it fails the test.

Gomoku is the kind of game I could play for hours. In fact, I have to force myself to stop. During the first few games I was beaten quickly and soundly, but after that I began to see how the computer set me up each time, and I managed to avoid the more obvious traps. Now the board becomes more crowded and there are more traps to look out for. I'm getting better but not good enough yet.

The computer is rather slow in determining its moves, but that is normal for a program written in BASIC that has to search a large number of possibilities for the correct position. The graphic layout of the board and the area where you and the computer enter your moves is well done. The computer even makes such intimidating remarks as: Tsk! Tsk!, Uh-huh, Chuckle and many more during play.

As I've said, I rate a game program on how well it holds my interest and whether it makes me want to play again. Based on that criteria, Gomoku rates very high.

Video Checkers

This is a product of Compu-

quote, 6914 Berquist Avenue, Canoga Park CA 91307. It sells for $14.95, which includes a complete BASIC listing.

The vendor claims that Video Checkers plays a legal game of checkers according to the international rules of the game. I can't dispute that claim, but that doesn't mean that it plays an intelligent game.

The graphics used to display the playing surface and the checker pieces are very well done. It is always easy to tell at a glance what the game situation is. I particularly like the fact that the board is displayed continually, as opposed to some programs in which the screen is scrolled each time that the board is updated. There is an area on the right side of the screen where the computer indicates whose turn it is and also prints each move as it is made. Messages such as Illegal Move, I win!, You win!, etc., are displayed there also.

There is one bug in the program that shows up occasionally. After the computer has printed and made a move, it will sometimes leave a letter (usually M or J) in the space where you enter your move. This results in an illegal move message one or more times before the letter is removed, and you can continue play. This is a minor bug because it does not affect the game. It is better than a bug that destroys a game after you have been playing for a long period of time.

Time is one thing that the computer seems to have plenty of. It takes a long time (40 seconds) to determine a move. This might not seem like a long time, but it is when you are sitting, staring at the screen. This time doesn't seem to vary even when the computer only has one move on the entire board. It does think a little faster when it is faced with a jump.

I mentioned earlier that a legal game is not the same thing as an intelligent game. While Video Checkers appears at times to be playing a smart game, it invariably makes stupid mistakes, and I've never lost to it yet. In one instance, it had two pieces ready to move in and be kinged; instead it moved in front of me and set up a double jump. Since the program is written in BASIC, there is an opportunity here for you to use your programming skills to try to improve its game.

The documentation that comes with Video Checkers admits that it does not play a clever game. I've yet to find anyone, including my kids, who can't beat it every time.

Android NIM

Available from 80-NW Publishing Co., PO Box 7112, Tacoma WA 98407, Android NIM sells for $8 on cassette and $13 on disk.

This is the only tape program that I had trouble loading. It is recorded on both sides of the cassette, and both recordings sounded very sick with a lot of quite audible speed variations. After my fifth try, I finally got a good load and immediately CSAVEd it back on the tape. After that, CLOADing was no longer a problem. I also do this with any program that does not load at my standard volume setting.

NIM is an old game in which the players are presented with three piles of objects. Each player in turn can remove as many objects as he likes from any one pile. The player who removes the last object wins.

Android NIM is played by the standard rules, except that it uses androids (robots) as the objects. At the start of the game, three rows of androids are displayed along with an executioner for each row. When either you or the computer indicates the row and number of objects to be removed, the executioner looks at the move, looks down his row, counts the androids left in that row and shakes his head either yes or no. If it's no, then you will have to change the number of objects to be removed.

If it is yes, the executioner looks down his row again, raises his phasor, and each android in the row turns his head toward him. He then blasts the required number of androids into stardust, retires his pistol and waits for the next move.

Android NIM is one of those games that you can win every time, once you figure out the strategy. However, it is well worth the price just for the graphics, which are the most imaginative that I have seen for the TRS-80. Not only do the executioners shake their heads yes or no, but all of the androids' arms, heads, eyes and ears are constantly in motion. This is one game program that every TRS-80 owner should have just for demonstration purposes, even if he is not interested in playing it.

Conclusion

All of these games come with adequate documentation. Each is also provided with a sturdy cassette storage box to help keep it clean. This is important because minute particles of dirt can cause data bit errors that will prevent you from loading properly.

A great deal of programming effort obviously went into each of the above games, and each has something to offer the computer hobbyist. I hope that I have provided you with enough information so that you can make up your own mind. ■

Hard copy for under $300 can be done, find the right combination and LPRINT away.

LPRINT "CHEAP"

Fred Blechman
7217 Bernadine Ave
Canoga Park CA 91307

Once I had gotten my TRS-80 up and running, I started thinking about a printer. Using the Radio Shack line printer required buying an expansion interface (about $300), plus the cost of the printer—a total of over $1300!

It had only cost me about $700 for my Level II 16K model (I added the memory myself), so it seemed ridiculous to spend almost *twice* that amount for an accessory.

So I asked questions (a procedure I recommend for those, like me, who are wandering aimlessly in the microcomputer world) and became more confused! Buzzwords like serial, parallel and interface were sprinkled liberally throughout my conversations. But I came to realize that the TRS-80 expansion interface required a parallel printer, while some of the best new printers—and many used printers—required a serial interface.

TRS232 Printer Interface

About that time, Small System Software and Hardware, Newbury Park, CA, came out with their TRS232 Printer Interface ($49.95 + $2 shipping costs), a small box that plugs into the TRS-80 and printer cables and drives any RS-232 (that's the computer code word for serial) printer. The TRS232 had its own power supply, so it was not necessary to monkey around with computer innards, or to hang on some external supply.

Well, this got me looking around for a serial printer. I wanted a cheap, fast, printer that produced at least 64 characters per line. I spotted an ad for a refurbished Scope Data Series 200 Printer that uses electrosensitive 8½" wide paper, and prints 85 upper and lowercase 7 × 9 dot matrix characters on a line at 120 characters per second with ASCII serial input. Even though I knew nothing about printers, the $200 price tag was so attractive I had it shipped from Florida.

I bought a TRS232 Printer Interface and connected it as shown in Fig. 1. The keyboard cable that normally goes to the cassette recorder AUX input, plugs into one TRS232 jack, and a cable, with miniature phone plugs on each end, connects to the other TRS232 jack and the recorder AUX input. A cable with a DB-25 male connector at each end goes from the TRS232 socket to the printer socket.

The printer cable for RS-232 use might contain as little as two wires—pin 3 for data transmission and pin 7 for signal ground. If pin 3 doesn't work, try pin 2 at the printer end. However, many printers require that pins 5, 6 or 8 also be connected. It doesn't hurt to connect all these wires between the DB-25 plugs at each end. Also, as you'll see later, you might want to include a wire between pin 20 of the interface and pin 20 of the printer. To drive a 20-mil current loop for operating Model-33 Teletypes or other similar terminals, pin 10 is the positive pin, and pin 9 is the return. The documentation that comes with the TRS232 is very complete, and even shows the schematic.

A cassette tape is included. After reserving memory space CLOAD the BASIC tape. The 1573 byte program loads in less than 30 seconds, when RUN pokes a 110 byte machine language program into high memory. You then load or enter the program you want printed. The software routine is used to generate the serial data stream through the TRS-80 cassette port and out the cable that normally goes to the cassette AUX input. The data stream voltage levels are converted to RS-232 compatible voltage levels (plus or minus 9 volts) by the TRS232 electronics.

The software routines converting parallel data to serial are thus the key to TRS232 operation, with timing loops used to

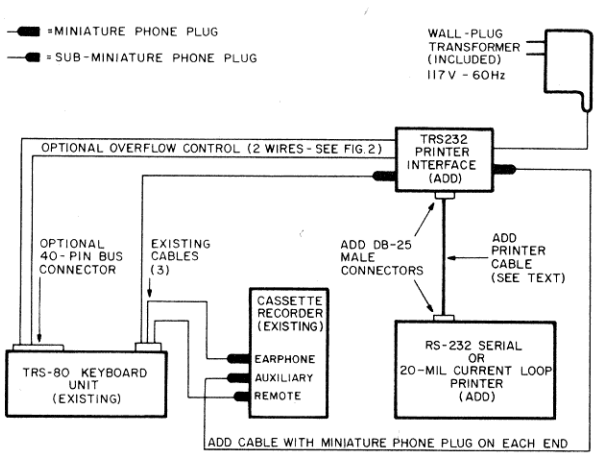


Fig. 1

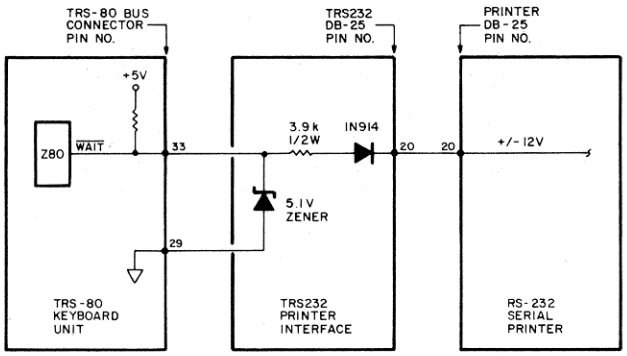


Fig. 2

generate virtually any desired baud rate to match your printer. The TRS232 is a one way send only device, from computer to printer.

I CLOADed and ran the software, then CLOADed a BASIC program for a listing. The printer worked like gangbusters—or so I thought! It's hard to describe the thrill of seeing the head zip across the paper, printing out all those things you've only been able to see on the screen. Listings, in particular, make programming a whole new ball game! And for tabulations there is just no substitute for a printout.

Printer Problems

However, I noticed on closer inspection that the printer had two problems. On any line exceeding 85 characters, the printer head hit a microswitch carriage return and lost 15 characters as it tried to continue printing the next line. In other words, the mechanical carriage return did not tell the computer to stop sending data. Second, I found that every once in a while the printer missed characters.

Fortunately, I had help from a neighbor. He had also purchased a Scope Data printer(they are no longer available) and he showed me how to find my missing characters (see Fig. 2).

It seems the printer has a buffer that holds only 64 characters in memory. So when the TRS-80 got more than 64 characters ahead of the printer some characters overflowed the buffer and were lost.

My neighbor, Jeff Mazur, has designed a simple circuit that allows a buffer full signal (– 12V on pin 20 of my Scope-Data printer) to pull the Z-80 $\overline{\text{WAIT}}$ line LO. This stops the Z-80 in its tracks, until the LO is removed. The 1N914 blocks +12 volts from the printer with the 3.9K resistor and 5.1V zener diode as additional insurance against high voltage on the Z-80 in case the 1N914 shorts out.

I installed these three components in the TRS232 box, with the cathode of the 1N914 soldered to pin 20 of the connector. The two wires going to pins 33 ($\overline{\text{WAIT}}$) and 29 (ground) were connected to the TRS-80 (left rear of the keyboard) using a soldertail 40-pin TRS-80 bus connector (available for $6.95 from Applied Invention, Hillsdale, NY). Though this overflow control circuit is not needed with most printers mine did need it, and I haven't missed a character since.

The loss of 15 characters can be overcome by wiring the microswitch to the proper point in the printer's complex circuitry so that the printer could be stopped during a carriage return. However, I found a simpler method—more about Formatter later.

I experienced other inconveniences as well. I had no way to page the output, that is, to print a certain number of lines in a given length. Some of my listings ran two or three feet with no convenient place to tear them into standard 8½ inch page lengths for filing. I could list the lines in groups, advancing the paper between groups, but that was not very convenient.

Also, either special gimmicks, imbedded in the program, were needed to print RUNs, or I had to change all PRINT to LPRINT statements. Keyboard debounce required entering a separate program on computer turn-on and whenever printing was directed to the computer I had no screen display.

While these inconveniences did not keep me from making extensive use of the printer, they were a constant annoyance and time-waster, and limited its applications.

Formatter

Well, the "geniuses" at Small Systems, apparently as a result of feedback from almost 2000 users of the TRS232 Printer Interface, now have available a new software tape called Formatter (less than $15).

With Formatter, you can set the number of lines printed on a page—a page contains up to 120 lines (although 66 lines on an 8½ inch long page is almost standard). When the specified number of lines are printed, the paper advances to the next page. No more long folded listings, with creases just where you don't want them!

You also set the number of characters on a line—from 25 to 250. When the printer reaches the maximum number it executes a carriage return and stops printing during the return.

Not only that, but you can choose "smart line termination," where the Formatter looks for a space, colon, semi-colon or comma near the end of the line and executes an early carriage return, breaking the line at a more readable location. The Formatter automatically indents continuation lines 5 spaces, again, for improved readability. this solves the printing during carriage return problem, and allows listing those programs with multiple lines per line number. Simply set the line length prior to carriage return.

Other Options

The Formatter allows all output to the printer to be displayed on the screen with its "echo" option. Another option allows programs strictly designed for the screen to be printed. Just enter LPRINTCRS$(1) from the keyboard, or imbed this command in the program, and all PRINT statements will output. Furthermore, if you've chosen the echo option, you'll still see the program running on the screen! LPRINTCHR$(2) returns PRINT commands to the screen only.

Do you go bananas with the fast automatic scroll listing in Level II? Using Formatter, you can list one line at a time! Just press the space bar to stop the printer at the end of a line. Each time you press the space bar you list one more line. Pressing any other key returns to continuous printing. Even if the printer is off, with the connected TRS232 hardware, the computer will think it is on so you can have single-line listing without using paper or time.

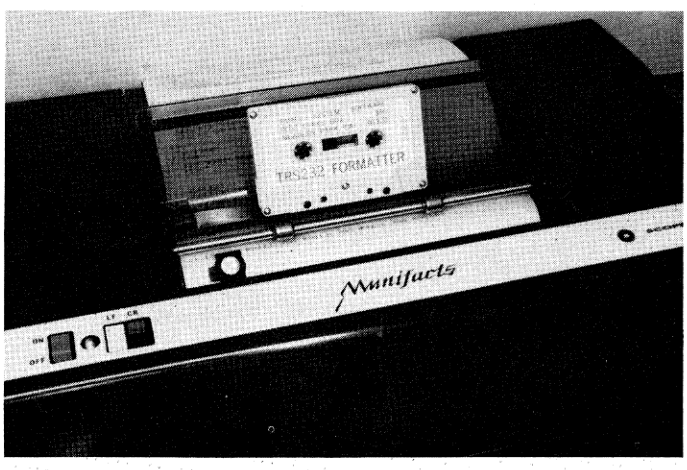


Photo 1

Formatter has a self-contained keyboard debounce program; you don't have to load in a separate tape. Also, you can select from 10 standard baud rates, or modify the program for a custom baud rate. You can also select line feeds after carriage returns if your printer needs them. Formatter uses a form feed character if your printer recognizes it, multiple line feeds if it doesn't.

Formatter loads in BASIC with CLOAD command after setting MEMORY SIZE? to allow for about 360 bytes at the top of memory. The clear documentation takes you step-by-step through each option and even shows how you can make your own customized version. Instead of answering twelve option questions after loading, you just CLOAD and RUN, then load or enter whatever program you want to print.

The documentation also covers the BASIC listing of Formatter (5970 bytes, but it only occupies 360 bytes after RUN POKEs machine language into high memory) and the assembly language source listing of the program that BASIC puts into memory.

Conclusion

The only problem I've found with the TRS232 Printer Interface is with the IC (a 741 operational amplifier). I've replaced it twice. This is a common 741 problem, not unique to this design. The first one lasted 4 months, the second one 2 weeks. The cause seems to be static discharge and/or line transients. The original 741 was soldered to the PC board. I replaced it with a socket, so the next failure only took a couple of minutes to fix. By the time you receive this, the units being delivered should have a socketed 741 and 2 diodes to protect it.

All technicalities aside, if you've been considering a serial printer, or have one that is limited, this TRS232/Formatter combination could make your printer a genius! ■

SPECIAL DELIVERY®

TRS 80® + Electric Pencil®

By adding SPECIAL DELIVERY to Electric Pencil you can realize the full potential of your TRS-80! A 100% *machine language* word processor!!

MAILFORM — Create MAILFILE: A complete Name and address list entry/editor program. Instant search on any field, complete cursor control, optional beeper to let you know something is wrong, active file always displayed, search can include numeric only as well as don't care characters, just FILL IN THE FORM!!

MAILRITE — Print letters written with the Electric Pencil inserting information from a MAILFILE into the letter for personalizing and addressing. True typist quality using your printer. Features: Indents, underscore, end of page stop, unlimited insertion from address list, address envelopes and MORE!

SORT — MAILFORM will sort an entire address list in seconds using any field as the key.

LABEL — MAILRITE prints mailing labels from MAILFILE.

CONVERT — Transform your Radio Shack Mailing list into a MAILFILE!

SPECIAL DELIVERY (Disk) 99.95
Electric Pencil (Disk) 150.00
Phone Orders (214) 357-8357
Or write for brochure of our complete line of fine software!

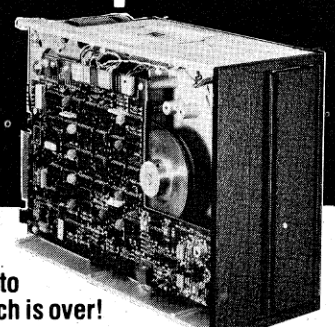
✓42

VISA · master charge

OFTWARE, ETC...
1839 CHAMBERLAIN DRIVE
CARROLTON, TX 75007

You don't have to buy it just for the low price.

You can buy it for the quality, too!



If you've been looking for a less expensive floppy disc drive, but not wanting to sacrifice quality - your search is over!

You get both in the Remex RFD1000B! For only $395 look at what you get: ■ 8" Floppy Drive ■ Single or Double Density ■ Hard or Soft Sectoring ■ Media Protection Feature ■ Single Density Data Separator ■ 90 Day Factory Warranty

REMEX RFD1000B
$395
VOLUME DISCOUNTS AVAILABLE

AVAILABLE OPTIONS/ACCESSORIES
☐ Door Lock, $19.95 ☐ Dual Drive Power Supply, $91.95 ☐ Interface Manual, $2.95
☐ Write-Protect, $19.95 ☐ Single Drive Power Supply, $69.95 ☐ Interface Adapter, $12.95 (Remex to Shugart)
☐ Connectors, $9.95 ☐ Drive Cabinet, $24.95

SIRIUS SYSTEMS, P.O. Box 9748, Knoxville, TN., 37920
Phone Orders accepted 9AM-7PM (E.S.T.): 615/577-1072
✓67
☐ Check ☐ Money Order ☐ C.O.D. ☐ MC ☐ VISA ☐ AE

NAME _____ CARD # _____
ADDRESS _____ EXPIRATION DATE _____
CITY _____ STATE _____ ZIP _____ CARDHOLDERS SIGNATURE _____

Add $7.00 per Drive for Shipping/Handling. Tennessee residents add 6% sales tax. Foreign orders add 10% (payment in U.S. currency only).

Double your memory . . . not the price!

COMPARE AT $695 . . .

$525

Remex RFD 4000/8" Floppy Disc Drive
Double the storage!
Double sided . . .
Double density!!

Offers quality and features found in drives costing much more! ■ Single or Double Density ■ Double-Sided Drive ■ Door Lock INCLUDED ■ Write-Protect INCLUDED ■ 180 Day Warranty ■ Compatible with Shugart 850/851 ■ Low Power Operation ensures LONGER LIFE!! ■ Model RFD 4001 offers Data and Sector Separator

AVAILABLE OPTIONS/ACCESSORIES
☐ Dual Drive Power Supply and Cabinet, $139.95 ☐ Single Drive Power Supply and Cabinet, $119.95
☐ Interface Manual, $2.95
☐ RFD 4000 Manual, $5.95 ☐ Drive Cabinet, $29.95

SIRIUS SYSTEMS, P.O. Box 9748, Knoxville, TN., 37920
Phone orders accepted 9 AM-7 PM (E.S.T.): 615/577-1072, please send the following
✓67
☐ RFD 4000, $525 ☐ RFD 4001, $539 / ☐ Check ☐ Money Order ☐ C.O.D. ☐ MC ☐ VISA ☐ AE

NAME _____ CARD # _____
ADDRESS _____ EXPIRATION DATE _____
CITY _____ STATE _____ ZIP _____ CARDHOLDERS SIGNATURE _____

Add $7.00 per Drive for Shipping/Handling. Tennessee residents add 6% sales tax. Foreign orders add 10% (payment in U.S. currency only).

Prevent problems caused by over adjustment of a process containing random errors.

Process Control

Robert G. Hoffmann
315 N.W. 109th Ave. Apt. 203
Miami FL 33172

In various kinds of manufacturing and testing, it is necessary to determine during production whether the process is performing properly. This may be done by accurately testing occasional individual specimens during production runs. If the test of each specimen indicates that the process is yielding the desired results, then it is left alone; otherwise, the process is "adjusted," so it is thought, to correct it.

Logical as the above may seem, its use may be catastrophic, as will be shown.

True values	Adjustment	Adjusted values
9	+1	-
11	−2	12
9	+3	7
11	−4	14

Table 1.

Blind use of the procedure ignores what is called random, or indeterminate error.

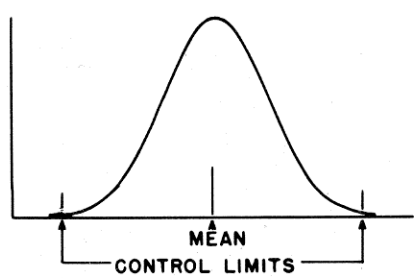
Indeterminate Error

The upper part of Fig. 1 shows a simulated distribution of random errors of a satisfactorily performing process. No process is perfect, so a certain amount of unavoidable error is always present. Such errors are not constant from specimen to specimen; otherwise, they could be corrected. This is why a distribution of errors is approximately like that shown in Fig. 1.

In practice at the beginning of a production run, several specimens are accurately tested, and the distribution of their results is plotted, as in Fig. 1. Production is set at their mean or adjusted accordingly. Assume that a process has been properly set up and that production has started. From time to time, individual production specimens are tested to determine whether the process is "in control." What is done with results of the tests is crucial.

Assume that the operator is unaware of the nature of random error, so he adjusts the process with results of each specimen he tests accurately. A simple example is shown in the lower part of Fig. 1 and in Table 1. The hypothetical process is expected to yield, on the average, a value of 10, and 95 percent of the random errors falls within the range 8 to 12.

The chart on the lower left in Fig. 1 and the left column of Table 1 show results that would be obtained if the operator had



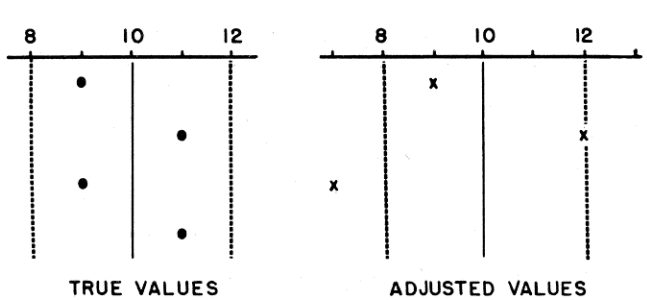


Fig. 1.

left it alone. All are well within the error or control limits. But look what happens if he adjusts the process with each result. The first result is 9, so to correct the process he says to himself, "It's running a bit low, so I'll change it to produce one more unit, since 10 is expected."

A bit later, he tests again. Had he left the process alone, a value of 11 would have been obtained, but instead, he gets a 12. "Wow! It's now too high," he mutters to himself as he adjusts the process downwards by two units. This now subtracts two units from the next "true value" of 9, so that 7 is actually obtained, and so on.

All results are shown in Table 1, and the erroneous adjustments are shown graphically in the lower right chart of Fig. 1. The last adjusted value is off the chart.

At this point, it may seem obvious that to adjust a process with random error is the height of folly, and that no one does it. But in real life, the consequences of such actions are not always so obvious, as the following examples indicate.

Examples of Erroneously Adjusting a Process

In his book on quality control, Grant describes a manufacturer who did exactly as outlined here[1]. He knew that something was wrong, because so many of the lots he produced were substandard. The more attention he gave to controlling the process, the worse the results. It was not until a consultant familiar with random error reviewed his practices that the source of the trouble was located and corrected.

In the medical field, the erroneous adjusting of equipment that tests blood specimens is rampant. Medical technologists have been trained, especially by instrument manufacturers, to correct the processes for drift, although the manufacturers never say what causes the drift nor how much is present[2].

Another source of difficulty in the medical field arises from the fact that medical technologists who test the specimens and physicians who use results of the tests have virtually no contact. Thus, only if results are so erroneous that physicians question them are mistakes sometimes detected[3].

The main point to the above is that the program described here can be used to provide information about real and continuing problems that arise in many situations.

The Program and Its Features

The program was written in Radio Shack Level II BASIC, although it should run in Level I if you add a subroutine to compute the square roots. To appreciate all of its features, you need some knowledge of statistics. For those who want it, a good, widely available book on basic statistics is recommended[4].

In statements 70 and 80 a random normal variable is generated. Its mathematically expected mean is 10, and its standard deviation is 1 (this is described later). The point at the moment is that this is the random error variable which is printed out by statement 90. The amount of the adjustment is computed by statement 150, and the adjusted value is printed out by statement 130. Of course, there is no adjusting the first result, so it prints out by itself; otherwise, the unadjusted and adjusted results print out in pairs.

Input 4s to at first study the results of the process. You can easily verify the results to see that the program is operating as it should. You will also find that in some cases the adjusting process is *apparently* successful. This is a trap for the unwary. It is impossible to improve a process that is operating to the limits of its accuracy and precision. To improve it, you must improve the process itself. For more extensive trials, inputs of 36 are convenient. All of the results can be visualized on the screen for study.

The random normal variable is produced as a result of the following. Obtaining random numbers with the RDN(x) statement amounts to obtaining samples from an infinite rectangular population, i.e., each number is equally likely to be obtained, as in throwing dice. With random numbers ranging from 1-11, the population mean is 6, and the standard deviation, s, is $\sqrt{10}$. Since 10 is a convenient number to keep track of for the mean, I added 4 to each result (statement 80). But we are drawing and averaging samples of size 10, and the standard deviation of the mean is given by: $s_X = s/\sqrt{n}$, so $s_X = \sqrt{10}/\sqrt{10} = 1$.

Furthermore, thanks to the phenomena described by the central limit theorem of statistics, the distribution of the means is not rectangular, but will be closely approximated by a theoretical normal or Gaussian distribution.

Anyone who wishes to assure himself of the truth of the assertions made here need only draw a sample of a thousand or more averages of RND(11), i.e., trials of the random normal variable, and study the results. This random normal generator should be used in many situations.

A Final Suggestion

If you have a friend who works in a clinical laboratory, invite him over for a demonstration of the program and see what it means to him. You may find that some people have rigid ideas—sometimes in conflict with the facts—about how the world should be. Remember, it is your blood specimens that they are testing. ■

```
20  REM SIMULATES ERRONEOUSLY ADJUSTING AN AUTOMATED PROCESS.
30 INPUT "NO. OF TRIALS WANTED"; K
40  A=0: S1=0: Q1=0: S2=0: Q2=0
50  FOR J=1 TO K
60  FOR I=1 TO 10
70  D=RND(11) : SUM=SUM+D : NEXT I
80  R=SUM/10 : SUM=0 : R=R+4
90  PRINT R;
100 IF J=1 THEN GOTO 150
110 S1=S1+R: Q1=Q1+R*R
120 R=R-A
130 PRINT R,
140 S2=S2+R: Q2=Q2+R*R
150 A=R-10
155 IF J=1 THEN PRINT " ",
160 NEXT J
170 L=K-1
180 M=S1/L : MA=S2/L
190 PRINT: PRINT"N IS --";K,"EXPECTED S.D. IS 1.00": PRINT
200 PRINT"TRUE MEAN";M, "ADJ. MEAN"; MA
210 V=(Q1-S1*S1/L)/(L-1) : S=SQR(V)
220 VA=(Q2-S2*S2/L)/(L-1) : SA=SQR(VA)
230 PRINT "S.D. TRUE ";S, "S.D. ADJ."; SA
240 PRINT "-----------------------------------------------"
250 GOTO 30
```

Program listing.

References

1. E. L. Grant and R. S. Leavenworth, *Statistical Quality Control*, 4th Ed. McGraw-Hill Book Co., New York NY (1972).
2. R. G. Hoffman, "Misuse of calibration sera in automated laboratory testing instruments." (Letter to the editor.) *Clinical Chemistry*, 22, 561 (1976).
3. National Bureau of Standards, Washington DC. Survey of Laboratories (1973).
4. H. T. Hayslett, Jr., *Statistics Made Simple.* Doubleday & Co., Garden City NY (1968).

A popular pastime, using the Microsoft ROMs for programming shortcuts.

ROM Routines

Wes Thielke
2930 Quakerbridge Rd.
Mercerville NJ 08619

I've owned a Radio Shack TRS-80 for over a year. Soon after I received my Level II unit I wrote a disassembler and began taking apart the TRS-80 monitor.

Since I prefer programming in assembly language to BASIC but don't like to waste time reinventing the wheel, I set out to find routines in the BASIC monitor which I could use in my programs. In this article I've compiled some of the more useful things I've learned.

The Keyboard

Striking each key on the TRS-80 causes one bit in a specific byte in the RAM to be turned on. Table 1 shows which bit is set by each key. (Please note that all numeric values shown in this article are hexadecimal unless otherwise indicated.)

bit	3801	3802	3804	3808	3810	3820	3840	3880
0	@	H	P	X	0	8	ENTER	SHIFT
1	A	I	Q	Y	1	9	CLEAR	
2	B	J	R	Z	2	:	BREAK	
3	C	K	S		3	;	↑	
4	D	L	T		4	,	↓	
5	E	M	U		5	–	←	
6	F	N	V		6	.	→	
7	G	O	W		7	/	SPACE	

Notes: (a) Location 387F may be tested for a non-zero value to determine if any data is present on the keyboard

(b) I am using the bit numbering traditionally used for the Z-80 and 8080. That is, the leftmost bit is 7. This is a caution to IBM 360/370 programmers, who are used to having the leftmost bit numbered zero.

Table 1. RAM Location

Appendix C of the Level II manual shows the codes for the alphanumeric keys as decoded by the BASIC monitor. However, the codes shown for the control characters, for the most part, apply only to video output. Here is how the control keys are decoded from the keyboard (codes shown in decimal):

2B	Test for keyboard input. If there is any, its decoded value is in the A register; otherwise A is zero. The data is *not* displayed on the video screen.
49	Wait for keyboard input. Continuous calls are made to 2B until a non-zero value is returned in A.
358	Saves DE register pair and calls 2B.
384	Calls 358 until a non-zero value is returned.
5D9	Input a line into the address in the HL register pair. This routine also displays each character as it is input and takes action on the control keys just as if you were entering a BASIC statement. Register B is assumed to contain the maximum length of the line. Control is returned when a terminator key (ENTER or BREAK) is pressed. On Return: HL — unchanged B — actual input length (excluding terminator) C — original contents of B A — terminating character (ENTER or BREAK)
361	Input a line into the buffer whose address is found at locations 40A7-40A8. Returns this address minus one in HL. The end of the input buffer is marked by a byte of zeros.

Table 2. Subroutines that may be used for input.

33	Display byte in A register at the current cursor position. This routine also performs the control functions as listed in Appendix C of the Level II manual. Note: the cursor position is saved at location 4020-4021.
1C9	Clear screen
22C	Flip flop asterisk in the right corner of the screen.
33A	Saves DE and displays byte in A at cursor (calls 33).
28A7	Displays the string at the address in HL. The string must be terminated by a byte of zeros. Control codes may be contained in the string.
150	SET/RESET subroutine. This is the common routine used by the SET/RESET commnd processors (after the x,y coordinates have been evaluated). The following code shows how to set up a call to this routine:

```
        LD   HL,RETRN
        PUSH HL              ;put return address on stack
        LD   A,set/reset code
        PUSH AF              ;put set/reset code on stack
        LD   A,x-value
        PUSH AF              ;put x-coordinate on stack
        LD   A,y-value       ;load y-coordinate
        JP   150H            ;go to set/reset routine
RETRN---
```

The set/reset codes are (decimal):

```
  1 -reset
128 -set
```

Table 3. Video Subroutines

9B4	Store the single-precision value in DE:BC at 4121
9C2	Load the single-precision value at the address in HL into DE:BC
9B1	Copy the single-precision value at the address in HL to 4121
9BF	Load the single-precision value at 4121 into DE:BC
9A4	Place the single-precision value at 4121 on the stack (DE then BC)
A9A	Store the integer value in HL at 4121 and set the type flag to 'integer'
E6C	Evaluate the numeric string at the address in HL and store it in the accumulator, setting the type flag. The routine stops as soon as it encounters a character that is not part of the number. It will accept signed values in integer, real and scientific notation. A value of zero is returned if the string is non-numberic.
1E5A	Evaluate the string at the address in HL for a positive integer value, stopping at the first non-numeric character. The result is returned in DE. A value of zero is returned if no numeric value is found. This routine is used by BASIC statements requiring a statement number, such as GOTO.
FBD	Non-formatted numeric edit routine. Converts the accumulator to display form, with the result string address returned in HL. The address of the end of the string + 1 is returned in DE.
FBE	Formatted numeric edit entry point (used by PRINT USING). Entry requirements: A register—flags as follows: bit meaning 7 perform edit 6 include commas in result 5 asterisk fill 4 precede by dollar sign 3 force sign (+ or −) 2 trailing sign 0 scientific notation B register—number of whole digits (including commas, excluding space and sign) C register—number of decimal digits (including decimal point) Note that FBD and FBE are the same routine. The instruction at FBD merely clears the A register so that no formatting is done.
2865	Creates a string vector using the output produced by routine FBE.

Table 4. Arithmetic Utility Routines

Key	No Shift	Shift
ENTER	13	13
CLEAR	31	31
BREAK	1	1
←	8	24
→	9	25
↑	91	27
↓	10	26

The way the keyboard is mapped into memory makes it easy to test for specific keys in interactive programs. For example, to test for the break key, one may simply code:

```
LD  HL,3840H  ;point to byte in which
              ;the BREAK key is stored
BIT 2,(HL)    ;test bit set by BREAK
              ;key
JR  NZ,BREAK  ;if bit set, go to 'BREAK'
```

Keys can also be tested in a

✓ Reader Service—see page 131

Abbreviations: ACC = accumulator, OP = operand

Routine Address	Operation	Precision
BD2	ACC = DE + HL	INTEGER
BC7	ACC = DE – HL	"
BF2	ACC = DE•HL	"
2490	ACC = DE/HL	"
716	ACC = OP + ACC	SINGLE
713	ACC = OP – ACC	"
847	ACC = OP•ACC	"
8A2	ACC = OP/ACC	"
C77	ACC = ACC + OP	DOUBLE
C70	ACC = ACC – OP	"
DA1	ACC = ACC•OP	"
DE5	ACC = ACC/OP	"
98A	ACC = SGN(ACC)	ANY (result integer)
B37	ACC = INT(ACC)	" " "
977	ACC = ABS(ACC)	ANY
13E7	ACC = SQR(ACC)	ANY (result single)
14C9	ACC = RND(ACC)	" " "
809	ACC = LOG(ACC)	" " "
1439	ACC = EXP(ACC)	" " "
1541	ACC = COS(ACC)	" " "
1547	ACC = SIN(ACC)	" " "
15A8	ACC = TAN(ACC)	" " "
15BD	ACC = ATN(ACC)	" " "
A7F	ACC = CINT(ACC)	ANY (result must be in integer range)
AB1	ACC = CSNG(ACC)	ANY
ADB	ACC = CDBL(ACC)	ANY
B26	ACC = FIX(ACC)	ANY (result must be in integer range)

NOTE: There is no direct way to compute a value raised to a power. This can be accomplished by using the following formula:

$$A^X = EXP(X \cdot LOG(A))$$

Table 5. Arithmetic and Function Routines

BASIC program using a similar technique, which, by the way, is much more efficient than using INKEY$. For example, try the following program which tests for the cursor keys:

```
10 DEFINT C
20 C=PEEK(14400)      : REM GET CONTENTS OF 3840H
30 IF C AND 8 THEN PRINT "UP";
40 IF C AND 16 THEN PRINT "DOWN";
50 IF C AND 32 THEN PRINT "LEFT";
60 IF C AND 64 THEN PRINT "RIGHT";
70 IF C AND 255 THEN PRINT
80 GOTO 20
```

Try holding down more than one key at the same time! Then modify your LUNAR LANDER program.

The subroutines in Table 2 may be used for input from the keyboard.

The Video Display

The video display is mapped into locations 3C00 through 3FFF. Except for the control and tabbing codes, all character codes shown in the Level II manual may be displayed by moving a byte to the desired location.

Each byte on the video display is subdivided into six "points" for graphics purposes. The value displayed is treated as a graphics character if bit 7 is set. The low order six bits then determine which points are turned on as follows:

0	1
2	3
4	5

Bit 6 is ignored when bit 7 is set.

Double width character mode is turned on by outputting a byte with bit 3 set to port FF. For example,

```
LD   A,8
OUT  (0FFH),A
```

It is reset by outputting a byte with this bit turned off.

Video subroutines are shown in Table 3.

Arithmetic Subroutines

Even a primitive understanding of the manner in which arithmetic is done by the BASIC monitor can save hours of programming in assembly language.

The monitor is capable of doing arithmetic in three modes: integer, single-precision floating point and double-precision floating point.

Integer values are stored in two's complement form as 16-bit integers (2 bytes). The high order bit is the sign bit.

Floating point data is stored as a normalized binary fraction, with an assumed decimal point before the most significant bit. In addition, the most significant bit doubles as the sign bit.

A binary exponent takes one byte in each floating point number. It is kept in excess 128 form; that is, 128 is added to the actual binary exponent needed. For example, the number 1 is $.1_2 \times 2^1$, so the exponent is 129 (81H). 2 is $.1_2 \times 2^2$, so the exponent is 130 (82H). (Note that $.1_2 = ½$. This will become clearer in a moment.) The value zero is stored as an exponent of zero.

The arithmetic subroutines use an 'accumulator' and sometimes an operand. A type flag, located in memory location 40AFH, indicates the current type of the value in the accumulator. Type conversion is performed automatically in most cases, based on the value of this flag. The values are:

2—integer
3—string
4—single-precision floating point
8—double-precision floating point

The accumulators for each type are:

Integer: Locations 4121 and 4122, least significant byte is stored first. For example, the value 1 would be stored as 0100. (This may seem strange at first, but data is stored this way because the Z-80 16-bit load instructions store data this way.)

String: Locations 4121 and 4122 contain the address of a length and address descriptor for the string. The first byte of the descriptor is its length and the following 2 bytes are the address. The string value itself is followed by a byte of zeros, ready for display using the routine at 28A7.

Single-Precision: Locations 4121 through 4124. Location 4124 is the exponent, 4123 the most significant byte (containing sign), 4122 the next most significant byte and 4121 the least significant byte. Here are some examples to clarify this representation:

Value (decimal)	4121	4122	4123	4124
0.5	00	00	00	80
1	00	00	00	81
–1	00	00	80	81
2	00	00	00	82
3	00	00	40	82
4	00	00	00	83
129	00	00	01	88
–129	00	00	81	88
256	00	00	00	89
257	00	80	00	89

Let's take a closer look at the number 257, in case you're still confused. 257 (decimal) is 101 in hexadecimal, 100000001 in binary.

You may think of the expo-

1A19 Reentry to BASIC

60 Time delay routine. Input consists of a loop counter in register pair BC. Here is the actual code, in case you want to figure out the number of T states:

```
0060   DEC  BC
0061   LD   A,B
0062   OR   C
0063   JR   NZ,0060H
0065   RET
```

1C90 Compare HL with DE, accomplished by computing HL minus DE in such a manner that the sign, carry and overflow flags are set as you would expect. This routine is normally called by executing RST 18H.

1D78 Fetches the next non-blank character at the address following that in the HL register pair (HL is incremented first). The character is returned in the A register. If the character is numeric (ASCII), the carry flag is set. If the byte is zero, the Z flag is set. (Normally called by executing RST 10H).

25D9 Tests the type flag at 40AF. Z flag is set if string, M if integer, P and C if single-precision, P and NC if double-precision. (Normally called by executing RST 20H).

Table 6. The above are Miscellaneous Subroutines and Entry Points

nent as the number of bits to the right of the most significant bit *after* which to place the binary decimal point. Then add this value to 128 to produce, in our example, 128 + 9 = 137 or 89 in hexadecimal.

Now, we normalize the number; that is, the most significant bit becomes the 'leftmost' bit in byte 4123. Our normalized number is 1000 0000 1. Grouping four bits together from the left, the result is the hexadecimal digits 808. The left two digits go in byte 4123 and the last hex digit goes in the leftmost part of byte 4122.

Lastly, we determine the setting of the sign bit. Since the number is positive, the sign bit is zero; therefore, we turn off the 80 bit in byte 4123.

Double-precision: Locations 411D through 4124. Data is stored just like single-precision except that there are more bytes of significance. Byte 411D is the least significant.

Where used, the operands are:

Integer: Register pairs DE and HL. (Used for add, subtract, multiply and divide only.)

Single-precision: Register pairs DE and BC, as obtained by executing:

```
LD  DE,(4121H)
LD  BC,(4123H)
```

Double-precision: Locations 4127 through 412E.

Table 4 lists arithmetic utility routines and 5 lists functions.

Exits

The writers of Level II BASIC were very clever in many ways. Did you ever wonder how they manage to provide all the additional functions of DISK BASIC without replacing the ROM monitor? What I am about to describe should enlighten you on that question. In fact, with this information and a little imagination, you can make your Level II machine do all sorts of new things.

I am referring to the

RST Operand	Vector Address	Initial	Contents
8H	4000H	JP	1C96H
10H	4003H	JP	1D78H
18H	4006H	JP	1C90H
20H	4009H	JP	25D9H
28H	400CH	RET	
		NOP	
		NOP	
30H	400FH	RET	
		NOP	
		NOP	
38H	4012H	EI	
		RET	
		NOP	

Table 7. Restart vectors and initial contents

Address (hex)	Command	Address (hex)	Command
4152	CVI	417C	FIELD
4155	FN•	417F	GET
4158	CVS	4182	PUT
415B	DEF	4185	CLOSE
415E	CVD	4188	LOAD
4161	EOF	418B	MERGE
4164	LOC	418E	NAME
4167	LOF	4191	KILL
416A	MKI$	4194	& •
416D	MKS$	4197	LSET
4170	MKD$	419A	RSET
4173	CMD	419D	INSTR•
4176	TIME$•	41A0	SAVE
4179	OPEN	41A3	LINE

•Items marked with an asterisk are called during expression evaluation.

Table 8. DISK BASIC Command Vectors

numerous 'exits' to fixed RAM locations that are built into the monitor (Table 6).

First of all, there are the restart vectors. The fixed memory locations, called by the RST instruction (with the exception of RST 0), contain jumps to fixed RAM locations which are initialized when power is turned on. The vectors and their initial contents are contained in Table 7.

These instructions can be modified in an assembly language program to provide whatever functions desired.

Then there are the DISK BASIC command vectors, which normally produce the L3 error when running without DISK BASIC by jumping to location 12DH.

These vectors can also be changed for whatever purpose you desire. For example, you may use them to obtain the equivalent of additional machine language functions in your BASIC programs or to keep a machine language routine on call from the BASIC monitor by the entry of a command (Table 8).

Finally, there are a series of exits, called by various routines in the monitor. The vectors for these exits start at location 41A6 and continue through 41E4. They each consist of three bytes, initialized with RET instructions. The functions of a few are listed in Table 9.

41AC	Called from 1A1C at BASIC reentry
41AF	Called at the beginning of the line input routine (361H)
41B2	Called after a BASIC statement line has been encoded (compressed). The HL register pair contains the address of the encoded line.
41BE	Called during error processing. This exit can be used to intercept errors such as overflow in an assembly language routine if locations 409B and 409C are first set to zero.
41C4	Called at 358H, the beginning of a routine used to get a byte from the keyboard.
41C7	Called by the RUN command processor when it is followed by something other than statement end.
41E2	Called at entrance to the SYSTEM command processor.

Table 9. Exits initialized by RET instructions.

Conclusion

As you can see, there's a lot more to the TRS-80 than meets the eye, or than Radio Shack cares to make generally known.

I hope the information I've presented here saves you a little time and helps to add some spice to your TRS-80 programming. ■

PROBLEMS?

Are you having trouble with advertisers or products?
If such is the case, please write to the firm
giving complete details and send a copy of your letter to
Wayne Green, *80 Microcomputing*,
Peterborough, New Hampshire 03458.
Be polite—no matter how badly you're shafted.

We're looking for names and addresses of all the **TRS-80 newsletters** out there. If you produce or are affiliated with such a newsletter, please drop a line to:

Jim Perry, Managing Editor
80 MICROCOMPUTING
Peterborough NH 03458

Thanks.

Not everyone has disk or $100 for a 'Pencil.' Try this for $10.

Poor Man's Text Editor

Fred Blechman
7217 Bernadine Ave.
Canoga Park CA 91307

One of the most significant emerging uses of microcomputers is word processing. By using keyboard commands to add, delete, modify or move letters, words, paragraphs and even pages, a user can output a fully edited letter, report or manuscript to the printer.

Complete word processing systems are available for several thousand dollars. Software programs for such processors are available from $25 to $100 or more. But the "Poor Man's Text Editor" for the Level II TRS-80, for under $10, is surprisingly versatile and might satisfy all your needs.

Features

This text editor program features upper and lowercase, a non-destructible easily maneuvered blinking cursor, editing and graphics options and will interface with either tape or printer. You can add, delete, or repeat any regular or graphic character. Furthermore, you can hold up to nine pages of text, at 16 lines per page (144 lines) in memory and when printing you can select the line spacing!

This program requires a 16K memory to utilize all the options. Also, to print or display lower case, your TRS-80 must be modified using one of several conversion kits available for $20 or less.

With only 13 commands using the Poor Man's Text Editor is simple and the instructions supplied with the cassette are excellent. First, answer the MEMORY SIZE? prompt with 21740. This preserves memory space for the program, nine pages of text and printer software, if needed. If printer software is required, enter it *first*.

Next you CLOAD the Poor Man's Text Editor and run it. After a short delay the screen clears and then asks "Printer On Line?" Type in Y if your printer is enabled, N if not. The computer then asks, "Is this a cold start?" Although the program is practically "bomb-proof," this input allows you to save previously stored text. Type in Y if you have no previously stored text on this run. Now the screen clears and a square blinking cursor appears in the upper left hand corner of the screen.

The Cursor

As you type on the keyboard the characters appear above the cursor, which then moves to the next space. At the end of a line, the cursor jumps to the beginning of the next line. At the lower right corner it disappears and stops one space beyond view.

The cursor can be easily moved around the screen with the arrow keys one increment at a time, up-down-left-right, from any position, and it will *not* blank out anything on the screen; that's why it's called a non-destructible cursor! If you want to automatically step the cursor in any direction just hold down the shift key when you press the arrow. Pressing any key will then stop the cursor. The cursor also stops, when in the automatic mode, at the top and bottom screen limits. Pressing the ENTER key moves the cursor to the beginning of the next line, except on the last line.

You can move the cursor to the "home position" (upper left corner) or the space following the last text entry. To do this, put the program in the COMMAND MODE by holding down the SHIFT key and pressing ENTER (SHIFT/ENTER). The cursor stops blinking. Now type H for sending the cursor "home" or X to move the cursor to the end of the text.

WHILE HOLDING DOWN SHIFT KEY, PRESS ENTER KEY. THIS PUTS THE PROGRAM IN THE COMMAND MODE. THEN:

KEY	COMMAND	ACTION	ESCAPE
C	CODE	Allows insertion of any of 64 graphic block combinations of 6 small blocks	X&ENTER*
D	DELETE	Deletes any number of characters	ENTER
E	END TEXT	Clears end-of-text of trailing blanks	NOT REQ'D
F	FILE	Holds page of text in memory	NOT REQ'D
G	GET	Retrieves page of text from memory	NOT REQ'D
H	HOME	Returns cursor to first position	NOT REQ'D
I	INSERT	Inserts any number of characters	ENTER
L	LOAD	Loads text from tape	NOT REQ'D
Pn	PRINT	Contents of screen to printer. "n" designates line spacing	NOT REQ'D
R	REPEAT	Repeats last character	ANY KEY
S	SAVE	Saves text on tape	NOT REQ'D
X	END (cursor)	Moves cursor to end of text	NOT REQ'D
CLEAR	CLEARS SCREEN	Clears screen & moves cursor to starting position.	NOT REQ'D

***C-mode blocks are repeated by hitting enter key alone.**

Table 1. Summary of commands

Editing is Simple

Okay, so now you've entered some text. How do you edit? Simple! Suppose a word or number contains the wrong character. Just maneuver the cursor under that character and type in the correct letter or number. See Table 1 for a summary of more elaborate editing commands.

Suppose one or more characters have been omitted. Move the cursor under the beginning of the omission. Put the program into the command mode with SHIFT/ENTER. The cursor stops blinking and the character above the cursor goes blank. Don't panic! The character is still there, you just can't see it. Now type I and the character re-

appears. Type the desired insertion and the characters appear as you type them, pushing any following text to the right or onto the next line. Press ENTER to return to the normal mode. If characters or spaces need to be deleted, press SHIFT/ENTER, then D, and any key or the space bar will delete one character per stroke; the screen shows you exactly what you're doing as you do it. Escape from this mode with ENTER.

The last-entered character (or space) can be repeated by simply pressing SHIFT/ENTER and R. The CLEAR key blanks out the screen for a new start with the cursor in the upper left corner.

As you add or delete characters all entries (including spaces) *beyond* your editing point are moved. This means that the beginning and ends of lines might need spaces added or deleted to prevent broken words or indentations. Therefore, always do the editing from the top of the screen down, and correct words and spelling as soon as noted.

Graphic blocks—64 different combinations from 1 to 6 blocks each—can be added to the display with SHIFT/ENTER and C, using the code supplied with the cassette instructions. Unfortunately, most printers can not print these blocks.

Printing and Storing

Once you've got exactly what you want on the screen, press SHIFT/ENTER and E. This clears the end-of-text of trailing blanks. Now you can preserve the screen contents on tape or direct it to a printer. To save on tape, use SHIFT/ENTER and S, and record on tape. To play it back from tape, use SHIFT/ENTER and L; the screen goes blank and then each character is displayed in real time as it comes from the tape! It's fascinating to watch as the letters zip across the screen—like a superfast typist (about 60 characters per second!).

To print the screen contents, press SHIFT/ENTER and then P followed by a number—1 for single line spacing, 2 for double spacing, 3 for triple-spaced lines, and so on. Whatever is on the screen (except the graphic blocks, with most printers) will print. A feature of this program is that trailing blanks trigger a line feed, as does a blank line. This saves considerable typing time compared to some other editors that "print" every space on the screen. The cursor does not print (it's actually a graphic block) so it can be anywhere on the screen.

If your printer adds extra spacing between lines, change the value of C = 138 in line 300 of the program to either C = 26 (ASCII linefeed) or C = 32 (ASCII space).

Now comes one of the big features: page storage. To save the screen display in memory, use SHIFT/ENTER and F (for FILE). If the memory is not full, the screen will be cleared as the text is stored in memory. If the text remains on the screen, no memory is available. Up to 9 full screens (144 lines, or 9216 characters and spaces) can be saved this way.

To retrieve the text, use SHIFT/ENTER and G (for GET). The next memory page will appear on the screen all at once, almost like magic! If it's not the page you want, refile and try again (poor man's approach). If the text remains on the screen, nothing is being held in memory.

You can't conveniently move blocks of text, or paragraphs, from one position or page to another; everything is accomplished with individual characters. Also, right-justification of the text is not automatic, but you can do it by inserting blanks on each line as needed. Hey, whaddya' want for under $10?

The "Poor Man's Text Editor", is available on cassette with instructions from Don Coon, 1228 Alpine, DeWitt, MI 48820. $9.95 postage paid. ■

Keep track of how much, to who and when—up to 32,767 accounts.

Household Accountant

David C. Andresen
3317 Morris Hill Rd.
Boise ID 83704

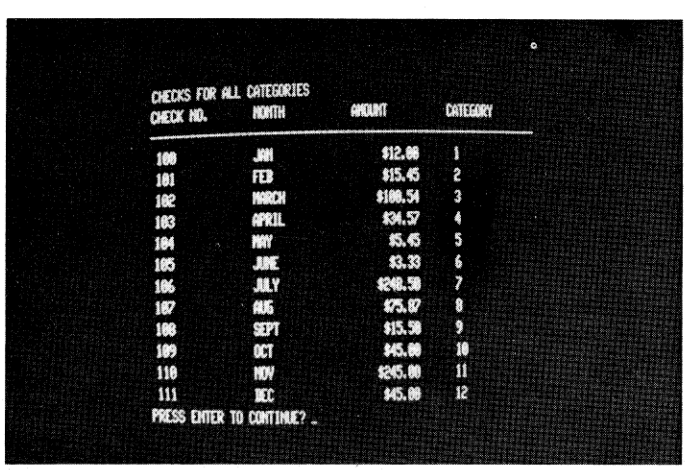
I have kept my family's expense records off and on for a number of years. They have been valuable in determining where all the money has gone and in making up budgets for the future. They have also been of great help at tax time.

Yet, as anyone who has ever done accounts would agree, the task of keeping and analyzing expense records is both tedious and time consuming when performed by hand. Somehow the columns and rows never seem to balance. And the more expense categories there are, the more difficult the whole problem becomes. That is the main reason that handling the family expense accounting has been a sometimes matter with me.

The simple program described in this article has changed all that. Now when the bank sends the family's canceled checks each month, I only have to carry out two simple steps to have them automatically recorded, analyzed, summarized and reported. First, I assign an expense category number to each check (32,767 categories are possible). Second, I type them into my TRS-80, indicating check number, the month, amount of check and expense category.

Code number	Category name
1	Wife's weekly check for cash
2	My weekly check for cash
3	House payment
4	Electricity
5	Telephone
6	Heating oil
7	Sewer
8	Garbage collection
9	Water
10	Taxes
11	Clothes
12	House maintenance
13	Car maintenance
14	Gifts
15	Subscriptions
16	Electronics
17	Professional dues
18	Doctors
19	Dentists
20	Medicines
21	Insurance
22	Contributions
23	Entertainment
24	Appliances/furniture
25	Charge accounts
26	Other

Table 1. Expense category codes assigned to checks. Consult books at your local library for a more complete listing. The basic idea in devising categories is to have enough detail to help you in a meaningful way. Avoid superfluous categories.



Typical display of audit trail with check numbers.

At that point the computer takes over, keeping track of the checks for up to a year, analyzing and summarizing them by months and categories and providing a reference or audit trail back to individual checks so there is hard documentation to back up the figures (the IRS likes that).

The program is very easy to use and understand. However, it does have limitations. Since it is check oriented, the program cannot readily handle cash transactions (always write checks), and it does not attempt to match checks with receipts, since there are simply too many complications in manually doing this in a simple home system. In my own situation,

these are minor inconveniences that are outweighed by the program's advantages.

Program Description

The program is written in Radio Shack Level II BASIC and occupies about 2000 bytes of memory without any REMARK or DATA statements. With 16K of memory, there is ample room for over 500 checks—more than my family writes in a year. I use tape as the storage medium.

When designing the program, I intended to keep it as simple and flexible as possible. This turned out to necessitate the use of DATA statements to hold check information and the use of numbers instead of words to designate expense categories.

DATA statements greatly simplify programming (less I/O) and tape handling (only one tape). They also significantly reduce tape-loading time on the TRS-80. The trade-off is a certain inelegance because it is necessary to leave BASIC to update the check file. This, however, poses no real problem for the home computer user.

The use of numbers to signify expense categories gives the user a great deal of latitude in the number and types of categories that can be accommodated. The limitation is that an index must be kept in hard copy to indicate the category name that goes with each number. Table 1 lists some of the categories I use.

In the program listing, all REMARK statements may be omitted without affecting program performance. Check information is entered in DATA statements beginning in line 10,000. I have found it useful to make the line number equal to 10,000 plus the check number. For example, check number 1527 is entered in line 10000 + 1527 = 11527. This makes it easy to find the proper line when data has to be corrected. This procedure also implies, however, that only one check is entered per line.

Lines 100-130 set up initial program conditions. CLEAR 63 reserves space for the string of dashes used in lines 740 and 940. It can be omitted if you don't want dashes (also delete lines 740 and 940). All variables beginning with the letters I, J, C, M or S are defined as integers by DEFINT in line 100. This conserves a little memory and, more importantly, speeds up program execution. It too can be omitted if desired or necessary. Line 110 defines the PRINT USING format used to align certain columns—leave it out if you want. Lines 140-180 print a brief description of the program.

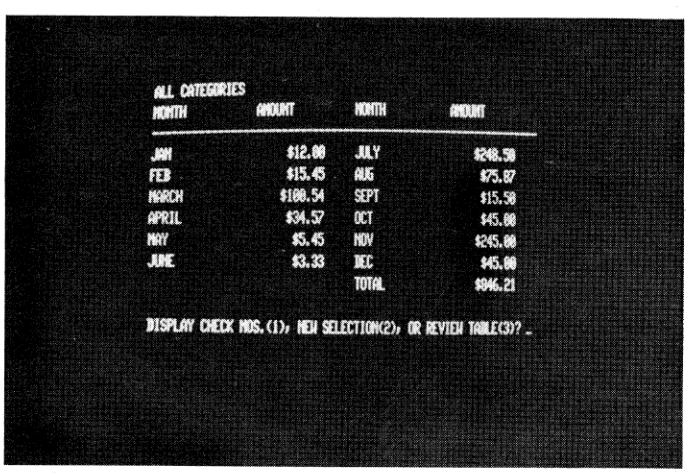
Lines 190-200 initialize several working variables. T# is the grand total of all monthly checking subtotals. The # indicates double precision, which is necessary since Level II BASIC single precision only displays six significant digits, i.e., a maximum of 9,999.99 without truncation. Nowadays six digits are not enough.

Lines 300-330 permit the option of analyzing the checks or first updating the file. Lines 400-495 comprise the file update routine. If you do not locate data in the 10,000 series lines, be sure to change line 495 appropriately.

Lines 500-780 analyze the checks for the category chosen and create the display table. Lines 540-640 are the heart of the routine; they search for and sum checks in the proper category. Specifically, lines 540-560 read the DATA statements (line 550 checks for end of data). Line 565 allows for the instance when the user wants all categories analyzed; line 570 is for a single category. Line 580 keeps a grand total of check amounts. Line 590 assigns check amounts (A) to

```
10 REM * SIMPLEST HOME EXPENSE PROGRAM *
20 REM * BY DAVID ANDRESEN *
30 REM * 3317 MORRIS HILL RD., BOISE, IDAHO 83704 *
99 REM * INITIALIZE PROGRAM *
100 CLEAR 63: DEFINT I,J,C,M,S
110 G$="$$###,###.##"
120 DIM M$(12),T(12)
130 M$(1)="JAN":M$(2)="FEB":M$(3)="MARCH":M$(4)="APRIL":M$(5)="MAY":M$(6)="JUNE":M$(7)="JULY":M$(8)="AUG":M$(9)="SEPT":M$(10)="OCT":
M$(11)="NOV":M$(12)="DEC"
135 REM * INTRODUCTORY REMARKS *
140 CLS
150 PRINT @ 15, "CHECK ANALYSIS PROGRAM -- 1979"
160 PRINT:PRINT "THIS PROGRAM CONTAINS ALL CHECKS WRITTEN IN 1979."
170 PRINT "THEY MAY BE ANALYZED BY CATEGORIES AND MONTHS."
180 PRINT "CHECK DATA MAY ALSO BE CHANGED OR UPDATED.":PRINT
185 REM * INITIALIZE VARIABLES *
190 T#=0
200 FOR I=1 TO 12:T(I)=0:NEXT I
299 REM * SELECT ANALYSIS OR FILE UPDATE *
300 INPUT "ANALYZE CHECKS (1) OR UPDATE FILE (2)";S
310 IF S=1 THEN 500
320 IF S=2 THEN 400
330 GOTO 300
399 REM * FILE UPDATE ROUTINE *
400 CLS
410 PRINT @ 25,"FILE UPDATE"
420 PRINT:PRINT "DATA IS ADDED OR CHANGED BY WORKING WITH 'DATA' STATEMENTS."
430 PRINT "THE PROGRAM WILL LIST ALL DATA STATEMENTS FOR YOU."
440 PRINT "ENTER NEW DATA ACCORDING TO THIS FORMAT:"
450 PRINT:PRINT TAB(3);"LINE NO.  'DATA'  CHECK NO., MONTH NO., AMOUNT, CATEGORY NO."
460 PRINT:PRINT "TO INDICATE END OF DATA, ENTER FINAL LINE AS FOLLOWS:"
470 PRINT:PRINT TAB(3);"LINE NO.  'DATA' -1
480 PRINT:PRINT "TO RUN THE PROGRAM, TYPE 'RUN'"
490 PRINT:INPUT "PRESS ENTER WHEN READY";A$
495 CLS:LIST 10000-
499 REM * SELECT CATEGORY FOR ANALYSIS *
500 S=0
510 INPUT "ANALYSIS FOR ONE (1) OR ALL (2) CATEGORIES";S
520 IF S<>1 AND S<>2 THEN 510
530 IF S=1: INPUT "WHICH CATEGORY NUMBER";S1
535 REM * ANALYSIS ROUTINE *
540 READ C
550 IF C=-1 THEN 700
560 READ M,A,C1
565 IF S=2 THEN 580
570 IF C1<>S1 THEN 600
580 T#=T#+A
590 T(M)=T(M)+A
600 GOTO 540
699 REM * DISPLAY TABLE *
700 CLS
710 IF S=2 PRINT "ALL CATEGORIES":GOTO 730
720 PRINT "CATEGORY";S1
730 PRINT "MONTH","AMOUNT","MONTH","AMOUNT"
740 PRINT STRING$(63,"-")
750 FOR J=1 TO 6
760 PRINT M$(J),:PRINT USING G$;T(J),:PRINT,:PRINT M$(J+6),:PRINT USING G$;T(J+6)
770 NEXT
780 PRINT ,, "TOTAL",:PRINT USING G$;T#
799 REM * NEW SELECTION ROUTINE *
800 PRINT:INPUT "DISPLAY CHECK NOS.(1), NEW SELECTION(2), OR REVIEW TABLE(3)";S2
810 RESTORE
820 IF S2=1 THEN 900
830 IF S2=2 THEN 190
840 IF S2=3 THEN 700
850 GOTO 800
899 REM * DISPLAY CHECKS BY NUMBER *
900 CLS: I=1
910 IF S=2 PRINT "CHECKS FOR ALL CATEGORIES":GOTO 930
920 PRINT "CHECKS FOR CATEGORY";S1
930 PRINT "CHECK NO.","MONTH","AMOUNT","CATEGORY"
940 PRINT STRING$(63,"-")
950 READ C
960 IF C=-1 THEN 800
970 READ M,A,C1
980 IF S=2 THEN 1000
990 IF C1<>S1 THEN 1050
1000 PRINT C,M$(M),:PRINT USING G$;A,:PRINT,:PRINT C1
1010 I=I+1
1020 IF I<13 THEN 1050
1030 INPUT "PRESS ENTER TO CONTINUE";A$:I=1
1040 CLS
1050 PRINT "CHECK NO.","MONTH","AMOUNT","CATEGORY"
1060 GOTO 950
10000 REM * DATA SECTION *
10010 REM * DATA FORMAT *
10020 REM * DATA, CHECK NO., MONTH NO., AMOUNT, CATEGORY NO. *
10030 REM * TO END DATA TYPE: DATA -1 *
```

Program listing.



Display of all expenditure categories by months.

A—Check amount
A$—Used in conjunction with INPUT to control scrolling
C—Check number
C1—Expense category number of check
G$—Employed with PRINT USING to align columns
I—Loop counter or index
J—Loop counter
M—Month number in which check was written
M$(n)—List of month names
S—Menu selector
S1—Expense category selector
S2—Menu selector
T(n)—Total of check amounts for each month
T#—Grand total of check amounts for all months in double precision

Table 2. Variables.

their corresponding months (M) and totals them by month in list T(M).

Line 760 contains a PRINT USING G$ statement to align columns. It can be replaced by a simple PRINT T(J) and PRINT T(J + 6) if you do not have PRINT USING. The same comment applies to line 780.

Lines 800-850 permit a new selection to be made. Lines 900-1060 select and display individual check information, including check number, to provide an audit trail back to hard data.

Lines above 10000 are DATA statements to hold check information for the program. They use the following format: Line No., DATA, Check No., Month No., Amount, Category No. (for example, 101000 DATA 100, 1, 10.25,3). End of data is indicated by an additional and final DATA statement: Line NO. DATA-1 (for example, 10200 DATA-1).

Conclusion

This program has greatly facilitated expense accounting for my family. It is now easy to keep track of precisely where all the money has gone. . .even if it is no less painful to find out.

Since I don't have a printer yet, I still have to transcribe information from the CRT screen if I want hard copy (a 13-column pad works ideally).

Although written for the TRS-80, the program is direct enough for conversion to other BASICs. The biggest obstacle, I suspect, will be the number of significant digits displayed—less than seven will cause truncation of the grand totals and impair the usefulness of the program.■

✓ Reader Service—see page 131

Take all the REM statements and/or spaces out of your BASIC programs.

Compress It!

David E. Powers
10 Wilben Ct.
New Hyde Park NY 11040

Two short programs help me edit and compact BASIC software. Sometimes I come dangerously close to the limits of my computer's memory. I have 32K of RAM, but a long disk-oriented program can gobble it up quickly, particularly when there are many string statements and arrays.

The two programs presented here are in TRS-80 Disk BASIC, but should be translatable into other dialects. The command OPEN "I",1,A$ (line 70 in Listing 1 and line 60 in Listing 2) establishes buffer 1 as an input buffer for sequential access to file 1, specified by the name in variable A$. OPEN "O",2,B$ assigns buffer 2 for sequential output to the file named in B$. Both programs require that the target program be saved in ASCII format rather than in the compressed format, which the TRS-80 selects by default. Therefore, a file to be edited must first be saved under the "SAVE 'FILESPEC',A" option. Of course, they may later be resaved in compressed format.

```
1 REM *** DEREMARK
2 REM *** PROGRAM REMOVES REMARK LINES
3 REM *** VERSION 1.0, 1 JUNE 1979
4 REM ***
5 REM ***
6 REM ***
10 CLEAR 2000
20 CLS: PRINT @ 220, "DEREMARK"
30 PRINT @ 320, ""
40 LINEINPUT "ENTER FILESPEC OF TARGET PROGRAM ==> "; A$
50 PRINT: LINEINPUT "ENTER FILESPEC FOR DEREMARKED PROGRAM ==> "; B$
60 CLS: PRINT @ 218, "READING FILE"
70 OPEN "I",1,A$ 'INPUT OF TARGET PROGRAM
80 OPEN "O",2,B$ 'OUTPUT OF DEREMARKED PROGRAM
90 C$="": D$=""
100 LINEINPUT#1,C$
110 D$=C$
120 IF C$="" THEN 100
130 IF F1%=0 THEN CLS: PRINT @ 216, "REMOVING REMARKS": F1%=1
140 FOR I%=1 TO 7 'LOOP FINDS FIRST CHARACTER OF PROGRAM STATEMENT AFTER LINE NUMBER
150 IF MID$(C$,I%,1)<>" " THEN NEXT
160 IF I%=7 THEN 90
170 C$=RIGHT$(C$,LEN(C$)-I%) 'LINE IS STRIPPED OF ITS NUMBER
180 IF LEFT$(C$,3)="REM" THEN C$="": GOTO 90  ' CHECK FOR "REM"
190 PRINT#2, D$ 'IF LINE DOES NOT BEGIN "REM" THEN PLACE IT SEQUENTIALLY IN NEW FILE, OTHERWISE SKIP TO NEXT LINE
200 IF EOF(1) THEN 220
210 GOTO 90
220 CLS: PRINT @ 218, "LOADING FILE"
230 CLOSE
240 LOAD B$
```

Listing 1. Deremark.

Using the Programs

Listing 1 shows Deremark, which automatically removes any line beginning REM from a file and stores it under a new name, supplied by the user in response to statement 50. I usually change the extension, so that a program named Myprog/Txt would be renamed

Myprog/Der. Note that the new program is saved in ASCII format and is immediately loaded into memory for execution or inspection.

Listing 2 is of Despace, which excises spaces from all program lines, except those spaces contained in quoted strings. Also, the program deletes unquoted line feeds if they are not accompanied by carriage returns. I use such line feeds to make listings more readable on the CRT, but they add nothing to the run and often make a hard-copy listing confusing or inelegant.

Like Deremark, Despace requests the name of the target program and asks for a name for the new file. I usually specify the extension "Des" for a program so altered. If I subject a program to both Deremark and Despace, then I often use the extension "Dsr."

Despace should be used with some caution. If, for example, the statement

```
900 FIELD 1, 25 AS COPY$, 10 AS D$
```

is despaced, an error will be generated because "25 AS COPY$" will be rewritten without space, "25ASCOPY$," creating an instance of the reserved word, "ASC." The program crashes when the computer encounters what it understands as the inexecutable statement, "ASCOPY$." Similar results could occur in Example 1.

```
1 REM *** DESPACE
2 REM *** VERSION 1.0.
3 REM ***
4 REM ***
5 REM ***
10 CLEAR 2000
20 CLS: PRINT @ 221, "DESPACE"
30 PRINT @ 320, ""
40 LINEINPUT "ENTER FILESPEC OF TARGET PROGRAM ==> "; A$
50 PRINT: LINEINPUT "ENTER FILESPEC FOR DESPACED PROGRAM ==> "; B$
60 OPEN "I", 1, A$ 'INPUT OF TARGET PROGRAM
70 OPEN "O", 2, B$ 'OUTPUT OF DESPACED PROGRAM
80 CLS: PRINT @ 218, "READING FILE"
90 C$="": D$=""
100 LINEINPUT#1,C$
110 IF C$="" THEN 100  'CHECK FOR NULL STRING
120 IF F2%=0 THEN CLS:  PRINT@ 220, "DESPACING": F2%=1
130 PRINT @ 320, CHR$(31): PRINT @ 320, C$: PRINT 'OPTIONAL LINE
140 FOR I%=1 TO LEN(C$)
150    D$=MID$(C$,I%,1)
160    IF F1%=1 AND ASC(D$)=34 THEN F1%=0: GOTO 180  'RESETS TO REMOVE SPACES AFTER QUOTED STRING
170    IF F1%=0 AND ASC(D$)=34 THEN F1%=1 'SETS FLAG TO BYPASS DESPACE IN QUOTED STRING
180    IF F1%=0 AND (D$=" " OR ASC(D$)=10) THEN D$="" 'REMOVES SPACES OR LINE FEEDS WITHOUT CARRIAGE RETURNS
190    PRINT D$; 'OPTIONAL LINE
200    E$=E$+D$
210 NEXT
220 F1%=0
230 PRINT # 2, E$
240 E$=""
250 IF EOF(1) THEN 270
260 GOTO 90
270 CLOSE
280 CLS: PRINT @ 218, "LOADING FILE"
290 PRINT:  PRINT
300 LOAD B$
```

Listing 2. Despace.

Despace takes some time to run, but it is faster than manual editing. While it is running, the user may watch it operating on the CRT. If the target program is long, you may wish to forego this option by deleting lines 130 and 190. Like Deremark, Despace saves the new file in ASCII format and loads it immediately for inspection or execution.

If you list a despaced program directly from DOS, you will notice that spaces after line numbers are also deleted. This does not represent a problem and is corrected when the program is loaded. Spaces are also reinserted after line numbers on subsequent saves.

These two programs provide more running room for a particularly long program. However, to avoid difficulty in future editing, I find it advisable to keep on hand a hard copy of the file that has not been despaced or deremarked. I have saved as much as 25 percent of RAM usage and disk space by careful use of Deremark and Despace.■

```
910 IF BE = LO GOTO 300                          (LOG)
920 IF D = R AND OM = 5 THEN PRINT R, OM  (RANDOM)
930 IF C = 0 D = 0                               (0D)
```

Example 1.

Format program printouts with this software routine.

LPRINT Formatter

E.M. McCormick
13100 Chapman, 3-113
Garden Grove CA 92640

This program allows you to get neatly paged program listings, instead of one long stack or roll of printout, from your TRS-80. It lists 50 lines of BASIC code per page complete with a header and page number. It even provides horizontal lines 11 inches apart to facilitate cutting the 8½-inch-wide Teletype paper into individual sheets. It is great for documenting programs.

The program listing was printed using a Teletype. The program is in BASIC and must be added to the BASIC program to be listed. A utility program such as RACET's Proload permits this without disturbing the existing program. Lacking this ability, you can start programming with this program as the starter. The print program uses about 1600 bytes.

It is executed by entering RUN 50000. (Obviously, it can be relocated.) The program asks for the header line to be printed on each page. It then asks for the lowest and highest line number to be printed and then prints out as many pages as needed.

This program takes care of the limitation of some printers in that it provides an automatic carriage-return/line-feed when the number of characters printed exceeds the print line length. Thus, long lines can be used in programs without unreadable black blobs on the right side of the page. This is illustrated by several lines in the program itself.

This program presumes that you use a printer that provides a line feed after each carriage return; a printer that did not do this would require program modifications. Note that this formatting program will read through any other data statements present until it finds its own data statements.

Incidentally, a Teletype provides an inexpensive means of printing since it can be driven from a TRS-80 without the expansion interface. I use Small Systems Hardware RS-232 printer interface, which sells for about $50. Other similar devices are available. They permit using the LLIST and LPRINT commands, but a program such as

Program listing.

```
 50000 REM PRINTS ØUT 50 LINES PER PAGE ØF BASIC CØDE WITH HEADER AND PA
GE NUMBER; WRAPS ARØUND ØN LØNG LINES.  E. M. MC CØRMICK, 7-9-79
 50010 CLEAR 200:DIM D$(124):E=17129:PØKE 16553,255
 50020 READ D$(1):IF D$(1)<>"END" THEN 50020
 50030 READ D$(2):IF D$(2)<>"FØR" THEN 50020
 50040 FØR K=3 TØ 124:READ D$(K):NEXT K
 50050 IF D$(123)<>"MID$" PRINT "DATA ERRØR":END
 50060 INPUT"HEADER";A$:INPUT "LØ,HI";LØ,HI
 50070 EN=0:PG=1:LPRINT CHR$(10)
 50080 LPRINT STRING$(72,"-"):LPRINT STRING$(3,10)
 50090 IF EN=1 LPRINT STRING$(8,10):END
 50100 LN=0:KL=36-LEN(A$)/2
 50110 LPRINT STRING$(KL," ");:LPRINT A$;:LPRINT STRING$(KL-2," ");
 50120 LPRINT USING "###";PG:LPRINT STRING$(3,10)
 50130 S=PEEK(E)+256*PEEK(E+1):L=PEEK(E+2)+256*PEEK(E+3)
 50140 IF L=4 ØR L>HI THEN 50330 :ELSE IF L<LØ THEN 50290
 50150 LN=LN+1:PC=LEN(STR$(L)):LPRINT L;
 50160 FØR M=E+4 TØ S-2:M1=PEEK(M)
 50170 IF M1<127 ØR M1>251 THEN 50220
 50180 PC=PC+LEN(D$(M1-127))
 50190 IF PC<72 THEN 50210
 50200 LPRINT:PC=LEN(D$(M1-127)):LN=LN+1
 50210 LPRINT D$(M1-127);:GØTØ 50270
 50220 IF M1=10 THEN 50250 :ELSE PC=PC+1
 50230 IF PC<72 THEN 50260
 50240 LPRINT
 50250 PC=0:LN=LN+1
 50260 LPRINT CHR$(M1);
```

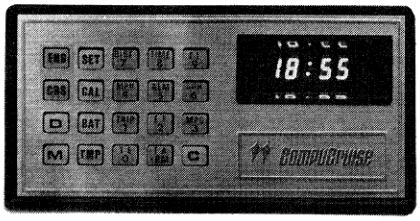
this is needed to format program listings.

The colons in lines 50140 and 50220 are not to be entered in the program. They are printed out to show how the TRS-80 handles the ELSE option.

Being in BASIC, this program slows down the print speed somewhat. However, with the Teletype, which is not noted for its speed anyway, it is well worth the extra time to get programs that come out neatly page by page. ■

```
 50270 NEXT M
 50280 LPRINT:IF LN=>50 THEN 50300
 50290 E=S:GØTØ 50130
 50300 KM=56-LN
 50310 FØR KL=1 TØ KM:LPRINT:NEXT KL
 50320 E=S:PG=PG+1:GØTØ 50080
 50330 EN=1:GØTØ 50300
 50340 DATA END,FØR,RESET,SET,CLS,CMD,RANDØM,NEXT,DATA,INPUT,DIM,READ,LE
T,GØTØ,RUN,IF,RESTØRE,GØSUB,RETURN,REM,STØP,ELSE,TRØN,TRØFF,DEFSTR,DEFIN
T,DEFSNG,DEFBDL,LINE,EDIT,ERRØR,RESUME,ØUT,ØN,ØPEN,FIELD,GET,PUT,CLØSE,L
ØAD,MERGE,NAME,KILL,LSET
 50350 DATA RSET,SAVE,SYSTEM,LPRINT,DEF,PØKE,PRINT,CØNT,LIST,LLIST,DELET
E,AUTØ,CLEAR,CLØAD,CSAVE,NEW,"TAB(",TØ,FN,USING,VARPTR,USR,ERL,ERR,STRIN
G$,INSTR,PØINT,TIME$,MEM,INKEY$,THEN,NØT,STEP,"+","-","*","/","[",AND,ØR
,">","=","<",SGN,INT,ABS
 50360 DATA FRE,INP,PØS,SQR,RND,LØG,EXP,CØS,SIN,TAN,ATN,PEEK,CVI,CVS,CVD
,EØF,LØC,LØF,MKI$,MKS$,MKD$,CINT,CSNG,CDBL,FIX,LEN,STR$,VAL,ASC,CHR$,LEF
T$,RIGHT$,MID$," "
```

If you enjoy driving, you're going to get a COMPUCRUISE. Once you see what it can do, you just won't be able to live without it.



This gadget fits into most dashboards . . . no strain even in a tiny sports car like the Mazda RX-7 . . . and once you have it, every trip is like flying a 747. The darned thing tells you the time, how fast you're going, how far you've been on this trip or since the last regassing, how many miles per gallon you're getting, either at the instant or the average on the trip . . . or gallons per hour at the moment or for the trip . . . temperature outside . . . inside (or coolant temperature, if you prefer) . . . oh, it has an elapsed time for the trip, a stop watch, lap time, an alarm . . . how much further for your trip, how many gallons more the trip will take, how much longer for the trip at your present average speed . . . yes, it gives you your average speed for the trip. You prefer it in metric, no strain . . . liters remaining, etc. Did we mention that it also has cruise control either at a speed set on the control board or at whatever speed you are traveling? The Compucruise will keep you busy and entertained during any trip . . . telling you more than you will ever want to know.

The Compucruise is not difficult to install . . . though it does connect to everything except the cigarette lighter. Until you've tried computerized travel, you haven't found out how much fun driving can be. It will work on any car not having fuel injection . . . and there is a front-wheel drive accessory gadget available for only $4.40-#P001 (regularly $5.50).

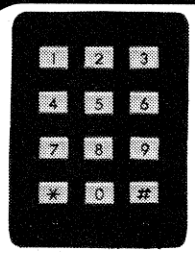
The price for the Compucruise is regularly $199.95 . . . and a bargain at that price. We'll sell you one of these fantastic gadgets for $159.95 with cruise control (Model 44-#P002), and $127.95 without (Model 41-#P003).Send money . . . and start having fun!

MOM'S

MAIL ORDER MICROS

158

Dept F80 • PO Box 427 • Marlboro NH 03455



Steal Stopper

Ever had your car stolen?
The first reaction is one of disbelief . . .
. . . you *know* it was *right there*!

What you want is a modern combination lock on your ignition . . . The Steal Stopper. It's easy to install and almost impossible to defeat. *You* can by-pass it, if you want, for parking attendants or a car wash. Other than that, you set up a secret four-digit code and only you will then be able to start the car . . . even if you leave the keys in the ignition. Of course, if you forget the number . . . . . .

This protection retails for $50 . . . but we have a special for you at $39.95. Don't procrastinate. Order #P004.

PROCESSOR TECH HARDWARE

9 **Processor Tech Video Display Module**—Memory mapped video for S-100, excellent condition. #D009-$144 each.

1 **Processor Tech ROM/RAM card**—Contains 1K RAM, empty slots for 12K ROM, S-100, excellent condition. #D007-$76.

3 **Processor Tech 3P + S I/O card**—Three parallel ports, one serial port, S-100, excellent condition. #D008-$116 each.

2 **Processor Tech 16K Static RAM card**—450 nanoseconds, S-100, excellent condition. #D010-$199 each.

4 **Processor Tech 2K ROM card (kit)**—Empty board, S-100, new. #D011-$17.20 each.

4 **Processor Tech 2K ROM card**—Empty board, S-100, excellent condition. #D012-$25.20 each.

PROCESSOR TECH SOFTWARE

2 **Processor Tech Extended Disk BASIC**—This is full disk BASIC on 8" disk for HELIOS II disk controllers with PTDOS and greater than 16K. #D015-$80 each.

1 **Extended Disk BASIC on cassette**—This is the same as previously mentioned for the Disk BASIC from Processor Tech. Needs more than 16K. #D017-$72.

8 **Extended Cassette BASIC**—This includes all file operations, advanced functions for doing more than playing games; for SOLOS, CUTTER, and CONSOL Monitors. #D016-$22 each.

19 **BASIC 5 from Processor Tech**—This is a simple BASIC for a SOLOS, CUTTER, or CONSOL Monitor and 8K of RAM. #D013-$11.60 each.

4 **Processor Tech GAMEPAC**-for above BASIC—Various simple games. #D014-$11.60 each

PROCESSOR TECH SOL

4 **SOL Computers**—8K Memory, S-100, excellent condition. #D004-$980 each.

19 **TREK-80** on cassette for SOL—This is one of the best real time space games available today; needs 8K. #D005-$11 each.

9 **Electric Pencil** on cassette for SOL—Word processor, needs 8K. #D006-$80 each.

MICRO TERM ACT TERMINAL

2 **MicroTerm ACT Terminal**—Need a video monitor, up to 600 Baud, good condition. #S035-$200 each. **SPECIAL PRICE.**

COMPUCOLOR HARDWARE

2 **Compucolor Computer 8001** (use as computer or 75 MHZ Color Monitor)—8K RAM, BASIC and DOS in ROM, good condition. #S025-$1500 each. **SPECIAL PRICE.**

2 **Compucolor MiniFloppy**—5¼ inch, good condition. #S026-$525 each. **SPECIAL PRICE.**

4 **Compucolor 8K RAM card**—Static RAM, good condition. #S027-$200 each. **SPECIAL PRICE.**

3 **Compucolor Floppy Tape Drive**—Uses eight track cartridges, good condition. #S028-$70 each. **SPECIAL PRICE.**

BALLY GAMES

2 Bally VideoCode Cassettes—They consist of two games: Speed Math and Bingo Math. #D029-$16 each.

ABACUS

112 Abacus Paperweight—Hefty, brass, excellent condition. #S024-$3 **SPECIAL PRICE.**

POLYMORPHIC HARDWARE

8 **PolyMorphic Video Terminal Interface**—Memory mapped video for S-100 bus, good to excellent condition. #S044-$150 each. **SPECIAL PRICE**

1 **PolyMorphic Video Terminal Interface** (used)—As above. #S045-$100.00. **SPECIAL PRICE.**

3 **PolyMorphic CPU**—8080 CPU for S-100 with Poly monitor in ROM, excellent condition. #S046-$100 each. **SPECIAL PRICE.**

4 **PolyMorphic Cassette Interface**—Plugs into Poly CPU only, new. #D047-$29.60 each.

4 **PolyMorphic Printer Interface**—Plugs into Poly CPU only, new. #D048-$29.60 each.

4 **PolyMorphic S-100 Cabinets**—Nice 5 slot S-100 mainframe, good to excellent condition. #D049-$248 each.

POLY-88 ACCESSORIES

3 **Parallel Keyboard**—Good condition. #S001-$50 each. **SPECIAL PRICE.**

1 **Software on 5¼"**—This is system software that requires a PolyMorphic Disk Controller in a System 88 Cabinet. #D002-$100.

1 **Electric Pencil**—#D003-$80.

NORTH STAR HARDWARE

1 **North Star Floating Point BASIC card**—With special BASIC, new. #D0059-$287.

2 **North Star Floating Point BASIC card (kit)**—With special BASIC, S-100, new. #D0060-$207.

8 **North Star Floppy Disk Controller card**—Single density, S-100, new. #D0061-$248.

120 **S-100 Edge Connector**—Gold Contacts, new. #D050-$.20 each.

6 **Extender Card for S-100 (kit)**—New. #D051-$24 each.

QUASAR VIDEO TAPE RECORDER

Records color or black and white up to 2 hours, off the air, or records one program while you watch another. Camera input to make your own tapes, pause control for editing. These $1,000 VTRs were demonstrated and come with new warranty. **SPECIAL PRICE** is $475—extra bonus—free $100 worth of new tape with each order. Limited supply. Order #P005.

MUSIC

2 **Software Technology S-100 Music system** on cassette—This is an S-100 Music system: contains the proper hardware. #D0058-$19.60 each.

COMPUTER TRAINER

1 **IASIS COMPUTER-IN-A-BOOK**—8080 Microcomputer, comes built into training manual, excellent condition. #D020-$240

ICOM DISK DRIVE ACCESSORIES

1 **ICOM Dual Disk Drive**—Single density, 512K storage, S-100 controller, includes CP/M ROM, good condition. #S030-$1500. **SPECIAL PRICE.**

1 **ICOM PROM and 8" Disk for SOL FDOS**—This disk requires an ICOM S-100 Disk Controller installed in an S-100. #D031-$160.

1 **ICOM CP/M on 8" Disk for S-100**—Requires an ICOM S-100 controller in an S-100 cabinet, #D032-$100.

2 **ICOM FDOS-II on 8" Disk for S-100**—Requires an ICOM S-100 controller in an S-100 cabinet, no documentation. #D033-$180 each.

1 **ICOM FDOS-II on 5¼" Disk for S-100**—Requires an ICOM S-100 Mini-Floppy Controller in an S-100 cabinet. #D034-$168.

MOM'S MAIL ORDER MICROS

Dept F80 • PO Box 427 • Marlboro NH 03455

✓158

20% OFF Inventory Clearance SALE

IMSAI HARDWARE

1 **IMSAI 8800 Mainframe S-100**—Excellent condition. #D0047-$839.
1 **IMSAI 80/15 S-100 Development System**—Partially assembled, needs a CPU card, excellent condition. #S0048-$525 as is. **SPECIAL PRICE.**
1 **IMSAI 80/15 S-100 Development System**—Kit, mainframe cover missing, needs a CPU card, excellent condition. #S0049-$500 as is. **SPECIAL PRICE.**
2 **IMSAI 4K RAM card**—S-100, good condition. #D0055-$89.60

IMSAI SERIAL I/O CARDS

1 **IMSAI Serial I/O card 2-2**—Two serial ports, full RS-232 control, S-100, good condition. #D0050-$188.
5 **IMSAI Serial I/O card 2-2 (kit)**—Two serial ports, full RS-232 control, S-100, new. #D0051-$124.
2 **IMSAI Serial I/O card 2-1 (kit)**—One serial port, full control RS-232 control, S-100, new. #D0052-$100.

IMSAI PARALLEL I/O CARDS

1 **IMSAI Parallel I/O card 4-4**—Four parallel ports, S-100, excellent condition. #D0053-$186.
1 **IMSAI Parallel I/O card 4-1 (kit)**—One parallel port, S-100, new. #D0054-$74.40.

IMSAI SOFTWARE

2 **IMSAI IMDOS V2.02** on 8" Disk for S-100—No documentation, but this is apparently IMSAI's version of CP/M for S-100 systems with an IMSAI Disk Controller. #D0056-$96 each.
2 **IMSAI BASIC 9A**—This BASIC is for an S-100 system with a TARBELL cassette interface. #D0057-$22.
1 **TARBELL Cassette Interface (kit)**—Kansas City Interface, Tarbell Phase encoding, S-100, new. #D0064-$96.

PHONE INTERFACE

6 **Novation Modem #3102A**—Connects to any phone, originate only, good condition. #S021-$165 each. **SPECIAL PRICE.**
2 **Novation Modem #3103A**—Connects to any phone, answer/originate, good condition. #S022-$200 each. **SPECIAL PRICE.**
1 **Novation Modem #43**—Connects to any phone, originate only, good condition. #S023-$165. **SPECIAL PRICE.**

VECTOR GRAPHIC S-100

3 **Vector Graphic ROM/RAM card**—12K empty ROM sockets, 1K RAM, excellent condition. #D0078-$119.60.
5 **Vector Graphic Analog Interface**—Allows hobbyist to interface analog experiments, S-100, new. #D0079-$79.20.
1 **Vector Graphic Analog Interface (kit)**—As above, S-100, new. #D0080-$55.60.

TDL ZAPPLE

2 **TDL Z-80 8K BASIC**—This is for a Z-80 system with a ZAPPLE Monitor. #D0071-$30.
1 **TDL ZAPPLE 2K Monitor**—Nice monitor to start a system with. #D0072-$16.
3 **TDL ZAP 1K Monitor**—Simple monitor. #D0073-$12.

TDL HARDWARE

2 **TDL ZPU Z-80 CPU**—S-100 CPU with adjustable clock 0-4 MHz, good condition. #D0062-$159.
1 **TDL Systems Monitor Board**—Contains system software including monitor, S-100, good condition. #D0063-$140.

HONEYWELL

15 **Honeywell ASR-33 Communications Consoles** with TTY, paper tape reader and punch. Used, working when removed from service. Shipped freight collect or you pick up. Weight 300 lbs. $395. Order #P006.

TDL TEXT PROCESSOR

2 **TDL Text Output Processor**—Text processor. #D0074-$16.
1 **TDL Text editor**—Nice editor. #D0075-$12.
1 **TDL Memory Test**—Simple utility. #D0076-$4.

TDL SOFTWARE-DISK

1 **TDL FDOS & SuperBASIC on 8" Disk**—This requires an ICOM Disk Controller and at least 20K of memory, plus a ZAPPLE Monitor in an S-100 Cabinet (Altair, IMSAI, etc.). #D0065-$137.
1 **TDL System Software on 5¼" disk**—This set of system software requires a North Star Disk Controller, a TDL Systems Monitor Board I, and consists of 12K BASIC, Relocator/Linking Loader, Z-80 Editor, and Text Processor. #D0066-$183.
1 **TDL System Software on 5¼" disk**—This is the same as above, but does not require the Systems Monitor Board I. #D0067-$200.
1 **TDL System Software on 5¼" disk**—Again, as above, but requires a HELIOS Disk Controller and the TDL Systems Monitor Board II (not I). #D0068-$183.

TDL SOFTWARE-CASSETTE

1 **TDL 12K BASIC V2.1 on cassette**—Same as disk version. #D0069-$22.
1 **TDL System Pack**—This is the same as the Disk Pack. #D0070-$136.

MOUNTAIN HARDWARE

7 **Mountain Hardware AC Controller**—Remote AC outlet control, S-100, new. #D040-$100 each.
4 **Mountain Hardware remote outlet**—Remote module for above, two channels, new. #D041-$72 each.

HEURISTICS SPEECH LAB

1 **Heuristics Speech Lab**—S-100, used, fair condition. #S042-$100 as is. **SPECIAL PRICE.**
4 **Heuristics Speech Labs**—S-100, new. #D043-$151 each.

SHUGART MINIDISK DRIVES

1 **Shugart MiniDisk Drives**—With case and power supply, includes cable for North Star, good condition. #S036-$320. **SPECIAL PRICE.**
4 **Shugart MiniDisk Drives**—No cabinet, good condition. #S037-$300 each. **SPECIAL PRICE.**
4 **Power Supplies** for above—good condition. #S038-$30 each. **SPECIAL PRICE.**

Prices include 20% discount. **SPECIAL PRICE** includes more than 20% discount.

Quantities are limited, immediate refund if ordered item is no longer available. Please read terms below.

TERMS: FOB Marlboro, NH USA. Limited stock; everything quaranteed as described; you pay postage on returns. PRINT orders clearly. Minimum order $10 plus $2.50 shipping and handling charge in USA only. DOUBLE THAT ELSEWHERE. Orders over $50 add 5% for shipping in USA; 10% elsewhere (we will refund excess). Orders shipped UPS or insured mail only. Send US funds by check or money order. For credit card purchases, add 4%, list AE, MC or VISA, number, and expiration date. Mail to MOM's, Department F80, PO Box 427, Marlboro NH 03455

Condition of Inventory:
New = original container
Excellent = new, but not in original container
Good = tested or used in store

Qty	Catalog #	Description	Unit Price	Total

Enclosed $ _____

Bill: ☐ AE ☐ MC ☐ VISA

MOM'S MAIL ORDER MICROS ✓158
Dept F80 • PO Box 427 • Marlboro NH 03455

Shipping & Handling _____

Credit Card (+4%) _____

Total _____

Signature _____ Exp Date _____

Name _____

Address _____

City _____ State _____ Zip _____

Ship: ☐ UPS ☐ Insured mail

✓ Reader Service—see page 131

A memory refresh for the absent-minded.

Disk Index

Ron Cheshire
1230 Mono Ct.
Ridgecrest, CA 93555

When I finally got my TRS-80 disk drive, I thought, no more waiting hours to load my programs from tape.

To load a program from disk into RAM, you must first go to the DISK BASIC mode. But if you have several programs on a disk and want to run different programs, you must go back to the DOS mode and look at the directory to find the file-spec names—after you return to DISK BASIC mode.

His Problem

My problem is a short memory and I sometimes forget the name of a program. It occurred to me that the computer is supposed to handle these kinds of problems. After all, isn't that why I bought it in the first place?

Well, if you have a similar problem, cheer up, Radio Shack did not forget us. The TRS-80 lets you use a RUN command as a program statement. This allows you to link together several programs.

Once I found this out, the rest came naturally. Since I have only several months experience with computers and programming, I assembled an index that is probably longer than need be, but I couldn't figure out how to shorten it. The program loads onto each disk and should be the first that you would run. All of the programs on that disk are displayed in the form of an index, and you may select any of them by entering a letter/number combination and hitting ENTER.

Further Step

One further step adds the statement "1000 RUN "INDEX" in place of the end statement for each program. This brings up the index for another selection.

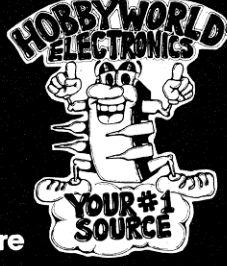
There are all sorts of possibilities for using the RUN as a statement in programs, can you think of some others? ■

Program Listing

```
5 CLS
10 PRINT"INDEX FOR DISK # 3:
20 PRINTTAB(10)"C1.  BBALL          BOUNCING BALL
30 PRINTTAB(10)"C2.  CIVILWAR       CIVIL WAR SIMULATION
40 PRINTTAB(10)"C3.  DECISION       DECISION MAKER
50 PRINTTAB(10)"C4.  LIFEXPCT       LIFE EXPECTANCY PROGRAM
60 PRINTTAB(10)"C5.  DAYWEEK        GIVES YOU THE DAY OF THE WEEK
70 PRINTTAB(10)"C6.  BANDIT         ONE ARMED BANDIT.( SLOTS )
80 PRINTTAB(10)"C7.  MADLIB         ADD THE WORDS FOR A STORY
90 PRINTTAB(10)"C8.  GUESSNUM       THE STANDARD GUESSING GAME
100 PRINTTAB(10)"C9.  NUTTY1         INTRODUCTION TO COMPUTERS
110 PRINTTAB(10)"C10.  NUTTY2","MORE OF THE SAME
120 PRINTTAB(10)"C11.  DEPTHCHG       DEPTH CHARGE GAME
125 PRINT:PRINT"TO RUN A PROGRAM ENTER IT'S 'C' LETTER/NUMBER
126 PRINT"COMBINATION.  ELSE TYPE 'NEXT'
130 GOSUB5000
140 CLS
150 GOTO5
5000 INPUT@$
5010 IF@$="C1"THENRUN"BBALL
5020 IF@$="C2"THENRUN"CIVILWAR
5030 IF@$="C3"THENRUN"DECISION
5040 IF@$="C4"THENRUN"LIFEXPCT
5050 IF@$="C5"THENRUN"DAYWEEK
5060 IF@$="C6"THENRUN"BANDIT
5070 IF@$="C7"THENRUN"MADLIB
5080 IF@$="C8"THENRUN"GUESSNUM
5090 IF@$="NEXT"THENRETURN
```

HOBBY WORLD® ELECTRONICS

America's Largest Mail-Order Computer Store



CALL TOLL FREE:
(800) 423-5387 USA
IN CALIF:
(800) 382-3651
LOCAL & OUTSIDE USA
(213) 886-9200

BBB MEMBER

CENTRONICS 730-1
MINIPRINTER
Provides greater flexibility than any other printer we've offered. Prints on either roll, sheet, or fanfold; allows you to print on your letterhead, payroll checks, etc, taking the place of many printers. Simple design and construction ensure reliability. Sold nationally by a large electronics chain for $100 over our price! 7 x 7 dot matrix for crisp, clean characters. Wt 12 lbs. **$995**
Cat No. 2213

16K MEMORY ADD-ON
$75 2 for $140
For TRS-80, Apple, Exidy
Installs in minutes, no special tools required. Complete with detailed instructions. Wt 4 oz.

Cat No.	Description
1156	For TRS-80 keyboard unit
1156A	TRS-80 without buffered cable
1156B	TRS-80 with buffered cable
1156C	For Apple
1156D	For Exidy Sorcerer

SARGON II
Hayden
The champ of champs! Surpasses Microchess, and even Sargon I! Offers complex moves, 7 levels of play, activity indicator, a special "hint" mode, plus more! The best chess program ever!
Cat No. 2082 TRS-80 $29.95
Cat No. 2083 APPLE II $29.95

PENTAMINOES
Programma
A fascinating puzzle game with WILD sound effects and FAST graphics. Four levels of play.
Cat No. 2136 TRS-80 L2 16K $9.95

ADVENTURE
Explore an almost endless maze of treasures and pitfalls. Challenging and fun!
TRS-80 L2, 16K.
Cat No. 1723 $14.95

MACHINE LANGUAGE MONITOR
RSM-2. Monitor and disassembler, interract directly with the Z80.
TRS-80 L2, 16K
Cat No. 1189 $26.95
TRS-80, L2, 16-48K, w/disk
Cat No. 1190 $29.95

NEWDOS+
Better than TRSDOS! Allows DIR while in basic, and then returning to basic, DIRCHECK, DISASSEM, EDTASM, and SUPERZAP! A must for disk owners!
Cat No. 1549 TRS-80, L2 + disk $99

BARRICADE
Similar to breakout. A real time game, with options of speed, balls, angle, etc.
TRS-80 L1/L2 16K.
Cat No. 1362 $14.95

CRIBBAGE
Plays according to Hoyle's Rules. You vs the computer.
Cat No. 1179 TRS-80, L2, 16K $9.95

PRINTERS

Cat No.	Description	Wt	Price
1342*	Anadex DP8000	49 lbs	$925
1844*	Centronics S1	13 lbs	$515
1843*	Centronics P1	13 lbs	$445
2210*	Centronics 700-1	75 lbs	$1895
2211*	Centronics 702-2	60 lbs	$1995
2212*	Centronics 761-1	85 lbs	$1950
2213*	Centronics 730-1	10 lbs	$995

Verbatim
5¼ DISKETTES
The most popular diskette in the world!
$29/box of 10
2 for $55

Cat No.	Type	Description
1147	525-01	Soft, TRS-80
1148	525-10	10 hole hard, Apple, Northstar, H8
1149	525-16	16 hole hard, Micropolis

SOUNDWARE
Add music and sound effects to your programs. Complete with software and hardware. Installs in seconds.
Cat No. 1898 TRS-80, L2 $29.95

TRS-80 UTILITY 2
CFETCH searches tapes for file manes, merges programs with consecutive line numbers. CWRITE combines subroutines, basic or machine language.
Cat No. 1984 TRS-80 L2, 16K $7.95

SYSCOP
Duplicates SYSTEM tapes (for backup).
TRS-80 L2, 4K.
Cat No. 1681 $9.95

TRS-232 SERIAL INTERFACE
Software driven RS232 output port for printers. Installs in seconds.
TRS-80 L2, 4-48K.
Cat No. 1199 $49.95

TRS-80 UTILITY I
Duplik, duplicates basic, assembly, and machine language programs, copies level 1 onto level 2. Renum, for renumbering basic programs.
Cat No. 1983 TRS-80 L2, 16K $7.95

ANDROID NIM II
Super improved version of NIM. Constant excitement!
TRS-80, L2, 16K.
Cat No. 1686 $14.95

LEVEL III BASIC
Gives your TRS-80 the power of a full size system. Disk commands, advanced editing, etc.
TRS-80 L2, 16K
Cat No. 1332 $49

STAR TREK III
The most advanced version we've seen!
TRS-80 L2, 16K
Cat No. 1041 $14.95

BRIDGE CHALLENGER
You and dummy play against the computer in regular contract bridge. Either you or comp sets up.
TRS-80 L2, 16K
Cat No. 1195 $14.95

TRIBBLE TRAP
Programma
All new tribble trap by the original author! High speed machine language graphics. Can you capture the elusive beast?
Cat No. 2139 TRS-80 L2 16K $6.95

PACHINKO
Programma
Simulates the Pachinko machine and the ball's random path downward. It isn't hard to run out of balls in the beginning, but if you get a good reserve built up, you can almost go on indefinitely. To break the bank, you must get 500 balls to your credit. $6.95
Cat No. 2143 TRS-80, L2, 16K

TRS-80 FORTH
Programma
A compact threaded language. Vocabulary based, user may tailor the system to the needs and structure of a specific application. User defines new words, which may in turn define still more complex applications.
Cat No. 2133 TRS-80 L2, 16K $34.95

MAILING LIST & LABELS
Programma
Cassette based, comprehensive. Features include automatic sort routines, additional sort field, hard copy print capability, and many edit features. $9.95
Cat No. 2144 TRS-80 L2 16K

HUNT
Programma
A new class of game where you invent your own characters, rules, and so on. All the best of Adventure, Dungeon, Quest, Dragons all rolled into one!
Cat No. 2145 TRS-80 L2, 16K $19.95

TRS-80 PIE
Programma
2 dimensional, cursor based editor. Features blinking cursor, cursor movement of up/down/left/right, plus tabs; character insert/delete; string search forwards and backwards; plus more and more! The best editor available for the TRS-80!
Cat No. 2134 TRS-80, L2, 16K $19.95

DATA BASE MANAGEMENT 5
Programma
A disk file system. Allows cross-filing, zip code filing. Sets up parameters for you. Requires 32K + disk + TRSDOS.
Cat No. 2146

TIELINE
Programma
Turns your TRS-80 into a timeshare mainframe, smart terminal, and allows you to send/receive BASIC and data via phone. Requires modem, RS232, 32K, DOS 2.1 or later.
Cat No. 2137 $6.95

MAZE-80
Programma
Maze runner is here! Try to beat the computer in running MAZE-80. Each one is different and more difficult. Features fast graphics and tension building sound! $9.95
Cat No. 2141 TRS-80 L2, 16K

TRS-80 SMART TERMINAL
- *Quickly convert your TRS-80 with no modifications!*
- *Use with or without disk!*

Allows you to gather and preformat data, store it (on disk with disk version), then transmit it to a timesharing computer for processing. Processed data from the timesharing computer may then be sent back to the TRS-80. Also allows your TRS-80 to set up a data base and transmit to another TRS-80 via telephone lines. Terminal program may be user-customized by redefining translation tables. Conversion takes only minutes!
Features: CONTROL key, ESC key, REPEAT key, a RUN key, and a functioning BREAK key. Also lets you list incoming data on your lineprinter. You can reprogram the RS-232-C switches from the keyboard, making baud rate changes simple. Fully upper/lower case keyboard and video driver are included, plus instructions on how to make a simple hardware modification to display upper/lower case letters (This change is optional. Unmodified TRS-80 will display capital (upper case) letters only.). The cursor control format includes clear screen, backspace, advance, down and up space, clear to end of line and home, using the most common control character format currently in use (similar to CDC terminals).

Cat No. 2216 TRS-80, L2, 16K, Modem, RS-232 $49.95
Cat No. 2217 TRS-80, L2, 32K, Modem, RS-232, disk $79.95

FULLER ELECTRONICS
TRS-80 DUAL CASSETTE CONTROL CENTER $74.95
- *Aids and simplifies cassette loading!*
- *Allows you to listen to cassettes as they are loading!*

No more plugging and unplugging cassette cables! Saves wear and tear on your computer system, adds the flexibility that you need for level II data storage! Internal speaker allows you to listen to CSAVE and CLOAD, also allows you to listen to fast forward and rewind (CTR-41). Independent volume control lets you adjust monitor volume without affecting loading volume. Built-in LED's let you know status of one or both recorders. Now you can simultaneously CSAVE with both recorders; CLOAD with either recorder; cross tape with both recorders. Tested and reviewed in March '79 Creative Computing. Compact, only 4" x 3¼" x 2¼" high. Completely assembled and tested Wt 16 oz.

Cat No. 1973 For CTR-41
*Cat No. 1974 For CTR-80 or CTR-21

FULLER ELECTRONICS RF-3
TRS-80 I/O & AUXILIARY CONTROL CENTER $39.95
- *Controls lights, video recorders, appliances thru your TRS-80!*
- *With complete application software and documentation*
- *Built-in relay protects your TRS-80 relay!*
- *Controls from fractions of a second to months!*

The most versatile TRS-80 accessory available today! Offers features found only in units costing much more. Now you can control lights when you're not home, at random times! Turn video recording equipment on and off at preset times, even months in advance. Internal beeper can be used to signal the end of a long sort. It also lets you know with one beep, two beeps, etc, exactly what part of the program you're in! Also allows you to check your RS-232 output without turning on the printer. Built-in internal relay protects and bypasses control current from your sensitive TRS-80 relay. Now you can even operate 6 or more cassette recorders safely for CSAVE. NOTE: 110VAC CONTROLLER IS NOT INCLUDED. A simple schematic and parts list is supplied. Size: 2" x 4" x 1½".

Wt 12 oz.
Cat No. 1972

HAZELTINE TERMINALS
Standard interfacing to most systems!

Cat No.	Model	Description	Price
2167*	1400	53 key, 80 x 24 display, dumb	$840
2168*	1500	Intelligent, 74 key, 80 x 24	$1195
2169*	1520	Intelligent, 81 key, 95 char.	$1588

HOW TO ORDER
Pay by check, Mastercharge, Visa, or COD. Charge orders please include expiration date. Foreign pay in U.S. funds. Order by phone or mail, or at our retail. MINIMUM ORDER $10. please include phone number and magazine/issue you are ordering from. Prices valid thru last day of cover date. SHIPPING: USA:: add $2.00 for the first 2 lbs. For ground add 35¢ for addt'l lb. For air add 75¢ for addt'l lb. FOREIGN: surface: add $3.00 for first 2 lbs, 60¢ per addt'l lb. Air: add $11.00 for first 2 lbs, $5 for each addt'l lb. CODs $1 addt'l. Guaranteed satisfaction for 120 days or your money back! Not responsible for typographical errors. Some items subject to prior sale. We reserve the right to limit quantities.

SEND FOR FREE CATALOG FEATURING:
Computers and accessories, disk drives, printers, integrated circuits, LEDs, semiconductors, books, software, connectors, plus more and more! The widest selection at the lowest prices! Circle our reader service number or phone/write today for your copy!

19511 Business Center Dr. Dept. V2 Northridge, Ca. 91324

Disks and Printers for the TRS-80 . . .

Add 1 or 2 megabytes of disk storage to your TRS-80 with a Micromation MEGABOX® interface — double-density 8'' drives

(2 megabytes by utilizing dual-headed double-sided drives).

The Micromation MEGABOX® provides for up to 4 megabytes of storage. It has provision for adding 32K of RAM to your TRS-80, giving you the opportunity of having a 48K system without purchasing an expansion interface. The Micromation MEGABOX comes with an RS232 serial interface, which can be used by a serial printer or a modem.

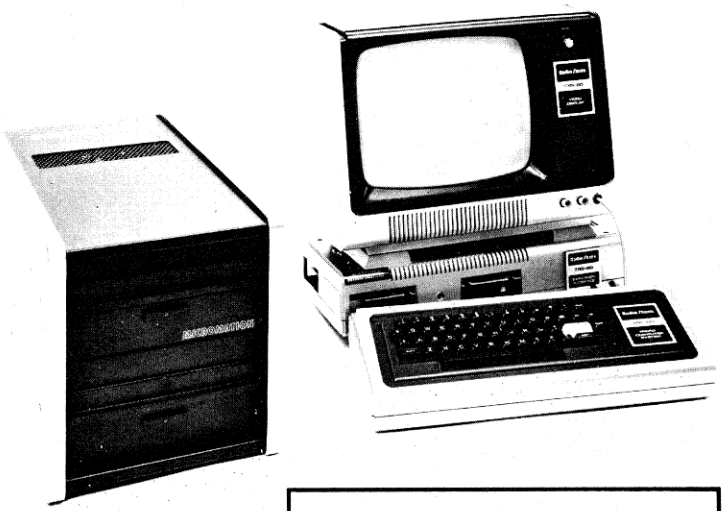
1-megabyte MBII-80 MEGABOX for the TRS-80
List $2295 . . . . . OUR PRICE $1949

2-megabyte MBIV-80 MEGABOX, List $3095 . . $2629

CP/M Operating System for above $150
Microsoft Patch to ROM BASIC $150

Lifeboat CP/M for TRS-80, List $145 . . . . OUR PRICE $129
CBASIC . . . . . ONLY $99
Microsoft software for TRS-80 (requires CP/M) — Fortran, Macro Assembler, Linking Loader, Subroutine Library, and Disk Text Editor, List $150 . . . . . $139

Full line of CP/M* Software available from MiniMicroMart!
**** CP/M is a trademark for the Digital Research Corporation***



PRINTERS

NEC SPINWRITER for TRS-80, 55 cps, letter quality
5530 RO/Centronics Interf., $3275 2475

ANADEX80-col dot matrix, $995 . . . $895

INTEGRAL DATA IP-125 w/1210 option NEW LOW $724
IP-225 w/1210 & 1250 options, List $988 . . . . . NEW LOW $834
IP225 w/1210, 1250, 1221, 1241(graphics) options, List $1098 . . NEW LOW $899

IDS440 Paper Tiger, List $995 . . . . $895
IDS440G Paper Tiger with graphics option including buffer, List $1194 . . . $1069

CENTRONICS 779-1, List $1245 . . $ 949*
779-2 w/tractor, List $1350 . . . . . 1049*
780-2 w/tractor, List $1725 1295
701-2 w/tractor, List $2205 1895
702-2 w/tractor VFU, List $2480 . . 1995
703-2 w/tractor VFU, List $2975 . . 2395
730-1 parallel interface, List $995 . . . 895†

T.I. 810 serial/parallel, List $1940 . . $1735
(T.I. 810 — printer performance exceeds Radio Shack TRS-80 Line Printer specs (150 cps).

Printers for TRS-80 require Level II machines. Printer cables extra. Call for price and order number.

* Same as Line Printer I † Same as Line Printer II

DISK DRIVES

Fully compatible with expansion interface and TRSDOS software, VISTA, PERCOM, LOBO, and others —

SAVE OVER $100 — YOUR CHOICE ONLY $369 each

CORVUS 10 megabyte hard disk — ONLY $4795

CONVERT YOUR EXISTING SELECTRIC TYPEWRITER TO A LETTER-QUALITY PRINTER . . .

Power Supply & electronics, assembled & tested. You make only a simple solenoid installation (or have the factory do it). Manufactured by Escon.
Parallel version, List $525 . . . . . $469
TRS-80 cable . . . . . $ 20

RADIO SHACK TRS-80's™, EXPANSION INTERFACES, AND DISK DRIVES at 10% DISCOUNT

Full warranty at any Radio Shack store

26-1051 4K RAM, Level-I BASIC, List $499 $449
26-1053 16K RAM, Level-I BASIC with numeric keypad, List $729 656
26-1054 4K RAM, Level-II BASIC, List $619 557
26-1056 16K RAM, Level-II BASIC with numeric keypad, List $849 764
26-1140 Expansion Interface with 0 Ram, List $299 269
26-1141 Expansion Interface with 16K RAM, List $448 403
26-1142 Expansion Interface with 32K RAM, List 597 537
26-1160 First Disk Drive (includes DOS, cable), List $499 449
26-1180 Voice Synthesizer, List $399 359

SHIPPING, HANDLING AND INSURANCE: Add $5 for Selectric Converter, $6 for disk drives, $10 for Megabox. Centronics printers shipped freight collect. Contact us for shipping information on other terminals and printers.

All prices subject to change and all offers subject to withdrawal without notice. Prices in this ad are for prepaid orders. Slightly higher prices prevail for other-than-prepaid orders, i.e., C.O.D., credit card, etc.

— WRITE FOR FREE CATALOG —

MiniMicroMart, Inc.

VISA master charge

✓24

1618 James Street, Syracuse, NY 13203 (315) 422-4467 TWX 710-541-0431

SMALL SOFTWARE SYSTEM ✓30 TRS-80 PRODUCTS SMALL SOFTWARE SYSTEM

MACHINE LANGUAGE GAMES

AIR RAID, BARRICADE or **RSL-1: - $10.00** each, all 3 for **$25.00**

Three popular machine language games now at new lower prices! All run on both Level-1 and Level-2 and require only 4K of memory. All may be converted for disk using DCV-1.

AIR RAID: Shoot down high speed aircraft with a steerable ground based missile launcher! Aircraft explode dramatically when hit, sometimes destroying other nearby planes! Score is tallied for each hit or miss, and the highest score is saved to be challenged by other players. Provides hours of fun for you, and a super program for entertaining friends!

BARRICADE: "BREAKOUT" for the TRS-80! Break through 5-wall Barricade with high-speed ball and keyboard controlled paddle! Trap the ball among the walls and watch it destroy the 100 blocks! Select 96 different options to challenge expert or beginner. 3 scores with the best of each saved to be challenged by other players.

RSL-1: Enter graphic patterns with repeating keyboard! Save patterns on tape (4 furnished). Play LIFE, a game of birth, growth and death of a colony of cells. FAST - about 1 second per generation! Hours of fascinating patterns!

ADVENTURE! $14.95 each, (3 or more, **$12.50** each)

Level-II 16K machine language versions of Adventure, the current rage of the big time-share computers! 6 versions: 1-Adventureland, 2-Pirate's Adventure, 3-Mission Impossible, 4-Voodoo Castle, 5-The Count, and 6-Strange Odyssey. Each is a challenge that can take weeks to solve! Furnished on tape; convert to disk with DCV-1.

UTILITIES

RSM-2: AN ADVANCED TAPE MONITOR FOR 16K TRS-80'S - $26.95
RSM-2D: THREE MONITORS FOR TRS-80 DISK SYSTEMS - 29.95

22 commands to control your TRS-80 Z-80 processor! Examine ROM's, test RAM, program in machine language, read/write machine language tapes, and much more! SYMBOLIC DUMP will disassemble memory into Z-80 mnemonics! HEX and two ASCII memory dumps; EDIT, MOVE, EXCHANGE, VERIFY, FILL, ZERO, TEST, or SEARCH memory, read/write SYSTEM tapes, enter BREAKPOINTS, PRINT with TRS232 or Centronics, and read/write disk sectors directly! Radio Shack uses RSM; see page 4-17 of your disk manual! RSM-2 tape loads at top of 16K LEVEL I or II; RSM-2D disk includes 3 versions for 16K, 32K and 48K.

RSM-2 RELOCATOR: PUT RSM-2/2D ANYWHERE IN MEMORY - 9.95

This new program loads with the RSM-2/2D "L" tape command, then asks you where you want RSM-2 located. Your answer causes a new version to be created and executed! Instructions included for saving your new versions.

DCV-1: CONVERT SYSTEM PROGRAMS TO DISK FILES -$9.95

EDTASM, The Electric Pencil, Air Raid, RSL-1, ESP-1, T-BUG, or nearly any SYSTEM tape can be executed from disk, even if it interferes with TRSDOS! DCV-1 loads system tapes into high memory and adds a block-move routine. TAPEDISK is then used to create a disk file. When accessed from disk, the program loads into high memory, moves itself to its correct address, then jumps there and executes! New version works with TRSDOS 2.2.

BASIC-1P: LEVEL-1 BASIC WITH PRINTING! - $19.95

Loads into the top 4K of 16K TRS-80's and uses any LEVEL-I BASIC program or DATA tape (up to 12K in length) without conversion! NEW commands, LPRINT and LLIST to print with either our TRS232 or the Radio Shack printer! Loads from tape or disk (furnished on tape). All LEVEL-I abbreviations and functions supported!

CALIFORNIA Residents please add 6% state sales tax.

MODEL-II TRS-80

Small System Software is currently developing several programs for the Model II TRS-80. An enhanced RSM monitor with many new features will be available about January. We are adapting CP/M (tm Digital Reaserch, Inc.) in conjunction with Lifeboat Associates. CP/M for the Model II will be a "standard" version and will run all existing CP/M software, including Cobol, Fortran, C-Basic, M-Basic, business and accounting packages, etc. Write for details!

PROFESSIONAL SOFTWARE

MICROSOFT SOFTWARE PACKAGES - $80.00 each, **$150.00** for both

ASSEMBLER PACKAGE: Macro Assembler uses Zilog mnemonics and produces relocatable code! Includes Linking Loader, Editor and Cross Reference Utilities. Requires 32K and 1 disk drive.

FORTRAN PACKAGE: A true Fortran Compiler (faster than Basic). Linking Loader combines Fortran, Assembly and Library modules into one program! Editor and extensive Library are included. Requires 32K and at least 1 disk drive.

THE ELECTRIC PENCIL FOR TRS-80 DISK SYSTEMS - $150.00
THE ELECTRIC PENCIL FOR TRS-80 TAPE SYSTEMS - 99.95

Write text, delete, insert, or move words, lines or paragraphs, save text on tape (or disk), then print formatted copy with our TRS232 or Centronics printer (RS-232-C with disk version). Right justification, page titling and numbering, transparent cursor and repeating keyboard. Upper case only, or lowercase with modification. Level-1 or 2 16K (tape version).

CP/M OPERATING SYSTEM WITH TRS232 SOFTWARE - $145.00

SMALL SYSTEM SOFTWARE/LIFEBOAT ASSOCIATES version of CP/M. Includes TRS232 and RS-232-C software, lower-case support, debounce, DCV-2 and other unique utilities. CP/M Editor creates and modifies all files. Assemble from disk, placing HEX and PRINT files back onto disk! Includes DDT (Dynamic Debugging Tool), PIP (Peripheral Interchange Program), and more! CP/M is a trademark of Digital Reasearch, Inc.

PRINTER SUPPORT

TRS232 PRINTER INTERFACE - $49.95 (+$2.00 shipping)

Assembled and tested output port for TRS-80 printing. Use Diablo, Teletype, TI Silent or any RS-232 or 20-mil current loop ASCII printer. Expansion interface not required. Use with LEVEL-II BASIC, CP/M, BASIC-1P, ELECTRIC PENCIL, RSM-2/2D or your own programs! Standard cassette software included, or order new "FORMATTER" (below) for enhanced printer control.

TRS232 "FORMATTER" SOFTWARE PACKAGE - $14.95

Enhanced software for with Level-2 Basic and our TRS232. Page and line length control, form feed function, printer pause, "smart" line termination, indented continuation lines, keyboard debounce, echo screen to printer, etc. Includes BASIC cassette and BASIC and machine language source listings.

PRINTER CONVERSION PACKAGES - $9.95 EACH

Many programs do not include provisions for printing with either our TRS232 or the Radio Shack RS-232-C. We currently offer the following tapes for adding printing functions:

RSM RS-232-C: Adds RS-232-C capability to RSM-2/2D
PENCIL RS-232-C: For cassette version of Electric Pencil
EDTASM PRINT: TRS232 and RS-232-C for disk/tape EDTASM

OTHER TRS-80 PRODUCTS

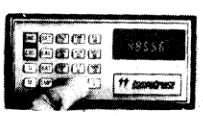
ESP-1: $29.95 Assembler, Editor & Monitor (8080 mnemonics)
LST-1: 8.00 Listing of Level-1 BASIC with some comments

SMALL SYSTEM SOFTWARE Your BANKAMERICARD welcome here **P. O. BOX 366** MasterCharge **NEWBURY PARK, CA 91320**

COMPUCRUISE

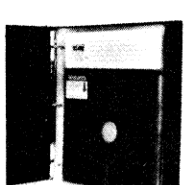
Put a computer in your car, which gives you the most effective and functional cruise control ever designed, plus complete trip computing, fuel management systems, and a remarkable accurate quartz crystal time system.

So simple a child can operate, the new CompuCruise combines latest computer technology with state-of-the-art reliability in a package which will not likely be available on new cars for years to come • Cruise Control • Time, E.T., Lap Timer, Alarm • Time, Distance, Fuel to Arrival • Time, Distance, Fuel to Empty • Time, Distance and Fuel on Trip • Current or Average MPG, GPH • Fuel Used, Distance since Fillup • Current and Average-Vehicle Speed • Inside, Outside or Coolant Temperature • Battery Voltage • English or Metric Display. $169.95 , without cruise control $129.95.

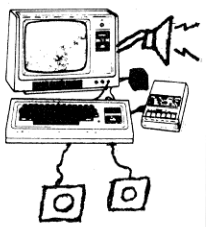


FLOPPY DISK STORAGE BINDER

This black vinyl three-ring binder comes with ten transparent plastic sleeves which accommodate either twenty, five-inch or ten, eight-inch floppy disks. The plastic sleeves may be ordered separately and added as needed. A contents file is included with each sleeve for easy identification and organizing. Binder & 10 holders $14.95 Part No. B800; Extra holders 95¢ each. Part No. 800



GAME PADDLES & SOUND For Your TRS 80 (Level II or Disk)

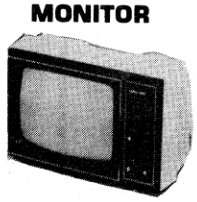


Includes: 2 game paddles, interface, software, speaker, power supply, full documentation including: schematics, theory of operation, and user guide; plus 2 games on cassette (Pong and Starship War). $79.95 Complete Part No. 7922C

DISKETTES

Box of 10, 5" $29.95, 8" $39.95.
Plastic box, holds 10 diskettes, 5" - $4.50, 8" - $6.50.

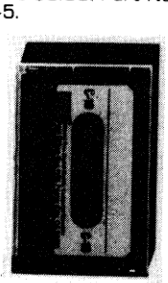
LEEDEX MONITOR



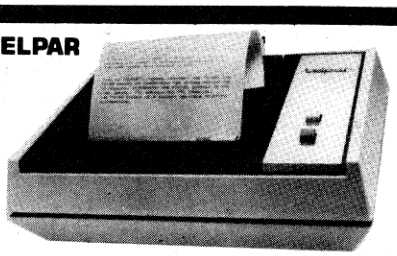
12" Black and White • 12 MHz Bandwidth • Handsome Plastic Case • $139.00

DIGITAL CASSETTE

5 min. each side. Box of 10 $9.95. Part No. C-5.



TELPAR



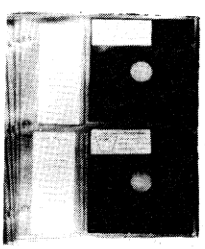
TTL parallel, TTL serial, RS232C serial, 20 mil loop serial. Quiet, clean — no ribbons or ink; 24 characters/second; Upper/lower case 96 ASCII characters; Signalling rates: Serial Mode — 110 or 300 Baud; Parallel mode — up to 960 characters/second; For applications over 48 columns, automatically prints on next line; Prints 1 character at a time — ideal for interactive applications; Local line feed for advancing paper; Parity check (odd, even or none); ASCII control codes: back space, carriage return, line feed (implemented as a new line), vertical tab (1 line), negative vertical tab (on receipt of ESC, VT), Bell (9 microsecond pulse on receipt of BEL) PS-48C Printer $699, with Keyboard, $895. Paper roll 5½"x164' $3.00

SARGON: A Computer Chess Program

Features the complete program that won the 1978 West Coast Computer Faire Tournament. Part No. 00603 — TRS-80 Level II; Part No. 00604 — Apple II (24K). $19.95

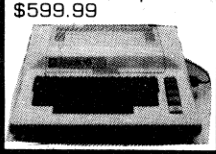
DISK JACKET™

Made from heavy duty .0095 matte plastic with reinforced grommets. The minidiskette version holds two 5-1/4 inch diskettes and will fit any standard three ring binder. The pockets to the left of the diskette can be used for listing the contents of the disk. Please order only in multitudes of ten. $9.95/10 Pack.



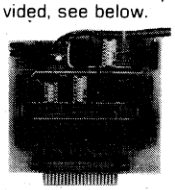
ATARI 800

Computer with 8K $995.00, disk drive $549.00, printer $599.99



TRS-80 SERIAL I/O

• Can input into basic • Can use LLIST and LPRINT to output, or output continuously • RS-232 compatible • Can be used with or without the expansion bus • On board switch selectable baud rates of 110, 150, 300, 600, 1200, 2400, parity or no parity odd or even, 5 to 8 data bits, and 1 or 2 stop bits. D.T.R. line • Requires +5, -12 VDC • Board only $19.95 Part No. 8010, with parts $59.95 Part No. 8010A, assembled $79.95 Part No. 8010 C. No connectors provided, see below.

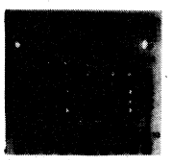


EIA/RS-232 connector Part No. DB25P $6.00, with 9', 8 conductor cable $10.95 Part No. DB25P9.

3' ribbon cable with attached connectors to fit TRS-80 and our serial board $19.95 Part No. 3CAB40.

RS-232/20mA INTERFACE

This board has two passive, opto-isolated circuits. One converts RS-232 to 20mA, the other converts 20mA to RS-232. All connections go to a 10 pin edge connector. Requires +12, and -12 volts. Board only $9.95, part no. 7901, with parts $14.95 Part No. 7901A.



COMPUCOLOR II

Model 3, 8K $13 95, Model 4, 16K $15 95, Model 5, 32K $18 95. Prices include color monitor, computer, and one disk drive.

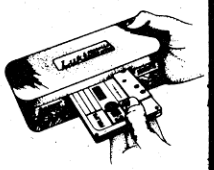
PET COMPUTER

With 32K & monitor - $1195. Dual Disk Drive - $1195.

apple II or APPLE II PLUS

16K - $995 , 32K - $1059, 48K - $1123. Disk & cont. $589

CASSETTE TAPE ERASER



REMOVES RECORDINGS IN ONE SECOND! The process eliminates static positive / negative ions and maintains original tone quality with minimal tape hiss • To improve tone quality • To reduce hissing • For quick and easy to erase • No battery or liquid required • Powerful and effective action • Unconditional 2 year guarantee. ERASER-8 $19.95.

16K RAMS

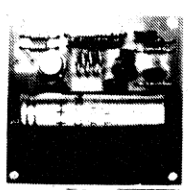
For the Apple, TRS-80 or Pet $8 each Part No. 4116/2117.

APPLE II HOBBY/ PROTOTYPING CARD

$14.95 Part No. 7907

T.V. INTERFACE

• Converts video to AM modulated RF, Channels 2 or 3. So powerful almost no tuning is required. On board regulated power supply makes this extremely stable. Rated very highly in Doctor Dobbs' Journal. Recommended by Apple • Power required is 12 volts AC C.T., or +5 volts DC • Board only $7.60 part No. 107, with parts $13.50 Part No. 107A



Game Playing with BASIC Tape 3

Features these 8 programs: Slot Machines, Blackjack, Roll the Dice, Tower of Hanoi, 15 Puzzle, Buried Treasure, Odd Cell Magic Square, and 4x4 Magic Square. Part No. 00401 — PET; Part No. 00403 — TRS-80 Level II; Part No. 00404 — Apple II. $9.95

To Order: Mention part no. description, and price. In USA shipping paid by us for orders accompanied by check or money order. We accept C.O.D. orders in the U. S. only, or a VISA or Master Charge no., expiration date, signature, phone no., shipping charges will be added. CA residents add 6.5% for tax. Outside USA add 10% for air mail postage and handling. Payment must be in U. S. dollars. Dealer inquiries invited. 24 hour order line (408) 448-0800

Send for FREE Catalog . . . a big self-addressed envelope with 41¢ postage gets it fastest!

ELECTRONIC SYSTEMS Dept. 80 P. O. Box 21638, San Jose, CA USA 95151

80 microcomputing T.M. bookshelf

● **INTRODUCTION TO MICROPROCESSORS**—by Charles Rockwell of MICROLOG, is an ideal reference for the individual desiring to understand the hardware aspects of microprocessor systems. Describes the hardware details of computer devices in terms the beginner can understand, instead of treating the micro chip as a "black box." Specific systems are not described and programming is only briefly discussed. $17.50 U.S. and Canada; $20 foreign (in US funds).* (BK1032)

● **BASIC NEW 2ND EDITION**—by Bob Albrecht. Self-teaching guide to the computer language you will need to know for use with your microcomputer. This is one of the easiest ways to learn computer programming. $6.95.* (BK1081)

● **HOBBY COMPUTERS ARE HERE!**—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.* (BK7322)

● **THE NEW HOBBY COMPUTERS**—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.* (BK7340)

● **PIMS: PERSONAL INFORMATION MANAGEMENT SYSTEM**—Learn how to unleash the power of a personal computer for your own benefit in this ready-to-use data-base management program. $9.95.* (BK1009)

● **HOW TO MAKE MONEY WITH COMPUTERS**—In 10 information-packed chapters, Jerry Felsen describes more than 30 computer-related, money-making, high profit, low capital investment opportunities. $15.00.* (BK1003)

● **INSTANT BASIC**—by Jerald R. Brown. For the personal computer enthusiast or the user of DEC's BASIC PLUS language, here is a new book to teach you BASIC. It teaches BASIC to beginners using interesting programming ideas and applications that will be easily understood by the home computer programmer. BASIC PLUS users know that the two languages are very similar, so this book can be used by them as well. This is an "active participation" workbook, designed to be used with your home computer so you can learn by doing! Ideas are slowly introduced in a nonmathematical context so the beginner can quickly learn good programming techniques. $6.00.* (BK1131)

● **HOW TO SELL ANYTHING TO ANYBODY**—According to *The Guiness 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.* (BK7306)

● **MICROCOMPUTER DICTIONARY**—This microcomputer dictionary fills the need to become quickly acquainted with the terminology and nomenclature of the revolution in computers. There is also a comprehensive electronics/computer abbreviations and acronyms section. $15.95.* (BK1034)

● **MICROCOMPUTER PRIMER**—by M. Waite and M. Pardee. Describes basic computer theory, explains numbering systems and introduces the reader to the world of programming. Describes the world of microcomputing in "real world" terminology. $7.95.* (BK1035)

● **THE SECRET GUIDE TO COMPUTERS** Parts 1, 2 and 3 by Russ Walter. **Part One** describes computers in general, and after reading for ten minutes you will be writing simple BASIC programs! **Part Two** discusses computer applications. It's one thing to master the syntax of the language such as BASIC and another to solve problems using the new tool. **Part Three** describes programming languages. Ever heard of APL and QLISP? BASIC is not the only language used to program computers. 7th Edition. Part I —$2.75—BK1050*; Part II—$2.50—BK1051*; Part III—$3.50—BK1052.*

● **SOME COMMON BASIC PROGRAMS**—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. $9.50 paperback.* (BK1053)

● **HOW TO BUILD A MICROCOMPUTER—AND REALLY UNDERSTAND IT**—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.* (BK7325)

● **BASIC COMPUTER GAMES**—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.* (BK1074)

● **WHAT TO DO AFTER YOU HIT RETURN**—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.* (BK1071)

● **YOUR HOME COMPUTER**—by James White, is an introduction to the world of personal microcomputing. This book tells you everything you want to know about home computing and gives the computer novice a painless introduction to microcomputer technology and terminology, beginning with what computers are and how they work. This basic book requires no prior knowledge or experience in electronics or computing. It provides information about home computer kits; guidelines for selecting and building your own microcomputer, how to use your home computer and what you can do with it; lists of computer stores, clubs, periodicals; and answers to many more of your questions about microcomputers and the jargon surrounding the personal computing scene today. $6.00.* (BK1172)

● **AN INTRODUCTION TO MICROCOMPUTERS, VOL. 0**—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.* (BK1130)

● **VOL. I**—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. $9.50.* (BK1030)

● **VOL. II**—(loose leaf) $25.00* BK1031; (with binder) $30.00* BK1040——Contains descriptions of individual microprocessors and support devices used only with the parent microprocessor. Volume II describes all available chips.

● **VOL. III**—BK1132 (loose leaf)—$15.00*; BK1133 (with binder)—$20.00.* Contains descriptions of all support devices that can be used with any microprocessor.

● **ADD 'N STAC**—If you've got programs lying around and want them organized in easy-to-locate fashion Add 'N Stac is the answer. Each unit stores eight cassettes. Each module locks together with the next and grows with your collection. Build yourself a software library with Add 'N Stac by ordering from the Book Nook. As your library grows you'll need more of these handy units so order more than one today. Several colors are available and you can mount them to your wall, desk, table or keep them loose for taking with you. Colors and prices are: Smoke $3.00*; Black, Dark Blue, Orange, Brown, White, Red $2.50.* Please specify color(s). (BK1003)

---

*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. All orders, add $1.00 handling.

FOR TOLL FREE ORDERING CALL 1-800-258-5473

80 PREVIEW

WHAT TO LOOK FOR IN THE MARCH 80

INVESTMENT

Here we have an in depth look at computer investment analysis. The bottom line is return on investment and the program discussed gives a comprehensive analysis of major investment considerations. Buy low—sell high with us next month.

MEMORY

Nagging doubts about that whizzbang expansion memory kit? Test out your RAM with this BASIC program (any suspect chips are indicated on the video display or printer) with a run time of less than 15 minutes for 16K.

lowercase

The first step in using your system for word processing is a lowercase modification to the keyboard. With just a little bit of electronic know-how, and some readily available parts you can install this mod in an evening. See how next month.

UTILITIES

If you've done any programming at all, you'll be familiar with the sinking feeling that occurs when there is no space left for that one extra line. Next month we have a review of the RACET computes Remodel/Proload utility package, with it your program can be renumbered plus subroutines can be inserted from tape.

PRINTERS APPRENTICE

If the prospect of consulting appeals to you, watch for this piece in next month's 80. The author details the steps involved in setting up a consulting service to printers in his area. A full program listing for providing print quotes is included.

EQUATIONS

Curious about the techniques you can use to solve algebraic equations? No? Well read this anyway, you never know when it may be useful!

THE 4K BRAIN?

Artificial intelligence, the ability of machines to do tasks which would require intelligent responses if performed by humans, is a fascinating area of study. Next month, in 4K Intelligence, we will explore the world of AI as it applies to 80 systems. A game called Hexpawn, in which the computer modifies its own program in response to opponent moves, is investigated and the fundamentals of AI program structure are discussed.

ALL THIS PLUS MUCH MORE!

80 ADVERTISERS

80 microcomputing™

READER SERVICE

To receive more information from any of the advertisers in this issue of 80 Microcomputing, circle the number on the Reader Service Card that corresponds with the Reader Service number on the ad in which you are interested. You will find these 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. If the attached card has been used, send your name, address & zip, plus the Reader Service number and issue date to 80 Microcomputing Reader Service Department, Peterborough NH 03458.

Use these post-cards to get free information from our advertisers and to order from the

80 microcomputing™

book shelf.

READER SERVICE REPLY CARD

Return this card to receive full information on the many products advertised in this issue.
☐ Subscriber ☐ Newsstand. This card is valid until March 31, 1980

1	16	31	46	61	76	91	106	121	136	151	166	181	196
2	17	32	47	62	77	92	107	122	137	152	167	182	197
3	18	33	48	63	78	93	108	123	138	153	168	183	198
4	19	34	49	64	79	94	109	124	139	154	169	184	199
5	20	35	50	65	80	95	110	125	140	155	170	185	200
6	21	36	51	66	81	96	111	126	141	156	171	186	201
7	22	37	52	67	82	97	112	127	142	157	172	187	202
8	23	38	53	68	83	98	113	128	143	158	173	188	203
9	24	39	54	69	84	99	114	129	144	159	174	189	204
10	25	40	55	70	85	100	115	130	145	160	175	190	205
11	26	41	56	71	86	101	116	131	146	161	176	191	206
12	27	42	57	72	87	102	117	132	147	162	177	192	207
13	28	43	58	73	88	103	118	133	148	163	178	193	208
14	29	44	59	74	89	104	119	134	149	164	179	194	209
15	30	45	60	75	90	105	120	135	150	165	180	195	210

This month's BEST article begins on page _____.

I own a: ☐ Level I ☐ Level II ☐ Model II ☐ No system yet.

Name _____
(Please type or print clearly)

Address _____

City _____ State _____ Zip _____

2/80

PLACE STAMP HERE

80 microcomputing™

peterborough nh 03458

Use these postcards to get free information from our advertisers and to order from the 80 Microcomputing bookshelf

80 microcomputing™

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 T.M.

peterborough nh 03458

BX1003 Add 'n Stac—$2.50
smoke color—3.00

BX1074 Basic Computer Games—$7.50

BK1081 Basic New 2nd Edition—$6.95

BK7322 Hobby Computers Are Here—$4.95

BK7325 How to Build A Microcomputer and Really Understand It—$9.95

BK1003 How To Make Money With Computers—$15.00

BK7306 How To Sell Anything to Anybody—$2.25

BK1131 Instant BASIC—$6.00

Introduction To Microcomputers:

- **BK1130** Vol. 0—$7.95
- **BK1030** Vol. I—$9.50
- **BK1031** Vol. II(loose leaf)—$25.00
- **BK1040** Vol. II(w/binder)—$30.00
- **BK1132** Vol. III(loose leaf)—$15.00
- **BK1133** Vol. III(w/binder)—$20.00

BK1032 Introduction to Microprocessors—$17.50 USA & Canada; $20 elsewhere

BK1034 Microcomputer Dictionary—$15.95

BK1035 Microcomputer Primer—$7.95

BK7340 New Hobby Computers, The—$4.95

BK1009 PIMS: Personal Information Management System—$9.95

BK1050 Secret Guide to Computers, The —$2.75

- **BK1051** Part II—$2.50
- **BK1052** Part III—$3.50

BK1053 Some Common BASIC programs—$9.50

BK1071 What To Do After You Hit Return—$10.95

BK1172 Your Home Computer—$6.00

80 microcomputing book order form

Please send me the following **80 Microcomputing** products (please print or type!)

	Cat. #	Description	Unit Price	Total
			Add $1.00 Handling	
			Total	

Enclosed $_____ ☐ Check ☐ Money Order

☐ Master Charge ☐ Visa ☐ American Express

Credit Card #_____ Exp. date_____

Signature _____

Name _____

Address _____

City _____

State _____ Zip_____

80 microcomputing T.M.

peterborough nh 03458

2/80

Ask for Instant Software at a computer store near you.

Alabama

Anderson Computers
3156 University Dr., Huntsville

The Computer Shack
913 Shadyview Lane, Adamsville

Computerland of Huntsville
3020 University Dr., Huntsville

Olensky Bros.
3763 Airport Blvd., Mobile

Arizona

Ham Shack
450 6-A N. 16th St., Phoenix

Millets TV & Radio
621 East Broadway, Mesa

Resalem Electronics
16610 Meadow Park Dr., Sun City

California

Amco Elect. Supply
635 E. Arrow Hwy., Azusa

Byte Shop of Fairfield
87 Marina Center St., Suisun City

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

Computer Components of Burbank
3808 West Verdugo Ave., Burbank

Computer Components of South Bay
15818 Hawthorne Blvd., Lawndale

Computers Made Easy
819 East Ave. Q-9, Palmdale

Computer World
6791 Westminster Ave., Westminster

Computerland
16720 S. Hawthorne, Lawndale

Computerland of San Francisco
117 Fremont St., San Francisco

Computerland of W. LA
6840 La Cienega Blvd., Inglewood

Coast Electronics
3118 No. Main St., Morro Bay

Hobby World
19511 Business Ctr. Dr., Unit 6
Borthridge

I.C.E. House Inc.
398 North E. St., San Bernardino

Marfam Co.
6351 Almadin Rd., San Jose

Microsun Computer Center
2989 North Main St., Walnut Creek

Opamp/Technical Books
1033 N. Sycamore Ave., Los Angeles

Radio Shack Dealer
8250 Mira Mesa Blvd., San Diego

Santa Rosa Computer Center
604 7th St., Santa Rosa

Silver Spur Elect. Comm.
3873 F. Shaefer Ave., Chino

The Computer Store
820 Broadway, Santa Monica

Colorado

Byte Shop
3464 S. Acoma St., Englewood

Colorado Computer Systems
311 W. 74th Ave., Westminster

Computerland of North Denver
8749 Wadsworth Blvd., Arvada

The Computer Store
2300 Welton St., Denver

Connecticut

American Business Computers
Rt. 184 & Rt. 117, Groton

Computerlab
130 Jefferson, New London

The Computer Store
43 South Main St., Windsor Locks

D.C.

The Program Store
4200 Wisconsin Ave., N.W.,
Washington, D.C.

Florida

Adventure International
200 Bald Cypress Ct., Longwood

Alvorado Export
8525 NW 56th St., Miami

AMF Electronics
11146 N. 30th St., Tampa

Boyd-Ebert Corporation
1328 West 15th St., Panama City

Computer Center
6578 Central Ave., St. Petersburg

Computerland of Boca Raton
500 E. Spanish River Blvd.
Boca Raton

Computerland of Ft. Lauderdale
3963 N. Federal Hwy., Ft. Lauderdale

Computerland of Jacksonville
2777-6 University Blvd. W.,
Jacksonville

Computer Shack
3336 Beach Blvd., Jacksonville

Curtis Waters Enterprises
236 Talbot Ave., Melbourne

Heath Kit Electronic
4705 W. 16th Ave. Center, Hialeah

Sound Ideas
2201-C N.W. 13th, Gainesville

Ukatan Computer Store
Airport Rd., Destin

Georgia

Atlanta Computer Mart
Atlanta

Computerland of Atlanta
2423 Cobb Parkway, Smyrna

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

Bloomington Normal Computer
Works
124 E. Beaufort, Normal

Computer Station
3659 Nameoki Rd., Granite City

Midwest Micro Computers, Inc.
708 S. Main St., Lombard

Indiana

Computer Center of South Bend
51591 US 31 North, South Bend

Iowa

Cyberia, Inc.
2330 Lincoln Way, Ames

Memory Bank
4128 Brady St., Davenport

Kansas

Central Kansas Computers
6 S. Broadway, Herington

Louisiana

Computer Shoppe Inc.
3225 Danny Park, Suite 222, Metairie

Maine

Radio Shack
315 Main Mall Rd., So. Portland

Maryland

Computers, Etc.
13 A. Allegheny Ave., Towson

Jack Fives Electronics
4608 Debilen Circle, Pikesville

The Comm Center
9624 Ft. Meade Rd., Laurel

Massachusetts

ComputerCity
5 Dexter Row, Charlestown

Computer Packages Unlimited
244 W. Boylston St., West Boylston

Lighthouse Computer Software
14 Fall River Ave., Rehobath

New England Electronics Co.
679 Highland Ave., Needham

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 Rochester
301 S. Livernois, Rochester

Computerland of Southfield
29673 Northwestern Hwy., Southfield

Computer Mart
560 W. 14 Mile Rd., Clawson

Golden Anvil
259 Broadway, South Haven

Hobby House
1035 W. Territorial Rd., Battle Creek

Minnesota

Zim Computers
5717 Xerxes Ave., N. Brooklin Center

Mississippi

Dyer's, Inc.
200 E. Main St., West Point

Missouri

Computervan, Inc.
51 Florissant Oaks Shopping Center,
Florissant

Consolidated Software
16501 Greenwald Court, Belton

Montana

Intermountain Computer
529 So. 9th St., Livingston

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

Nevada

Century 23
4566 Spring Mountain Rd., Las Vegas

Home Computers
1775 Tropicana #2, Las Vegas

New Hampshire

ComputerCity
1525 S. Willow, Manchester

Computerland of Nashua
419 Amherst St., Nashua

Portsmouth Computer Center
31 Raynes Ave., Portsmouth

New Jersey

Computer Encounter
2 Nassau St., Princeton

Radio Shack/J&J Electronic
Mansfield Shopping Ctr.
Rt. 57 Allen Rd., Hackettstown

The Computer Emporium
Bldg. 103, Avenues of Commerce
2428 Route 38, Cherry Hill

The Bargain Brothers
Glen Roc Shopping Center
216 Scotch Road, Trenton

New Mexico

South West Computer Center
121 Wyatt Drive, Suite 7, Las Cruces

New York

Aristo Craft
314 Fifth Ave., NYC

Automatic Systems Developers
Industry St., Poughkeepsie

Bits & Bytes
2800 Straight Rd., Fredonia

Computer Corner
200 Hamilton Ave., White Plains

Computer Factory
485 Lexington Ave., NYC

Computer House, Inc.
721 Atlantic Ave., Rochester

Computerland
225 Elmira Rd., Ithaca

Computerland of Nassau
79 Westbury Ave., Carle Place

Computer Shop
Rte. 28, West Harley

Comtek Electronics, Inc.
2666 Coney Island Ave., Brooklyn

Comtek Electronics, Inc.
Staten Island Mall
Store 220A, Staten Island

Home Computer Center
671 Monroe Ave., Rochester

Key Electronics
Schenectady

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

North Carolina

Byte Shop of Raleigh
1213 Hillsborough St., Raleigh

Ohio

Astro Video Electronics
504 E. Main St., Lancaster

Cincinnati Computer Store
4816 Interstate Dr., Cincinnati

Computerland
1288 Som Rd., Mayfield Heights

Computer Store of Toledo
18 Hillwyck Dr., Toledo

Forbees Microsystems Inc.
35 N. Broad, Fairborn

Heath Kit Co.
2500 Morst Rd., Columbus

Microcomputer Center
7900 Paragon Rd., Dayton

Micro-Mini Computer World
74 Robinwood, Columbus

Micro Processor Systems
Wadsworth

Modern Communications
681 Hamilton Cleves Pike, Hamilton

21st Century Shop
16 Convention Way, Cincinnati

Oklahoma

Vern Street Products
Radio Shack Dealer
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

Pennsylvania

Artco Elect.
302 Wyoming Ave., Kingston

Artco Elect.
Back Mountain Shop. Ctr.,
Shavertown

Computerland of Harrisburg
4644 Carlisle Pike, Mechanicsburg

Erie Computer Co.
2127 West 8th St., Erie

Personal Computer Corp.
24-26 West Lancaster Ave., Paoli

Personal Computer Corp.
Frazer Mall, Lancaster Ave., Frazer

The Computer Workshop of
Pittsburgh
4170 William Penn Hwy., Murrysville

Wes Fasnacht
8 York Town Ave., West Chester

Rhode Island

Digital World Inc.
329 Bald Hill Rd., Warwick

South Carolina

Seely Communications
1084 Broad St., Sumter

South Dakota

CB Radio Shack
21st and Broadway, Yankton

Tennessee

Computerlab
671 S. Menden Hall Rd., Memphis

H & H Electronics Inc.
509 N. Jackson St., Tullahoma

Texas

Computercraft Inc.
3211 Fondren, Houston

Computer Port
926 N. Collig, Arlington

Houston Microcomputer Tech.
5313 Bissonet, Bell Aire

Interactive Computers
7620 Dashwood Rd., Houston

K.A. Elect.
9090 Stemmons Frwy., Dallas

Pan American Elect. Inc.
1117 Conway, Mission

Ram Micro Systems
6353 Camp Bowie Blvd., Ft. Worth

Reb's Mail Order Electronics
5439 Doliver, Houston

Virginia

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

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

Wyoming

Computer Concepts
617 W. 16th St., Cheyenne

Puerto Rico

The Microcomputer Store
1568 Ave. Jesus T. Pinero
Caparra Terrace

Guam

The Fun Factory
851 Marine Dr., Tamuming

Canada

Computerland of Winnipeg
715 Portage Ave., Winnipeg, Man.

Compumart
411 Roosevelt Ave., Ottawa, Ontario

Computer Mart, Ltd.
1055 Yonge St., Suite 208
Toronto, Ontario

Galactia Computers
103rd Ave., Edmonton, Alberta

Micromatic Systems Inc.
101 8136 Park Rd., Richmond, B.C.

Micron Distributing
409 Queen St., W. Toronto, Ont.

Micro Shack of W. Canada
333 Park Street, Regina, Sask.

Orthon Holdings Ltd.
12411 Stony Plain Road
Edmonton, Alberta

Total Computer Systems
Ajax, Ontario

England

Tamays & Farr Ltd.
4 Morgan St., London

France

Sideg
45 Rue de la Chapelle, Paris

Sivea s.a.
20, Rue de Leningrad, Paris

Italy

HOMIC s.r.l.
Piazza De Angeli 1, Milano

West Germany

Electronic Hobby Shop
Kaiserstr. 20, Bonn

MicroShop Bodensee
Markstr. 3, 7778 Markdorf

Australia

Computerware
62 Paisley St., Footscray VIC

Deforest Software
36 Glen Tower Drive
Glen Waverly, VIC.

Softronics Micro Systems
Lindfield

Sure-Load Software
P.O. Box 26, Weston, A.C.T.

South Africa

Eddie Talberg
P.O. Box 745, Johannesburg

Instant Software™ Inc.

Peterborough, N.H. 03458 603-924-7296

STARTER KIT

EXATRON STRINGY FLOPPY FOR THE TRS-80

Recommended initial purchase:

Exatron Stringy Floppy with Level III BASIC	$249.50
5 Wafers each: 5', 10', 20', 50'	40.00
Bus Extender, 2-for-1	15.00
ESF Machine Language Monitor	9.95
(Plus shipping and tax if applicable)	$314.45

SPECIAL PRICE FOR THIS STARTER KIT	$299.50
Sales Tax (California only)	
Shipping and Handling	3.00
TOTAL	

For more information see the current Exatron Stringy Floppy Owners Association Newsletter in *Microcomputing*.

If you have any questions about the product, about Exatron, or ESFOA, please call the Hot Line. Address letters to ESFOA, 3559 Ryder St., Santa Clara, CA 95051.

Stringy Floppy is a trademark of Exatron Corporation.

HOT LINE (For Calls Outside CA) 800-538-8559

excellence in electronics

exatron

3555 Ryder Street ▪ Santa Clara, California 95051
(408) 737-7111